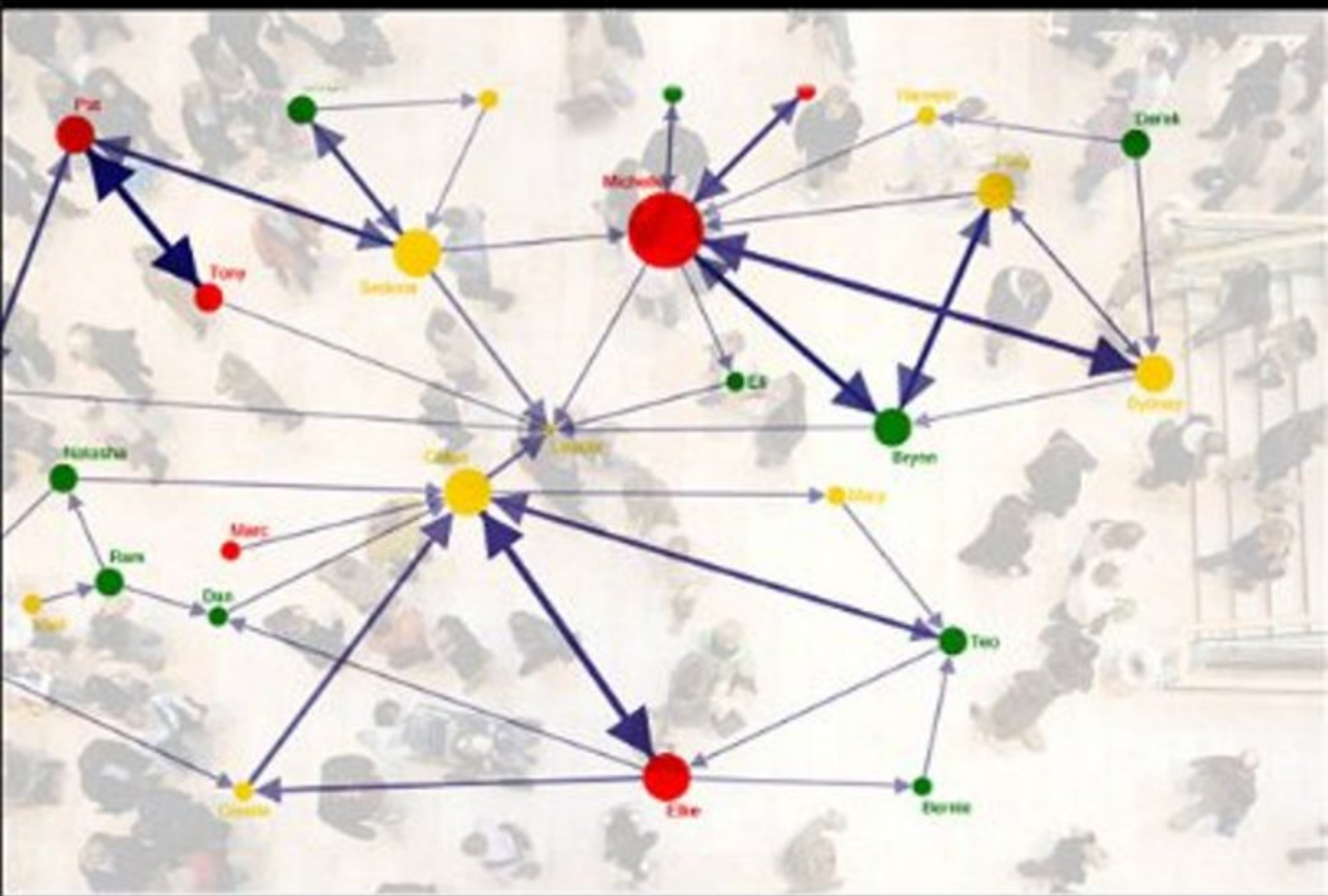
Derek Hansen Ben Shneiderman Marc A. Smith



Analyzing Social Media Networks with NodeXL

Insights from a Connected World

MK
MORGAN KAUFMANN

ANALYZING SOCIAL MEDIA NETWORKS WITH NODEXL

INSIGHTS FROM A CONNECTED WORLD

ANALYZING SOCIAL MEDIA NETWORKS WITH NODEXL

INSIGHTS FROM A CONNECTED WORLD

Derek L. Hansen
Ben Shneiderman
Marc A. Smith

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann is an imprint of Elsevier

MK MORGAN KAUFMANN

Acquiring Editor: Mary James
Development Editor: David Bevans
Project Manager: Julie Ochs
Designer: Dennis Schaefer

Morgan Kaufmann is an imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

Copyright © 2011 Elsevier Inc. All rights reserved

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods or professional practices, may become necessary. Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information or methods described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
Hansen, Derek L.
Analyzing social media networks with NodeXL : insights from a connected world/Derek L. Hansen, Ben Shneiderman, Marc A. Smith.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-12-382229-1 (pbk. : alk. paper) 1. Data mining—Computer programs. 2. Information visualization-Computer programs. 3. Webometrics—Computer programs. 4. Online social networks. 5. NodeXL. I. Shneiderman, Ben. II. Smith, Marc A., 1965– III. Title.
QA76.9.D343.H45 2010
006.7′54—dc22 2010019806

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-382229-1

Printed in China

10 11 12 13 14 10 9 8 7 6 5 4 3 2 1

Working together to grow libraries in developing countries
www.elsevier.com | www.bookaid.org | www.sabre.org
ELSEVIER BOOK AID International Sabre Foundation

For information on all MK publications visit our website at www.mkp.com

Derek: To Maren Hansen
Ben: To Jennifer Preece
Marc: To my strong ties to Madeline, Eli, and Christine

Preface

We live in the era of networks. Occasionally the aspirations of academic researchers are in harmony with the needs of commercial developers, business entrepreneurs, and government agency staffers. In our case, the authors brought complementary backgrounds in information studies, computer science, and sociology, as well as a shared interest in the interdisciplinary topics of human-computer interaction, network analysis, social media, and information visualization.

At the same time, these topics were gaining interest from the growing community of entrepreneurs who were coping with the success of their social media commercial platforms such as Twitter, Facebook, Flickr, YouTube, and the equally remarkable open source communities that produced valuable resources such as Wikipedia. These entrepreneurs perceived of their creations as networks, drawing on the language of social communities and asking questions about motivation, influence, and social roles.

The tools for social media network analysis and visualization have been emerging from many research groups and startup companies. These pioneering network analysis tools often require programming skills and knowledge of technical network terminology, making it a challenge for those without programming skills to import and make sense of network data. By 2000, network metrics had become a mature topic, but research on layout and clustering algorithms has expanded dramatically in the past decade, producing breakthroughs that raised the quality of what was possible to visualize. Similarly, strategies for filtering, highlighting, and de-cluttering networks have matured as more users tackle a broader variety of problems with increasingly large networks.

The authors have been fortunate to be part of a team with unique skills that developed the NodeXL tool. Funded by Microsoft Research External Research Projects group, the team is distributed across the United States and Europe with links to others around the world. Our weekly phone calls over more than 2 years developed requirements and set priorities so that we could substantially lower the barrier to entry for social media network analysis while at the same time raising the power offered to users seeking network insights. We chose Excel 2007 as our first platform for data import and manipulation because of its widespread acceptance in the business community, rich set of features, and convenience in extending it. Then we took the best features from existing analysis and graphics tools, closely integrating statistical methods with information visualization strategies while keeping in mind the goal of being easy to learn by a wide range of users.

The need for precise management of the visual properties of a network chart led to rich controls for vertex layout, size, color, shape, images, labels, and opacity, and parallel attributes such as thickness for edges. The desire for rapid filtering was satisfied with dynamic queries based on the ambitious use of double-slider range controls. The aspiration to support multiple sources of data led to a strong effort to import multiple networks from common platforms. The desire to gain insights into social media networks led to integration of several "spigots" that open streams of network data from popular social media network sources like YouTube, Twitter, Flickr, and personal email.

As NodeXL emerged in release after release, sometimes only weeks apart, we tested it on our own social network research projects and then used it in three academic courses, plus professional tutorials at conferences. These experiences led us to improve usability aspects and also led to important clarifications of the analysis process that improved our teaching strategies. In each stage, we raised our aspirations about what we and our users could accomplish in terms of gaining actionable insights from social media analytics. Although we are pleased with what NodeXL users are able to accomplish, we are humbled by the richness and diversity of social media analytical possibilities. The opportunities and challenges are substantial, which will keep researchers and developers productively engaged for many years to come.

Acknowledgments

The authors would like to thank the many people who have made this document and the NodeXL project possible. First, the members of the NodeXL design and development team include Natasa Milic-Frayling, Eduarda Mendes Rodrigues, Janez Brank, and Annika Hupfeld from Microsoft Research in Cambridge, England, and Jure Leskovec at Standford. We thank Tony Capone from Microsoft Research in Redmond, Washington, for his remarkable programming prowess, thoughtful discussions about features, and always courteous help to us and the NodeXL user community.

Support for NodeXL development has been generously provided by Tony Hey, Daron Green, and Dan Fay from the Microsoft Research External Research Programs group in Redmond, Washington.

We thank Eric Gleave from the University of Washington; Vladimir Barash from Cornell University; and Cody Dunne, Udayan Khurana, and Adam Perer from the University of Maryland for their intellectual contributions to our grand adventure.

We thank SeriousEats (www.seriouseats.com), which has allowed us to use data collected from its fascinating online community. We also thank Emily Mason, Chad Doran, and Rachel Collins, who collected data sets used in the book as part of their coursework and came up with compelling analyses of them. Special thanks to Chris Wilson of *Slate Magazine* for sharing the Senate 2007 voting data.

Our many users have provided remarkable feedback, but Pierre de Vries merits a specific mention for pushing NodeXL beyond our expectations. Our research collaborators Dana Rotman and Elizabeth Bonsignore have made it possible to field-test NodeXL and carefully document the results. The students of several classes who were assigned projects with NodeXL have been patient and forgiving as we refined the rough edges. Many thousands of people have downloaded NodeXL, and several have created research and business results using the tool. We appreciate the time and attention our users give the tool and the project and hope they will continue to upgrade with us as the project grows. We are grateful to these and many other people for their efforts to make NodeXL an easy and useful tool for understanding complex networks.

Andrew Fiore, the University of California at Berkeley
Barry Wellman, University of Toronto
Bill Johnston, ForumOne Communications
Cliff Lampe, Michigan State University
Dave Ashton, OsVisNet Project
David Shamma, Yahoo! Research
Eleanor Wynn, Intel Corporation
Elizabeth Bonsignore, University of Maryland
George Barnett, University of California at Davis
Han Woo Park, Yeungnam University
Howard Rheingold, social media author and teacher
Jes Koepfler, University of Maryland
John Kelly, Morningside Analytics
Julian Hopkins, Monash University
Keith Hampton, University of Pennsylvania
Lada Adamic, University of Michigan
Libby Hemphill, Illinois Institute of Technology
Mathieu Bastian, Project Gephi
Martha Russell, Stanford Media-X program
Nicky Van Zyl, Vodacom South Africa
Nosh Contractor, Northwestern University
Phillip Howard, University of Washington
Pierre de Vries, independent telecommunications policy researcher
Randy Farmer, social media strategy consulting, WikiAnswers
Rob Cross, University of Virginia
Ton Zijlstra
Yann Leroux

Our Twitter fans!

About the Authors

Derek L. Hansen is an assistant professor at the University of Maryland's iSchool and director for the Center for the Advanced Study of Communities and Information (http://casci.umd.edu), a multidisciplinary research center focused on harnessing the power of novel social technologies to support the needs of real and virtual communities. He is also an active member of the Human Computer Interaction Lab (www.cs.umd.edu/hcil).

Dr. Hansen completed his Ph.D. from the University of Michigan's School of Information where he was an National Science Foundation–funded interdisciplinary STIET Fellow (http://stiet.si.umich.edu) focused on understanding and designing effective online sociotechnical systems. His research and teaching focus on mass collaboration, information reuse, consumer health informatics, and social network analysis of online communities. One line of research focuses on helping community analysts make sense of the mass of social data available through social media sites. Another line of research applies those methods to improve our understanding of best practices for supporting mass collaboration in the medical, scientific, and entertainment domains. Finally, Dr. Hansen is involved with designing novel tools that take advantage of the unique properties of information technologies. For example, he has done work on "collection recommender systems" and social networking applications designed to anonymously disseminate information about stigmatized illnesses through social networking sites.

Ben Shneiderman (www.cs.umd.edu/~ben) is a professor in the Department of Computer Science and founding director (1983–2000) of the Human-Computer Interaction Laboratory (www.cs.umd.edu/hcil) at the University of Maryland. He was elected as a Fellow of the Association for Computing (ACM) in 1997 and a Fellow of the American Association for the Advancement of Science (AAAS) in 2001. He received the ACM SIGCHI Lifetime Achievement Award in 2001.

Shneiderman is the co-author, with Catherine Plaisant, of *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (5th ed., April 2009, www.awl.com/DTUI). With S. Card and J. Mackinlay, he co-authored "Readings in Information Visualization: Using Vision to Think" (1999). With Ben Bederson he co-authored *The Craft of Information Visualization* (2003). His book *Leonardo's Laptop: Human Values and the New Computing Technologies* appeared in October 2002 (MIT Press) (http://mitpress.mit.edu/leonardoslaptop) and won the IEEE book award for Distinguished Literary Contribution.

Marc A. Smith is a sociologist specializing in the social organization of online communities and computer-mediated interaction. He founded and managed the Community Technologies Group at Microsoft Research in Redmond, Washington, and led the development of social media reporting and analysis tools for Telligent Systems. Smith leads the Connected Action consulting group and lives and works in Silicon Valley, California.

Smith is the co-editor, with Peter Kollock, of *Communities in Cyberspace* (Routledge), a collection of essays exploring the ways identity, interaction, and social order develop in online groups.

Smith's research focuses on computer-mediated collective action: the ways group dynamics change when they take place in and through social cyberspaces. Many "groups" in cyberspace produce public goods and organize themselves in the form of a commons (for related papers see http://delicious.com/marc_smith/Paper). Smith's goal is to visualize these social cyberspaces, mapping and measuring their structure, dynamics, and life cycles. At Microsoft, he developed the "Netscan" web application and data mining engine that allows researchers studying Usenet newsgroups and related repositories of threaded conversations to get reports on the rates of posting, posters, crossposting, thread length, and frequency distributions of activity. Smith applied this work to the development of a generalized community analysis platform for Telligent, providing a web-based system for groups of all sizes to discuss and publish their material to the web and analyze the emergent trends that result. He contributes to the open and free NodeXL project (www.codeplex.com/nodexl) that adds social network analysis features to the familiar Excel spreadsheet. A tutorial on social network analysis is evolving into a book and is freely available (http://casci.umd.edu/NodeXL_Teaching). NodeXL enables social network analysis of email, Twitter, Flickr, and other network data sets.

The Connected Action consulting group (www.connectedaction.net) applies social science methods in

general and social network analysis techniques in particular to enterprise and Internet social media usage. SNA analysis of data from message boards, blogs, wikis, friend networks, and shared file systems can reveal insights into organizations and processes. Community managers can gain actionable insights into the volumes of community content created in their social media repositories. Mobile social software applications can visualize patterns of association that are otherwise invisible.

Smith received a B.S. in International Area Studies from Drexel University in Philadelphia in 1988, an M. Phil. in social theory from Cambridge University in 1990, and a Ph.D. in Sociology from UCLA in 2001. He is an affiliate faculty at the Department of Sociology at the University of Washington and the College of Information Studies at the University of Maryland.

Contributors

Robert Ackland Australian Demographic and Social Research Institute, Australian National University, Canberra, Australia

Vladimir Barash Information Science, Cornell University, Ithaca, NY

Laura W. Black School of Communications Studies, Ohio University, Athens, OH

Dan Cosley Information Science, Cornell, University, Ithaca, NY

Jennifer Golbeck University of Maryland, College Park, MD

Scott Golder Department of Sociology, Cornell University, Ithaca, NY

Bernie Hogan Oxford Internet Institute, University of Oxford, Oxford, UK

Eduarda Mendes Rorigues Department of Informatics Engineering, University of Porto, Porto, Portugal

Natasa Milic-Frayling Microsoft Research Ltd., Cambridge, UK

Dana Rotman University of Maryland, College Park, MD

Patrick Underwood Department of Sociology, University of Washington, Seattle, WA

Howard T. Welser Department of Sociology and Anthropology, Ohio University, Athens, OH

PART I

GETTING STARTED WITH ANALYZING SOCIAL MEDIA NETWORKS

This volume is organized in the form of a tree with roots, a trunk, and branches. The roots (Part I: Chapters 1 through 3) provide grounding in the history and core concepts of social media and social network analysis. The trunk (Part II: Chapters 4 through 7) focuses on the practical details of operating the free and open source NodeXL extension of the familiar Microsoft Excel spreadsheet application used for all exercises in this volume. And the branches (Part III: Chapters 8 through 15) each focus on one form of social media by describing each system, the nature of the networks that are created when people interact through it, and the kinds of analysis that can be performed to identify key people, documents, groups, and events. The results are actionable insights that can guide community managers, marketers, organizations, and members as they try to improve the quality and value of their social media initiatives.

Part I discusses the novel ways people connect through social media tools and the formal network analysis techniques that can elucidate those connections. Chapter 1 provides a high-level overview of social media initiatives and network science, explaining why the two can be so effectively combined. Chapter 2 describes the design space of social media tools, providing examples of different types of connections created by popular tools. Finally, Chapter 3 introduces the core concepts of social network analysis and visualization, which will be put to work in the remainder of the book to understand social media networks. The chapters assume no prior knowledge of these topics.

CHAPTER

1

Introduction to Social Media and Social Networks

OUTLINE

1.1 Introduction 3
1.2 A Historical Perspective 3
1.3 The Rise of Social Media as Consumer Applications 4
1.4 Individual Contributions Generate Public Wealth 5
1.5 Who Should Read This Book 5
1.6 Applying Social Media to National Priorities 6
1.7 Worldwide Efforts 7
1.8 Practitioner's Summary 8
1.9 Researcher's Agenda 8

1.1 INTRODUCTION

Billions of people create trillions of connections through social media each day, but few of us consider how each click and key press builds relationships that, in aggregate, form a vast social network. Passionate users of social media tools such as email, blogs, microblogs, and wikis eagerly send personal or public messages, post strongly felt opinions, or contribute to community knowledge to develop partnerships, promote cultural heritage, and advance development. Devoted social networkers create and share digital media and rate or recommend resources to pool their experiences, provide help for neighbors and colleagues, and express their creativity. The results are vast, complex networks of connections that link people to other people, documents, locations, concepts, and other objects. New tools are now available to collect, analyze, visualize, and generate insights from the collections of connections formed from billions of messages, links, posts, edits, uploaded photos and videos, reviews, and recommendations. As social media have emerged as a widespread platform for human interaction, the invisible ties that link each of us to others have become more visible and machine readable. The result is a new opportunity to map social networks in detail and scale never before seen. The complex structures that emerge from webs of social relationships can now be studied with computer programs and graphical maps that leverage the science of social network analysis to capture the shape and key locations within a landscape of ties and links. These maps can guide new journeys through social landscapes that were previously uncharted.

1.2 A HISTORICAL PERSPECTIVE

Network science focuses on the study of patterns of connection in a wide range of physical and social phenomena. Network researchers have explored foundational physical systems created by chemical and genetic connections, webs of consumption of which animals eat which others, and profound distributed human social phenomena such as collective action, empathy, social cohesion, privacy, responsibility, markets, motivation, and trust. In the past few decades, network researchers have developed new data collection methods, innovative mathematical techniques, and surprising predictive

theories. Just as Lord Kelvin (1824–1907) encouraged careful measurement as the method of advancing science, the new sciences of collective action, collaboration, and productive communities require new forms of measurement. Similarly, where Newton (1643–1727) and Leibniz (1646–1716) created the mathematical methods of calculus to grasp the physical world of objects in motion, social scientists are developing advanced mathematical methods for capturing social network evolution, diffusion, and decay. Like Galileo's telescope (1564–1642), Hooke's microscope (1635–1703), or Roentgen's (1845–1923) x-rays, new information analysis tools are creating visualizations of never before seen structures. Jupiter's moon, plant cells, and the skeletons of living creatures were all revealed by previous technologies. Today, new network science concepts and analysis tools are making isolated groups, influential participants, and community structures visible in ways never before possible.

Social network analysis is the application of the broader field of network science to the study of human relationships and connections. Social networks are primordial; they have a history that long predates systems like Facebook and Friendster, and even the first email message. Ever since anyone exchanged help with anyone else, social networks have existed, even if they were mostly invisible. Social networks are created from any collection of connections among a group of people and things. Social network science is itself relatively new, with roots in the early twentieth century that built on two centuries of work in the mathematics of graphs and topology. In the twenty-first century, network science has blossomed alongside a new global culture of commonplace networked communications. With widespread network connectivity, within just the past few decades, billions of people have changed their lives by creatively using social media. We use social media to bring our families and friends closer together, reach out to neighbors and colleagues, and invigorate markets for products and services. Social media are used to create connections that can bind local regions and span continents. These connections range from the trivial to the most valued, potent collaborations, relationships, and communities. Social media tools have been used successfully to create large-scale successful collaborative public projects like Wikipedia, open source software used by millions, new forms of political participation, and scientific collaboratories that accelerate research. Unheard of just a few years ago, today systems such as blogs, wikis, Twitter, and Facebook are now headline news with social and political implications that stretch around the globe. Despite the very different shapes, sizes, and goals of the institutions involved in social media, the common structure that unifies all social media spaces is a social network. All of these systems create connections that leave traces and collectively create networks.

1.3 THE RISE OF SOCIAL MEDIA AS CONSUMER APPLICATIONS

Social media are visible in the form of consumer applications such as Facebook and Twitter, but significant use of social media tools takes place behind the firewalls that surround most corporations, institutions, and organizations. Inside these enterprises employees share documents, post messages and engage in extensive discussions, document annotation, and create extensive patterns of connections with other employees and other resources. Networked communication has become an indispensable link to customers and partners and a critical internal nervous system required for every aspect of commerce. Social media tools cultivate the internal discussions that improve quality, lower costs, and enable the creation of customer and partner communities that offer new opportunities for coordination, marketing, advertising, and customer support.

As enterprises adopt tools like email, message boards, blogs, wikis, document sharing, and activity streams, they generate a number of social network data structures. These networks contain information that has significant business value by exposing participants in the business network who play critical and unique roles. Some employees act as bridges or brokers between otherwise separated segments of the company. Others have patterns of connection that indicate that they serve as sources of information for many others. Social network analysis of organizations offers a form of MRI or x-ray image of the organizational structure of the company. These images illuminate the ways the members of the organization are actually structured in contrast to the formal hierarchies of traditional "org-charts."

Technology consulting firms have recently started to highlight the value of analyzing patterns of connection within an organization. The Gartner Group reported that social network analysis would prove to be a strategic advantage for a corporation, calling it an "untapped information asset."[1] They recommend the analysis of "business intelligence on the ties, information flows and value exchanges" within a corporation. Network analysis can be focused, they argue, on three separate regions of commerce: organizational network analysis, value network analysis, and influence analysis, which map loosely to internal, vendor, and consumer populations. In each segment, network analysis is a useful method for identifying choke points and positions of leverage, locating expertise, and enhancing innovation.

[1]www.gartner.com/it/page.jsp?id=1239913

1.4 INDIVIDUAL CONTRIBUTIONS GENERATE PUBLIC WEALTH

Social media collective goods are a remarkable story of bottom-up individual initiative that leads to the creation of public value and wealth. Collections of individual social media contributions can create vast, often beneficial, yet complex social institutions. The intriguing challenge for the authors of this book and for a growing circle of social media analysts is to focus on individual behaviors while recognizing the emergent, collective properties of social media contributions. Seeing the social media forest, and not just the trees, branches, and leaves, requires tools that can assemble, organize, and present an integrated view of large volumes of records of interactions. Building a better view of the social media landscape of connection can lead to improved user interfaces and policies that increase individual contributions and their quality. It can lead to better management tools and strategies that help individuals, organizations, and governments to more effectively apply social media to their priorities.

Many utopian commentators have reported and proclaimed the benefits of social media. However, dangerous criminals, malicious vandals, promoters of racial hatred, and oppressive governments can also use social media tools to enable destructive activities. Critics of social media warn of the dangers of lost responsibility and respect for creative contributions, when vital resources are assembled from many small pieces [1]. These dangers heighten interest in understanding how social media phenomena can be studied, improved, and protected. Why do some groups of people succeed in using these tools while many others fail? Community managers and participants can learn to use social network maps of their social media spaces to cultivate their best features and limit negative outcomes. Social network measures and maps can be used to gain insights into collective activity and guide optimization of their productive capacity while limiting the destructive forces that plague most efforts at computer-mediated communications. People interested in cultivating these communities can measure and map social media activity in order to compare and contrast social media efforts to one another.

Around the world, community stakeholders, managers, leaders, and members have found that they can all benefit from learning how to apply social network analysis methods to study, track, and compare the dynamics of their communities and the influence of individual contributions. Business leaders and analysts can study enterprise social networks to improve the performance of organizations by identifying key contributors, locating gaps or disconnections across the organization, and discovering important documents and other digital objects. Marketing and service directors can use social media network analysis to guide the promotion of their products and services, track compliments and complaints, and respond to priority customer requests. Community managers can apply these techniques to public-facing systems that gather people around a common interest and ensure that socially productive relationships are established. Social media tools have become central to national priorities requiring government agency leaders to become skillful in building and managing their communities and connections. Governments at all levels must learn to optimize and sustain social media tools for public health information dissemination, disaster response, energy conservation, environmental protection, community safety, and more.

In this book we explore the social structure and organization of these social media phenomena through the application of the methods and concepts of social network analysis. Network analysis is a relatively recent scientific method for describing and analyzing a web of links among entities, including people. Network analysis provides powerful ways to summarize networks and identify key people or other objects that occupy strategic locations and positions within the matrix of links. Visualizations that map these structures complement numerical measures to provide intuitions and insights into the shape, size, density, sub-regions, and key locations within a connected population. Over decades, dedicated scientists, technologists, and entrepreneurs have dramatically improved the tools, analytic methods, visualization approaches, and sources of data for social network analysis. Increasingly potent software applications are available to study these phenomena and uncover useful, actionable insights. Until recently, these tools demanded significant programming and data management skills that excluded many interested users. We focus on a social network analysis tool designed for ease of use and customized for application to social media, the free and open Network Overview Discovery and Exploration add-in for Excel 2007 (NodeXL).

1.5 WHO SHOULD READ THIS BOOK

Practitioners, researchers, and students interested in the study of social media can benefit from this book. Using this volume, business leaders, instructors, and students can shift their attention from a focus on mastering social network tools and concepts to their application in measurement, analysis, and interpretation. Readers will learn how to extract insights from networks to reveal internal business activity, external customer communities, and the competitive landscape.

Professors and instructors in a range of disciplines may find this volume useful in semester-long courses as well as shorter units related to computing, business,

and social sciences. Technical classes in computer science/engineering, information science, human computer interaction, information visualization, and even social physics have been increasingly focused on the topic of "social computing." In business and management schools, enterprise collaboration and customer communities have become hot topics. In the social sciences, sociologists, anthropologists, historians, economists, demographers, political scientists, and other students of collective intelligence, collective action, and communities of practice have a historical opportunity to do data analysis at a scale never before possible. Digital humanities scholars and instructors have increasingly used social media to understand and enable novel connections between people and cultural artifacts.

1.6 APPLYING SOCIAL MEDIA TO NATIONAL PRIORITIES

Government agencies around the world are attracted to the possibilities of improved delivery of services at reduced costs but challenged by the loss of content control, liability for libel, pornography, or terrorist use. Open access to vast stores of government data expands their value while potentially calling attention to unfavorable information that could be used by political adversaries. Government professionals are excited by the prospects of increased citizen engagement but concerned by what that engagement may mean for their control over the flow of information and their obligations to protect privacy, avoid censorship while preventing libel, and other inappropriate uses of government information technology resources. Across the planet, developed and developing countries are attracted to the potential to use social media to change their societies, from promoting energy conservation and smoking cessation to new levels of political engagement with citizens.

Early efforts by innovative citizen/residents gave encouragement that existing social media platforms could be harnessed for national priorities, such as disaster response. In the aftermath of Hurricane Katrina in New Orleans in August 2005, volunteers created web sites to coordinate assistance, offer food, provide housing, and eventually help rebuild. When 32 Virginia Tech campus members were killed by a lone shooter on April 16, 2007, 1500 people self-organized and composed a Wikipedia page to inform outsiders what was happening. Each summer's California wildfires have stimulated innovative uses of Flickr to collect photos and Twitter for communication among firefighters and residents. But these and other uses could have been far more effective if designers had anticipated these needs and made modest design changes so that photos or tweets could be conveniently organized by groups, location, or time.

Disaster response is typically seen as a federal government responsibility, but other services such as community safety, are seen as a local government responsibility. Here too innovative individuals have created web sites and services to enable resident-to-resident assistance, such as the Nation of Neighbors project,[2] which describes its effort as follows:

> Real-Time Collaborative Neighborhood Watch and Reporting: Nation of Neighbors' web-based community networks enable neighbors to share local crime, suspicious activity and other community concerns in real time. Our mission is to enable citizens and law enforcement to work together to fight and deter crime and build strong communities. You can participate at the individual level or as a member of a Community Group or Agency.

Another successful community safety effort has been Amber Alert,[3] named after a child who was abducted and murdered in 1996. This alerting system, now coordinated by the U.S. Department of Justice, claims to have directly assisted in almost 500 safe returns of abducted children. It may also have raised awareness enough to have prevented many other abductions.

Resident reports are being solicited for tornadoes, earthquakes, floods, or other natural disasters as well as for reporting on fraud, abuse, and waste of government funds. Positive contributions such as fixes to the Library of Congress Card Catalog or the National Library of Medicine's PubMed service lead the way in suggesting further possibilities. The innovative Peer-to-Patent system (see Noveck's 2009 book on *WikiGovernment* [2]) invited specialists in certain technical areas to contribute information on prior art related to patent applications. Then a group discussion ranked the 10 items for submission to the patent examiners so as to speed up and improve their work at the U.S. Patent and Trademark Office. Noveck summarizes her case this way: "Ordinary citizens have more to offer than voting or talking. They can contribute their expertise and, in so doing, realize the opportunity now to be powerful… Collaborative governance is an idea whose time has come." [2, p. 190].

Volunteers to museums, parks, hospitals, or schools could also improve public services at national and local sites. Government-run web sites such as http://serve.gov and http://nationalservice.gov now facilitate these service efforts, while data at www.volunteeringinamerica.gov indicate that more than 38 percent of residents

[2] www.nationofneighbors.com

[3] www.amberalert.gov, America's Missing: Broadcast Emergency Response

volunteer in the top three states of Utah, Nebraska, and Minnesota, while the bottom ranking states of Florida, Nevada, and New York have volunteer rates of under 20 percent. Could increased visibility of volunteer efforts and greater public recognition increase participation in volunteer programs?

Web services can also be used to distribute tasks among volunteers who prefer to stay at home and contribute their technology and skills, as in the Search for Extraterrestrial Intelligence:

> SETI@home is a scientific experiment that uses Internet-connected computers in the Search for Extraterrestrial Intelligence (SETI). You can participate by running a free program that downloads and analyzes radio telescope data.
>
> *http://setiathome.berkeley.edu/sah_about.php*

The early success of the National Aeronautics and Space Administration (NASA) in getting "clickworkers" to help with the image analysis of Martian craters led to the recent "Be a Martian" effort to help map features and count craters.[4] Similarly the Smithsonian Institution provides a home for the ambitious Encyclopedia of Life[5] to achieve Edward O. Wilson's goal of a web page for each of the 1.8 million species on earth. The payoffs in scientific knowledge to support biodiversity and environmental preservation are potentially large, but the challenges of getting professional scientists, citizen scientists, and nature enthusiasts to work together are substantial. Designers and community managers have yet to find the motivational structures and recognition strategies to gain broader participation.

Another powerful use of social networks is to support rapid dissemination of public information (e.g., on flu vaccinations, weather alerts, or community safety threats). Many people are more likely to trust and act on cell phone calls or email messages from friends and family than from pronouncements by public officials on television programs or newspaper reports. Public officials could also disseminate less time-sensitive information on energy conservation strategies, environmental protection initiatives, or health alerts about obesity reduction or smoking cessation. Citizen-generated YouTube videos often have more impact than carefully scripted appeals by professionals at press conferences. Counts of retweets and video downloads can show how effective various strategies are in reaching different demographic segments.

Leaders of many nongovernmental community groups have come to appreciate the growing power of social media and ubiquitous cell phones with their increasingly rich services. Communities can become energized by modern technology-mediated versions of parent-teacher associations, neighborhood watches, and disaster planning teams. Even smaller groups such as book clubs, high-school orchestras, or local birdwatchers benefit from use of communications tools such as Twitter feeds, Facebook pages, or Google groups.

1.7 WORLDWIDE EFFORTS

Although this book emphasizes examples that we are more familiar with in the United States, there are worldwide efforts to apply social media strategies and encourage further research and development. The European-based Institute for Prospective Technological Studies produced two thoughtful reports on The Impact of Social Computing on the EU Information Society and Economy[6] and Public Services 2.0: The Impact of Social Computing on Public Services.[7]

The first report suggests that "social computing could play an increasingly important role in re-engaging citizens in political debate, in securing social cohesion and harmony, and it could provide a platform for dialogue on the grand challenges of the EU [European Union] and the rest of the world." The second report encourages "policy makers to seize the opportunities of social computing but also to mitigate any undesirable effects" and laments the "limited provision of citizen-centered public services by governments." The report further warns "that the empowerment and transparency characteristics of social computing initiatives seem to disrupt existing power balances."

There have been other efforts as well, such as that of the European Society of Socially Embedded Technologies (www.eusset.eu), which produced a manifesto seeking to shift the attention of researchers and policy makers from technology to social issues:

> Many people have been involved in the attempt to shift the focus of computing—and informatics more generally—away from a purely technical approach concerned with hardware and software only, to one that considers the human activities of design and use of information systems as being of central concern … how can we design in support of cultural diversity, social inclusion, as well as social and environmental sustainability; how can we design for new forms of engagement and participation?
>
> *http://www.eusset.eu/index.php?id=5*

We are also aware of strong efforts by the Chinese Academy of Sciences, which launched a social computing laboratory focusing on (1) testbed and simulation

[4] http://beamartian.jpl.nasa.gov
[5] www.eol.org
[6] http://ftp.jrc.es/EURdoc/JRC54327.pdf
[7] http://ftp.jrc.es/EURdoc/JRC54203.pdf

tool development; (2) foundations of social computing; and (3) social, cyber, and physical systems.

The international Web Science Trust (www.webscience.org) takes as a central tenet that the web is socially embedded, and among the key research questions is "How can we develop interdisciplinary epistemologies that will enable us to understand the web as a complex socio-technical phenomenon?" Its position papers [3,4] affirm the need to develop a strong scientific foundation that integrates sociotechnical systems thinking. Hendler et al. [5] point out how difficult this challenge is:

> The social model enabled by humans interacting in ways allowed by that technology is more difficult to explain … the success or failure of the sites hinges on the rules, policies, and user communities they support. Given that the success or failure of Web technologies often seems to rely on these social features, the ability to engineer successful applications requires a better understanding of the features and functions of the social aspects of the systems.

The Web Science effort is in harmony with the long-standing efforts to create a science of sociotechnical systems (STSs), which began at London's Tavistock Institute as a way of generating fresh thinking about workplace experiences. It has broadened to address the experience of individuals, teams, and communities as they apply technology in organizations and beyond. The integrative view of STS thinking addresses the whole task and clarifies responsibility for decisions in ways that increase motivation and enrich the experience for all participants. While some see the STS more narrowly as a management and work design topic, the term has come to reflect a balanced view of how technology design can be successfully attuned to human needs and values.

Many social scientists have taken an interest in sociotechnical systems design. Applied anthropologists use ethnographic methods for mobile phone companies to better understand how these devices are changing family, business, and social structures. Similarly, some sociologists and political scientists are studying how information and communications technologies are changing societies around the world.

The authors of this book see many opportunities for diverse academics and professionals to contribute to our understanding of how social media are already changing our world. We also see an active role for them in the design of future technologies, as well as future social, economic, and political systems. Technology promoters succeed more often when they address usability and sociability, are alert to human needs and values, and are sensitive to balancing policies and norms.

1.8 PRACTITIONER'S SUMMARY

Existing social institutions, educational curricula, business plans, and government policies are shifting as a result of social media tools and their application. Forward-looking universities are adding courses on the study of social computing, social informatics, new media, digital society, and other terms. Journal editors, conference organizers, and national science funding agencies are working to take advantage of the opportunities of using these new tools and techniques.

Individuals, organizations, and government agencies are devoting resources to using social media for their benefit while avoiding the dangers. Understanding how these social media networks thrive, change, or fail is a substantial challenge to researchers and professionals. Researchers in social network analysis have provided a set of concepts and metrics to systematically study these dynamic processes. Innovators in information visualization have also contributed to helping users to discover patterns, trends, clusters, gaps, and outliers, even in complex social networks. Each day solutions are being found that bring competitive advantages to business product developers, opportunities for government agency staffers, and new possibilities for nongovernmental social entrepreneurs.

1.9 RESEARCHER'S AGENDA

Now is an exciting time for those involved in the emerging discipline of social media network analysis. Researchers are designing novel collaborative technologies and social strategies that enable new forms of working and playing. They are also analyzing existing communities to find out what strategies and design decisions lead to success. Although social media success stories abound, there are countless examples of failed attempts to effectively apply social media to achieve desired goals. Social network analysis offers a systematic method to evaluate social media efforts, replacing anecdotes with scientifically based evidence. Unfortunately, many observers see no urgency in changing business strategies, marketing plans, research directions, curricula, or government programs. We hope this book will change their minds by showing the compelling business opportunities, attractive research challenges, strong educational needs, and important national priorities that social media can address and network analysis can elucidate.

Just as the physicists of the 1940s were challenged and troubled by the awesome forces they unleashed, researchers studying these social phenomena may yet create technologies that release human chain reactions, harnessing vast amounts of human energy to overcome the social problems that challenge our world. In the past 400 years, scientists have focused on fundamental physical phenomena, such as gravity, magnetism, nuclear forces, and genetic information. Their work

has produced profound changes in human life as we gain insights and control over core physical forces. Cellular communication networks, material sciences, and nuclear power are leading examples of the accomplishments of this vast intellectual endeavor. Similarly, biologists have revealed the core processes of all life, exposing the structure of DNA and opening the door to powerful techniques and practices that are only just unfolding. Because powerful technologies are eagerly sought by those who might put them to destructive purposes, urgent efforts are needed to ensure constructive outcomes. We believe that open discussion, broad participation, and respect for individual rights can help lead the way to more beneficial results.

References

[1] J. Lanier, You Are Not a Gadget: A Manifesto, Knopf Publishers, New York, 2010.
[2] B. Noveck, Wiki Government: How Technology Can Make Government Better, Democracy Stronger, and Citizens More Powerful, Brookings Institution Press, Washington, DC, 2009.
[3] T. Berners-Lee, W.T. Hall, J.W. Hendler, N. Shadbolt, D. Weitzner, Creating a science of the web, Science 313 (5788) (2006) 769–771.
[4] N. Shadbolt, T. Berners-Lee, Web science emerges, Scientific American (October 2008) 32–37.
[5] J.W. Hendler, N. Shadbolt, W.T. Hall, T. Berners-Lee, D. Weitzner, Web Science: An Interdisciplinary Approach to Understanding the World Wide Web, Communications of the ACM 51 (7) (2008).

Additional Resources

Benkler, Y. (2005). *The wealth of networks: How social production transforms markets and freedom*. New Haven, CT: Yale University Press.
Castells, M. (1996). *The rise of the network society*. Malden, MA: Blackwell.
Christakis, N., & Fowler, J. (2009). *Connected: The surprising power of our social networks and how they shape our lives*. New York: Little, Brown.
Cross, R.L., Parker, A., & Cross, R. (2004). *The hidden power of social networks: Understanding how work really gets done in organizations*. Boston, MA: Harvard Business Press.
Heath, C., & Luff, P. (2000). *Technology in action*. Cambridge, UK: Cambridge University.
Kleinberg, J. (2008). The convergence of social and technological networks. *Communications of the ACM*, *51*(11), 66–72.
Preece, J., & Shneiderman, B. (2009). The reader-to-leader framework: Motivating technology-mediated social participation. *AIS Transactions on Human-Computer Interaction*, *1*(1), 13–32. Available at: http://aisel.aisnet.org/thci/vol1/iss1/5.
Putnam, R.D. (2000). *Bowling alone: Collapse and revival of the American Community*. New York, NY: Simon and Schuster.
Shneiderman, B., & Plaisant, C. (2009). *Designing the User Interface: Strategies for effective Human-Computer Interaction* (5th ed.). Boston, MA: Addison-Wesley.
Surowiecki, J. (2004). *The wisdom of crowds*. New York, NY: Anchor Books.
Vieweg, S., Palen, L., Liu, S.B., Hughes, A.L., & Sutton, J. (2008). Collective intelligence in disaster: Examination of the phenomenon in the aftermath of the 2007 Virginia Tech shootings (pp. 44–54). Proc. 5th International ISCRAM Conference, May 5–7, Washington, DC.
Whitworth, B., & de Moor, A. (Eds.), (2009). *Handbook of research on socio-technical design and social networking*. Hershey, PA: IGI Global.

CHAPTER

2

Social Media
New Technologies of Collaboration

OUTLINE

2.1 Introduction 11

2.2 Social Media Defined 12

2.3 Social Media Design Framework 12
- *2.3.1 Size of Producer and Consumer Population* 13
- *2.3.2 Pace of Interaction* 14
- *2.3.3 Genre of Basic Elements* 15
- *2.3.4 Control of Basic Elements* 15
- *2.3.5 Types of Connections* 16
- *2.3.6 Retention of Content* 17

2.4 Social Media Examples 17
- *2.4.1 Asynchronous Threaded Conversation* 19
- *2.4.2 Synchronous Conversation* 20
- *2.4.3 World Wide Web* 21
- *2.4.4 Collaborative Authoring* 22
- *2.4.5 Blogs and Podcasts* 22
- *2.4.6 Social Sharing* 23
- *2.4.7 Social Networking Services* 24
- *2.4.8 Online Markets and Production* 25
- *2.4.9 Idea Generation* 26
- *2.4.10 Virtual Worlds* 26
- *2.4.11 Mobile-Based Services* 27

2.5 Practitioner's Summary 28

2.6 Researcher's Agenda 28

2.1 INTRODUCTION

One of the marvels of our time is the unprecedented development and use of technologies that support social interaction. Social mediating technologies have engendered radically new ways of working, playing, and creating meaning, leaving an indelible mark on nearly every domain imaginable. Billions of people now weave a complex collection of email, Twitter, mobile short text messages, shared photos, podcasts, audio and video streams, blogs, wikis, discussion groups, virtual reality game environments, and social networking sites like Facebook and MySpace to connect them to the world and the people they care about. Increasingly, people access these tools using mobile devices that can tie content to locations in real time. Behind organizational firewalls, a host of "Enterprise 2.0" social media tools echo the "Web 2.0" tools in the public Internet. The novel ways that people have adopted and adapted these technologies to their particular needs is a testament to human ingenuity and sociability.

Despite the growing ubiquity of social technologies, their potential has hardly been tapped. Effectively using and improving social technologies is far from trivial. Within them a complex interplay between social practices and technological infrastructures takes place. Architects will tell you that the physical design of a building or city can dramatically influence the ways in which people interact with one another. Teaching a course in a room with seats arranged in a circle versus seats arranged in rows facing the front invites a different form of participation from students. Although the physical layout does not wholly determine the forms of interaction, it does make certain interactions easier and others more challenging.

Similarly, the sociotechnical infrastructure, or platform, that underlies online activity influences social interaction. Admitting so is not technological determinism. Rather it is a solid materialism that recognizes that

technologies change the fabric of the material world, which in turn changes the social world. For example, microblogging sites like Twitter enable short exchanges ideal for efficiently pointing out resources or knowing what conferences people are attending, while discouraging in-depth discussion and analysis on the platform itself. In contrast, traditional blogs without length limitations and with their support for sharing multimedia content and comments are better suited for more in-depth presentations and conversations. Other media including books, newspapers, wikis, email, social networking sites, and so forth each have a set of properties that create a unique terrain of interaction. Learning to effectively meet your objectives using social media requires an understanding of that terrain and the social practices that have grown up around its use.

One of the most exciting aspects of online social media tools is that they produce an enormous amount of social data that can be used to better understand the people, organizations, and communities that inhabit them. More specifically, they create relational data: information about who knows or is friends with whom, who talks to whom, who hangs out in the same places, and who enjoys the same things. These relational data provide a wealth of new opportunities to understand and improve the social worlds we inhabit, as discussed throughout this book.

The purpose of this chapter is to introduce some of the important social media systems and provide a language and framework to talk about their key properties. It is also intended to informally introduce the relationship between social media systems and the networks they implicitly and explicitly create. This chapter begins with a definition of social media, followed by a framework for characterizing types of social media, and then takes you on a whirlwind tour of several important social media technologies that have emerged in the recent past.

2.2 SOCIAL MEDIA DEFINED

Social media refers to a set of online tools that supports social interaction between users. The term is often used to contrast with more traditional media such as television and books that deliver content to mass populations but do not facilitate the creation or sharing of content by users. Social media is about "transforming monologue (one-to-many) into dialog (many-to-many)."[1] In practice, it is a catchall phrase intended to describe the many novel online sociotechnical systems that have emerged in recent years, including services like email, discussion forums, blogs, microblogs, texting, chat, social networking sites, wikis, photo and video sharing sites, review sites, and multiplayer gaming communities. Related terms that describe many of these systems include Web 2.0, the read/write web, social computing, social software, collective action tools, sociotechnical systems, computer-mediated communication, groupware, computer supported cooperative work (CSCW), virtual or online communities, user-generated content, and consumer-generated media.

Pioneers of the information age such as Vannevar Bush who envisioned a hypertext-like device called the "memex" [1] and Douglas Engelbart who saw a future of graphical interfaces (i.e., windows), computer mice, and multiple-authored digital content [2] decades before it was realized, were interested in augmenting human intellect. In other words, they wanted to develop systems that "increase the capacity of man to approach a complex problem situation, to gain comprehension to suit his particular needs, and to derive solutions to problems" [2]. These goals have slowly been realized through remarkable developments in hypertext, human-computer interaction, the World Wide Web, and mobile technologies [3].

As the world has become increasingly connected, the focus has shifted to augmenting social experience and collective intelligence. Social media tools allow users to collaboratively create, find, share, evaluate, and make sense of the mass of information available online. They also allow users to connect, inform, inspire, and track other people. The new blend of social action and technological infrastructure allows entirely new ways of collaborating. Users can receive personalized recommendations based on the prior purchasing habits of thousands of other "similar" people, identify high-quality news stories based on real-time voting by the crowd, collaboratively author the world's largest and most-read encyclopedia, and instantly notify hundreds of followers about an online video presentation they found insightful.

2.3 SOCIAL MEDIA DESIGN FRAMEWORK

Social media systems come in a variety of forms and support numerous genres of interaction. Although they all connect individuals, they do so in dramatically different ways depending in part on the technical design choices that determine questions like these: Who can see what? Who can reply to whom? How long is content visible? What can link to what? Who can link to whom? As discussed in the introduction, these design choices can influence the social interactions that they enable and mediate. In addition, social practices, personalities, and history heavily influence how social media systems are used. If designers have learned anything

[1] www.webpronews.com/blogtalk/2007/06/29/the-definition-of-social-media

from successful social media systems like email and discussion forums, it is that they can be adapted to meet a surprisingly wide array of individual and community needs. Despite the adaptability of many social media systems, it is important to distinguish among systems as different as email, wikis, and massively multiplayer video games while recognizing their similarities.

One way to make sense of the bewildering proliferation of systems and services is to consider a set of key dimensions along which many social media services can be located. This approach provides a language and framework for comparing social media tools. This section considers six key dimensions:

- Size of producer and consumer population
- Pace of interaction
- Genre of basic elements
- Control of basic elements
- Types of connections
- Retention of content

These are not the only dimensions of possible interest, but they capture many of the important differences between social media tools. They also help lay the groundwork for the remainder of the book, which will use more formal methods to analyze the networks that are implicitly or explicitly created by the various social media platforms.

2.3.1 Size of Producer and Consumer Population

In most social media systems, producers and consumers are drawn from the same set of users. Users are producers one moment and consumers the next. However, differentiating between those who produce and consume content can be useful in comparing social media systems, even if the set of producers and consumers are not mutually exclusive.

Social media services vary in terms of their intended number of producers and consumers. An email is usually authored by just one person, whereas a wiki document is likely to be authored by several or even hundreds of people. An individually authored email might be sent to just one other person or be broadcasted to thousands. More generally, social media tools support different scales of production and consumption of digital objects. Table 2.1 provides some examples of social media systems, as well as some traditional media systems, and where various actions related to them fall within the producer and consumer size dimensions. You may notice that some systems show up in different places based on their usage scenario or the features that are being discussed.

Many social media tools help individuals or small groups interact. Instant messaging (IM), video chat, and

TABLE 2.1 Examples of Social Media and Pre-digital Media Systems Organized by the Size of Producer and Consumer Populations

Size of Consumer Population	**Size of Producer Population**		
	Small	Medium	Large
Small	Instant messaging Personal messaging (e.g., within Facebook) Video chat Phone call Face-to-face office meeting	Committee report to a decision maker Online survey Social networking friend feed Twitter homepage showing tweets of people you follow	Professional services reports for decision makers Personalized suggestions based on recommender systems
Medium	"Social" or family blog Profile page on community site or social network Departmental email list Tweet sent to followers Wall post on Facebook	Group blog on niche topic Internet relay chat room Internal department wiki Facebook group Niche YouTube channels Local markets (e.g., Craigslist)	Professional report for specialty group NASA clickworkers[1] Idea-generation sites (e.g., IdeaConnection[2])
Large	Popular blog, podcast, or webcast Message to large forum or email list Popular Twitter user's tweet Popular YouTube video Company web site Novel	News rating site (e.g., Digg) Wikipedia page Television program Popular discussion forum Online user-generated databases (e.g., IMDB) or marketplace (e.g., Threadless)	Large online marketplaces (e.g., eBay) Wikipedia encyclopedia YouTube video sharing Flickr photo sharing Popular massively multiplayer game

[1] *www.scienceofcollaboratories.org/resources/collab.php?317*
[2] *www.ideaconnection.com/crowdsourcing/procter-gamble-00007.html*

personal messaging within general-purpose social networking sites provide intimate communication channels comparable to phone calls and face-to-face office meetings. Social media can help individuals reach out to medium-sized groups of friends or acquaintances by broadcasting a personal message (e.g., a tweet sent to a user's followers on Twitter; a post sent to a departmental email list) or allowing others to overhear a comment (a post to someone's Facebook wall). They can also allow individuals to reach large groups through popular blog posts, podcasts, videos posted on sites like YouTube, or updates on Twitter by companies or celebrities with numerous followers.

Other social media tools help medium-sized groups reach out. Friend feeds on social networking sites and Twitter homepages allow a group (i.e., your friends or those you follow) to implicitly "create" a personalized stream of information just for you. Other tools like online surveys help aggregate information from many people for a small group of people analyzing the data. Group blogs or collections of related blogs about niche topics within the blogosphere allow a medium-sized number of bloggers and commenters to interact with one another. A host of tools facilitate interaction between medium-sized groups whether they are part of a Facebook group, YouTube channel, Internet Relay Chat (IRC) room, or combinations of tools as in a Ning social networking site on a niche topic. In some cases, a department or workgroup wiki can allow members or co-workers to co-author materials that are of interest to their group. Finally, some tools enable medium-sized producer groups to reach large consumer groups in a way similar to TV programs that take considerable effort to produce, but often reach the masses. Some of these include online databases such as the Internet Movie Database (IMDB), where user-generated movie content is shared with the world; news rating sites like Digg, where dozens to hundreds of people recommend a given news article that is consumed by much larger populations of read-only users; discussion forums, where posts by a few dozen active members may be seen by thousands of readers; and Wikipedia pages that are edited by dozens of people and read by thousands.

Some of the most interesting social media tools are those that help harness the power of the masses. For example, some recommender systems (e.g., Last.fm; MovieLens) provide personalized suggestions of books, movies, or songs by comparing your ratings with ratings' of other users. Other large groups help generate ideas that are used by medium-sized groups such as small businesses, corporate departments, or government agencies. For example, Procter & Gamble's IdeaConnection allows anyone to contribute ideas to solve their problems, whereas NASA's ClickWorkers help identify craters on Mars. Many of the most well-known social media sites allow large producer groups to interact with large consumer groups. Although there are many more Wikipedia readers than contributors, both groups are enormous. Online marketplaces like eBay allow the masses to sell and purchase goods. Meanwhile, social sharing sites like YouTube and Flickr make it easy for large numbers of producers and even larger numbers of consumers to interact. While these sites often facilitate smaller group interaction, they also aggregate those interactions so that people can search and navigate through large corpora of user-generated content. Massively multiplayer games rely on having large numbers to produce content and social experiences that make use of the entertaining environment.

2.3.2 Pace of Interaction

The pace at which interaction occurs is another important dimension along which social media systems can be organized. Traditionally, a distinction has been made between asynchronous and synchronous communication. Asynchronous systems like email, discussion forums, and voicemail presume a staccato pattern of interaction spread out over hours or days or weeks. Though less immediate, these systems have the advantage of allowing each participant to schedule their participation without much coordination with other people who may be in a wide range of time zones. They also encourage more careful contributions. In contrast, synchronous systems, like chat, instant messaging, videoconferencing, and graphical worlds, require that partners interact at the same time, as in face-to-face interactions and telephone calls. Although they require temporal coordination, they create a richer environment for interaction as participants quickly react and adjust to one another's signals in near real time. The pace of interaction has implications for the kinds of groups that form using each kind of tool. Global collaborations are often easier using asynchronous tools that don't require some people to change their sleeping habits. But some interactions need more rapid turn-taking to accomplish their goals.

More recently, the distinction has become blurred. For example, Twitter users often reply within minutes to another's tweet, but it is completely acceptable to reply a day later as well. Replying to a wall post or status update on a social networking site is similar in this regard. Tools like Google Buzz and Google Chat are now integrated with the widely used Gmail web email system, again blurring the distinction between synchronous and asynchronous modes of communication. However,

users' varied expectations about the pace of interaction within these tools remains important for understanding social media environments.

2.3.3 Genre of Basic Elements

Digital objects, the basic elements of social media systems, vary in size and type. Twitter posts (i.e., tweets) are limited to 140 characters, whereas email messages are typically a few lines to a few paragraphs in length. This difference in size produces dramatically different patterns of interaction. Although size limits of instant messaging are not typically enforced, design choices such as the size of the text box and messaging window promote brevity. Meanwhile, MediaWiki (the wiki platform used by Wikipedia) supports six levels of headers and automatically generates a table of contents, making it relatively easy to create large pages.

The various types of digital objects supported by social media tools are another way to understand the similarities and differences among them. Social media systems have often evolved around a distinct type of digital object: videos at YouTube, photos at Flickr, bookmarks (i.e., web site URLs) at Delicious, books at Amazon, music or podcasts at iTunes, TV shows at Hulu, people at Facebook, tweets at Twitter, messages at discussion forums or email lists, pages at Wikipedia, products at eBay, presentations at SlideShare, 3D objects in Second Life, and career professionals at LinkedIn. Each of these provides different levels and mechanisms of engagement. For example, virtual worlds more closely model embodied physical interactions, where avatars can convey meaning through proximity and orientation [4]. They also introduce many of the burdens of face-to-face interaction, demanding attention to successfully puppet the avatar in interactions with other partners [5]. Although these differences may relate to the type of media (e.g., video, audio, text, 3D model), there are further distinctions within each type. Wikis support structured text elements like tables and bullets, whereas email does not. Some virtual worlds like Webkinz use cartoon characters, while multiplayer games like World of Warcraft include realistic-looking creatures. Of course some social media systems like Facebook include many basic elements: profile pages, wall posts, personal messages, applications, instant messages, notes, groups, photos, tags, status updates, and so on. Wikipedia has user pages, talk pages, articles, edits, categories, and so forth. Even in these systems, identifying the basic elements of the system is important because they are the building blocks of the interactions. They are also the building blocks of networks when they are connected together or exchanged, as you will learn throughout this book.

2.3.4 Control of Basic Elements

Social media systems provide different levels of control over their basic elements. They can restrict who can create, edit, read, invite, respond to, subscribe to, and share content of various types. Some systems differentiate between anonymous users, registered users, and those with special privileges such as administrators. For example, some discussion forums require that users log in before they post, but they allow anyone to read the messages created by the community. This helps reduce spam by creating a higher barrier to entry, while still allowing anyone access to the content. It also allows users to exclude participants they define as social deviants. In other discussion communities of a more sensitive nature (e.g., patient support groups), access to content can be limited along with contribution until a person is registered, a process that may require some type of approval process by current administrators. Other systems like eBay require users to provide validated credit card information before they can sell items. The more open a community, the more potential there is for deviant behavior as evidenced by the frequent spam sent to wikis and Usenet. However, closing a community off too much may reduce the number of contributors, whereas openness may attract high-quality contributions that include combating the effects of spam and abuse. This is evidenced in the many high-value pages in Wikipedia, where poor edits left by non-registered users are quickly reverted by other registered and non-registered users.

Deciding the right types of barriers to entry can be an important part of online community building and one that deserves careful consideration [6]. For example, in studying real-world communities, Ostrom [7] found that successful communities had clearly defined boundaries, largely to overcome problems associated with outsiders taking advantage of internally produced or maintained resources. Boundaries are also important in that they encourage frequent, ongoing interaction among group members. This is critical because repeated interaction is perhaps the single most important factor in encouraging cooperation [8]. If individuals are not likely to interact in the future, there is a huge temptation to behave selfishly and free ride. On the other hand, knowing that one will be interacting with others on a continual basis can lead to the creation of reputations and serve as a powerful deterrent to short-run, selfish behavior. Boundaries can have an impact on the kinds of interactions people are willing to engage in because of the ways they shape the expected audience. Some media, like telephones and post offices, encourage the expectation that only a specific group of selected others will be the audience to your message. Other media, like public notices or messages on web discussion boards, are likely to be seen by any number of unknown people. Further, some media prevent the

identity of message creators to be known with certainty, if at all (e.g., pay phones or anonymous letters).

Control structures can heavily influence governance structures and the distributed or centralized nature of the environment. Although email lists, message boards, and Usenet newsgroups are all examples of asynchronous threaded conversation (see Section 2.4.1), their control structures are different. In centralized systems like email lists and many web boards, all communication flows through a single point, which is controllable by a single person (administrator) or small group who can wield dictatorial control over resources (messages) and access (who can subscribe). As a result, mail list owners can remove people for inappropriate conduct or prevent spam or other inappropriate messages from being sent out or stored in the archive. They often serve as "benevolent dictators." In contrast, distributed systems like Usenet are composed of hundreds of thousands of interlinked systems all interacting with a set of neighbors. The lack of a central point of control makes it impossible for a member to exclude others or remove content.

The granularity of control is another important factor. Users of a wiki can edit individual characters of a shared document, whereas other systems limit users to authoring entire messages that cannot then be edited. Twitter users can follow another user but then must receive all tweets from that user. In contrast, Delicious users can follow another user or can follow a user/tag combination so they only receive messages from the user that use a tag of interest. In many systems, users can only edit their own content, whereas in other systems, such as wikis and Google Documents, users can edit others' messages, documents, or objects. The level of granularity may differ for different user groups within the same system. For example, discussion boards on many web sites create a preset number of containers for interaction or leave control over the creation of new spaces in the hands of a small number of administrators. In contrast, email lists allow anyone to start new threads. Of course, systems in which users can create spaces with little restraint often contain many more of such spaces.

The pace of interaction can be crossed with the granularity of user control to characterize systems (see Table 2.2).

2.3.5 Types of Connections

There are many ways that the basic elements of social media systems can be connected. It is important to understand these connections or ties in order to construct and understand networks from each kind of social media system. We will describe the theory and language of networks in the next chapter. This section describes the many types of connections that exist in social media systems and explores the ways those collections of connections create a larger social system that we can now analyze with the math, tools, and insights of social network analysis.

The basic elements of many social media systems can be connected to one another explicitly or implicitly. Users intentionally and knowingly create explicit connections, whereas implicit connections are inferred from online behaviors. Perhaps the most common type of explicit social media connection is friending on social networking sites, where both people must approve the connection before it is realized. Other examples of explicit connections are following another user on Twitter, hyperlinking a wiki page to another page, tagging two photos or videos with the same tag, and adding someone to an IM buddy list.

Implicit connections can be inferred when a person sends another person an email message, "favorites" content (and by extension its author), replies to a discussion post, or "pokes," "waves," or "throws sheep" at another user as some sites allow. Although these actions were intentional, they were not performed with the explicit intention of creating a connection with the person. Other more subtle implicit connections can be identified, such as connecting people who "hang out" in the same discussion forums or Facebook groups or who edit the same wiki pages. These individuals may not know one another, but they are connected by their shared interests

TABLE 2.2 Examples of Social Media Categorized by the Pace of Interaction and the Granularity of Control over Content

Pace of Interaction	Granularity of Control		
	Fine: Users can directly control smallest units of content (characters, pixels, bytes)	Medium: Users control medium-sized blocks of content (objects, attributes, tracks, players) that they can only indirectly alter or that can be altered by other users	Coarse: Users control large block of content (documents, messages, blog posts, photos), rarely edited or modified by others
Synchronous	Real-time shared canvas	Virtual worlds, multiplayer games, real-time networked musical jamming	Chat, instant messaging, texting, Twitter
Asynchronous	Shared documents (e.g., Google Docs), source code, Wikipedia	Contribution to collected works like an album, anthology, report section, discussion group, or photosets	Email; blog posts and comments; sharing of links, photos, videos, and documents; turn-based games

and activities. Other connections can be inferred from data that are often not public but are available to the hosts or owners of social media systems such as reading patterns of discussion forums, music downloads, patterns of telephone calls, and location information. In innumerable ways, users now leave behind traces that form an intricate web connecting those users with the people, locations, and digital objects around them.

Another important distinction is between directed and undirected connections. If you and another person become friends on Facebook, the connection is a mutual one. In other words, it is undirected. Likewise, if you both are tagged as an "expert," then you are connected by an association that is mutual and thus undirected. In contrast, some systems like Twitter allow people to follow other users without first gaining those users' approval. This creates a different type of tie, where the directionality of the tie is important (i.e., who is following whom). Directed ties are also created when a person invites another person, favorites content, and creates a hyperlink from one page pointing to another page. In all of these cases, connections flow from one person or object to another and may not necessarily be reciprocated. Therefore, the relationships that are reciprocated are different and perhaps stronger than those that are not.

Finally, connections mean different things and can have different weights and values. For example, two people on Facebook can either be friends or not be friends; it is a binary connection that is either on or off. In contrast, two Facebook friends may send each other personal messages. The strength of their messaging connection could be measured based on the number of messages or the number of different days they each sent one another messages. This is an example of a weighted connection that varies in intensity. These weights often contain important information about the strength of a tie. For example, if Marc sent 10 messages to Ben last week and only 1 to Derek, it is probably safe to say that last week Marc was more strongly connected to Ben than to Derek (at least via that messaging medium).

The examples shared so far primarily connect people to each other, objects to each other, or people to objects. Recently, location has become an expanding part of social media services, allowing connections to be created between people, objects, and places. Smart mobile devices are opening a new era of social media that integrates information about location and activity in novel and powerful ways. New kinds of ties are being formed by just being in the same place as someone else, even at different times. Just having a laptop nearby someone else's can create new kinds of ties called "hyper-ties" that are like hyperlinks but applied to people.

2.3.6 Retention of Content

Social media systems also vary in how long content is retained. On one end of the spectrum are systems like wikis that typically create a permanent history of all actions that occurred in the system. Not only is each action recorded and stored, it is made available on article history pages and user contribution pages. At the other end of the spectrum, some instant messaging or voice-over Internet Protocol (IP) systems do not centrally record the interactions at all, allowing for fleeting exchanges more reminiscent of most face-to-face conversations. End points to these conversations can, of course, record them but extra effort must be taken. Many social media systems fall somewhere in the middle. For example, as of this writing, searches of the Twitter network only pull from the most recent 1.5 weeks. The horizon of the past in these systems is in flux as data volumes grow along with information processing capacity. The desire to add social media data to our long-term cultural memory has prompted interesting partnerships, as evidenced by the recent agreement between Twitter and the Library of Congress.

Some types of social media systems vary in their retention policies depending on the specific product or user settings. For example, some instant messaging clients do not archive conversations, whereas other clients retain them by default. Likewise, some email lists create a searchable archive of prior messages sent to the list, whereas others do not. However, it is important to realize that even if there is no centralized archive, individuals at the end points of these services may archive content and make it public at a later date. People can collect email messages, record Skype calls, log chat sessions, and collect most digital content fairly easily. We are now living in a world of easy data collection, retention, analysis, and publication suggesting prudence in using social media systems. Choose your words carefully; they may outlive you.

2.4 SOCIAL MEDIA EXAMPLES

This section provides a brief description of some popular types of social media as of this writing. Table 2.3 lists the social media systems and categorizes examples of each. It also serves as an index to this section. For a much more comprehensive list of social media tools, see Go2Web20[2] and Wikipedia. Because of the focus of this book, we highlight the types of networks that these social media tools create by discussing their basic elements and types of connections.

[2] www.go2web20.net

TABLE 2.3 Types of Social Media Listed with Example Services

Social Media Type	Examples
ASYNCHRONOUS THREADED CONVERSATION	
Email	Gmail, Hotmail, AIM Mail, Yahoo! Mail, MS Outlook
BBS, discussion forums, Usenet newsgroups, email lists	Slashdot, Google groups, Yahoo! Groups, Yahoo! Answers, Listserv
SYNCHRONOUS CONVERSATIONS	
Chat, instant messaging, texting	UNIX Talk, IRC, Yahoo! Messenger, MSN Messenger, AIM, Google Talk, ChaCha
Audio and videoconferencing	Skype, Gizmo, iChat, Window's Live
WORLD WIDE WEB	**TRADITIONAL WEB SITES, HOMEPAGES, AND DOCUMENTS**
Corporate, organizational, and government websites and documents	Ford.com, UMD.edu, Prevent.org, Serve.gov; Data.gov
Homepages	Faculty member websites, artists' portfolio websites, family history websites
COLLABORATIVE AUTHORING	
Wiki	Wikipedia, Wikia (Lostpedia), pbwiki, wetpaint
Shared documents	Google Docs, Zoho, Etherpad
BLOGS AND PODCASTS	
Blogs	LiveJournal, Blogger, WordPress
Microblogs and activity streams	Twitter, Yammer, Buzz, Activity Streams
Multimedia blogs and podcasts	Vlogs (video blogs such as Qik), photo blogs (Fotolog, FAILblog.org), moblog (mobile blogging such as moblog.net), podcasts (iTunes, NPR)
SOCIAL SHARING	
Video and TV	YouTube, Hulu, Netflix, Vimeo, Chatroulette
Photo and art	Flickr, Picasso, deviantART
Music	Last.Fm; imeem; Sonic Garden
Bookmarks, news, and books	Delicious, Digg, Reddit, StumbleUpon, Goodreads, LibraryThing, citeulike
SOCIAL NETWORKING SERVICES	
Social and dating	Facebook, MySpace, BlackPlanet, Tagged, eHarmony, Match
Professional	LinkedIn, Plaxo, XING
Niche networks	Ning (e.g., classroom 2.0), Ravelry, Grou.ps
ONLINE MARKETS AND PRODUCTION	
Financial transaction	eBay, Amazon, craigslist, Kiva
User-generated products	Instructables, Threadless, TopCoder, Sourceforge, Codeplex
Review sites	ePinions, Amazon, Angie's List, Yelp
IDEA GENERATION	
Idea generation, selection, and challenge sites	IdeaConnection, Chaordix, IdeaScale, Imaginatik
VIRTUAL WORLDS	
Virtual reality worlds	Second Life, Club Penguin, Webkinz, Habbo
Massively multiplayer games	World of Warcraft, Lord of the Rings Online, Aion
MOBILE-BASED SERVICES	
Location sharing, annotation, and games	Foursquare, Gowalla, Loopt, MapMyRun, Geocaching, Letterboxing, SCVNGR

2.4.1 Asynchronous Threaded Conversation

Asynchronous threaded conversations take on many forms such as email, email lists, Usenet newsgroups, discussion forums, and web boards. They have been the backbone of online communities since before the Internet and continue to play essential roles in a variety of settings both within organizations and on the public web. Even newer forms of social media such as social networking sites, social sharing sites, and virtual worlds often have asynchronous threaded conversations embedded within them to facilitate discussions. Although there are several flavors of threaded conversation, they share some key properties in that they are asynchronous, messages are sent with an address associated, typically, with a single author and authors can reply to messages and reply to others' replies creating conversation "threads." Authors do not typically edit one another's messages or even their own messages after (and some say before) it has been initially contributed. This basic structure has proven to be extremely versatile, supporting a wide range of individual, organizational, and community needs.

Threaded conversations, in all their forms, create implicit ties that connect senders and receivers of content in what is often called a "reply network" or "reply graph." These reply networks can be analyzed to identify important relationships, distinctive patterns of connections that reflect social roles, subgroups or clusters of people, interdepartmental connections, and many other important relationships. Next are brief descriptions of some of the most important asynchronous threaded conversation systems. See Chapters 8 and 9 for a more complete history and description.

Email

Email messaging was introduced in the late 1960s and the familiar "@" was introduced by Ray Tomlinson in 1971. Email lists were first introduced in 1972, the same year as PONG and the end of the moon landings. Today, email is almost universal with hundreds of millions of daily users exchanging tens of billions of messages a day. Because of the ubiquity of email, the implicit reply network created by its exchange is often an authentic representation of real-world connections. In Chapters 8 and 9 we discuss how to analyze three types of email collections: personal email collections (e.g., your own email archive), organizational email collections (e.g., your company's email traffic), and community collections (e.g., email lists).

Email Lists

Email lists turn email into a community experience by allowing people to send a message to a single email list address, which is then forwarded to everyone who has subscribed to the list.

This allows large groups of individuals to contribute and, more often, listen in on the conversations of others. These collective email discussions are widely used in enterprise discussion lists or Internet Listservs covering a nearly unlimited array of topics. Email lists facilitate discussions on a topic of interest, technical support, neighborhood gatherings and advocacy, workgroup interactions, internal communities of practice, and even the exchange of goods (e.g., FreeCycle). They are particularly good for reaching less tech savvy users such as older adults who are familiar with email but not more advanced technologies. They differ from web boards and forums in that they are a push technology that shows up in your inbox rather than requiring you to visit a site to get the latest information. Although it is hard to measure, they are likely among the most common forms of group interaction on the Internet, with some of them including thousands or tens of thousands of members.

Usenet Newsgroups

Usenet newsgroups were first created in 1979 at the University of North Carolina. The Usenet grew over decades to contain tens of thousands of newsgroups, each devoted to a variety of topics and containing chains of messages in reply to one another in structures called threads. Like a collection of email lists, Usenet newsgroups can foster the collective construction of discussion threads through the individual creation of messages in reply to one another. From politics to cats, from cooking to computers, Usenet organized billions of messages into millions of conversations sorted into tens of thousands of newsgroups. Usenet newsgroups are distinguished from email lists largely in terms of their comparative lack of centralized control and weak boundaries. Anyone can post a message to any newsgroup without regard to membership or the desires of others receiving the messages. Like email and email lists, newsgroups contain a core social network structure called a "reply graph" created when authors are connected with those they reply to in a thread. Although Usenet newsgroups are not the cutting-edge social platform they once were, they did help inspire innovations that continue to live on in other forms. For example, the Microsoft Research Netscan project demonstrated the value of visualizing social interaction on Usenet, a promise only now being realized to its full potential [9]. Usenet newsgroups also helped inspire one of the first collaborative filtering systems called GroupLens, which made personalized recommendations of content you are likely to enjoy based on the preferences of like-minded people [10].

From Bulletin Board Systems (BBS) to Discussion Forums

In the late 1970s,[3] dialup bulletin board systems (BBSs) hosted a wide range of message boards that

[3] www.bbsdocumentary.com

allowed people to interact in various ways. Using telephone connections and acoustic modems, thousands of individual BBSs run on early desktop personal computers exchanged digital data including early forms of networked threaded message discussions. Messages traveled more slowly through the BBS network than through current social media services: messages traveled from one computer to another over days not seconds. But like later generations of discussion technologies on the World Wide Web, these systems allowed for threaded conversations and some levels of centralized control over boundaries. BBS managers selected who could access their services and what content would be retained, exchanged, and copied from other systems.

Decades after dialup BBS systems, modern World Wide Web discussion board applications transplant many aspects of email, email lists, newsgroup, and discussion threads into a web-based environment. A web forum or discussion board is now a common feature of many sites that allow their users to create messages in multiple categories (called topics or threads) and link them to one another in chains of reply. Often a conversation thread can be found hanging beneath articles in newspaper and blog sites along with video services like YouTube, expanding further the scope of implicit conversational social networks. Now that message boards and other interaction media have integrated with the interface tools made available through the World Wide Web, conversation interfaces and databases have been refined into elaborate discussion systems like Slashdot.org, Plastic.com, and Fool.com where discussions about technology, popular culture, and financial markets, respectively, are held within a system that awards points and cleverly filters content. These sites provide web interfaces that offer new ways of organizing conversational content beyond the long-standing defaults of time and alphabetization. Through the use of backend databases that allow readership data and reviews of content to be tracked, these systems can dynamically filter content based on personal preferences to help overcome information overload.

2.4.2 Synchronous Conversation

Synchronous conversations such as text messaging, chat, instant messaging, and audio and videoconferencing differ from asynchronous conversations in that they occur in real time. Precursors to these Internet-based conversations occurred via telegraph, phone, two-way radios, and similar technologies. UNIX talk messaging, first used in the early 1970s, was among the first instantiations of text-based synchronous conversations based on computer networks. This simple system is the early precursor to chat and instant messaging, allowing two people to initially share a text stream, both users typing characters that would appear intermingled in the same space. Later innovations and refinements separated the text streams and evolved into the many forms of short messaging and texting services available today. Synchronous conversations today differ in that some such as instant messaging or videoconferencing support one-on-one, or small group conversations, whereas others such as chat enable more open conversations with anyone "tuned" to a specific "channel."

Chat

Chat was one of the most popular forms of interaction on the early Internet and accounted for up to a third of the revenue of the original commercial online providers such as America Online and CompuServe. In the late 1970s, "Citizen's band" (CB) radio in the United States was an analog technology that opened up new space for social interaction that was later mirrored by Chat services. CB radios, though restricted to a limited geographical space, organized discussions into a series of a few dozen "channels" located at frequency intervals allowing for open discussions and debates among people who might never have otherwise met. Most chat systems support a great number of "channels" dedicated to a vast array of subjects and interests. The IRC network of "Internet Relay Chat" servers remains a thriving and teeming space filled with chat from many people on numerous topics streaming nearly continuously around the world.

To the outsider, text chats can be confusing and chaotic places. But experienced users can accomplish a great deal of communication in these media. In contrast to Usenet, most text chat systems are organized around a centralized server that grants the server owner a great deal of power over access to the system and over individual channels. In commercial chat services, chat channels are frequently policed by the provider's staff or by appointed volunteers. In the largest noncommercial system, IRC, each channel has an owner who can eject people from the channel, control who enters the channel, and decide how many people can enter. Because of a lack of explicit links tying specific comments to one another, it can be hard for humans and computers to know who is talking to whom. This means that the reply networks created by chat may be error prone or probabilistic in nature.

Instant Messaging

Instant message text chat clients offer a private, often one-on-one, chat environment. Sometimes referred to as "buddy lists," these tools allow people to keep a list of their friends and contacts who also use the same or compatible tool. The messenger software indicates which if any of a person's "buddies" are available at that moment for possible conversation. Selecting a person on

your buddy list opens up a private window for exchanging short lines of text in real time. More recent tools like Skype merge the buddy list and text chat with full voice telephony, blurring the distinctions between these modes of communication further. Two primary networks connect users of these services. One network is a friendship network that connects users to those in their buddy list. Another network is a conversation network that connects people based on how often they talk with one another. Organizations that provide instant messaging services can use these networks to capture latent and active internal connections.

Texting

Text chat has a mobile form called Mobile Short Messaging Service (SMS), which has taken the world by storm, becoming the most widely used form of electronic communication. Billions of texts are exchanged each day among almost 3 billion users of mobile phones and other devices. Text messaging lacks many of the features of richer forms of message exchange. Where a desktop, laptop, or "smart phone" device routinely now handles audio, video, and document attachments, text messaging is limited to 160 characters of text. The simplicity and low bandwidth requirements make it almost universally available, relatively cheap, and accessible from mobile devices. Texting has become a basic entry point for Internet services and the means of weaving together family, work, and civic relationships. SMS generates communication networks connecting phones to each other based on the number of messages exchanged.

Audio Conferencing

More than even the most engaged chat session, audio and videoconferencing are highly synchronous forms of social interaction. People interact in near real time, speaking and replying and gesturing in a nearly continuous loop of symbolic exchange similar to face-to-face encounters. Audio conferencing using standard phones has grown steadily to become a widely used service for small teams, as well as for training or marketing sessions for hundreds of users. The simplicity of use, low cost, and emphasis on human voice has turned phone conferences into widely used and productive applications. In an audio conference, neither speakers nor listeners need to worry about their dress, facial expressions, or eye contact, and therefore can engage in other tasks simultaneously without offending others. Audio conferencing is now increasingly taking place through Voice over Internet Protocol (VoIP) connections with dramatic reductions in cost and expansion of use around the planet. Tools like Skype now make an audio conference among a dozen or more people in as many time zones as simple and a text chat.

Videoconferencing

Videophones have been the promised "vision of the future" since the 1940s but have failed to reach mass market production. High costs of early system hardware and connections were multiplied by the social awkwardness that video connections impose. Unlike audio only connections, video requires that people comb their hair, straighten their tie, or put on a dress and organize at least the area the camera can see. Video requires a continuous display of engagement, more like a face-to-face encounter, without the flexibility of a phone call that allows wandering attention and multitasking. Despite these hurdles, as hardware and connection costs have plummeted to near zero, the vision of widely used videoconferencing is now starting to be realized. Early Internet-based phone systems such as CUSeeMe from Cornell University developed an enthusiastic community of users starting in 1992. Inclusion of built-in video cameras and the iChat program on Apple computers and video services in Skype and other instant messaging clients have triggered a much wider community of users, including grandparents seeing distant grandchildren and distributed musical bands and project work teams. Corporate use of videoconferencing has increased steadily and has been integrated with screen and document sharing services using dedicated communication lines to ensure higher-quality sound and images and improved reliability. Video cameras on video game consoles and smart phones have further expanded the scope of videoconference users.

Social networks based on who converses with whom via audio or videoconferencing can be created relatively easily if the data are available. However, more fine-grained analysis of the conversations themselves (e.g., who replies to whom) is challenging because the data are difficult to capture automatically. Advances in automated speech to text conversion and video facial recognition may make it possible to efficiently extract these network exchanges in the years to come.

2.4.3 World Wide Web

The largest public, machine-readable network is the World Wide Web where web content, such as web pages and images (identified by their Uniform Resource Locator or URL), is connected together by hyperlinks. The World Wide Web, WWW, or just the web, was initially conceived by Tim Berners-Lee in the 1980s, but it was not realized until the 1990s. For the first time it married the concept of hypertext, developed in the 1960s by Ted Nelson (Xanadu) and Douglas Engelbart (oN-Line System [NLS]), with the Internet [11]. The result was a highly flexible platform that allowed people to view web content hosted on servers throughout

the world with the use of a web browser. Today the web is the primary platform upon which most of the social media tools are built. In this section, and in Chapter 12, we focus primarily on traditional web sites such as corporate, organizational, and government web sites, homepages, and documents (e.g., images and pdf files).

Although you may not realize it, network analysis plays a role in millions of people's everyday lives when they search the web via Google. As the amount of content on the web increased, the role of search engines became more essential for content on the web to be accessible. Early search engines looked only within the text of each web page (and its associated metadata fields) to determine its relevance. The first generation of search engines, like Alta Vista, built an index of all the words on millions of pages and matched them to search queries. Google made a breakthrough in the quality of its results by developing its PageRank algorithm, which determines how important a page on the web is based on its position in the web's network. At its core, the PageRank algorithm views a link to a page as a "vote" for that page's importance, so that pages with many incoming links score well. In addition, it considers the importance of each of the incoming links: receiving a link from a highly linked-to site counts more than receiving a link from an unknown site. This and related concepts are captured in various measures of the "centrality" of a point within a network as discussed in Chapter 5. A detailed analysis of several networks found within the World Wide Web is found in Chapter 12, which also introduces a "web crawler" tool that can generate networks of interconnected web pages.

2.4.4 Collaborative Authoring

Word processing is an innovation for those who grew up with typewriters. Although many believe creating and editing documents on a computer is a clear improvement over the use of typewriters, it often remains a solitary process. A further refinement of electronic writing is not just to improve the layout and fonts used but to change the way that groups of people collectively author and consume documents. Several social media tools facilitate the collaborative authoring of documents and repositories, enabling small groups and even communities of thousands to effectively create, maintain, and organize documents and document repositories.

Wikis

The most widely known example of collective document creation is Wikipedia, which is only one example of thousands of smaller wikis scattered throughout the web. Created by Ward Cunningham in the early 1990s, wikis are tools that allow a group of people, potentially any Internet user, to access and edit a shared collection of documents in the form of web pages. There are many "wiki engines" (i.e., platforms) including open source products like MediaWiki that are used by Wikipedia, as well as corporate products like Wetpaint, PBWiki, and SocialText. Wikis are used to create encyclopedias (e.g., Wikipedia), fan or game sites, corporate intranet content, and information repositories on topics ranging from educational resources to technical documentation to patient support information. Despite many differences among implementations, all wiki engines track each edit of each wiki page, creating a detailed page history. These edits can be reversed, creating a social version of the "undo" function of a word processor. People can follow updates to content on the RecentChanges page, get notified of changes to specific pages after subscribing to (i.e., "Watching") them, or view people's "user contribution pages" that chronicle all of their edits.

Wikis include many implicit networks. Some networks connect pages to other pages through hyperlinks or when pages are grouped together into categories. Other networks connect people to people when, for example, a person posts on another person's user page or when two people co-edit the same page. These networks can be mined to better understand content relationships or social roles as described in Chapter 15.

Shared Documents

The idea of collaboration through shared documents such as word processing and spreadsheet files is well established and goes beyond the community approach provided by wikis. Collaborative document creation continues to grow in popularity as corporations, governments, and community organizations discover that they can conveniently share and edit documents through Google Docs, Windows LiveSync, or DropBox. Users who edit the same documents form a network that may reveal patterns of cooperation, shared interest, or opportunities for new collaborations, although these networks are not yet explicit.

2.4.5 Blogs and Podcasts

Blogging is a special form of web page publication. Deceptively simple, a blog is essentially a low-cost mechanism for publication of rich digital content. Early blogs presented a series of text messages or "posts" in reverse chronological order so the most recent post was always at the top of the page. Today, a blog is a rich platform for content presentation and commentary. Blogs may contain text messages that are now complemented with pictures, video, and audio. Blogs present this content along with search, feeds such as Really Simple Syndication (RSS) for subscribing to the content, functionality for readers to comment on each blog

post, tags to categorize posts, pointers to related blogs (i.e., blogrolls), and a range of applications and widgets. Popular tools like LiveJournal, Blogger, and WordPress make blogging essentially free and widely available. They are also commonly embedded within larger sites.

Blogs and by extension bloggers are now often able to build audiences that rival predigital media and challenge many more established information providers, particularly in the news arena where current information is paramount. Blogs and bloggers are seen as potentially powerful makers and breakers of brands, political candidates, and news stories. They also serve as micropublishing fronts for families sharing stories and photos of their children and niche interest groups exploring an obscure topic. A single blog may be authored by an individual or a handpicked set of authors. Others participate via comments or by linking their own blog to other blogs, creating a monolithic collection of interconnected blogs often referred to as the blogosphere. Specialized search engines such as Technorati scour the blogosphere for recent blogs, using the unique properties of blog links (called trackbacks) and number of comments to assign credibility scores to blogs. Because the connections can be automatically captured using web crawlers, researchers have analyzed the blogosphere to better understand issues like the nature of political discourse in the highly bifurcated United States arena [12], as well as other areas such as Iran [13].

Microblogs and Activity Streams

The Twitter microblogging system burst into public view only a few years ago and has now become a worldwide phenomenon. It is similar to traditional blogs in its focus on recent posts, but differs in that its posts, called "tweets," are restricted to 140 characters of text. Twitter takes advantage of the idea of blog feeds by allowing you to subscribe to, that is, "follow" any other Twitter user. A user's personalized feed shows the most recent tweets of all individuals he or she is following, creating a live stream of bite-sized information nuggets. A number of competing services exist, some of which overlay the service on top of other services. For example, Facebook and LinkedIn have status messages that serve as microblogs that are broadcast to friends. More recently, Google Buzz and the open source identi.ca provide similar services.

Microblogging sites create several interesting social network structures. The most obvious network is the one created by the "follows" and "is followed by" relationships. Unlike Facebook, these "follow" relationships are potentially directed: you can follow people who don't follow you and visa versa. This is in contrast to the undirected ties present in Facebook and LinkedIn. Other networks are created that connect users together based on the number of times they reply to others' microblog posts or repost messages they come across (i.e., "retweet" or RT). A detailed analysis of several networks found within Twitter is found in Chapter 10.

Multimedia Blogs and Podcasts

As bandwidth and multimedia support has increased, a variety of services related to blogs have appeared including video blogs (vlogs), photo blogs, mobile blogs (moblogs), and audio blogs (typically called podcasts). These multimedia blogs may focus on a specific topic or may focus on the everyday experiences of a specific individual as a lifelogging/lifeblogging process. Typically, people can comment in text to the initial posts, and occasionally systems allow you to reply using multimedia as with vlogs that encourage video replies to other videos. Some multimedia sites, such as the humorous photo blog Failblog, encourage submissions of content from readers but are vetted by those in charge of the site before being posted. Other multimedia blogs are authored by an individual or small group, which are read by small to large sized groups. Moblogs are unique in that mobile devices such as smart phones are used to upload photos, videos, and text and often automatically tagged with location information (e.g., see moblog.net and Qik). Podcasts may include audio or video content and like traditional blogs can be subscribed to using tools like iTunes so that new content is automatically downloaded. They differ in that the facilities for commenting on podcasts are not as common, although they may be provided as part of a web site. Technological improvements will likely make search tools and annotation of multimedia content such as videos, images, and audio more common in the future.

The networks created by multimedia blogs and podcasts are similar to those created by blogs and microblogs. They connect content to content and, by extension, content authors. People can also connect to others who use similar tags to describe multimedia posts. In addition, moblogs can connect individuals based on shared locations.

2.4.6 Social Sharing

Social sharing sites are designed to allow individuals to share content, typically of a certain type (e.g., videos, photos, web sites). They provide an alternative to purposeful searching for content on search engines by allowing a community of peers to collaboratively identify and share interesting content. They are a modern incarnation of browsing, where the masses decide what is on the shelf. Although social sharing occurs in other social media tools such as social networking sites (e.g., photo sharing and tagging on Facebook), social sharing sites are typically more content-centric and stand-alone tools.

Video and TV

Since the widespread use of digital video cameras, people have been uploading their videos to share with others. Sites like YouTube and Vimeo allow the masses to freely upload and share video content and link to it or embed it within other web sites such as blogs. Corporations, universities, and media outlets post content on their own YouTube "channels" and occasionally participate in YouTube's partnership program. Sites like Google Video, Metacafe, Hulu, and Netflix streaming service allow users to search for and view TV shows, movies, and shorter video clips. These sites are home to a number of social network structures. For example, YouTube allows users to become "friends" or contacts with one another. In addition, relationships can be created between users when they comment on one another's videos, make a video a "favorite," or subscribe to a user's stream of uploaded videos. Networks of videos that relate to one another are also created based on having shared tags or shared viewers. A detailed analysis of several networks found within YouTube is found in Chapter 14.

Photo and Art

Since the invention of cameras, people have shared photos via albums, scrapbooks, and fridge magnets. Likewise, art is largely created so it can be shared. Social media tools enable the sharing of photos and digital copies of artwork with a select group of other people or the world at large. One of the most popular services is Yahoo's Flickr, which hosts a vast collection of digital photographs that are attached to individuals (e.g., photographers who upload them), groups, and tags that describe them. Like most social sharing sites, Flickr allows users to create networks of contacts (like friends in social networks sites) and limit the distribution of photos to just those individuals or to the world at large. Many similar services exist, such as Picasa, SmugMug, PhotoBucket, and DeviantART, and some sites like Slide that allow you to arrange photos in a slideshow with accompanying music. There are also a number of stock photo and vector art sites like stock.xchng and iStockphoto that allow individuals to purchase content one image at a time. Additionally, sites like Facebook include photo sharing and tagging elements.

The richly annotated content enables the construction of many types of networks. Some networks connect people who appear in photos together, whereas other networks connect people who follow others' art or are in the same group. Implicit networks can be generated that connect people who use similar tags, comment on others' photos, favorite others' photos, take pictures in similar locations, or use the same type of camera. A detailed analysis of several networks found within Flickr is found in Chapter 13.

Music

A number of social sharing sites revolve around music, including sites like Last.Fm, imeem, and Sonic Garden. These sites share many properties with video and photo sharing sites such as the ability to friend others, post comments, and navigate the site via various metadata fields such as tags and artist. One way that some sites, such as imeem, differ is their capacity to create explicit playlists, a modern incarnation of the mix tape, recognizing the value that comes from just the right collection of songs. Some sites like Last.Fm utilize collaborative filtering technologies to help recommend music that you are likely to enjoy based on music you currently enjoy. Collaborative filtering tools, a subset of recommender systems, use data from other users to create personalized recommendations. If you like songs that a certain group of users also likes, then you will probably also like songs they like that you have not yet heard. The networks created by music sharing sites are similar to those of other sharing sites.

Bookmarks, News, and Books

As users review content on the World Wide Web, it is common to want to save a web page or a pointer to it for later use. These collections of bookmarks can be valuable for other people as well. Several services have emerged to allow users to save a bookmark pointer to a web site and share that link with others. Services like deli.cio.us, digg, reddit, and stumbledupon provide a range of tools for users to collect pointers to useful and interesting material on the web, annotate it in various ways, and publish it to select others or the public. A related set of tools allows users of these sites to filter, search, and sort the accumulated links from many other users. Many users are eager to recall useful material on the web and are often willing to signal their interest or appreciation for certain web sites to others. Sites such as digg refined the idea of sharing bookmarks by adding a voting mechanism to the process to further sort quality content. These services have developed a strong following of users who signal their interest in web sites or blog postings, which are then aggregated and ranked for others to see. Bookmark sharing services provide a rapid way to identify novel and interesting information, build a historical trace, and form communities of shared interest. Similar sites exist for non-web-based content such as books (e.g., Goodreads, LibraryThing) and journal articles (citeulike, connotea), although they often include more extensive user-contributed reviews. These systems include similar networks to other content sharing sites, as well as networks based on voting behavior.

2.4.7 Social Networking Services

In 1971, Les Earnest wrote the "finger" program that allowed users of a system to check on another user's

status. When a user requested it through the finger program, a file named ".plan" would be displayed to other users.[4] This file soon became the business card and office door for many early users of the Internet and the networks that preceded it. Some users even updated the file regularly to note their current location, activity, and state of mind. This simple status and profile system evolved over time and inspired the creation of systems that allow people to present themselves to others. Modern incarnations, called social networking services, allow people to share contact information, text, images, and videos about themselves with their self-identified friends or followers. Earlier examples like Friendster established the basic outlines of the social network feature set in which users "friend" other users, enabling them to share content and receive updates about each others' activity.

Social and Dating

Services such as Facebook and MySpace have come to dominate social networking services, even though they are just a few years old. With 400+ million users in 2010, many of whom are regularly active, Facebook contains one of the largest machine readable "social graphs" on earth. There are many ways people connect to one another in Facebook, from the obvious "friending" that starts a Facebook relationship, to the many ways people can subsequently interact by writing on one another's "wall," indicating that they "like" other people's content, sending messages, tagging photos, and joining common fan clubs or groups. Facebook and related systems are rich sources of social network data as a result.

Other services have come to dominate specific geographical regions like the prominence of Orkut in Brazil and India and Bebo in the UK and StudioVZ in Germany. Some social sites have a focus on dating such as Tagged, eHarmony, and Match, or a specific demographic group like BlackPlanet. These often include detailed profile information not available on more general-purpose social networks so that members can find "compatible" companions.

Many of these social networking services, such as Facebook, impose restrictive terms of use for their data. In contrast to the mostly open and free model that surrounds Twitter, Facebook constrains what data can be accessed and the duration that the data may be used. As a result, analysis of Facebook social networks can be challenging. Individuals may extract some Facebook data related to their own interactions and social network, but even that data may only be used for short periods and for specific purposes. See Chapter 15 for a discussion of how to analyze your personal Facebook network.

Professional

Services like LinkedIn, Plaxo, and XING provide a social network feature set tuned to the self-presentation of career professionals engaged in business networking. Users can post their resume, receive and send targeted job invitations, recommend co-workers, introduce a colleague to another colleague, exchange private messages, and join groups such as university alumni associations or special interest groups. These networks are becoming a vital part of the job search process in many industries.

Niche Networks

More recently, sites like Ning and Grou.ps make it possible for users to easily create their own niche social networking site based around particular topics. For example, educators interested in using new technologies meet at the Classroom 2.0 Ning site, and people who cope with diabetes meet at TuDiabetes. These sites can be open to anyone to join or closed so that only invited members can join. In addition, a host of other online communities incorporate the core social networking features such as profile pages, while integrating them with community content. For example, Ravelry, a community of knitters, shows users current yarn stashes, favorite plans, and completed knitting projects. In Chapter 9 we explore Ravelry.

2.4.8 Online Markets and Production

Many social media sites facilitate the creation, evaluation, and exchange of goods and services.

Financial Transaction

Networks of exchange have always been at the core of marketplaces where buyers and sellers meet, exchange news, and make trades, purchase goods, or form plans for future activities. There has been enormous demand for online marketplaces in the form of auction sites such as eBay and Amazon or advertising sites like craigslist that facilitate location-specific ads for products, services, apartments, and jobs. These services generate communities of buyers and sellers who share an interest in the same products. Many small businesses and professionals, such as artists, craftspeople, or photographers, routinely advertise their products through personal or collective web sites, along with service providers, consultants, and personal trainers. These independent small businesspeople can reach a wide audience and develop credibility through reputation system tools like eBay's feedback mechanism. These marketplaces create networks that connect sellers and buyers through transactions, creating a trade network.

[4] www.rajivshah.com/Case_Studies/Finger/Finger.htm

Related services such as Kiva allow users to donate money to entrepreneurs in developing countries, facilitating microloans, and then follow their progress via blog posts and public repayment statistics. Finally, a growing number of services called "prediction markets" allow people to buy and sell assets whose cash value is tied to a future event (e.g., a Republican will be the next U.S. president). The market prices are interpreted as the probability of the event occurring. Examples include Intrade, Betfair, and more specific markets such as the Hollywood Stock Exchange. These financial transaction services create a wealth of network data on purchasing patterns that can be used to understand market dynamics, or as the basis for recommender systems like those found at Amazon.

User-Generated Products

A host of social media sites focus on collaboratively developing, sharing, or selling products. The open source software movement is an excellent example, where users contribute code to develop software tools that are then made freely available. Sites like Sourceforge and Codeplex provide tools to support developer communities by tracking changes to the software, monitoring the number of downloads, and providing basic discussion capabilities. Other sites like TopCoder allow users to compete against each other in programming tasks, where the winner gets paid by the organization or individual who posted the challenge. These sites create networks that connect people based on shared projects and challenges. Amazon's Mechanical Turk provides a platform for supporting a host of "human intelligence tasks" such as classifying items and identifying images. This "marketplace for work" allows people from around the world to perform these tasks for micropayments that can add up over time. Other communities focus on a specific product. For example, at Threadless, users can purchase T-shirts from a host of user-created designs. These sites generate trade networks, as well as networks that connect people who work on similar projects or use similar tags to describe their products.

Review Sites

Many social media sites allow people to post reviews of products or services. Some sites like ePinions and Amazon, support written reviews as well as ratings of almost every conceivable product. Local versions of review sites, such as Angie's List and Yelp, focus on location-based services such as restaurants, shopping, and nightlife, or service providers such as doctors, contractors, and service professionals. These sites create explicit networks when people friend each other (e.g., on Yelp), as well as implicit networks when users favorite places or services. They also create networks that connect people who provide similar ratings as others.

2.4.9 Idea Generation

Organizations are increasingly looking for ways to benefit from the collective intelligence of the masses. Several social media sites use "idea generation" tools to help solicit and evaluate new ideas. Companies like IdeaConnection allow organizations to post proprietary challenges to a community of problem solvers. If someone solves the problem, that person is awarded a specified dollar amount. If nobody solves the problem, no money is exchanged. Other tools by companies like Chaordix and IdeaScale allow users to post ideas and vote on others' ideas, helping the best ones bubble to the top. These services create networks that connect people based on who voted on whose ideas. They also create networks that connect ideas to other ideas based on the number of people who liked both ideas.

2.4.10 Virtual Worlds

Virtual worlds, graphical worlds, and massively multiplayer games attempt to model physical places as well as face-to-face interaction. Modern virtual worlds allow users to build new spaces, create objects, and use powerful programming languages to automate their behavior. These sophisticated forms of social media create remarkably rich collections of networks. Even services offering relatively simple game experiences like card games and backgammon offer sophisticated ways of creating friend networks, teams, and rankings. Game systems commonly allow users to create affiliation networks when players join clubs, guilds, tribes, or teams. Within the game play are other processes that create networks as records are created when users shoot or kill one another or trade less lethal materials.

Virtual Reality Worlds

Although many multiplayer games continue to focus on combat role playing, many "social" virtual worlds have become a means for widely dispersed groups to maintain personal contact. These include systems like Second Life and The Sims, designed for adults, and far more popular systems designed for children and youth such as Club Penguin, Webkinz, and Habbo. Virtual worlds typically offer a range of traditional communication channels, as well as the ability to manipulate "avatar" bodies that pose near one another. Like text chat, these systems support synchronous communication. Virtual worlds allow a number of people who occupy the same "room" to meet and "talk" by speaking, posing,

gesturing, and sending lines of text or shared spatialized audio conferencing with one another. Because this interaction happens in real time, all the participants must be active at the same time. But in return, virtual worlds provide a powerful sense of social and physical presence that is absent in asynchronous media. Virtual worlds often support simulations of the multichannel quality and nuances of face-to-face interaction by integrating lines of text with gesture, pose, and voice. For children and youth, they provide an engaging environment where users can earn virtual cash by playing games or completing virtual jobs and use their virtual cash to decorate their virtual home or feed their virtual pets.

Virtual worlds and the social data created in them are typically owned by the company that provides them. Thus, owners of virtual world servers have had monopoly control over their systems making it hard to access data for analysis purposes. However, when data are available, it is a rich source of network connections, which are created in virtual worlds whenever users exchange text or virtual items, are near one another, or interact with the same objects.

Massively Multiplayer Games

Massively multiplayer online games (MMOs) are video games that include hundreds or thousands of players who interact simultaneously in a persistent virtual world. They utilize the Internet, although they may be played on a computer or game console such as Xbox 360 and PlayStation 3. There are many types of MMOs including role-playing games (e.g., World of Warcraft, Everquest), strategy games (Mankind, War of Legends), first-person shooter games, racing games, and many more. They may take place in fantasy worlds or virtual worlds that correspond with specific locations such as World War II battlefields or cities. Many MMOs include complex social arrangements such as guilds, tribes, or teams; sophisticated collaboration tools including live audio-feeds; and virtual currencies that allow players to purchase items needed to complete quests or build their empires.

2.4.11 Mobile-Based Services

The next step from virtual reality is into physical reality. Recent waves of social media tools integrate hardware and software tools to enable users to annotate physical locations. Mobile computer devices like the Apple iPhone and Motorola Droid phones are now integrated with location detection features based on Global Positioning System (GPS) and cell tower location services. Combined with compasses, these devices can reliably locate and orient a user in space. Once the device has an accurate sense of its location, that information can be associated with all the digital objects created with the device. Photos can be easily linked to the place on earth where they were taken. Restaurant reviews can be associated with a map of the location of the restaurant. A comment can be linked to where it was made. In addition, mobile devices lead to the creation of more social media content because it can be created and captured in spare moments and when notable events unfold.

Location is key to social media services that want to provide contextual information about the world immediately around users. When a device is in a known location, simply asking for content associated with "here" can quickly deliver relevant material. Furthermore, a host of additional sensors can be used to provide functionality that can be called upon by new social media services. These include still and video cameras, audio and motion sensors, accelerometers that detect motion, and tools that enable device-to-device connections. This combined functionality and the proliferation of smart devices promises a bright future for mobile social media tools. Already we see companies making their services easy to access and digest using small screens and relatively slower network and computing resources. As more users access social media services from mobile devices, new opportunities will emerge. Although we are still in the early stages of mobile-based social media services, some interesting examples exist.

Location Sharing, Annotation, and Games

Services such as Google Latitude allow users to sense and plot their location on a map and then share that location with other authorized users. The position of other users who have selected to share their location can be seen on the map. Social media services like Gowalla, Loopt, and Foursquare allow users to share their current location with friends (often via other sites like Facebook), leave virtual messages or gifts for others, and write reviews of locations such as restaurants, bars, and parks. Foursquare even adds a game layer with score keeping based on patterns of location sharing. For example, the person who "checks in" from the same location more than anyone else is declared the "mayor" of that place. Mobile, social games like Geocaching and Letterboxing encourage people to hide "caches" that often include small awards or stamps that other players can find, often using GPS tools. SCVNGR provides a more general geogaming platform that lets anyone create location-based mobile games and tours that those with a mobile device can play. Finally, sites like MapMyRun allow users to create and share jogging trails on maps, as well as log training miles. Similar sites exist for bikers, hikers, and even golfers, some of which use data from GPS devices such as Garmin products that allow users to upload data to a computer about their heart rate, elevation, and speed.

These location sharing and annotation services often contain network structures similar to those found in other social media services, but with geographical location as an additional dimension. Place joins the set of other entities found in many social media services like people, tags, dates, and connections. This allows you to create networks that connect people to each other based on who is within a certain distance or who frequents the same locations.

2.5 PRACTITIONER'S SUMMARY

Social media tools have become ubiquitous, despite their recent development. Business success strategies that make good use of the power of social networks have been reported [14–16], whereas others that have shown positive effects when applied to government projects [17, 18] are evolving rapidly. However, for every successful example, there are numerous failed attempts suggesting the need for more systematic methods for analyzing and understanding social media environments.

There is no agreed-upon taxonomy of social media tools or characteristics. Yet it is essential that practitioners spend time understanding which services match their personal and professional needs. Corporate and government decision makers who are seeking to use social media for advertising and promoting their products and services will be more successful if they learn which mechanism best reaches their desired audience, and what constitutes acceptable etiquette in those communities. As a starting point, we discussed the following six key dimensions that help characterize social media systems:

- Size of producer and consumer population
- Pace of interaction
- Genre of basic elements
- Control of basic elements
- Types of connections
- Retention of content

In addition we briefly introduced some of the more popular social media systems, which are outlined in Table 2.3. In doing so, we highlighted the types of networks each of them create, laying the groundwork for the rest of the book, which will discuss how to gain actionable insights from the analysis and visualization of those networks.

2.6 RESEARCHER'S AGENDA

The widespread adoption of social media tools has begun to usher in a golden age of social science research. Social media systems provide a wealth of data about communication patterns, location information, friendships, and other social arrangements. Mining this data is bound to provide numerous insights into human nature. There are also many important questions that need answering to help us effectively utilize social media tools to achieve our goals. For example, we need to understand how to motivate voluntary participation [19], develop persuasive systems [20], govern social media communities [21], organize activities to meet specific goals, find the limits of scalability, and develop tools to better visualize and understand social activity. Related issues of trust, empathy, responsibility, and privacy have strong research foundations, which can be helpful to a wide range of practitioners. Addressing these issues will help designers and community managers make well-informed decisions rather than simply relying on intuition and anecdotes.

The rapid pace of commercial development offers new challenges to the research community to evaluate the impact of design changes, novel policies, and evolving norms. What forms of recognition or reward are appropriate for different domains? How can communities that involve participants with different expectations, skills, and experience be accommodated? How can malicious behavior be reduced? Can envisioned benefits to health, education, energy, or international development become a reality?

References

[1] V. Bush, "As We May Think," Atlantic Monthly, 1945.
[2] D. Engelbart, A conceptual framework for augmentation of man's intellect, in: P.W. Howerton, D.C. Weeks (Eds.), Vistas in Information Handling, vol. I, Spartan Books, Washington, 1963, pp. 1–29.
[3] B. Shneiderman, C. Plaisant, Designing the User Interface: Strategies for Effective Human-Computer Interaction: Fifth ed., Addison-Wesley Publ. Co., Reading, MA, 2010.
[4] E. Hall, The Silent Language, Doubleday Press, New York, 1990.
[5] M. Smith, S. Farnham, S. Drucker, The Social Life of Small Graphical Chat Spaces, Proceeding ACM CHI 2000 Conference, The Hague, Netherlands, March 2000, New York: ACM Press, 2000.
[6] D. Powazek, Chapter 8: Barriers to Entry … Making Them Work for It, In Design for Community, Waite Group Press, 2001.
[7] E. Ostrom, Governing the Commons: The Evolution of Institutions for Collective Action, Cambridge University Press, New York, 1990.
[8] R. Axelrod, The Evolution of Cooperation, Basic Books, New York, 1984.
[9] M. Smith, Tools for navigating large social cyberspaces, Commun. ACM 45 (4) (April 2002) 51–55.
[10] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, J. Riedl, GroupLens: An Open Architecture for Collaborative Filtering of Netnews, in: ACM Conference on Computer Supported Collaborative Work Conference, 10/1994, Chapel Hill, NC, 1994, pp. 175–186.
[11] T. Berners-Lee, Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web, Harper Paperbacks, 2000.

[12] L.A. Adamic, E. Adar, Friends and neighbors on the web, Soc. Networks 25 (3) (2003) 211–230.
[13] J. Kelly, B. Etling, Mapping Iran's Online Public: Politics and Culture in the Persian Blogosphere, Berkman Center Research Publication No. 2008-01, 2008, Available at: http://cyber.law.harvard.edu/publications/2008/Mapping_Irans_Online_Public.
[14] C. Li, J. Bernoff, Groundswell: Winning in a World Transformed by Social Technologies, Harvard Business Review, Boston, MA, 2008.
[15] J. Porter, Designing for the Social Web, New Riders, Berkeley, CA, 2008.
[16] D. Tapscott, A.D. Williams, Wikinomics: How Mass Collaboration Changes Everything, Portfolio, Penguin Books, London, UK, 2007.
[17] T. Kalil, Leveraging cyberspace, IEEE Commun. Mag. 34 (7) (1996) 82–86.
[18] B. Noveck, Wiki Government: How Technology Can Make Government Better, Democracy Stronger, and Citizens More Powerful, Brookings Institution Press, Washington, DC, 2009.
[19] K. Ling, G. Beenen, P. Ludford, X. Wang, K. Chang, X. Li, et al., Using social psychology to motivate contributions to online communities, J. of Computer-Mediated Commun. 10 (4) (2005) article 10.
[20] B.J. Fogg, Persuasive Technology: Using Computers to Change What We Think and Do, Morgan Kaufmann, San Francisco, CA, 2002.
[21] J. Preece, Online Communities: Designing Usability, Supporting Sociability, John Wiley & Sons, Chichester, UK, 2000.

Additional Resources

Easley, D., & Kleinberg, J. (2010). *Networks, crowds, and markets: Reasoning about a highly connected world*. New York: Cambridge University Press.
Rheingold, H. (2002). *Smart mobs: The next social revolution*. Cambridge, MA: Basic Books.
Shirky, C. (2008). *Here comes everybody: The power of organizing without organizations*. New York, NY: The Penguin Press.
Smith, M., & Kollock, P. (Eds.), (1999). *Communities in Cyberspace*. London, UK: Routeledge.
Wenger, E., White, N., & Smith, J.D. (2009). *Digital Habitats; stewarding technology for communities*. Portland, OR: CPsquare.

CHAPTER

3

Social Network Analysis

Measuring, Mapping, and Modeling Collections of Connections

OUTLINE

3.1 Introduction 31

3.2 The Network Perspective 32
- *3.2.1 A Simple Twitter Network Example* 33
- *3.2.2 Vertices* 34
- *3.2.3 Edges* 34
- *3.2.4 Network Data Representations* 34

3.3 Types of Networks 36
- *3.3.1 Full, Partial, and Egocentric Networks* 36
- *3.3.2 Unimodal, Multimodal, and Affiliation Networks* 36
- *3.3.3 Multiplex Networks* 37

3.4 The Network Analysis Research and Practitioner Landscape 37

3.5 Network Analysis Metrics 39
- *3.5.1 Aggregate Networks Metrics* 40
- *3.5.2 Vertex-Specific Networks Metrics* 40
- *3.5.3 Clustering and Community Detection Algorithms* 41
- *3.5.4 Structures, Network Motifs, and Social Roles* 41

3.6 Social Networks in the Era of Abundant Computation 44

3.7 The Era of Abundant Social Networks: From the Desktop to Your Pocket 46

3.8 Tools for Network Analysis 47

3.9 Node-Link Diagrams: Visually Mapping Social Networks 47

3.10 Common Network Analysis Questions Applied to Social Media 47

3.11 Practitioner's Summary 48

3.12 Researcher's Agenda 49

3.1 INTRODUCTION

Human beings have been part of social networks since our earliest days. We are born and live in a world of connections. People connect with others through social networks formed by kinship, language, trade, exchange, conflict, citation, and collaboration. Computer technologies used to create social networks are relatively new, but networks of social interactions and exchanges are primordial. Simply stated, a network is a collection of things and their relationships to one another. The "things" that are connected are called nodes, vertices, entities, and in some contexts people. The connections between the vertices are called edges, ties, and links. Many natural and artificial systems form networks, which exist in systems from the atomic level to the planetary level. *Social* networks are created whenever people interact, directly or indirectly, with other people,

institutions, and artifacts. Social network theory and analysis is a relatively recent set of ideas and methods largely developed over the past 80 years. It builds on and uses concepts from the mathematics of graph theory, which has a longer history. Using network analysis, you can visualize complex sets of relationships as maps (i.e., graphs or sociograms) of connected symbols and calculate precise measures of the size, shape, and density of the network as a whole and the positions of each element within it.

The recent proliferation of Internet social media applications and mobile devices has made social connections more visible than ever before (Chapter 2). The idea of networks, whether they are composed of friends, ideas, or web pages, is increasingly an important way to think about the modern world. Social network analysis helps you explore and visualize patterns found within collections of linked entities that include people. From the perspective of social network analysis, the treelike "org-chart" that commonly represents the hierarchical structure of an organization or enterprise is too simple and lacks important information about the cross connections that exist between and across departments and divisions. In contrast with the simplified tree structure of an org-chart, a social network view of an organization or population leads to the creation of visualizations that resemble maps of highway systems, airline routes, or rail networks (See Chapter 8). Social network maps can similarly guide journeys through social landscapes and tell a story about how some points are at the center or periphery of the network. Transportation networks where distance is measured in number of flights or roads from one city to another city are familiar. They inspire application to less familiar networks of electrical connections, protein expression, and webs of information, conversation, and human connection.

Social network analysis and metrics are described in several excellent books and journals [1–6]. This chapter touches on the key historical developments, ideas, and concepts in social network analysis and applies them to social media network examples. We have left details of advanced topics and mathematical definitions of various concepts to the many fine technical works. The following is intended as an introductory survey of the core network concepts and methods used in subsequent chapters, which focus on the networks that can be extracted from social media sources like Twitter, Facebook, email, discussion forums, YouTube, Flickr, wikis, and the web.

3.2 THE NETWORK PERSPECTIVE

Network analysts see the world as a collection of interconnected pieces. Those studying social networks see *relationships* as the building blocks of the social world, each set of relationships combining to create emergent patterns of connections among people, groups, and things. The focus of social network analysis is between, not within people. Whereas traditional social science research methods such as surveys focus on individuals and their attributes (e.g., gender, age, income), network scientists focus on the connections that bind individuals together, not exclusively on their internal qualities or abilities. This change in focus from attribute data to relational data dramatically affects how data are collected, represented, and analyzed. Social network analysis complements methods that focus more narrowly on individuals, adding a critical dimension that captures the connective tissue of societies and other complex interdependencies.

Network analysis shares some core ideas with the real estate profession. In contrast to approaches that look at internal attributes of each individual, network analysis shares the real estate focus on location, location, location! The interior of a house may be a liability, but where a property is located matters far more when trying to get a good sale price. The network perspective looks at a collection of ties among a population and creates measurements that describe the location of each person or entity within the structure of all relationships in the network. The position or location of a person or vertex in relation to all the others is a primary concern of social network analysis. Many network explanations look for causes of outcomes in the patterns of connections around an individual instead of their personal characteristics. "Know who" is often more important in network explanations than "know how." Network approaches observe that different people in similar circumstances and social positions often act in similar ways. Positions within networks may be as significant a factor as any aspect of the people who occupy them. Network analysis argues that explanations about the success or failures of organizations are often to be found in the structure of relationships that limit and provide opportunities for interaction [7].

Many network concepts are intuitive and echo familiar phrases like "friend of a friend," "word of mouth," and "six degrees of separation." Other network terms like "transitivity," "triadic closure," and "centrality" (see Section 3.5) may be unfamiliar terms for familiar social arrangements. Many of us recognize social network differences among people: we know some people who are "popular" and have connections to many others. We may also know some people who may be less "popular" but are still "influential," connecting to a smaller number of people who have "better" connections. Network analysis recognizes these and other less intuitively sensed patterns in social relationships, like measuring the number of your friends who know each

other and how much a person occupies a gatekeeper or bridge role between two groups. The network analysis approach makes the web of interconnections that bind people to one another visible, creating a mathematical and graphical language that can highlight important people, events, and subgroups.

3.2.1 A Simple Twitter Network Example

To better understand the network perspective, consider the social network of Twitter users shown in Figure 3.1 (see Chapters 2 and 10 for a description of Twitter). It is an example of a sociogram, also called a network graph, which is a common way of visualizing networks. Like all networks, it consists of two primary building blocks: vertices (also called nodes or agents) and edges (also called ties or connections). The vertices are represented by images of the Twitter users, and the edges are represented by the lines that point from one vertex to another.

This simple graph paints a picture of the social relationship of the Twitter users who tweeted about a 2009 workshop on information in networks at New York University[1] by including the text string "#WIN09." The size of each Twitter user's profile image is determined by the user's total number of tweets as reported by the Twitter Application Programmer Interface (API), which gives sophisticated users access to powerful services. This is one example of how attribute data (e.g., data that describe a person) can be overlaid onto a network. A line, or edge, exists between two people when one "follows" the other or if one user "mentions" or "replies" to the other. All of these connections in aggregate reveal the emergent structure of two distinct groups with few connecting links. This accurately represents the way the workshop brought together previously separate clusters of people from different disciplines. It also helps identify individuals who fill important positions in the network, such as those who many people follow and those who are connected to both clusters. This and following

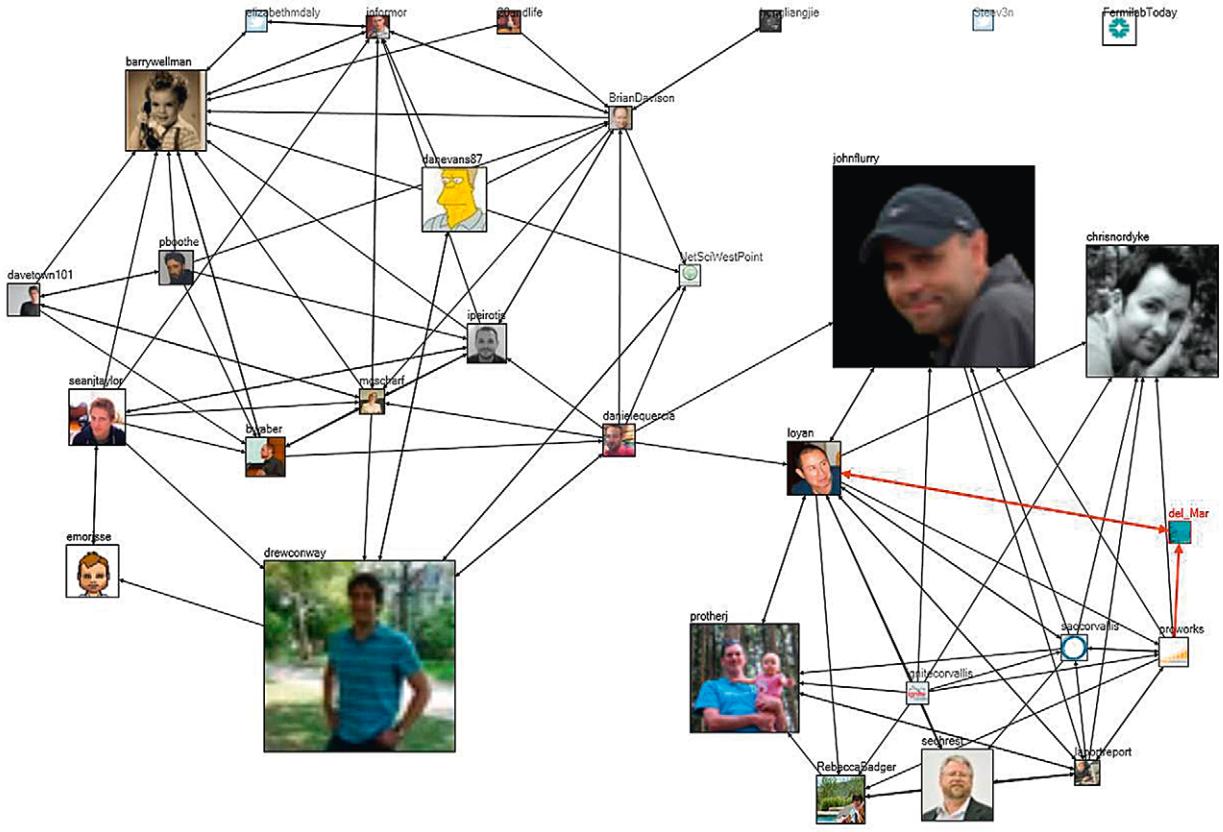


FIGURE 3.1 A NodeXL social media network diagram of relationships among Twitter users mentioning the hashtag "#WIN09" used by attendees of a conference on network science at New York University in September 2009. The size or each user's vertex is proportional to the number of tweets that user has ever made.

[1]http://winworkshop.net

chapters will provide a guide to creating maps like these from Twitter and other social media platforms and data sources. For now, let's consider the major components of a network in a bit more detail.

3.2.2 Vertices

Vertices, also called nodes, agents, entities, or items, can represent many things. Often they represent people or social structures such as workgroups, teams, organizations, institutions, states, or even countries. At other times they represent content such as web pages, keyword tags, or videos. They can even represent physical or virtual locations or events. They often correspond with the primary building blocks of social media platforms as described in Chapter 2: pages in wikis, friends in social networking sites, and posts or authors in blogs.

Although not necessary for network analysis, having attribute data that describe each of the vertices can add insights to the analysis and visualizations. For example, Figure 3.1 used descriptive attribute data about the total number of posts to convey a sense of who is most active on Twitter. Other attribute data from Twitter, such as the number of followers, people they follow, and their join date, can also be mapped to visual attributes (see Chapter 10). More generally, attribute data may describe demographic characteristics of a person (age, gender, race), data that describe the person's use of a system (number of logins, messages posted, edits made) or other characteristics such as income or location. In network visualization tools such as NodeXL, attribute data can be mapped to visual properties such as the size, color, or opacity of the vertices (see Chapter 4).

3.2.3 Edges

Edges, also known as links, ties, connections, and relationships, are the building blocks of networks. An edge connects two vertices together. Edges can represent many different types of relationships like proximity, collaborations, kinship, friendship, trade partnerships, citations, investments, hyperlinking, transactions, and shared attributes. A tie can be said to exist if it has some official status, is recognized by the participants, or is observed by exchange or interaction between them. A tie is any form of relationship or connection between two entities.

Network scientists have developed a language to describe different types of edges. In Section 2.3.5 of Chapter 2, we introduced the core types of connections that occur in social media networks. Here we describe how those concepts map to network and graph theory concepts more generally.

Undirected or *directed* edges are the two major types of connections. Directed edges (also known as asymmetric edges) have a clear origin and destination: money is lent from one person to another, a Twitter user follows another user, an email is sent to a recipient, or a web page links to another web page. They are represented on a graph as a line with an arrow pointing from the source vertex to the recipient vertex (see Figure 3.1). Directed edges may be reciprocated or not. If I sent you a message you may send one back in return, or not. An undirected edge (also known as a symmetric edge) simply exists between two people or things: a couple is married, two Facebook users are friends, or two people are members of the same organization. No origin or destination is clear in these mutual relationships. They cannot exist unless they are reciprocated. Undirected edges are represented on a graph as a line connecting two vertices with no arrows.

Edges can be represented by different types of data. The simplest type of edge, an *unweighted* edge or binary edge, only indicates if an edge exists or not. For example, a friendship tie between Facebook users either exists or it does not. In contrast, a *weighted* edge includes values associated with each edge that indicate the strength or frequency of a tie. For example, a weighted edge between two Facebook users may indicate the number of photo comments exchanged or the duration of a friendship. Weighted edges are often represented visually as thicker or darker lines or as more or less opaque lines. Including weighted data is preferable because they provide additional information about each tie. However, many social network analysis metrics (discussed later) are designed for unweighted networks. Fortunately, any weighted network can be converted to an unweighted one by choosing a cutoff point. For example, an unweighted edge could be shown between individuals who exchanged at least 10 email messages, with no edge between people who exchanged fewer than 10 messages.

3.2.4 Network Data Representations

Because network data differ from attribute data, there are different ways of representing it. With attribute data, it is common to create a data matrix where each row represents an individual and each column represents individuals' characteristics, behaviors, or answers to survey questions. A related approach can be used to represent relational data. Like attribute matrices, each row represents an individual in the network. However, unlike attribute matrices, each column also represents an individual as shown in Table 3.1.

Different types of edges can be represented in network matrices. Table 3.1 describes a directed network because not all connections are reciprocated. For example, Ann "points to" Bob as shown in row 1, but Bob

ADVANCED TOPIC
The Foundations of Graph Theory

Network analysis is rooted in the work of the mathematician Leonhard Euler who in 1736 studied whether a single path could be walked over the Seven Bridges of Königsberg that connected islands in the river Pregel (which flows through what was then Prussia and is now Kaliningrad in Russia) without crossing any bridge more than once. By reimagining the problem in terms of vertices and edges, he showed it is impossible to cross each bridge just once. Although the problem seems abstract, its solution led to the development of the mathematics of graph theory and, notably, hundreds of years later, the mathematical work of Paul Erdös and Alfréd Rényi on random graphs in the 1950s, an important theoretical development that allows for the generation of a graph from random processes. Social network analysis builds on these concepts and extends them to capture the nonrandom connections that occur among groups of people.

TABLE 3.1* A Network Represented as a Matrix

	Ann	Bob	Carol
Ann	0	1	1
Bob	0	0	0
Carol	1	0	0

**This network is a directed network, as it is not symmetrical (i.e., Ann points to Bob in row 1, but Bob doesn't point to Ann in row 2). It is a simple binary network: either a tie exists (value = 1) or not (value = 0).*

TABLE 3.2* A Network Represented as an Edge List

Vertex1	Vertex2
Ann	Bob
Ann	Carol
Carol	Ann

**Individuals in the Vertex1 column "point to" those in the Vertex2 column in this directed network. The network is implied to be a binary network. Additional columns could be used to describe each edge. For example, an Edge Weight column could be added with values representing the strength of various ties.*

does not "point to" Ann as shown in row 2. If it were an undirected network it would be a symmetric matrix; if Ann points to Bob then Bob must necessarily point to Ann. This network is a binary network because it only includes 1s and 0s, where a 1 indicates that there is a connection and a 0 indicates that there is no connection. Allowing additional values would create a weighted network. For example, the 1s could be replaced with the number of email messages sent to the other person. Finally, notice that the diagonal of the matrix connects each person with himself or herself. In this network, like most networks, the diagonal values are 0 indicating that a person does not "point to" herself. However, in some networks a "self-loop" connecting a person to herself can exist. For example, a person may send herself an email message as a reminder. Network matrices are powerful forms of representation that lend themselves to efficient mathematical manipulation for those inclined. However, they can also become quite large and challenging to navigate, particularly when networks have relatively few connections.

An alternate network representation is called an "edge list." Like its name suggests, it is simply a list of all edges in the network as shown in Table 3.2. This is the same network as shown in Table 3.1. Individuals in the Vertex1 column "point to" those in the Vertex2 column. Unless data describing the value of each edge are provided in additional columns, the network is implied to be a binary one. Self-loops are possible to represent in edge lists by having a row with the person's name repeated in both columns. Throughout this book, we will use edge lists instead of matrices.

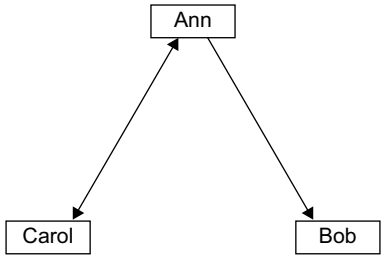


FIGURE 3.2 The directed, binary network described in Tables 3.1 and 3.2 represented as a network graph. Arrows indicate the direction of the connection (e.g., from Ann to Bob).

The final method for representing networks is through network graphs. Figure 3.2 is a network graph based on the data in Table 3.2. It makes immediately clear that the relationship between Ann and Carol is reciprocated (i.e., there are arrows on both sides of the line connecting them) and that there is no connection between Bob and Carol. Our earlier analysis of Figure 3.1, another network graph, demonstrates how network graphs can lead to insights that are hard to identify in tabular data, particularly when large networks are presented. However, many network graphs require significant preparation to assure that they are readable as described in Section 3.9.

3.3 TYPES OF NETWORKS

Social networks range in size from a handful of people to national and planetary populations. They also differ in the types of vertices they include, the nature of the edges that connect them, and the ways in which they are formed. In this section we introduce some of the distinctions that network scientists have identified to describe different types of networks. These distinctions affect the metrics and maps generated from them, as well as their interpretation.

3.3.1 Full, Partial, and Egocentric Networks

It is often useful to consider social networks from an individual member's point of view. Network analysts call the individual that is the focus of attention "ego" and the people he or she is connected to "alters." Some networks, called egocentric networks, only include individuals who are connected to a specified ego. For example, a network of your personal Facebook friends would be an egocentric network because you are, by definition, connected to all other vertices. Other egocentric networks and their associated "subgraphs" (see Chapter 6) may extend out from an ego, reaching not only friends, but also friends of friends. More generally, egocentric networks can extend out any number of "degrees" from ego. The basic "1-degree" ego network consists of the ego and their alters. The "1.5-degree" ego network extends the 1-degree network by including connections between all of the alters. For example, a Facebook 1.5 degree ego network would characterize which of your friends know each other (see Chapter 11). The "2-degree" ego network extends the 1.5-degree network by including all of the alters' own alters (i.e., friends of friends), some of whom may not be connected to ego. These three ego networks allow you to look at increasingly larger, but still "local" neighborhoods around a particular individual in a social network. Higher-degree networks (e.g., 2.5, 3) are feasible to create but not used as often in practice because they can quickly become intractable.

A full or complete network contains all the people or entities of interest and the connections among them. All egos are treated equally. A full network is often created and available when a single system, such as a social media platform, acts as a hub among a group of connected people or groups. For example, the Twitter network includes all users of the service and the connections between them. In practice, it is not always feasible (or particularly insightful) to analyze a full network. Instead, analysts create a partial network by selecting a sample or slice of the full network. For example, Figure 3.1 showed the slice of the Twitter network that included people who used the hashtag "#WIN09." This partial network was not egocentric. Rather it was topic centric. Other partial networks may be created to include a subgroup of users (e.g., all conference attendees), only people and connections that occurred within a specified time frame, or people who have certain characteristics (e.g., CEOs of Fortune 500 companies).

3.3.2 Unimodal, Multimodal, and Affiliation Networks

Up until this point we have only considered networks that connect the same type of entity. These standard networks are called *unimodal networks* because they include one type (i.e., mode) of vertex. They connect users to users or they connect documents to documents, but they don't include both users and documents. However, networks can include different types of vertices creating *multimodal networks*. For example, a network may connect users to discussion forums and blog posts they have commended on. Each vertex on the graph would represent a user, a forum, or a blog post, which could be visually distinguished by different colors or shapes. The SeriousEats Network discussed in Chapter 6 is an example. The rich sets of intersecting networks that form in social media environments include connections between people, photos, videos, messages, documents, groups, organizations, locations, and services. In many cases, these multimodal networks have to be transformed into simpler unimode networks to perform meaningful network analysis, as most network metrics (see Section 3.5) are designed for unimodal networks.

A common type of multimodal network is a *bimodal network* with exactly two types of vertices. Data for these networks often include individuals and some event, activity, or content with which they are affiliated, creating an *affiliation network*. For example, an affiliation network may connect users with wiki pages they edit. People are affiliated with pages. In this network, no two users would directly connect to each other. Likewise, no two wiki pages would directly connect to each other in this type of network.

Bimodal affiliation networks can be transformed into two separate unimodal networks: a user-to-user network and an affiliation-to-affiliation network (e.g., article–to–article network in a wiki) (see Chapter 6, Advanced Topic, Transforming Multimodal Affiliation Networks into Unimodal Networks for details). The user-to-user network connects people based on their links to one another. For example, in a wiki co-edit affiliation network Derek and Marc would be strongly connected because they both edit many of the same wiki

pages. The affiliation-to-affiliation network connects the affiliations based on the number of shared users. For example, a pair of wiki pages would be closely connected if many people edited both of the pages (see Chapter 15). More generally, this approach can be used to relate objects of all types (e.g., books, photos, and audio recordings) based on users' behaviors (e.g., purchasing or reading habits) and preferences (e.g., ratings). Affiliation networks are the raw material of many recommender systems that recommend items of interest, such as Amazon's "Customers Who Bought This Item Also Bought" feature. A network data structure can return results to queries like "people who linked to this document also linked to these documents" or "if you link to this document, you may want to link to these people."

3.3.3 Multiplex Networks

Although it is common for two people to be connected in many different ways (for example, by exchanging phone calls, emails, sharing group membership, and being married), most networks only include one type of connection or edge. However, it is possible to consider networks with multiple types of connections, called *multiplex networks*. For example, the Twitter network shown in Figure 3.1 includes three types of directed edges: following relationships, "reply to" relationships, and "mention" relationships. The graph could have uniquely represented each type of edge by using color, different edge types (e.g., dotted lines, solid lines), or edge labels (see Chapter 4). In the case of Figure 3.1, the type of edge was not deemed important, so the multiplex network data were condensed into a standard network that showed a single directed edge if one or more of the three types of connections were present. This strategy of combining multiple types of edges is a common one that allows for the use of network metrics, which are mostly based on standard networks.

3.4 THE NETWORK ANALYSIS RESEARCH AND PRACTITIONER LANDSCAPE

You can find network scientists in nearly every academic discipline and an increasing number of practitioner communities. Network concepts and techniques are now widely found throughout a range of disciplines including sociology, anthropology, communications, computer science, education, economics, physics, management, information science, medicine, political science, public health, psychology, biology, and the humanities. In the past several decades, social scientists have shown that network structures have an influence on health, work, and community. Getting a job, being promoted, catching an illness, adopting an innovation, and many more activities and processes have been explained in terms of social networks. Network structures are important in the biological sciences where research is focused on connections between metabolic and genetic processes. The shape and function of networks can have great consequences as ideas, genes, innovations, or pathogens diffuse through populations. Researchers now apply network theory and methods to understanding how Supreme Court decisions relate to previous cases, how the United States Senate votes (see Chapter 7), how epidemics spread within cities, and how characters in a novel relate to one another (see Chapter 7). Networks are formed from many physical processes and are echoed in a number of structures created inside information systems such as the collection of linked documents within the World Wide Web or an enterprise's collections of files. Information scientists use these links to identify high-quality web pages (e.g., Google's PageRank algorithm), or use the citations from research articles to identify high-impact articles and authors.

Network methods are diffusing beyond academic research, becoming an important tool for managing organizations, markets, and movements. Entrepreneurs apply network analysis techniques to understand how to leverage the powerful effects of word-of-mouth marketing as their customers spread news about their new products to one another. Many politicians recognize the potential power of a connected network of supporters who can be turned into contributors, volunteers, and voters. Engineers use network analysis to build more effective power grids, computer networks, and transportation systems. Law enforcement officers and lawyers analyze email networks to identify and defend potential criminals. And the intelligence community hunts down terrorists by looking at networks created by money trails and kinship. Having at least a basic understanding of network thinking and concepts is a core literacy of our time. Like statistics, network analysis has countless applications to a number of fields.

This book primarily focuses on social network analysis, a subfield of network sciences that focuses on networks that connect people or social units (i.e., organizations, teams) to one another (see Advanced Topic: Early Social Network Analysis). We are also interested in networks that connect human-generated content or artifacts together, such as web sites or cell phones.

ADVANCED TOPIC
Early Social Network Analysis

The social science roots of social network analysis can be found in the early 1800s in the work of the person credited with being the first sociologist, Auguste Comte, and later in the early 1900s in the work of the sociologist Georg Simmel. Both saw patterns of social ties as the main focus of sociology in contrast to the study of individuals and their attributes. Early in the nineteenth century, Comte defined society as more than simply a group of people. He argued that a population became a society only when people had influence on one another and considered the choices and interests of others as part of their own choices. Simmel echoed these ideas at the turn of the twentieth century, focusing social science on the study of how people come together and form groups and associations. These sociologists imagined society as composed of a web of relationships—more than a mass of individuals; they saw societies as networks of reciprocal influence.

The idea of connected actions linking people to one another has remained at the core of the social sciences, but efforts to create a systematic language to record social relationships started only in the twentieth century. Anthropologists studying the range of kinship systems they documented in fieldwork from around the world created symbol systems that are related to social network analysis. Their maps of who is related to whom were early forms of social networks focused on just the subset of social ties that are considered to be "family." The core concepts and methods of modern social network analysis date from the 1930s and the pioneering work of Jacob Moreno and his many collaborators. Researchers at New York University, Columbia, and Harvard created the first scholarly works featuring the distinctive core components of modern social network theory: measures, maps, and models. Moreno and his research partners created the first pictures of patterns of people and their partners, using visual maps with symbols that represented individuals with different types of lines connecting them to others that represented different kinds of relationships.

Moreno documented relationships among schoolchildren and the way an innovative behavior, running away, moved through chains of student connections. In 1934, Moreno [8] published "Who shall survive," which catalyzed work among a group of scholars who refined his approach and added critical mathematical elements that today are a standard part of network analysis. These approaches were applied to various settings, and revealed the key roles a few people played in their networks and often the presence of subgroups of distinct people. For example, in the 1930s, Davis et al. collected detailed records of observed attendance at 14 social events by 18 southern women, and the graph of that data revealed two distinct groups with minimal overlap [9]. Moreno, who developed sociometry and is often considered the founder of the sociogram, studied relationships among members of a football team and found patterns of friendship and animosity (see Fig 3.3) (as produced in Freeman [10]).

At Harvard in the 1930s, a group formed around W. Lloyd Warner and Elton Mayo to explore interpersonal relationship in workplaces. Early social network analysis work focused on connections in small work groups in industrial factory settings. For example, Roethlisberger and Dickson [11] studied the Western Electric Wiring room, documenting the ways individuals within a group worked with one another. As seen in Figure 3.4, some workers in the study emerged as the most connected, whereas others appeared as peripheral or isolated. Another data set was created that represented the relationships among 14 employees of the Western Electric Hawthorne Plant. Employees and two inspectors were observed, and each contact among them was coded. When employees played games with one another, argued, were openly friendly, confrontational, or helpful a note and tie was recorded. The result were six networks, which led to a seminal work by the Harvard sociologist George Homans [12] and later more mathematical work that focused on automatically finding clusters or groups within these data

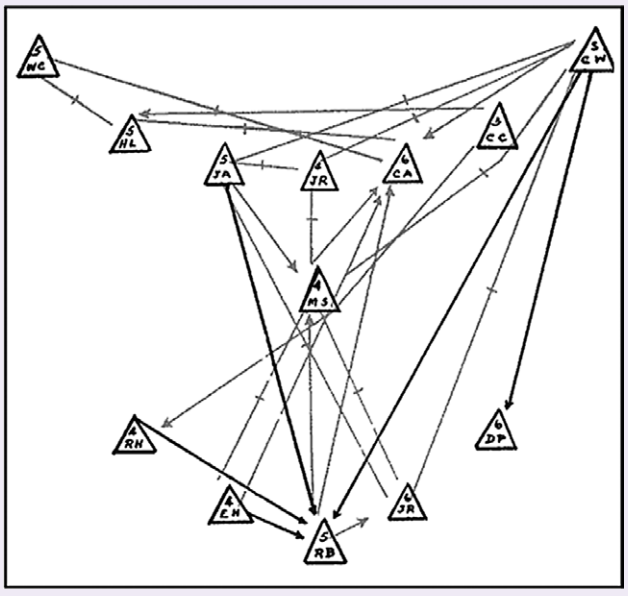


FIGURE 3.3 Jacob Moreno's early social network diagram of positive and negative relationships among members of a football team. Originally published in Moreno, J. L. (1934). *Who shall survive?* Washington, DC: Nervous and Mental Disease Publishing Company.

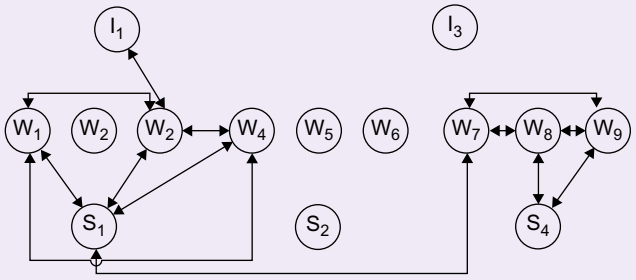


FIGURE 3.4 An early social network diagram of relationships among workers in a factory illustrates the positions different workers occupy within the workgroup. Originally published in Roethlisberger, F., and Dickson, W. (1939). *Management and the worker.* Cambridge, UK: Cambridge University Press.

sets [13]. In the 1950s, Nadel wrote about social roles and the social structures that define them [14]. He saw that the patterns of connections people had might be similar, even if they were connected to different people. These patterns, Nadel suggested, could be studied systematically, but in the 1950s the data and computational resources made that ambition a challenge.

Over time, Moreno's colleagues, including Paul Lazarsfeld, added key ingredients of the modern form of social network analysis: metrics and algorithms for calculating important network properties of the graph as a whole and for each individual in the graph (see Freeman [10] for details).

3.5 NETWORK ANALYSIS METRICS

Social scientists, physicists, computer scientists, and mathematicians have collaborated to create theories and algorithms for calculating novel measurements of social networks and the people and things that populate them. These quantitative *network metrics* allow analysts to systematically dissect the social world, creating a basis on which to compare networks, track changes in a network over time, and determine the relative position of individuals and clusters within a network.

Social network measures initially focused on simple counts of connections and became more sophisticated as concepts of density, centrality, structural holes, balance, and transitivity developed. Some metrics describe a network as a whole. For example, network density captures how highly connected vertices are by calculating the percentage of all possible connections that are realized. Other metrics are calculated for each vertex in a network. For example, centrality measures, of which there are many, capture how "important" (central) a vertex is within the network based on some objective criteria. Some people sit at the edge or periphery of their networks, whereas others are firmly at the center, connected to all the other most connected people. Even among a highly connected network, some pairs are not directly connected. When a third person bridges their connection, we can think of that person as a broker or connector. When that person is missing, we can think of a structural hole, a gap in which there is a missing connector. The following sections describe some of these metrics in more detail. Chapter 5 introduces some of the core metrics found in NodeXL through hands-on exercises.

ADVANCED TOPIC
Historical Obstacles to the Development of Network Analysis

Following the rapid development of the major elements of social network analysis in the 1930s there was a period of stagnation and neglect. For a variety of reasons, from Moreno's own personal and professional conflicts to the cost and lack of available network data sets and computing resources, social network analysis languished for decades.

The early social network literature was built on manually collected and processed data about social ties. Researchers would typically observe or survey population members, asking each to list those they came in contact with regularly for a variety of tasks and purposes. The prohibitive cost of this approach was a major limiting factor in the widespread application of social network analysis in enterprises and organizations. The recent explosion of computer-mediated social relationships and the associated drop in the costs of creating network data sets have made network approaches increasingly practical. As more details about our interactions and associations are tracked and captured by mobile devices and social media services, network analysis becomes increasingly useful.

Network analysis is computationally intensive: to generate many network metrics can require millions of calculations even when managing modest sized data sets. The recent explosion of computing power and the associated drop in costs have made network approaches increasingly practical, even if network methods remain among the most computationally intensive in use.

3.5.1 Aggregate Networks Metrics

A number of metrics describe entire networks. In some cases, a single network is broken into several disconnected pieces, called *components*. Some aggregate network metrics only work on networks where all of the vertices are connected in a single component, whereas others can be applied to entire networks even if they are split up. Here we describe just a few aggregate network metrics to give a flavor for what is possible, leaving a fuller discussion for Chapter 5.

Density is an aggregate network metric used to describe the level of interconnectedness of the vertices. Density is a count of the number of relationships observed to be present in a network divided by the total number of possible relationships that could be present. It is a quantitative way to capture important sociological ideas like cohesion, solidarity, and membership.

Centralization is an aggregate metric that characterizes the amount to which the network is centered on one or a few important nodes. Centralized networks have many edges that emanate from a few important vertices, whereas decentralized networks have little variation between the numbers of edges each vertex possesses.

Other metrics integrate attribute data with network data. For example, metrics that measure homophily look at the similarity of people who are connected. Studies typically show that people are connected to others who are similar to themselves on core attributes like income, education level, religious affiliation, and age.

3.5.2 Vertex-Specific Networks Metrics

Another set of metrics identifies individuals' positions within a network. Paramount among these is the set of centrality measures, which describe how a particular vertex can be said to be in the "middle" of a network. In the 1970s and 1980s, the sociologist Philip Bonacich developed a refined measure of centrality that took into consideration the different value a well-connected person can have in contrast to people with few connections. Network theorists noted that simply having many connections, called "degree centrality," was only one way to be "at the center" of things. A person with fewer connections might have more "important" connections than someone with more connections. One connection can be more important than another in different ways. Some are better because they bridge across otherwise separated portions of the network, whereas others are important because they connect to well-connected people. The following centrality metrics provide quantifiable measures for these concepts (see Chapter 5 for more details).

Degree Centrality

Degree centrality is a simple count of the total number of connections linked to a vertex. It can be thought of as a kind of popularity measure, but a crude one that does not recognize a difference between quantity and quality. Degree centrality does not differentiate between a link to the president of the United States and a link to a high school dropout. Degree is the measure of the total number of edges connected to a particular vertex. For directed networks, there are two measures of degree. In-degree is the number of connections that point inward at a vertex. Out-degree is the number of connections that originate at a vertex and point outward to other vertices.

Betweenness Centralities: Bridge Scores for Boundary Spanners

The notion of paths is central to the study of networks. Perhaps one of the most natural questions to ask about any two people in a network it is "How far apart are they?" This distance is measured simply: the distance between people who are not neighbors is measured by the smallest number of neighbor-to-neighbor hops from one to the other. For instance, people who are not your neighbors, but are your neighbors' neighbors, are a distance 2 from you, and so on. The shortest path between two people is called the "geodesic distance" and is used in many centrality metrics. For example, betweenness centrality is a measure of how often a given vertex lies on the shortest path between two other vertices. This can be thought of as a kind of "bridge" score, a measure of how much removing a person would disrupt the connections between other people in the network. The idea of brokering is often captured in the measure of betweenness centrality.

A structural hole is a missing bridge. Wherever two or more groups fail to connect, one can argue that there is a structural hole, a missing gap waiting to be filled. Burt provides compelling evidence that individuals who bridge structural holes are promoted faster than others [15]. Social network analysis has many strategic applications when people in an organization can analyze their position and the position of others. Managers and leaders can recognize gaps or disconnections within organizations and devote resources to traversing the divide. People may be able to apply social network analysis to identify locations in which a gap exists and elect to fill them, recognizing the value they can generate as broker between two otherwise separate groups.

Closeness Centrality: Distance Scores for Broadly Connected People

Closeness centrality takes a different perspective from the other network metrics, capturing the average

distance between a vertex and every other vertex in the network. Assuming that vertices can only pass messages to or influence their existing connections, a low closeness centrality means that a person is directly connected or "just a hop away" from most others in the network. In contrast, vertices in very peripheral locations may have high closeness centrality scores, indicating the number of hops or connections they need to take to connect to distant others in the network. Think of closeness, paradoxically, as a "distance" score. Some people are just a few miles from the big city, others must drive for hours: similarly, people with high "closeness" centrality scores have many miles or rather personal connections that they must travel to reach many other people in the network. Note that in some cases the inverse of the average distance to others in the network is used as a measure of closeness centrality. In that case, higher values indicate a more central position.

Eigenvector Centrality: Influence Scores for Strategically Connected People

Eigenvector centrality is a more sophisticated view of centrality: a person with few connections could have a very high eigenvector centrality if those few connections were themselves very well connected. Eigenvector centrality allows for connections to have a variable value, so that connecting to some vertices has more benefit than connecting to others. The PageRank algorithm used by Google's search engine is a variant of Eigenvector Centrality.

CLUSTERING COEFFICIENT: HOW CONNECTED ARE MY FRIENDS?

The clustering coefficient differs from measures of centrality. It is more akin to the aggregate density metric, but focused on egocentric networks. Specifically, the clustering coefficient is a measure of the density of a 1.5-degree egocentric network. When these connections are dense, the clustering coefficient is high. If your "friends" (alters) all know each other, you have a high clustering coefficient. If your "friends" (alters) don't know each other, then you have a low clustering coefficient. People have different measures for their clustering coefficient depending on the ways they cultivate connections to others and the environments they are in.

3.5.3 Clustering and Community Detection Algorithms

A network approach contrasts with those that presume the existence and boundaries of groups. In a network perspective, people occupy many relationships and are potentially members in many groups and less defined clusters. Defining exact boundaries in networks may be difficult, reflecting the reality of multiple and shifting memberships. From a network perspective, a group is a collection of vertices that are more connected to one another than they are to others. Relatively more cohesive or densely connected sets of vertices form regions, also called clusters, that may reflect the existence of groups without regard to whether they are officially recognized or even if members recognize their connections to one another. A rapidly growing body of research describes clustering algorithms, also called community detection algorithms, that automatically identify these clusters based on networks structures. We discuss these in more detail in Chapter 7.

3.5.4 Structures, Network Motifs, and Social Roles

Two people within a network may sometimes share a pattern of connection to other people, even if they do not connect to the same people. Certain professions have distinct patterns of connections, either linking with many others (real estate agents, and other retail professionals) or few (reclusive authors and artists, peripheral workers, and other people focused on things rather than people). In addition to the number of connections, some people share the pattern of connections among the people they connect. In some cases people are connected to people who are strangers to one another, in other cases a group may be densely connected to one another. These secondary patterns of connection are a distinctive feature of network analysis approaches: networks are as much about the attributes and patterns of connection among neighbors as they are about the attributes and connections of any individual.

Social roles are complex cultural and structural features of social life. An example social role like "father" is explicitly recognized in society, has a wide set of culturally shared meanings and expectations, is associated with particular goals and interests, and is partly defined by the content and structure of actions directed toward other distinctive role holders. Although social roles may not be as clearly defined or explicitly recognized by all the actors in a given social setting, they have identifiable content, behavioral, and structural features.

Studies of social media have illustrated the ways contributors create distinctive network patterns that reflect their role or status within the community (e.g., Welser, Gleave, and Smith [16]). These patterns are evidence of specialization of behavior in these social spaces. An example of a role in a social media space is the "answer person" who disproportionately provides the answers to questions asked in message board environments, "discussion people" who engage in extended exchanges of

messages in large and populous threaded discussions, "discussion starters" who demonstrate influence over the topics discussed by the "discussion people," "influential" people who are well connected to others who are more highly connected than they are, and boundary spanners who bridge between unconnected subgroups. These roles are described in greater detail in Chapter 9, devoted to email lists and discussion groups.

ADVANCED TOPIC
A Renaissance of Network Research and Data

Since the 1960s, network analysis has blossomed. New research and methods have flourished and social networking has developed a new prominence in mainstream culture. Despite early challenges, in the past several decades a healthy and growing subfield has reemerged around social network analysis. New network tools and concepts have been created and applied to a wide and growing range of domains. Mathematical sociology has developed as a major subdiscipline in the social sciences, dedicated to finding elegant descriptions of complex social phenomena. Starting 80 years ago with simple hand-drawn charts and diagrams that described small groups of people and their connections, network science concepts, methods, and tools are used today to calculate a range of measures that describe the shape, structure, and dynamics of potentially multimillion or billion vertex networks. New methods have been developed for automatically organizing and displaying visualizations of the links among large populations. This combination of structural models, visualizations, and metrics forms the key features of modern social network analysis.

In the late 1960s, Stanley Milgram explored the idea of small world networks in a study that came to be referred to as "Six Degrees of Separation" [17], which later inspired the 1990 John Guare play and 1993 movie of the same name. The study explored the question of how connected any two people selected at random might be. Milgram sent a collection of letters to people around the country asking them to send the message to someone they knew who could move their letter closer to the target, a stock broker in Massachusetts. On average, the letters took six steps to arrive at their destination. The "six degrees" or steps suggested that even in large networks where most people are not directly connected, people can be reached from every other person through a small number of steps.

Sampson's study in the late 1960s of relationships among members of a residential monastery captured social network data during an event in which several members were expelled or chose to leave [18]. A series of social network data sets were collected by asking participants about who they liked and spent time with. Social network analysis of this data allowed Sampson to identify the future lines of division among the members of the network. The idea that members of a network can be grouped based on how densely they are connected is an important concept in network analysis. These groups can be important divisions with consequences for the future of the network. For example, a notable study by Zachary in the 1970s mapped the structure of a Karate club based on affinities and connections between students and teachers. These maps predicted the ways the club eventually split when a new teacher, in conflict with the owner, left the studio and took many students with him [19].

The sociologist Barry Wellman demonstrated in the 1970s that real-world communities are composed of interlocking social networks of specialized relationships that changed dramatically in composition over a period of years.[2] He proposed that society was now characterized by networked individualism in contrast to the group memberships and identities of prior periods. Rather than defining oneself in professional or political terms, people create personal networks in which they occupy distinct locations and roles. He later applied these techniques to study online networks [20]. In 1977, Wellman founded a social network analysis professional association, the International Network for Social Network Analysis (INSNA). INSNA now has more than a thousand members, many of whom have gathered for more than 20 years for an annual conference ("Sunbelt") on social network analysis research.[3] Journals and publications devoted to social network analysis include *Social Networks*, *Connections*, and the *Journal of Social Structure*. Social network data, methods, and visualizations appear across a much wider spectrum of journals and conference publications.

[2]www.chass.utoronto.ca/~wellman/vita/index.html
[3]www.insna.org

In the early 1970s, the sociologist Mark Granovetter did research on the employment market, looking at how people discovered new job opportunities. He observed that, in contrast to the view held by classical economics, people were not freely floating independent actors in the labor market. They were embedded in a set of different relationships with particular people. Granovetter found that job news passed through connections that were not the closest and most intense relationships [21]. A person's "weak ties" brought news from distant parts of the social network to which "strong ties" did not have access because they occupied such a similar place in the network as the job seeker. Thus weak ties proved particularly useful for finding novel information, such as information about job prospects. Because weak ties were less intense, they were also less costly to maintain in terms of time and attention. As a result, it is possible to have many weak ties but only a few strong ties.

Armed with new network metrics and the means to calculate them, network analysts have focused on a variety of data sources and questions. Social networks have been applied to historical studies using records of investments, marriages, and memberships in elected positions. In the 1400s in the city of Florence, the Medici and Strozzi families struggled for domination. These families, like many others, were locked in political struggles. In the 1970s, John Padgett collected records of the social relations among Renaissance Florentine families that he extracted from historical documents. Families were often connected through a variety of ties, relations, and business connections. A data set was created that represented the financial loans, credits and joint partnerships, and marriages that bound families to one another. The resulting data set included information about each family as well as their links to others. Each family had a value representing its net wealth in the year 1427, the number of seats it held in the local government between the years 1282 and 1344, and the number of business or marriage ties among the population of 116 families. Analyzing these data, Padget found that the Medici held great power because, he argued, they sat at the center of business and family networks, brokering connections that no other family could equal [22, 23].

A more modern version of the study of historical Florentine politics can be found in the study of interlocking directorships in modern corporations. Many corporations and other institutions have a board of directors, some of whom serve on more than one board. When board members serve on two or more boards, they link those corporations and, in aggregate, create interlocking directorships that combine to form even larger meta-institutions. By building on research on interlocking directorships in U.S. corporations [24, 25], modern web sites like "They Rule" provide an interactive map that displays the common links between major corporations.[4]

In 1992, Robin Dunbar famously argued that people have an innate ability to handle a number of social relationships but not an endless number of them. Remembering people's names may have a biological limit as our brains evolved over long periods in which there were rarely more than a few hundred people within any region, group, or tribe. The number 150 has been loosely associated with the idea of a "Dunbar" number, an upper limit on the number of relationships a person can normally manage.[5] This number can be expanded with augmentation, through analog technologies like diaries, address books, and the "filo-fax." More recently, social media tools like Facebook and email contact lists extend our ability to maintain more relationships. These additional relationships can be said to be "weaker" than the core 150 "organic" relationships, but as Granovetter has shown, weak ties can collectively be of enormous value.

Business Applications of Social Network Analysis

Social network analysis has historically been an academic endeavor, but as network analysis tools and data sets become more available, pioneering businesses are applying it to help manage business challenges, gain insight into markets and communities, and build more robust industry relationships. For example, the work of Rob Cross and the Network Roundtable focuses on several practical applications of social network analysis for corporations and other large organizations, highlighting differences between healthy and underperforming divisions and the value of organization spanning connections [26, 27]. Others apply network analysis to the improvement of corporate structures and processes [28]. In the early 1990s, Monge and Contractor [29] documented the many forms of social network patterns that emerge inside of organizations and institutions.

Social networks have been shown to have a significant influence on the adoption of new technologies or social practices. The sociologist Everett Rogers described the concept of the "diffusion of innovations," arguing that

[4]www.theyrule.net

[5]www.lifewithalacrity.com/2004/03/the_dunbar_numb.html

(*Continued*)

ADVANCED TOPIC (*Continued*)

people with particular patterns of connections to others played pivotal roles in the success or failure of a new idea or message being adopted or distributed through the network [30]. Networks with different patterns of connection have different properties in terms of how they propagate a new message, rumor, or product and how they resist being dissolved when vertices are removed from the graph. These observations have significant implications for interventions into disease and rumor propagation and the cultivation of innovation [31].

Networks play an important role in e-commerce where collaborative filtering powers the familiar list of "books that people who liked this book also liked." Businesses are also interested in learning the requirements of viral marketing. We will discuss diffusion and marketing in more detail in the discussion of Twitter in Chapter 10, but for now the important thing to know is that diffusion can often lead to "cascades" where an unknown, even marginal idea can spread rapidly throughout the entire network and become the norm. Memes are a commonly-cited example of contagion, as are viral products, such as viral videos on YouTube that go from dozens to millions of viewers in a few months or even weeks.

3.6 SOCIAL NETWORKS IN THE ERA OF ABUNDANT COMPUTATION

The widespread adoption of networked communication technologies has significantly expanded the population of people who are both aware of network concepts and interested in network data. Although the idea of networks of connections of people spanning societies and nations was once esoteric, today many people actively manage an explicit social network of Internet friends, contacts, buddies, associates, and addresses that compose their family, social, professional, and civic lives. Email messages forwarded from person to person have become a common and visible example of the ways information passes through networks of connected people. The notion of "friends of friends" is now easy to illustrate in the features of Internet social media applications like Facebook, MySpace, and LinkedIn that provide explicitly named "social networking" services. Viral videos and chain emails illustrate the way word of mouth has moved into computer-mediated communication channels. The idea of "six degrees of separation" has moved from the offices of Harvard sociologists to become the dramatic premise of a Broadway play to now appear as an expected feature of services that allow people to browse and connect to their friend's friends.

As network concepts have entered everyday life, the previously less visible ties and connections that have always woven people together into relationships, cliques, clusters, groups, teams, partnerships, clans, tribes, coalitions, companies, institutions, organizations, nations, and populations have become more apparent. Patterns of sharing information, investments, personal time, and attention have always generated network structures, but only recently have these linkages been made plainly visible to a broad population. In the past few decades, the network approach to thinking about the world has expanded beyond the core population of researchers, analysts, and practitioners who have applied social network methods and perspectives to understand their businesses, communities, markets, and disciplines. Today, because many of us manage many aspects of our social relationships through a visibly computer-networked social world, it is useful for many more people to develop a language and literacy in the ways networks can be described, analyzed, and visualized. Visualizing and analyzing a social network is an increasingly common personal or business interest. The science of networks is a growing topic of interest and attention, with a growing number of courses for graduates and undergraduates and even becoming the topic of a television documentary.[6]

The availability of cheaper computing resources and network data sets has enabled a new generation of researchers access to studies of the structures of social relationships at vastly larger scale and detail. Since the late 1960s, as computing resources and network data sets have grown in availability and dropped in cost, researchers began developing tools and concepts that enabled a wider and more sophisticated application of social network analysis.

[6]"Connected: The Power of Six Degrees," http://ivl.slis.indiana.edu/km/movies/2008-talas-connected.mov

ADVANCED TOPIC
Social Network Analysis Research Meets the Web

As access to electronic networks grew in the 1970s, academic and professional discussions and collaborations began to take place through them. Systems to support the exchange of messages and the growth of discussions and even decisions became a major focus of systems development and the focus of study itself. Freeman and Freeman [32] collected data from the records of the Electronic Information Exchange System (EIES) that itself hosted a discussion among social network researchers. Two relations were recorded: the number of messages sent and acquaintanceship. These systems became the focus of the first systematic research into naturally occurring social media. Even before the Internet, early computer network applications supported the creation of exchanges, discussions, and therefore social networks, built by reply connections among authors.

Early proprietary systems evolved into the public World Wide Web. In the 1990s, the computer scientist Jon Kleinberg identified the patterns of links between high-quality web pages, an algorithm that went on to inspire Stanford graduate students who founded the Google corporation. Kleinberg described different locations in a population of linked documents on the World Wide Web. On the World Wide Web, a document or page can link to another page, forming a complex network of related documents. Some documents contain many pointers to other documents, whereas others have many documents point at them. These hubs and authorities defined two broad classes of web pages that offered a path to identifying high-quality content. Links from one page to another are considered to be indicators of value. Refinements of the HITS algorithm made use of eigenvector centralities to implement the page rank algorithm that is the core of the Google web search ranking method [33].

Network researchers studying social networks and the Internet found that empirical networks often exhibit "small-world properties": most nodes are not neighbors with each other, but nodes can be reached from every other node in a small number of hops. In the late 1990s, the physicist/sociologist Duncan Watts, working with the mathematician Steven Strogatz, created mathematical models of "small world" networks and contrasted them with purely random networks such as those proposed by Erdos and Renyi [34]. Their model captured the natural properties of social networks far better than those that assumed a purely random or normal distribution of links. Although most connections are to others who are local, a few connections importantly can jump far from the individual. Many of our friends are likely to live or work near us, but a few may be very far away. These relatively rare far-reaching links can dramatically change the properties of the network, making the widespread transmission of messages much easier. This model significantly improved on earlier models of network growth and structure, better approximating the observed structure of naturally occurring social networks. Later researchers have built upon their work to devise models that generate "small world" networks that more closely match empirical networks, helping us to understand how networks may have become the way they are. For example, Barabasi and Albert have developed a family of models of preferential attachment that can generate "scale-free" networks, which are a common feature of social networks [35]. Scale free networks have a power law degree distribution, meaning that there are a few key hubs in a network and many poorly connected vertices. While none of these models perfectly predict social networks, they provide a method for systematically comparing networks and focus attention on the processes that may have led to the characteristics that we see in networks.

In the past few years, researchers have begun to study large web-based networks. For example, Leskovec and Horvitz calculated metrics for a graph that includes more than 300 million users of the Microsoft Messenger service [36]. Each user typically had one or more "buddies" with whom he or she might send one or more messages and receive some in return. Buddies often listed their locations, allowing these linkages to be aggregated into a complex map of the world and the flow of conversation around it. Others have reported on the hyperlink network created by web pages hyperlinking to other web pages (e.g., Park and Thelwall [37]). A number of studies have examined the blog network. For example, Adamic and Adar [38] showed how political blogs are divided into two clear clusters with minimal overlap that represent the left and right political populations in the United States. More recently, Kelly and Etling mapped Iran's blogosphere, identifying more than 20 subcommunities of bloggers who wrote in Farsi for an Iranian audience.[7]

Another line of research has focused on visualizing social networks. A representative paper by Heer and Boyd [39] described a tool called Vizster that allowed users to navigate through their friends from a social networking site to explore social connections.

[7] http://cyber.law.harvard.edu/publications/2008/Mapping_Irans_Online_Public

3.7 THE ERA OF ABUNDANT SOCIAL NETWORKS: FROM THE DESKTOP TO YOUR POCKET

We now live in a new era of network data abundance. Network data collection was once a time-consuming and laborious process that yielded small data sets at great cost. Observations, surveys and interviews took many days or weeks to perform, could not be repeated frequently, required many people to produce, and often yielded low rates of participation with inherent biases and errors. Asking people about their relationships with others continues to have benefits and offers unique sources of insight, but people have been shown to be a poor source of accurate information as bias and faulty memory warp what people report about who they know and with whom they interact. The challenge of creating a data set that spanned long periods or large numbers of people or contained records of many events proved insurmountable using traditional methods.

Today, interactions between people increasingly take place through computing systems. Users create many types of networks in a machine-readable form each day as our interactions are documented in a computer. When we use these communication tools, databases are created and maintained with records and log files that document the details of the time, place, and participants of each interaction, whether via computers or telephones and even televisions. These event logs describe many different kinds of connection but share a common structure in which one person or entity is linked to another by some relationship.

The creation of these machine-readable network data sets mean that long periods of time or large populations connected by many events can now be studied using widely available computing equipment and data sources.

Like a jump from Galileo's handmade telescope to the orbiting *Hubble*, network science has made a vast leap in scale and scope as we create a digitally networked world around ourselves.

As the historical drought of social network data has ended with a flood of sources of network data, the challenge has been to rapidly develop the tools and concepts needed to process and analyze them. Technical methods for building multiterabyte databases have shifted to the even vaster task of managing petabytes of data. New methods of harnessing thousands and even millions of computers in parallel have been driven by the growing need to manage vast data stores growing from the web. The challenge is likely to grow steeper as new sources of network data come pouring out off an emerging class of sensor-rich devices that record vast streams of data from millions to billions of people, devices, and locations. The early wave of this surge of data can be seen in new sources of data from everyday life that are being captured and recorded with mobile devices, creating a new stream of archival material that is richer than all but the most obsessively observed biographies. It has become common in recent years that the most timely and well-placed photographs and video recordings have come from everyday individuals with phones and computers rather than from news photographers and reporters.

The coming wave of mobile technologies is likely to deepen this trend, with new ways for phones or other devices to capture information about their users and the world around them. Research projects like SenseCam from Microsoft Research, which captures a continuous stream of photographs and temperature and motion data, are now becoming products,[8] and services like nTag, Spotme, Loopt, Foursquare, and Google Latitude using devices like iPhones and Droids are weaving location into every application (see Chapter 2).

As phones are increasingly able to notice each other, a new set of mobile social software applications are becoming possible, as evidenced by new services such as Bump.[9]

A service like SenseNetworks[10] is a good example of a mobile data collection, analysis, and presentation service. Other services and products like CureTogether[11] and FitBit[12] are examples of social movements that are enabled by web applications integrated with devices that provide self-monitoring medical tracking. These communities overlap with the trail-based exercise communities of runners, bikers, skaters, hikers, and skiers, some with artistic inclinations (they hike in paths that resemble drawings when seen on a map). A new wave of devices is emerging that extensively quantifies your "self" and "others," allowing you to perhaps swap sensor data with other people. The result could be an aggregated map of the heath and environmental conditions of the planet, not unlike early examples of collectively authored road maps of whole nations accomplished by the Open Street Map project.[13]

[8]www.nytimes.com/2010/03/09/health/09memory.html
[9]http://bu.mp
[10]www.sensenetworks.com
[11]www.curetogether.com
[12]www.fitbit.com
[13]www.openstreetmap.org/

3.8 TOOLS FOR NETWORK ANALYSIS

The growth of interest in network analysis has been dramatic, but until recently the development of social network analysis tools has lagged, and they remained a challenge to use for many people. Applying network approaches has been traditionally a challenge that involved much more than simply mastering a new set of concepts and ideas that focus on relationships and patterns. Network data have traditionally been difficult to create and collect, and the tools for analyzing and visualizing networks have demanded significant technical skill and often mastery of programming languages. Many tools that exist to support network analysis demand significant commitment to learn and master. Existing network tools that are relatively easier to use have typically lacked support for easily importing social media network data. In the past few years, many network analysis projects and research papers have focused on computer-mediated networks of people, documents, and systems. Only recently have new tools made it simpler for people to extract data from major social media network sources and to perform a basic network analysis workflow without requiring programming skills or using a command line interface.

Social media network data collection, scrubbing, analysis, and display tasks have historically required a remarkable collection of tools and skills. A great example of the variety of tools that can be used in concert to extract, analyze, and display social media networks can be found on Drew Conway's blog.[14] This is a powerful set of tools for those who can master the demands of python or other programming languages and the application programmer interfaces (API) that give sophisticated users special access to resources. In contrast, this book focuses on a single tool designed for nonprogrammers, NodeXL, because of its relative ease of use, support for rich visuals and analytics, and integration with the ubiquitous Excel spreadsheet software. The python path is certainly the high road for experts and those with demanding volumes of data or esoteric data analysis requirements. But for the noncoding user, NodeXL may be one of the easiest ways to both manipulate network graphs and get graphs from a variety of social media sources. A detailed step-by-step guide to the core features of NodeXL can be found in Part II of the book.

3.9 NODE-LINK DIAGRAMS: VISUALLY MAPPING SOCIAL NETWORKS

One of the key elements that characterizes modern social network analysis is the use of visualizations of complex networks. Compared to staring at edge lists or network matrices (see Section 3.2.4), looking at a network graph can provide an overview of the structure of the network, calling out cliques, clusters, communities, and key participants. It could be said that a graph is worth a thousand ties. Not only can network visualizations inspire understanding and insights, they can also be appealing and even beautiful. They can serve as persuasive tools that demonstrate important points about networks. The ability to map attribute data and network metric scores to visual properties of the vertices and edges (see Chapters 4 and 5) makes them particularly powerful.

However, network visualizations are often as frustrating as they are appealing. Network graphs can rapidly get too dense and large to make out any meaningful patterns as illustrated in Figure 3.5. Many obstacles like vertex occlusions and edge crossings make creating well-organized and readable network graphs challenging. There is an upper limit on the numbers of vertices and edges that can be displayed in a bounded set of pixels; typically only a few hundred or thousand vertices can be meaningfully and distinctly represented on average-sized computer screens. In his appeal for better-quality network visualization, Shneiderman [40] has suggested that we aspire to reach the worthy but not always attainable goal of "netviz nirvana" in which the following goals are proposed:

- Every vertex is visible.
- Every vertex's degree is countable (i.e., the number of connections that start or end at that vertex).
- Every edge can be followed from source to destination.
- Clusters and outliers are identifiable.

To approach netviz nirvana, careful preparation, layout, and filtering techniques must be used. In practice, network visualizations often fall far from the mark. However, the graphs shown throughout this book illustrate the value of carefully crafting network graphs. We hope they will inspire network analysts to take the care needed to create substantive, understandable, and aesthetically pleasing graphs.

3.10 COMMON NETWORK ANALYSIS QUESTIONS APPLIED TO SOCIAL MEDIA

Once a set of social media networks has been constructed and social network measurements have been calculated, the resulting data set can be used for many applications. For example, network data sets can be used to create reports about community health, comparisons of subgroups, and identification of important

[14] www.drewconway.com/zia/?p=204

individuals, as well as in applications that rank, sort, compare, and search for content and experts.

The value of a social network approach is the ability to ask and answer questions that are not available to other methods. This means focusing on relationships. Although analysts, marketers, and administrators often track social media participation statistics, they rarely consider relationships. Traditional participation statistics can provide important insights about the engagement of a community, but can say little about the connections between community members. Network analysis can help explain important social phenomena such as group formation, group cohesion, social roles, personal influence, and overall community health. Combining traditional participation metrics with network metrics provides the best of both worlds and allows you to answer important questions such as the following:

- What kinds of social roles are being performed within a social media repository? Does a community have enough people filling the important roles?
- Which individuals play important social roles within a group or collection? Who would make a good administrator based on that person's network position?
- What subgroups exist? Do connections between subgroups exist? Who plays the bridge roles that connect otherwise unconnected groups?
- How do new ideas propagate through a network? Who are the influencers?
- How do the overall structures of a social network change after a particular event (e.g., a company social, a round of new hires or layoffs)?

3.11 PRACTITIONER'S SUMMARY

The opportunities for practitioners to apply network analysis to contemporary business, community management, political influence, and team collaboration have dramatically increased in recent years. The once esoteric

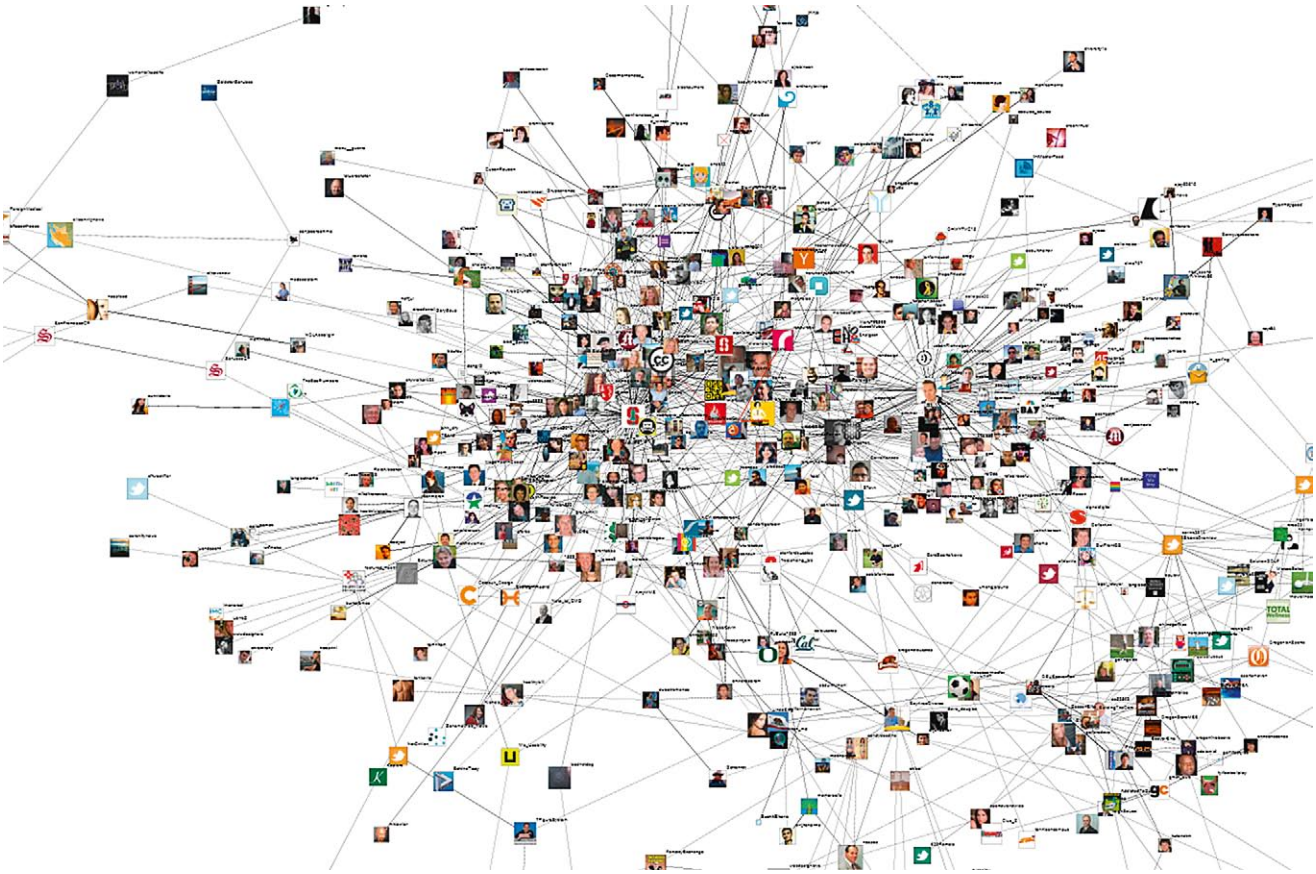


FIGURE 3.5 A medium-sized node-link network diagram visualization of Twitter users linked by patterns of following. This sized graph illustrates many issues with a network graph containing more than a few dozen vertices. Many vertices sit on or overlap with other vertices. The number of edges associated with some vertices is impossible to count, whereas other edges cannot be traced from source to destination. Improvements to network layout are an active area of research.

concepts and metrics of network analysis have become talk show and airport lounge topics. The difficulties in collecting and analyzing network data have been dramatically reduced by powerful database methods and well-designed network analysis and visualization tools. There is still a lot of work to be done, but practitioners now have the potential to make more effective decisions based on network analyses of their own data conducted in a few hours, rather than a few months.

Learning the concepts and tools is a necessary first step, but the payoffs are large. The growing industry of social media and networking consultants complemented by a vast array of books plus informative web sites, online seminars, and Wikipedia pages, makes the necessary training widely available. At the same time, network analysis methods are spreading through university curricula rapidly and filtering into high school courses.

Attending public seminars and professional conferences provides other means to acquire skills and make valuable connections. Your first steps may be a struggle, but we hope that with each step the processes become smoother and the professional benefits larger.

3.12 RESEARCHER'S AGENDA

The research progress on network analysis has been dramatic in the past few decades, transforming an exotic research topic into a thriving research community in academia, government, and industry. The existing metrics, clustering, and layout algorithms are stabilizing, but innovative approaches are still emerging to trigger bursts of new research. As practitioner pressure builds to apply network analysis to ever larger data sets, researchers have developed remarkably more efficient algorithms, while hardware developers have produced powerful graphics processors (based on gaming computers), huge arrays of computers, and scalable cloud computing services. Meanwhile, new social media services generate more relational data than ever before, ushering in a golden era of social science research on human relationships and collaboration.

The algorithms and hardware provide the platforms, but the concomitant development of vastly improved user interfaces for network analysis has begun to enlarge the community of users from the dedicated sociologists who are also programmers to the broad segment of business analysts who use spreadsheets or simplified web-based tools. Packaging the complex processes of frequently applied network analyses into a few clicks is the next challenge in many fields, thereby inspiring other researchers and developers to simplify the processes even further, while increasing the power offered to users. The best is yet to come.

References and Resources

[1] J.P. Scott, Social Network Analysis: A Handbook, Sage, Thousand Oaks, CA, 2000.

[2] D. Easley, J. Kleinberg, Networks, Crowds, and Markets: Reasoning about a Highly Connected World, Cambridge University Press, Cambridge, UK, 2010.

[3] M. Newman, A.-L. Barabasi, D.J. Watts (Eds.), The Structure and Dynamics of Networks, Princeton University Press, Princeton, NJ, 2006.

[4] P. Carrington, J. Scott, S. Wasserman (Eds.), Models and Methods in Social Network Analysis (Structural Analysis in the Social Sciences), Cambridge University Press, Cambridge, UK, 2005.

[5] W. de Nooy, A. Mrvar, V. Batageli, Exploratory Social Network Analysis with Pajek (Structural Analysis in the Social Sciences), Cambridge University Press, Cambridge, UK, 2005.

[6] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences), Cambridge University Press, Cambridge, UK, 1994.

[7] B. Wellman, Structural analysis, in: B. Wellman, S.D. Berkowitz (Eds.), Social Structures, Cambridge University Press, Cambridge, UK, 1988, pp. 19–61.

[8] J.L. Moreno, Who shall survive? A new approach to the problem of human interrelations, Nervous and Mental Disease Publishing Co., Washington, 1934.

[9] A. Davis, B.B. Gardner, M.R. Gardner, Deep South: A social Anthropological Study of Caste and Class, University of Chicago Press, Chicago, Ill., 1941.

[10] L.C. Freeman, The Development of Social Network Analysis: A Study in the Sociology of Science, BookSurge, LLC, North Charleston, SC, 2004.

[11] F. Roethlisberger, W. Dickson, Management and the Worker, Cambridge University Press, Cambridge, UK, 1939.

[12] G. Homans, The Human Group, Harcourt-Brace, New York, 1950.

[13] R. Breiger, S. Boorman, P. Arabie, An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling, J Math Psychol 12 (1975) 328–383.

[14] S.F. Nadel, The Theory of Social Structure, Cohen & West, London, 1957.

[15] R. Burt, Brokerage and Closure: An Introduction to Social Capital, Oxford University Press, Oxford, 2005.

[16] H. Welser, E. Gleave, M. Smith, Visualizing the signatures of social roles in online discussion groups, J. Soc. Struct. 8 (2) (2007).

[17] S. Milgram, The small world problem, Psychology Today 2 (1967) 60–67.

[18] S. Sampson, Crisis in a cloister. Unpublished doctoral dissertation, Cornell University, 1969.

[19] W. Zachary, An information flow model for conflict and fission in small groups, J Anthropol Res 33 (1977) 452–473.

[20] B. Wellman, An electronic group is virtually a social network, in: Kiesler Sara (Ed.), Culture of the Internet, Lawrence Erlbaum, Mahwah, NJ, 1997.

[21] M. Granovetter, The strength of weak ties, Am J Sociol 78 (6) (1973) 1360–1380.

[22] J. Padgett, C. Ansell, Robust action and the rise of the medici, 1400-1434, Am. J. Sociol. 98 (6) (1993) 1259–1319.

[23] D. Kent, The Rise of the Medici: Faction in Florence, 1426-1434, Oxford University Press, Oxford, 1978.

[24] M. Mizruchi, L.B. Stearns, A longitudinal study of the formation of interlocking directorates, Adm. Sci. Q 33 (2) (1988) 194–210.

[25] B. Mintz, M. Schwartz, The Power Structure of American Business, University of Chicago Press, Chicago, 1985.

[26] R. Cross, R.J. Thomas, Driving Results through Social Networks: How Top Organizations Leverage Networks for Performance and Growth, Jossey-Bass, San Francisco, CA, 2009.
[27] R. Cross, R.J. Thomas, Driving Results through Social Networks: How Top Organizations Leverage Networks for Performance and Growth, John Wiley & Sons, San Francisco, CA, 2009.
[28] M. Kilduff, W. Tsai, Social Networks and Organizations, Sage, Thousand Oaks, CA, 2003.
[29] P.R. Monge, N. Contractor, Theories of Communication Networks, Oxford University Press, New York, 2003.
[30] D.E.M. Rogers, Diffusion of Innovations, fifth ed., Simon and Schuster, New York, 2003.
[31] M.E.J. Newman, The structure and function of complex networks, SIAM Review 45 (2003) 167–256.
[32] L.C. Freeman, Centrality in social networks: conceptual clarification, Social Networks 1 (1979) 35–41.
[33] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, in: Proc. 7th World-Wide Web Conference (WWW7), BrisBane, Australia, April 1998.
[34] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks, Nature 393 (6684) (4 June 1998) 440–442.
[35] A.L. Barabasi, R. Albert, Emergence of scaling in random networks, Science 286 (15 October 1999) 509–512.
[36] J. Leskovec, E. Horvitz, Planetary-scale views on a large instant-messaging network, in: Proc. 17th international Conference on World Wide Web (Beijing, China, April 21–25, 2008). WWW '08. ACM, New York, NY, 2008, 915–924.
[37] H.W. Park, M. Thelwall, Hyperlink analyses of the world wide web: a review, Journal of Computer Mediated Communication 8 (4) (July 2003).
[38] L.A. Adamic, E. Adar, Friends and neighbors on the web, Social Networks 25 (3) (July) (2003) 211–230.
[39] J. Heer, D. Boyd, Vizster: Visualizing Online Social Networks, in: Proc. 2005 IEEE Symposium on Information Visualization (October 23-25, 2005), INFOVIS. IEEE Computer Society, Washington, DC, 2005.
[40] B. Shneiderman, A. Aris, Network visualization with semantic substrates, IEEE Symposium on Information Visualization and IEEE Trans, Visualization and Computer Graphics 12 (5) (2006) 733–740.

Additional Resources

Barabasi, A.L. (2003). *Linked: How everything is connected to everything else and what it means*. New York: Penguin Group.
Bonacich, P. (1987). Power and centrality: A family of measures. *The American Journal of Sociology*, *92*(5), 1170–1182.
Borgatti, S., Mehra, A., Brass, D., & Labianca, G. (2009). Network analysis in the social sciences. *Science*, *323*(5916), 892–895.
Brandes, U., & Erlebach, T. (Eds.), (2005). *Network analysis: Methodological foundations*. Berlin, Heidelberg: Springer-Verlag.
Buchanan, M. (2002). *Nexus: Small worlds and the groundbreaking theory of networks*. New York, NY: Norton.
Burt, R. (1995). *Structural holes: The social structure of competition*. Cambridge, MA: Harvard University Press.
Burt, R.S. (1992). *Structural holes*. Cambridge, MA: Harvard University Press.
Haythornthwaite, C. (1996). Social network analysis: An approach and technique for the study of information exchange. *Library and Information Science Research*, *18*(4), 323–342.
Johnson, S. (2002). *Emergence: The connected lives of ants, brains, cities, and software*. London, UK: Penguin.
Knoke, D., & Yang, S. (2007). *Social network analysis (Quantitative Applications in the Social Sciences)*. Thousand Oaks, CA: Sage.
Nooy, W., De, Mrvar, A., & Batagelj, V. (2005). *Exploratory social network analysis with pajek*. Cambridge, UK: Cambridge University Press.
Watts, D. (1999). *Small worlds*. Princeton, NJ: Princeton University Press.
Watts, D. (2003). *Six degrees*. New York: Norton.
Wellman, B., & Berkowitz, S.D. (1988). *Social structures: A network approach*. Cambridge, UK: Cambridge University Press.

PART II

NODEXL TUTORIAL

Learning by Doing

This section (Chapters 4 through 7) introduces the NodeXL tool for analyzing networks using a hands-on approach. It assumes no knowledge of network analysis tools and assumes only a basic familiarity with social network analysis concepts (see Chapter 3). The section has been used as the basis for workshops and classroom lab sessions and takes approximately 8 hours to complete with participants following along. It also serves well as a walk-through of NodeXL, introducing all of the basic functionality. We recommend working through the examples using the data sets available at the book's accompanying web site. The examples found in the chapters in Part III assume you have read the Part II chapters.

CHAPTER

4

Getting Started with NodeXL, Layout, Visual Design, and Labeling

OUTLINE

4.1 Introduction 53

4.2 Downloading and Installing NodeXL 54

4.3 Getting Started with NodeXL 54
- *4.3.1 Data Entry* 54
- *4.3.2 Showing the Graph* 54
- *4.3.3 Highlighting an Edge* 54
- *4.3.4 Importing an Edge List* 55
- *4.3.5 Resizing and Moving the Graph Pane* 57

4.4 Layout: Arranging Vertices in the Graph Pane 57
- *4.4.1 Automatic Layout* 57
- *4.4.2 Directed Graph Type* 57
- *4.4.3 Updating the Graph Pane* 59
- *4.4.4 Manual Layout* 59
- *4.4.5 Preserving Manual Layout* 59
- *4.4.6 Zooming and Scale* 59

4.5 Visual Design: Making Network Displays Meaningful 59
- *4.5.1 Vertex Colors* 59
- *4.5.2 Adding Descriptive Data* 60
- *4.5.3 Changing Vertex Size (and Other Visual Properties)* 61
- *4.5.4 Autofilling Columns* 61
- *4.5.5 Legend* 63
- *4.5.6 Changing General Graph Appearance* 63

4.6 Labeling: Adding Text Labels to Vertices and Edges 63
- *4.6.1 Adding Vertex Labels* 63
- *4.6.2 Viewing Hidden Columns* 64
- *4.6.3 Adding Labels as the Shape* 64
- *4.6.4 Adding Labels Alongside a Shape* 65
- *4.6.5 Adding Tooltips* 65
- *4.6.6 Adding Edge Labels* 65
- *4.6.7 Saving a NodeXL File* 66

4.7 Practitioner's Summary 66

4.8 Researcher's Agenda 67

4.1 INTRODUCTION

As discussed in Part 1 billions of people worldwide use social media tools such as email, discussion forums, blogs, microblogs, and wikis. As people communicate through these media via desktop and web-based applications on fixed and mobile devices, they create multiple complex social network structures. The lively interaction and networks of relationships created through these technologies is of growing importance to individuals, organizations, and communities. Understanding how these social media user communities grow, change, fail, or succeed is a growing concern to researchers and professionals (Chapter 2). The field of social network analysis (Chapter 3) provides a set of concepts and metrics to systematically study these dynamic processes. The methods of information visualization have also become valuable in helping users to discover patterns, trends, clusters, and outliers, even in complex social networks.

The profusion of software tools for social network analysis and visualization demonstrates the strength of interest, but many of these tools are difficult to use,

particularly for those who lack experience with programming languages. Wikipedia offers a current review of software in the article on Social Network Analysis Software.[1] NodeXL, an open source software tool, was designed especially to facilitate learning the concepts and methods of social network analysis with visualization as a key component [1].

The NodeXL Template for Microsoft Excel 2007 is a free and open source extension to the widely used spreadsheet application that provides a range of basic network analysis and visualization features. NodeXL uses a highly structured workbook template that includes multiple worksheets to store all the information needed to represent a network graph. Network relationships (i.e., graph edges) are represented as an "edge list," which contains all pairs of vertices that are connected in the network (see Chapter 3). Other worksheets contain information about each vertex (i.e., node) and cluster (i.e., group). Visualization features allow users to display a range of network graph representations and map data attributes to visual properties including shape, color, size, transparency, and location.

NodeXL is designed to support students who are learning social media network analysis and professionals who are interested in applying network analysis to business problems. It builds on the familiar spreadsheet paradigm to provide an easy-to-use tool for nonprogrammers that allows a variety of visual properties, supports powerful filtering, calculates frequently used network metrics, and offers rich support for diverse visual network layouts. NodeXL includes powerful automated features, while allowing for manual control of individual vertex placement, labeling, color properties, and the like. NodeXL integrates metrics, statistical methods, and visualization to gain the benefit of all three approaches. The tool supports work with modest-sized networks of several thousand vertices, although some users have successfully dealt with tens of thousands of vertices.

4.2 DOWNLOADING AND INSTALLING NODEXL

You can download NodeXL from Microsoft's Codeplex site, www.codeplex.com/NodeXL. NodeXL works with Excel 2007 and will work with Excel 2010 but not with earlier versions of Excel or with Mac versions of Excel. The source code is also available on the Codeplex site. Programmers are invited to participate in the project by adding new Data Importers, developing new layout algorithms, or adding new metrics (see Appendix).

NodeXL is constantly being updated, so you will want to check to make sure you have the latest version. If you are teaching a course based on NodeXL, you may want to have students install the version of NodeXL used in this book, version 1.0.1.113, and also available for download from the Codeplex web site at http://nodexl.codeplex.com/releases/view/40939 If you decide to use a more recent version, there will be some discrepancies between the examples in the book and the version you are using. You can visit the NodeXL Release History at http://nodexl.codeplex.com/wikipage?title=CompleteReleaseHistory for a list of changes since version 1.0.1.113.

Once you download the NodeXL .zip file, unzip it into any folder, make sure Excel is closed, Run Setup.exe, and delete the unzipped files. You need to be an administrator on the machine you are installing it on. As a result, you'll likely need an IT support staff member to install it on lab computers. If it is the first time you have installed an Excel 2007 template, the setup program will install several prerequisites.

4.3 GETTING STARTED WITH NODEXL

There are two ways to open the NodeXL template. You can access it through the Start menu: All Programs → Microsoft NodeXL → Excel 2007 Template. Or, while running Excel 2007, create a new document and select "My templates…" and choose NodeXLGraph. A blank NodeXL template shows the usual Excel menu bar across the top, a blank workbook on the left, and a blank graph pane on the right (Fig. 4.1). NodeXL allows users to fill or paste in columns of relationship data (i.e., edge list data) in the Edges worksheet. Each row represents a vertex pair that is connected to each other.

4.3.1 Data Entry

One way to begin using NodeXL is to type in your own edge list. For example, you might type the name of people who are friends in each row filling in the Vertex 1 and Vertex 2 columns (Fig. 4.2).

4.3.2 Showing the Graph

Click on the Show Graph button (directly above the graph pane) to show the network of friendships (Fig. 4.3). The example assumes undirected relationships; that is, Ann is a friend of Bob, and Bob is a friend of Ann.

4.3.3 Highlighting an Edge

Click one of the workbook rows to highlight the corresponding edge and the two vertices in the graph. For example, clicking on row 5 highlights the edge connecting Ann to Carol (Fig. 4.4). You can even click on multiple rows and all related edges and vertices will be highlighted.

[1] http://en.wikipedia.org/wiki/Social_network_analysis_software

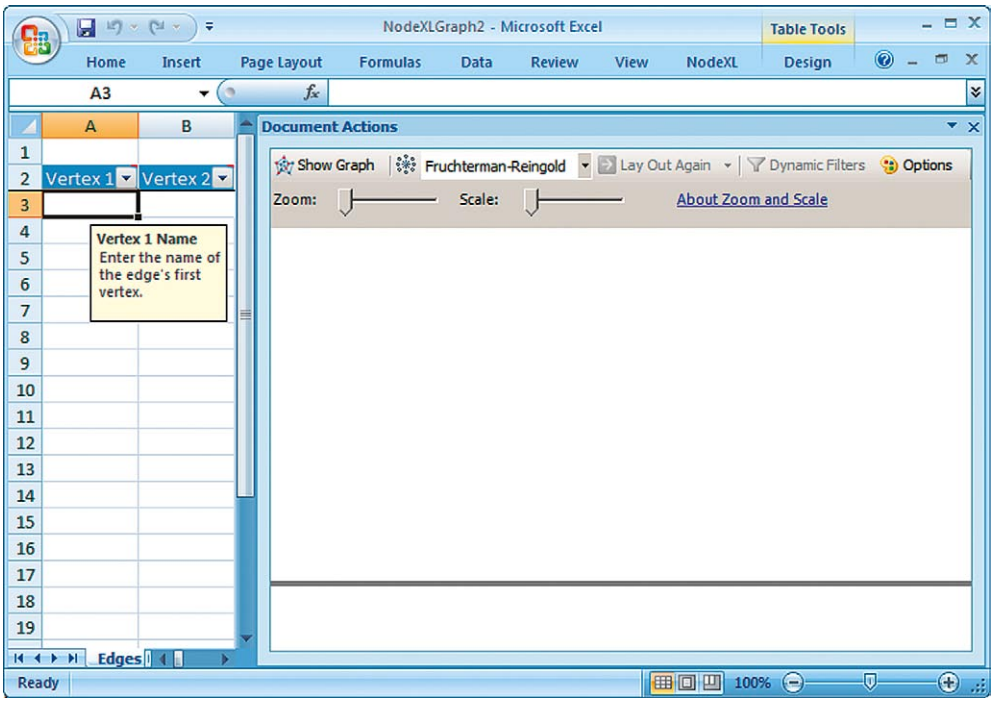


FIGURE 4.1 Start with an empty NodeXL Edges worksheet (left) and graph pane (right).

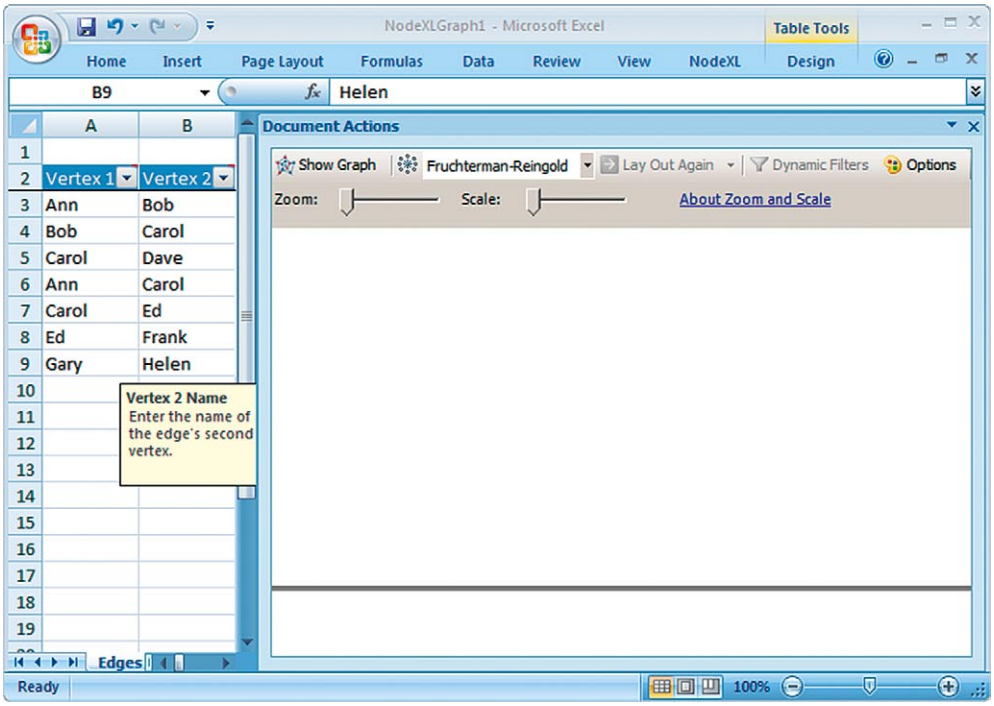


FIGURE 4.2 Seven friendships typed by hand into the Vertex 1 and Vertex 2 columns in NodeXL. For example, Ann and Bob are friends.

4.3.4 Importing an Edge List

Another way to begin using NodeXL is to use the Import command to load relationship data from an existing file or data source. The Import command is found on the NodeXL ribbon (Fig. 4.5) along with other NodeXL-specific commands. Someone may provide you with relationship data in the form of an edge list or an "adjacency" matrix (see Chapter 3 and [2]). NodeXL can import files from another social network analysis program such as Pajek or UCINET or from an other open Excel workbook. Networks stored in the GraphML format can also be imported. Alternatively, you can cut and paste from another Excel spreadsheet to fill in the edge list. NodeXL also includes import tools that grab network data directly from social sites such as Twitter and YouTube or from email collections stored on your computer (see Part III).

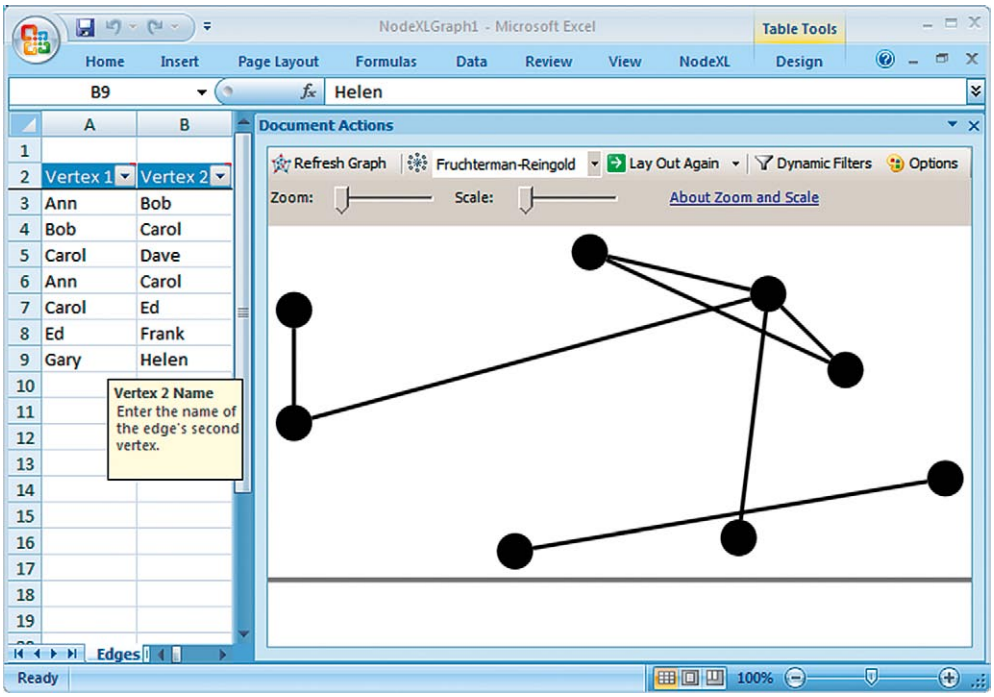


FIGURE 4.3 Your first NodeXL graph using the Fruchterman-Reingold layout shows the eight friends and seven friendships.

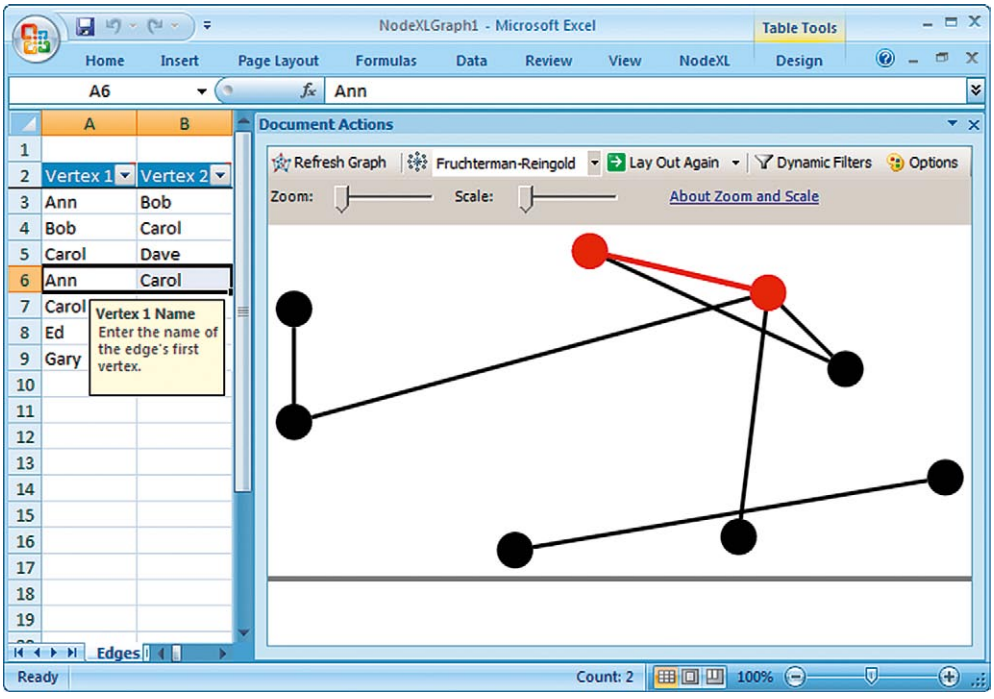


FIGURE 4.4 Clicking on row 5 in NodeXL (Ann and Carol) highlights their friendship edge in the graph pane.

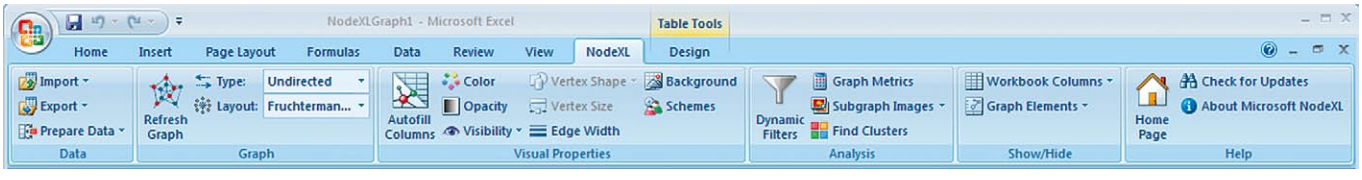


FIGURE 4.5 The NodeXL menu ribbon has sections for Data, Graph, Visual Properties, Analysis, Show/Hide, and Help.

The NodeXL menu ribbon (Fig. 4.5) provides access to the core NodeXL features. Hovering over some buttons displays additional information about that feature. Some features are accessible by right-clicking. You'll be using the NodeXL controls to create meaningful layouts of vertices, control the visual properties of vertices and edges (e.g., color, size, opacity), and calculate network metrics that describe your network data.

4.3.5 Resizing and Moving the Graph Pane

As you work with the data, you may want to resize the graph pane by moving the cursor to the left side of the pane until you see the ↔ symbol and then dragging it to the desired size. It is also possible to move the graph pane to the left, above, or below the worksheet data by clicking on the title that reads "Document Actions" and dragging it around. You can even drag the graph pane outside the Excel window. When used on a computer with a large monitor or two or more monitors, the NodeXL graph pane can be moved to occupy one full screen while the spreadsheet is fully visible in another screen.

4.4 LAYOUT: ARRANGING VERTICES IN THE GRAPH PANE

4.4.1 Automatic Layout

NodeXL offers several automatic layouts that can be selected from the control in the graph pane or in the NodeXL ribbon. The default layout type for NodeXL is called Fruchterman-Reingold [3]. In our example of the small friendship network, the Fruchterman-Reingold was not particularly useful as it included unnecessary edge crossings (i.e., times when two edges cross one another). Edge crossings can make it harder to track connections between vertices and in general should be minimized. Choose the Circle layout from the drop-down menu in the graph pane and choose Lay Out Again to see a more readable and appealing visual layout (Fig. 4.6).

Experimenting with different layout types (e.g., Sugiyama, Harel-Koren Fast Multiscale) can reveal useful patterns, relationships, or unusual features in the dataset being analyzed.

4.4.2 Directed Graph Type

In NodeXL, the default graph type is undirected, which means the relationship between Vertex 1 and Vertex 2 is symmetric. In the example with friendships, if Ann is a friend of Bob, Bob is also a friend of Ann. This is shown in the graph pane as an edge connecting Ann and Bob. The NodeXL ribbon allows you to specify the graph type as directed, which means that relationships go only in one direction. For example, the relationship might be that Ann invited Bob to a party. In this case, knowing who extended the invitation is important. For directed graphs, Vertex 1 is the source and Vertex 2 is the destination. An arrow is shown leading into the destination vertex (i.e., an arrow points from Ann to Bob). Click on Refresh Graph (in the graph pane) to display the directed network (Fig. 4.7) after choosing Type: Directed from the dropdown menu on the Graph section of the NodeXL Ribbon (Fig. 4.5).

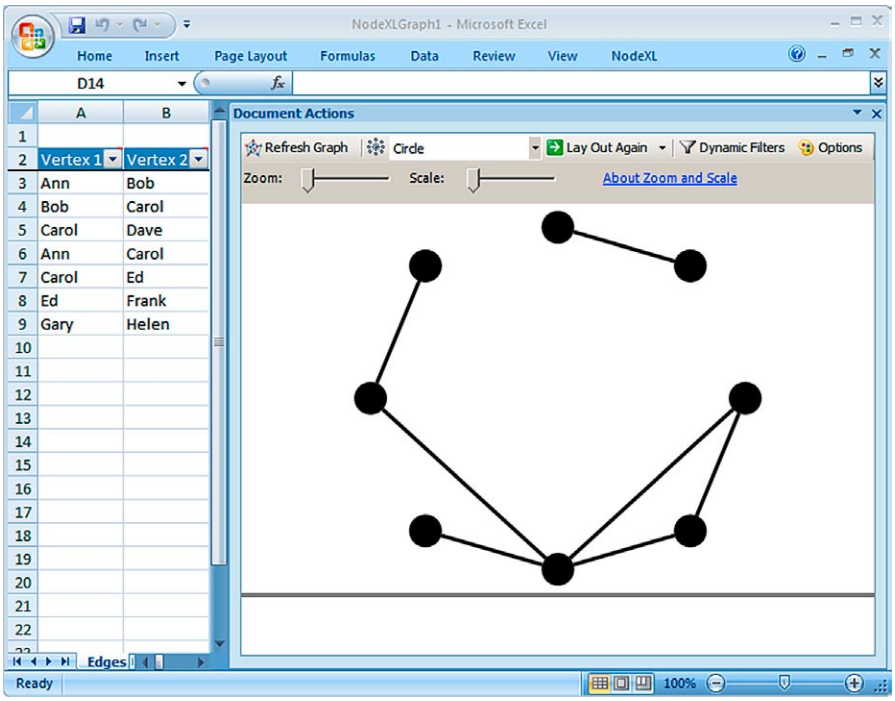


FIGURE 4.6 Vertices for eight friends arranged automatically using the Circle layout in NodeXL.

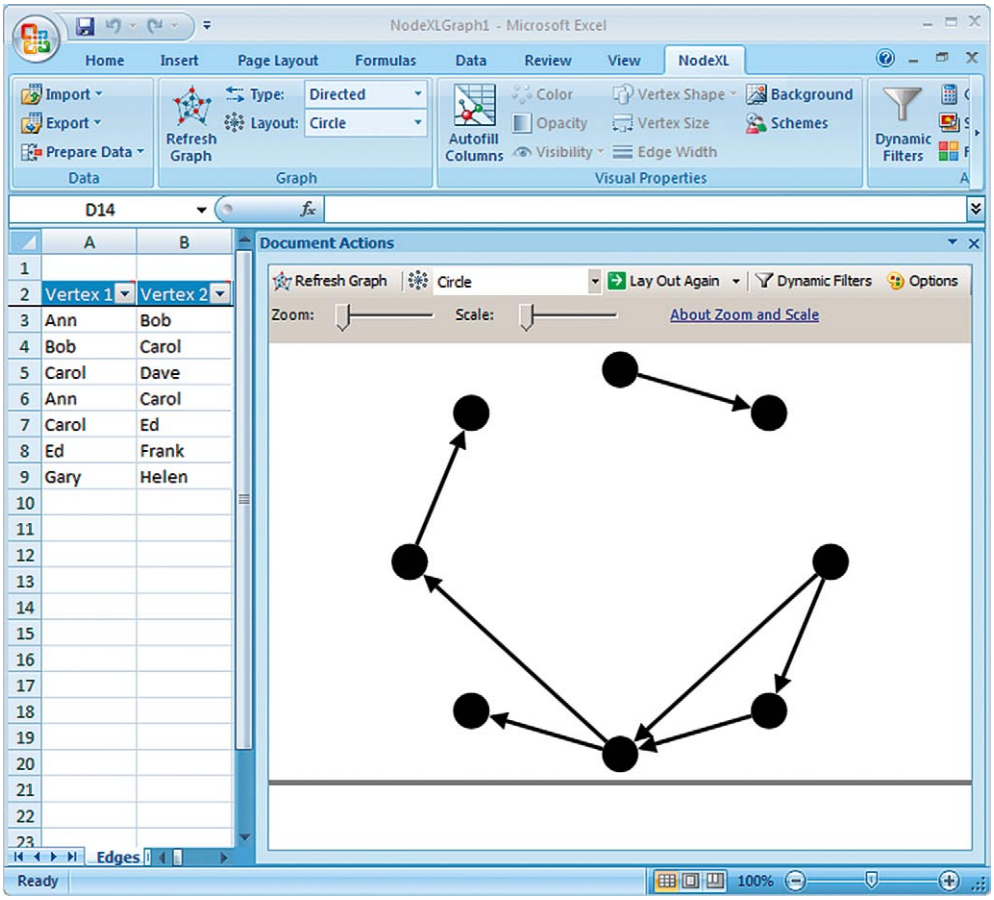


FIGURE 4.7 An invitation graph in NodeXL representing directed relationships shown as edges with arrows. For example, an arrow points from Ann (Vertex 1 column) to Bob (Vertex 2 column) indicating that she invited him to a party.

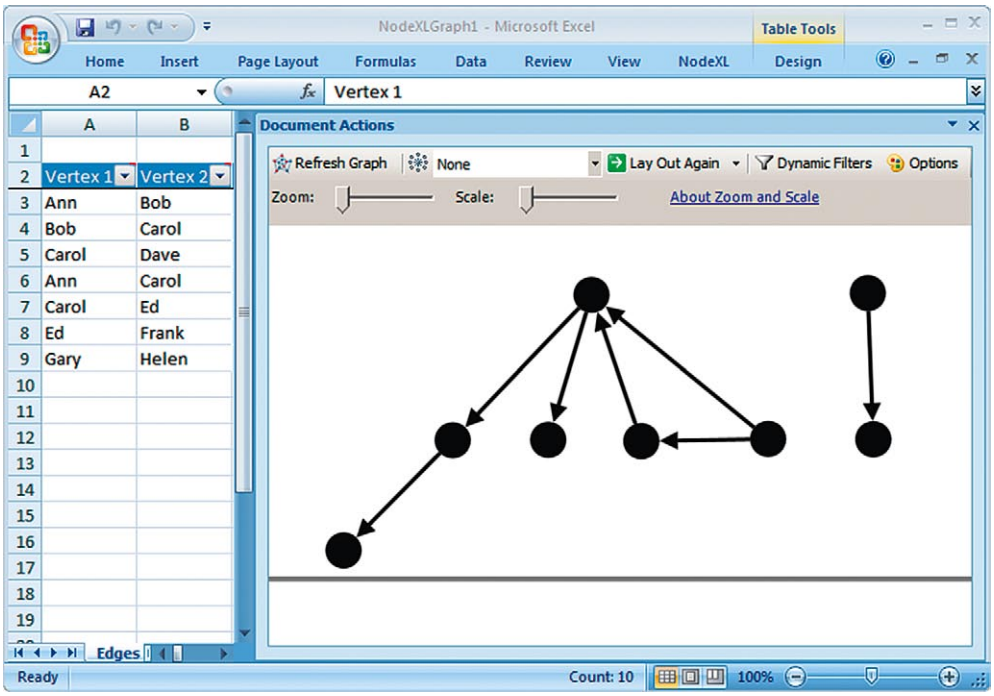


FIGURE 4.8 This manually arranged network graph in NodeXL shows two separated groups (often called components) and emphasizes the importance of Carol, who has given and received two invitations.

ADVANCED TOPIC
Fixing Vertex Placement

You can fix the placement of vertices so they do not change when you click on Refresh Graph, even if an automatic layout other than None is chosen. To do so, first highlight the vertices you want locked in place by selecting them in the graph pane. Next, right-click in the graph pane and choose Edit Selected Vertex Properties, and then choose Yes from the drop-down menu next to the word "Locked?" This will add information to a column in the Vertices worksheet called "Locked?" To view that column, click on the Workbook Columns button on the Show/Hide section of the NodeXL ribbon and check Layout from the list (see Fig. 4.16 presented later in the chapter). This will display the Layout related columns in the Vertices worksheet that are hidden by default. You will see the "Locked?" column that has been populated with "Yes" for each of the vertices you locked in place. You can manually change which vertices are locked by editing this information. You can also use the two columns labeled X and Y to fine-tune vertex placement if desired. For example, you could set the Y values of certain vertices to the same number to ensure that they line up perfectly.

4.4.3 Updating the Graph Pane

Any time you change the underlying data or features that affect the layout of the network (e.g., directed versus undirected), you *must* click on the Refresh Graph button to update the graph. If you just want to change the layout, you can select a new layout type and click on Lay Out Again to reduce processing time.

4.4.4 Manual Layout

In our example network of party invitations, you may want to manually position the vertices to gain a better understanding of the relationships. You can click and drag the vertices one at a time to create arrangements that emphasize structures or create a more orderly display (Fig. 4.8). You can even select multiple vertices by drawing a box around them or clicking on additional vertices while holding down the Control key. Multiple selected vertices move together when dragged.

4.4.5 Preserving Manual Layout

After working to get a layout that shows important relationships, you probably want to preserve that layout. In the layout selection menu choose "None," which keeps your manual layout, even after selecting Refresh Graph. Another more permanent method for fixing vertex placement is described in the Advanced Topic: Fixing Vertex Placement box.

4.4.6 Zooming and Scale

To get a closer look at a subsection of a graph, you can use the Zoom slider (or a mouse scrollbar in the graph pane). Once you are zoomed in, you can pan across the graph by holding down the spacebar, clicking the mouse button, and dragging the cursor in the direction you want to pan. You can also use the Scale slider to change the size of the vertices and edges all at once.

4.5 VISUAL DESIGN: MAKING NETWORK DISPLAYS MEANINGFUL

Graphs can reveal patterns, relationships, and interesting features that may be hard to spot in a tabular edge list. NodeXL lets you create a rich variety of possible drawings for a graph by changing visual attributes such as the color and size of edges and vertices.

4.5.1 Vertex Colors

You may want to change the colors of vertices. For example, in the party invitation graph, you can color vertices that represent men with blue and women with pink. Look at the worksheets on the lower left and click on the Vertices worksheet, which will bring up the list of eight vertices in our party invitation data set. The contents of the Vertices worksheet were generated automatically from the Edges data when you initially chose Show Graph. Type in the color you want for each person using standard CSS color names.[2] Alternatively, after selecting a person, click on the Color button in the NodeXL ribbon's Visual Properties section and select the color you want from the color palette. You can select multiple vertices using the Control or Shift keys and set all their visual properties together in this manner. Once you've populated the Color column, click on Refresh Graph to redisplay the graph pane (Fig. 4.9).

[2] www.w3schools.com/css/css_colornames.asp

4.5.2 Adding Descriptive Data

You may have additional information about the people in the data set (i.e., attribute data). You can add your own columns of data by typing or pasting it in from another source. To record the age of each person, scroll the Vertices worksheet to the right until you see the column header "Add Your Own Columns Here." Place the cursor on this header to get further instructions. If you select the next free column, you can type an attribute name (e.g., age) and then enter values for each person. Add two new columns, one for age and one for the number of prior parties the individual has attended since the beginning of the year (Fig. 4.10). By default, new columns are formatted as text. In this case, it makes sense to reformat the columns as General or Number before entering the numerical data. To more easily see

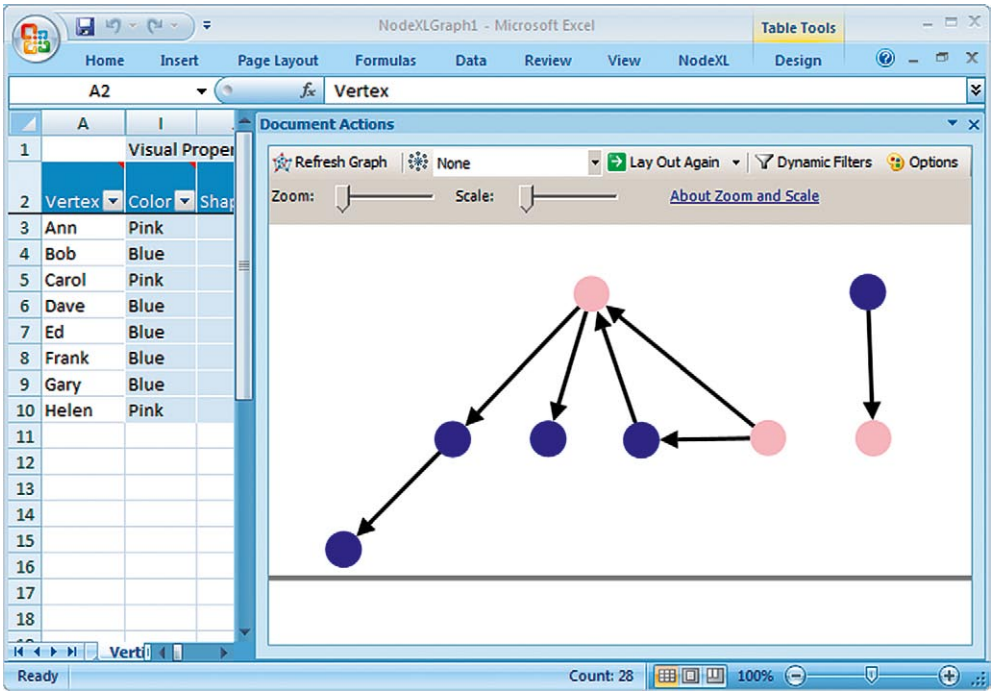


FIGURE 4.9 Color coding vertices in NodeXL helps quickly identify women (pink) and men (blue).

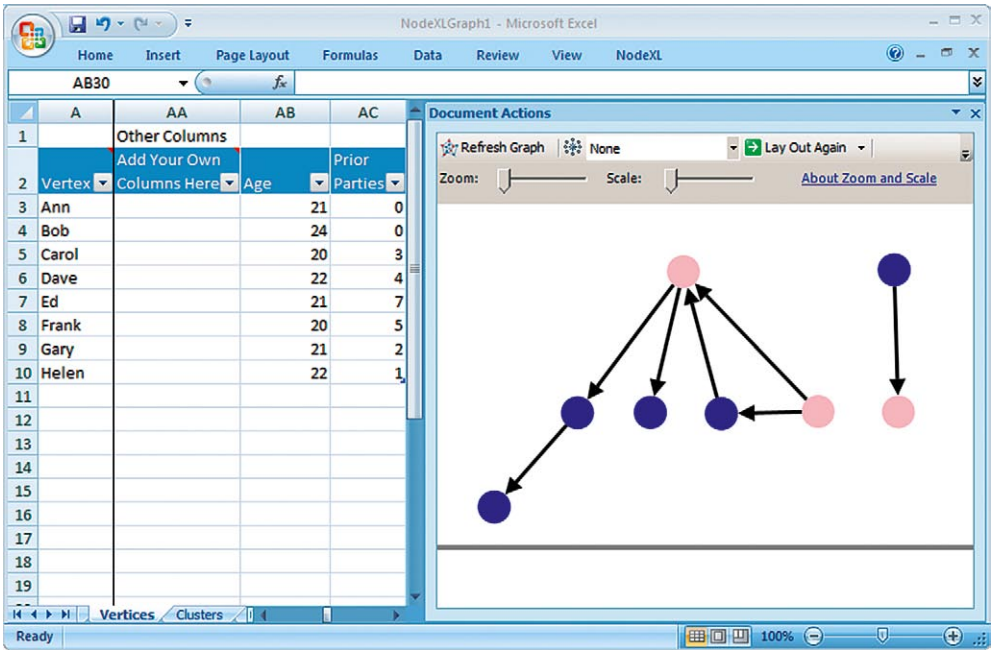


FIGURE 4.10 The NodeXL Vertices worksheet now includes user-supplied columns for age and number of prior parties.

how the attribute data maps to individuals, freeze the first column on the Vertex worksheet (the vertex names) so it always shows up by choosing Freeze Panes → Freeze First Column from the Excel 2007 View ribbon.

4.5.3 Changing Vertex Size (and Other Visual Properties)

Another visual property that can be used to represent attribute data is vertex size, which is controlled by the Size column in the Vertices worksheet. Put your cursor over the Size column header to show the type of data that must be entered—in this case numbers 1 through 10. Use this same approach to see what type of data to enter into any of the different Visual Properties fields such as Color, Shape, and Opacity (Fig. 4.11).

There are three ways to enter numbers into the Size column (or other visual attributes such as opacity or color): (1) you can manually type them into the Size column, (2) you can enter a formula into the Size column that calculates a number based on some other data (e.g., the Prior Party column you entered earlier), or (3) you can automatically populate the Size column based on data from another column (e.g., prior parties) by using NodeXL's Autofill Columns feature. Figure 4.12 shows the result of using the NodeXL Autofill Columns feature to automatically fill in the size numbers based on the prior parties data you entered earlier.

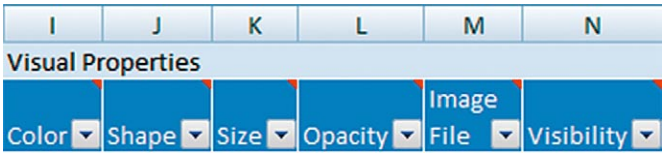


FIGURE 4.11 In NodeXL, vertices can have properties such as Color, Shape, Size, and Opacity.

4.5.4 Autofilling Columns

To recreate Figure 4.12, first click on the Autofill Columns button in the NodeXL ribbon. The resulting dialog window (Fig. 4.13) offers a set of drop-down boxes to allow you to select data you have entered in as additional fields or network metrics that can be calculated for each vertex (see Chapter 5). The dialog allows you to provide instructions on how NodeXL should fill in the worksheet columns such as those relating to size and shape. Click on the drop-down menu ▼ symbol next to Vertex Size to see all of the data columns available and choose Prior Parties. This tells NodeXL to base the Vertex Size on the data in the Prior Parties column. You can do the same for many other visual attributes of the vertices as well as the edges. Those associated with vertices populate columns in the Vertices worksheet, whereas those associated with edges populate columns on the Edges worksheet. When you click on the Autofill button at the bottom, the data in the visual attributes columns will be populated and the graph will be refreshed to reflect the changes.

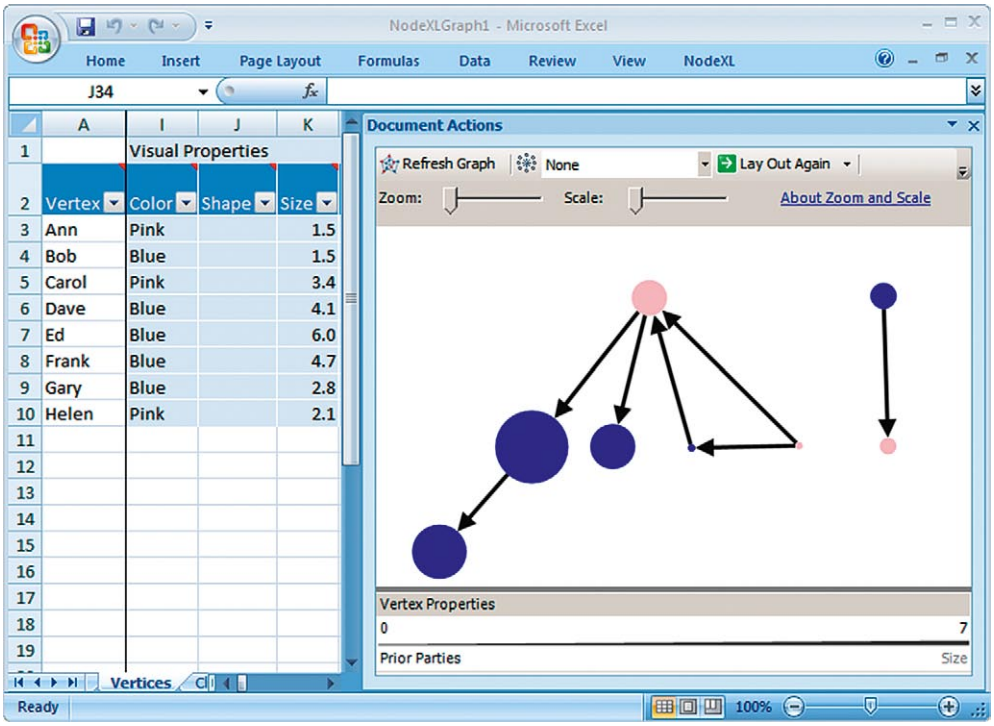


FIGURE 4.12 Vertex sizes have been populated using the NodeXL Autofill Columns feature based on the number of prior parties attended, revealing the wide disparity in social activity. The legend at the bottom of the graph pane shows the range of values for prior parties (0 to 7) and their mapping to size (vertex size of 1.5 to 6).

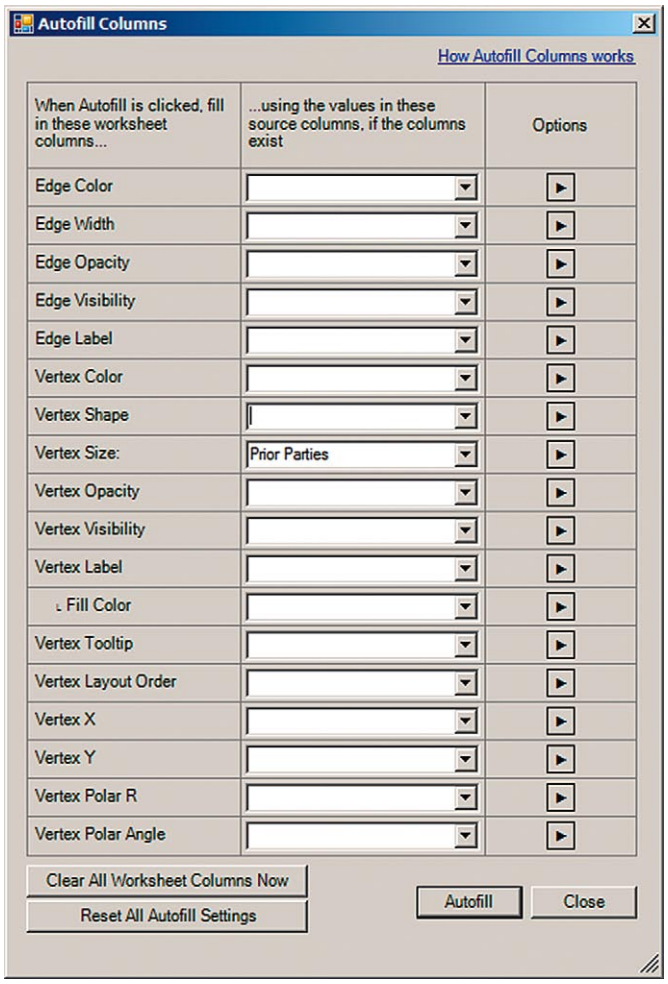


FIGURE 4.13 The Autofill Columns dialog used to set Vertex Size values based on the number of prior parties. Selecting the Autofill button populates the Size column and refreshes the graph.

Each attribute has an associated Options page that allows you to fine-tune some of the attributes. In our example, we want to ensure that the vertices are large enough to view well but not too large that they obscure other vertices. Click on the ▶ button in the Options column (Fig. 4.13) on the Vertex Size row and choose "Vertex Size Options…" to bring up a dialog (Fig. 4.14). Change the maximum vertex size to 6.0 and select OK and then Autofill to update the graph, which should now look like Figure 4.12. This will ensure that the maximum prior party value (which is 7) is mapped to the vertex size of 6.0. The Advanced Topic: Vertex Size Advanced Options box describes the other options available.

If you want to reset all of the options to their default settings, click on the Reset All Autofill Settings button in the Autofill Columns dialog (Fig. 4.13). This will not affect the data in the worksheet itself. The Autofill Columns dialog is merely instructions on how to populate the worksheet columns. To clear the actual worksheet data in a specific column such as the Size column on the Vertices worksheet, click on the Options arrow on the Vertex Size row and choose Clear Vertex Size Worksheet Column Now. You can use the Clear All Worksheet Columns Now button (bottom of Fig. 4.13) to delete content from all of the "autofillable" worksheet columns, regardless of whether they have been autofilled or filled manually. This should be used carefully, as you cannot undo these deletions. This will not delete the edges or vertices data (e.g., Vertex column data) or any data you have added in additional columns, but it will delete the other fields that can be populated using the Autofill Columns feature.

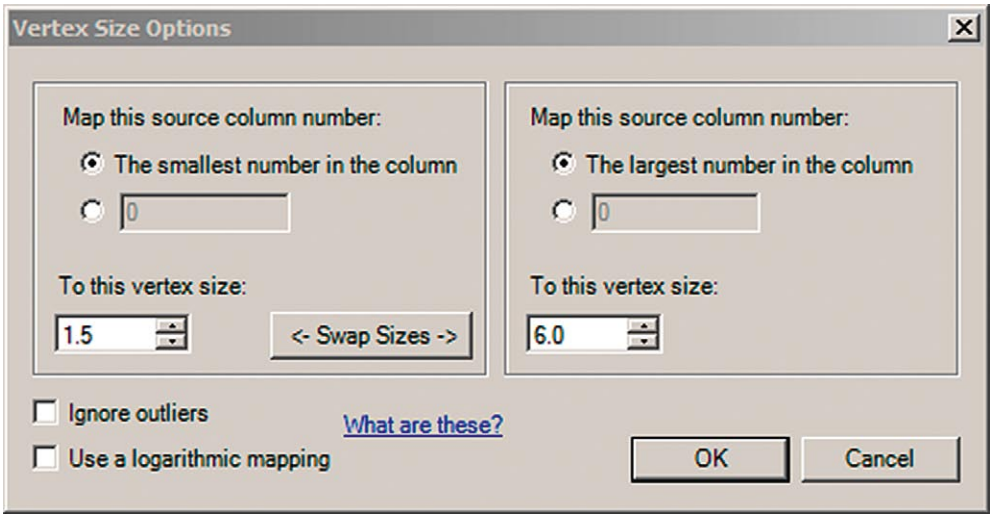


FIGURE 4.14 The NodeXL Vertex Size Options dialog allows you to set the range for sizes. Setting the range to be from 1.5 to 6.0 ensures that all vertices are visible and avoids overlap of vertices in the small network shown in Fig. 4.12.

ADVANCED TOPIC
Vertex Size Advanced Options

The Vertex Size Options dialogs (e.g., Fig. 4.14) provide several options for mapping attribute data to visual properties such as size. By default, attribute data (e.g., prior parties) will be linearly mapped so that the smallest number (0) corresponds with the vertex size indicated on the left-hand side (1.5) and the largest number (7) corresponds with the vertex size indicated on the right-hand side (6.0). Some data can be excluded from this mapping by manually setting a minimum or maximum data value to consider in the mapping. For example, values that are 0 or negative could be ignored by setting the "Map this source column number:" to 1 instead of "the smallest number in the column." Checking the "Ignore outliers" box will automatically identify outliers (i.e., numbers that are more than one standard deviation higher or lower than the average) when mapping numbers to the attribute. NodeXL also allows you to use a logarithmic mapping rather than the default linear mapping. This is useful when dealing with participation data that often follows a power law or logarithmic distribution.

4.5.5 Legend

Each time you use Autofill, NodeXL adds to the legend, which can be shown at the bottom of the graph pane. This legend helps you and other viewers understand the visual properties of the graph. In our example, the vertex size property was set by Autofill, so the legend shows the range of prior party values (0 to 7) and the size of the vertices that they map to (1.5 to 6) (Fig. 4.12). Because color settings were entered manually, these values do not show up in the legend. The only way to get categorical data to show up in the legend is to use the Categorized Graph Scheme (see Advanced Topic: Schemes).

4.5.6 Changing General Graph Appearance

Another way of setting visual features is to go to the graph pane and click on the Options button (or right click in the graph pane and select Options). This brings up the NodeXL graph pane Options dialog (Fig. 4.15), which offers controls for setting the default visual features for Vertices, Selected vertices, Edges, Selected edges, Fonts, Margins, and so on. Default visual properties (e.g., color, shape, opacity) will be superseded by data in the corresponding columns on the Vertices or Edges worksheets if they are populated.

4.6 LABELING: ADDING TEXT LABELS TO VERTICES AND EDGES

Textual labels can help clarify and add meaning to your graphs. In most cases, labels are applied to the vertices. For example, in a social network, people's names may be used as the labels. In a few cases, labels are applied to edges to describe the type of connections. For example, if a person delivered a package to another person, the directed edge connecting them could be labeled with the package number.

4.6.1 Adding Vertex Labels

NodeXL offers three ways to display vertex labels:

- *Labels as the shape.* The label text (e.g., vertex name) is surrounded by a rectangular box, which becomes the vertex. The Shape column must be set to Label for this to work.
- *Labels alongside a shape.* The label text appears alongside a specified shape such as the default disk shape. The cardinal and semicardinal position of

ADVANCED TOPIC
Schemes

NodeXL includes a tool called Schemes that you can access from the NodeXL ribbon's Visual Properties section (Fig. 4.5). Clicking on the Schemes button will open a dialog where you can easily set visual properties of a graph based on the type of data that you are using. It includes three options for schemes: Categorized Graph Scheme, Weighted Graph Scheme, and Temporal Graph Scheme. For an example of how to use the Schemes feature, see the discussion of Facebook in Chapter 11.

each vertex's label can be specified to reduce overlap and clutter.

- *Tooltip.* The tooltip text appears as a popup only when your cursor hovers over the vertex. This keeps the graph pane uncluttered while allowing you to see text associated with one vertex at a time. Tooltip text can be used in place of or in addition to labels.

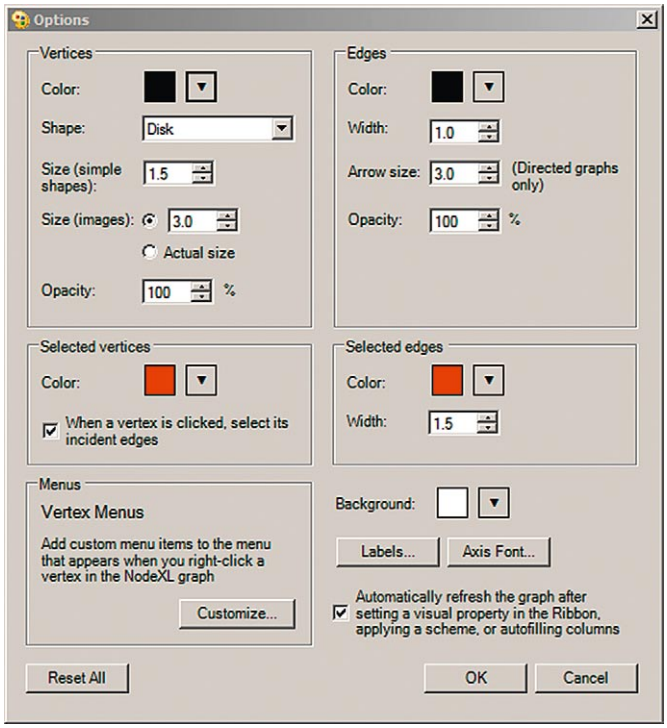


FIGURE 4.15 The NodeXL Graph Options dialog box shows current values for the visual properties of vertices and edges.

4.6.2 Viewing Hidden Columns

NodeXL stores many columns of data related to visual properties, labels, layout, and graph metrics, making navigation of the worksheets challenging. As a result, many columns are hidden by default. To show or hide these columns, select the Workbook Columns button in the Show/Hide group of the NodeXL ribbon (Fig. 4.5) and check the columns you want displayed (Fig. 4.16). Check the Labels box to unhide the relevant columns.

4.6.3 Adding Labels as the Shape

You can use the Autofill Columns feature to fill the Label column with the names from the Vertex column, or any other column you desire. Alternatively, you can copy and paste data directly into the Label field in the

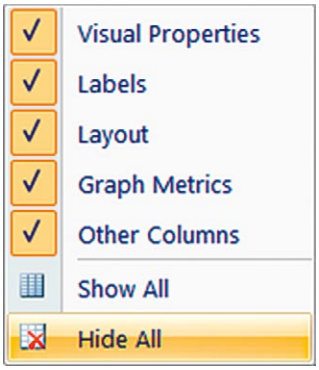


FIGURE 4.16 Groups of related NodeXL workbook columns can be shown or hidden by checking and unchecking the appropriate boxes.

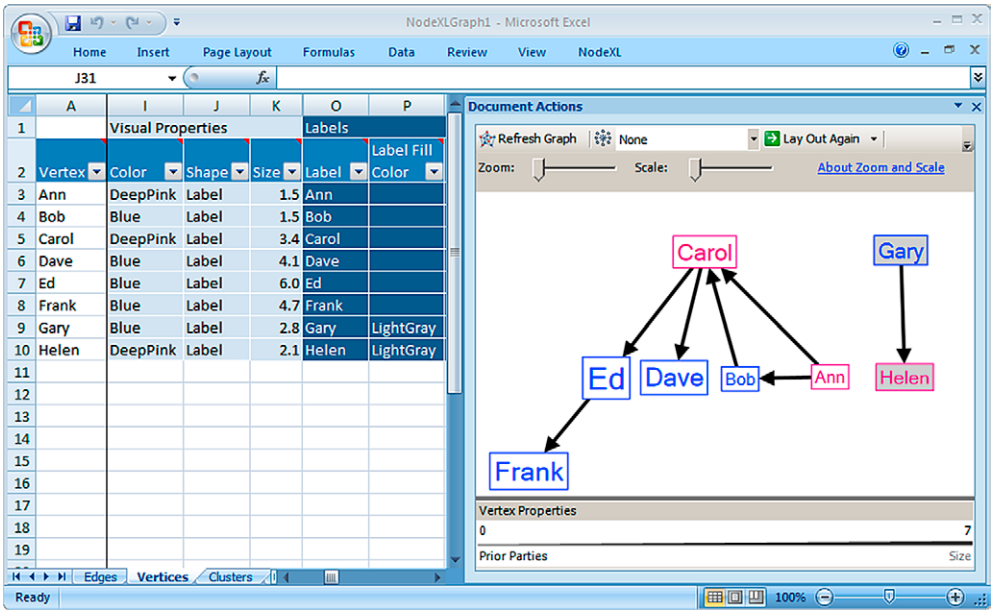


FIGURE 4.17 Labels are used as the shape after using Autofill columns to populate the vertex names in NodeXL. The label fill color is manually set to light gray to clearly show the separation of Gary and Helen from the rest of the group.

ADVANCED TOPIC
Adding Images as Vertices

NodeXL allows you to use images as vertices. There are two steps involved. First, set the Shape column on the Vertices worksheet to Image. Next, populate the Image File column with the image's full file path, relative file path, or a URL to an image on the Internet. Many social media systems have user profile images that can be used in place of usernames or images for blogs or forums. See Chapters 10, 13, and 14 for examples.

Vertex worksheet or enter formulas that combine other fields (see Advanced Topic: Adding Images as Vertices). Once you've populated the Label field, choose Label from the drop-down menu in the Shape column (Fig. 4.17). The background color of the rectangular shape surrounding the label can be changed by choosing a label fill color. For example, you can set the background color to light gray for Gary and Helen to emphasize their separation from the other group.

The other visual attributes including size, color, and opacity will continue to apply. Notice, for example, that Ed shows up larger than all other vertices. Sometimes lighter colors such as pink make the text hard to read, so you may want to change them to a darker color such as deep pink.

4.6.4 Adding Labels Alongside a Shape

You can show labels alongside another shape instead, allowing for more clarity when comparing attributes such as size. To re-create Figure 4.18, delete the content in the Shape column (or set them to Disk) and keep the information in the Label column. At first, you may have labels that overlap edges or other vertices. To reduce this problem, you can set the label position on each vertex (Fig. 4.17). Note that the Label Fill Color option is no longer applicable because the label is not chosen as the shape and there is no rectangle surrounding the labels.

4.6.5 Adding Tooltips

You can also add data that only show up when you mouse over a vertex. This is called a tooltip. In Figure 4.18, the Autofill Columns feature has been used to associate the Tooltip column with the Age column. Hovering over Helen's vertex with the mouse pointer will show her age (22 in this case).

4.6.6 Adding Edge Labels

Adding edge labels is relatively simple. There is a Label column on the Edges worksheet where you can manually add labels. You can also use the Autofill feature or formulas to fill the content of this column. In Figure 4.19 information about the medium through which the invitation was extended is included in the Label column.

ADVANCED TOPIC
Using Formulas to Create Labels

Excel has many formulas in the Text category that can be useful when creating labels. If you are going to use a formula, you will need to first reformat the column as "General" instead of "Text," otherwise the formula you enter will be displayed instead of the result of the formula. After changing to General format, you may need to go to the cell you want updated, click inside the Formula bar, and hit Enter to make the change go into effect. One useful trick is to combine information from multiple fields, such as a person's name (Vertex column) and attribute data (Age column). The following formula uses the "&" symbol to combine the Vertex and Age data and the " " symbols to add a space between them:

```
= Vertices[[#This Row],[Vertex]]&"
"&Vertices[[#This Row],[Age]]
```

The result for the first row would be "Ann 21."

If names are too long, you can use the LEFT function to only include a specified number of characters. The following formula can be used to include the first six characters of the Vertex column:

```
= LEFT(Vertices[[#This Row],[Vertex]],6)
```

Additional text functions such as TRIM, UPPER, PROPER, LOWER, and CLEAN can also be used to ensure consistency and clean data.

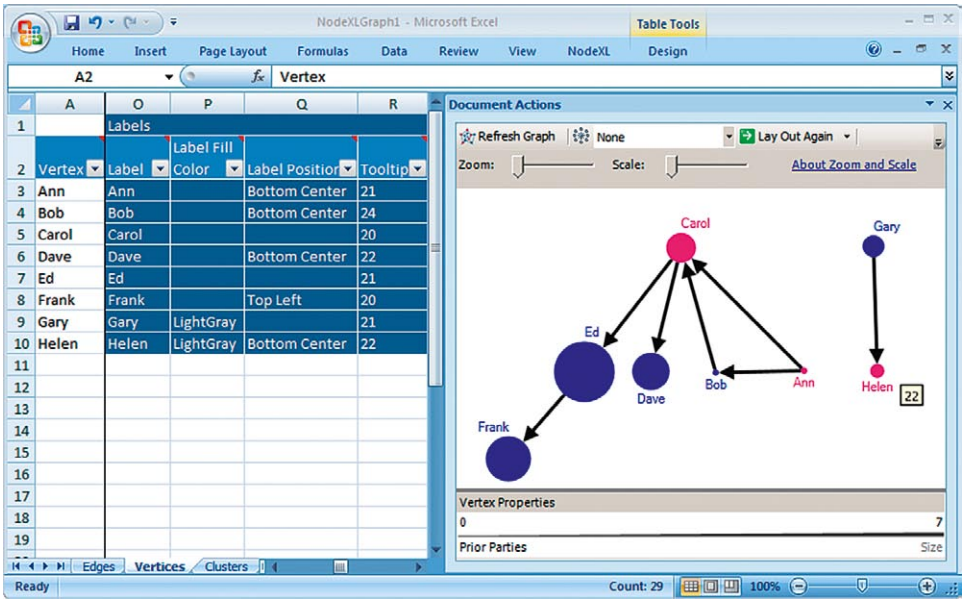


FIGURE 4.18 NodeXL can display labels outside the vertices, making the size information more easily comparable. Labels are positioned so they don't overlap with edges. Helen's tooltip of 22 (her Age) is shown when hovered over.

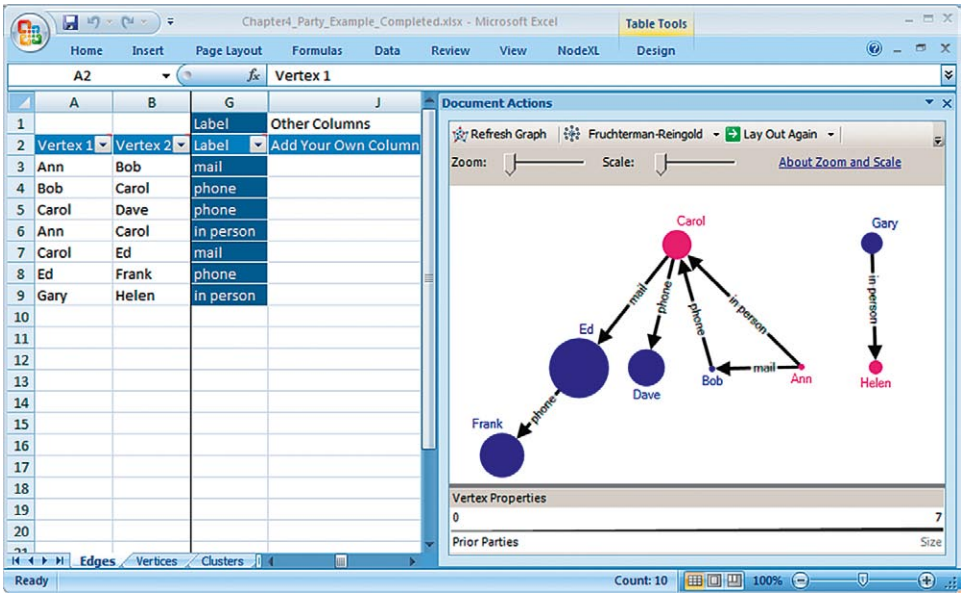


FIGURE 4.19 Edge labels, entered into the Label column on the Edges worksheet, indicate the medium through which a party invitation was extended (e.g., phone, mail, or in person).

4.6.7 Saving a NodeXL File

You are now done with the party example. To save the NodeXL file, save it as you would any other Excel file making sure to select the standard Excel Workbook (with a .xlsx extension). Do not save it as an Excel 97-2003 Workbook, a Macro-Enabled Workbook, or a Binary Workbook.

4.7 PRACTITIONER'S SUMMARY

NodeXL is an Excel 2007 template that supports network analysis and visualization in a spreadsheet environment. This chapter introduced the basics of NodeXL using a simple friendship network. The spreadsheet includes an Edges worksheet that contains a row for each edge and a Vertices worksheet that contains a row for each vertex. Attribute data that describes edges or vertices can be added and automatically mapped to visual attributes such as size, color, and opacity using the Autofill Columns feature. Vertices can be arranged on the graph using a number of automatic layouts or can be manually positioned and fixed into place if desired. Labels can be used in place of other vertex shapes or in addition to them. Vertex label position and color can be customized to ensure clarity and readability. The Tooltip

Label feature can be used to reveal additional data when a vertex is hovered over with the mouse pointer. Edge labels can be added to indicate the type of connection. The rich array of visual attributes will be particularly powerful when combined with network metrics as discussed in the next chapter.

4.8 RESEARCHER'S AGENDA

Designers of network-drawing programs must make many decisions as they develop their tools [4, 5]. Although traditional aesthetic measures have emphasized making an appealing visual display and network layout [1, 6–11], the contemporary view is more guided by user performance on benchmark tasks, such as comparing the degree of two vertices, identifying common properties of vertices, or spotting missing relationships [12–15]. More sophisticated tasks such as finding cliques, cut vertices, or chains are also being used in studies. Increasing attention is given to avoiding overlap, making labels readable, enabling users to follow edges from vertex to vertex, and showing connected components while pushing isolated vertices and small components to the side. The research questions become more challenging as the size and complexity of the network grow, leaving designers and users to make tradeoffs that may facilitate some tasks while making others more difficult [16]. Even more challenging is to have the visual design and network layout be done automatically while producing an effective presentation [14, 17]. Fortunately, these topics are an active area of research, which is leading to clarification of what primary tasks users have for directed/undirected networks that emerge from differing application domains. User control is emerging as the solution, but that requires giving users a clear understanding of the implications of each choice.

References

[1] M. Smith, B. Shneiderman, N. Milic-Frayling, E. Mendes-Rodrigues, V. Barash, C. Dunne, et al., Analyzing (social media) networks with NodeXL, in: Proc. Communities & Technologies Conference, State College, PA, June 2009.

[2] R. Hanneman, M. Riddle, Chapter 5: Using Matrices to Represent Social Relations, in: Introduction to Social Network Methods, University of California, Riverside, Riverside, CA (published in digital form at http://faculty.ucr.edu/~hanneman)

[3] T.M.J. Fruchterman, E.M. Reingold, Graph drawing by force-directed placement, Software Pract. Ex. 21 (11) (1991) 1129–1164 .

[4] G.D. Battista, P. Eades, R. Tamassia, I.G. Tollis., Graph Drawing: Algorithms for the Visualization of Graphs, Prentice Hall, New York, 1998.

[5] C. Ware, Visual Thinking for Design, Morgan Kaufmann Series in Interactive Technologies, Morgan Kaufmann, Burlington, MA, 2008.

[6] M.K. Coleman, D.S. Parker, Aesthetics-based graph layout for human consumption, Software Pract. Ex. 26 (12) (1996) 1415–1438.

[7] T. Dwyer, B. Lee, D. Fisher, K.I. Quinn, P. Isenberg, G. Robertson, et al., A comparison of user-generated and automatic graph layout, IEEE Trans. Vis. Comput. Graph. 15 (6) (Nov/Dec 2009) 961–968.

[8] W. Huang, S.-H. Hong, P. Eades, How people read sociograms: a questionnaire study, in: APVis '06: Proc. 2006 Asia-Pacific Symposium on Information Visualisation, 2006, pp. 199–206.

[9] W. Huang, S.-H. Hong, P. Eades, Effects of sociogram drawing conventions and edge crossings in social network visualizations, J. Graph Algorithms Appl. 11 (2) (2007) 397–429.

[10] H.C. Purchase, The effects of graph layout, in: OZCHI '08: Proc. 2008 Australasian Computer Human Interaction Conference, Nov.-Dec. 1998, pp. 80–86.

[11] H.C. Purchase, D. Carrington, J.-A. Allder, Empirical evaluation of aesthetics-based graph layout, Empirical Software Eng. 7 (3) (2002) 233–255.

[12] P. Eades, K. Sugiyama, How to draw a directed graph, J. Inf. Proc. 13 (4) (1990) 424–437.

[13] W. Huang, Using eye tracking to investigate graph layout effects, in: APVis '07: Proc. 2007 Asia-Pacific Symposium on Information Visualisation, 2007, pp. 97–100.

[14] B. Shneiderman, A. Aris, Network visualization with semantic substrates, in: IEEE Symposium on Information Visualization and IEEE Trans. Vis. Comput. Graph. 12 (5) (2006) 733–740.

[15] F. van Ham, B.E. Rogowitz, Perceptual organization in user-generated graph layouts, IEEE Trans. Vis. Comput. Graph. 14 (6) (2008) 1333–1339.

[16] C. Ware, H.C. Purchase, L. Colpoys, M. McGill, Cognitive measurements of graph aesthetics, Inf. Vis. 1 (2) (2002) 103–110.

[17] T. Dwyer, Y. Koren, K. Marriott, IPSep-CoLa: An incremental procedure for separation constraint layout of graphs, IEEE Trans. Vis. Comput. Graph. 12 (5) (2006) 821–828.

NodeXL Papers

Bonsignore, E.M., Dunne, C., Rotman, D., Smith, M., Capone, T., Hansen, D.L., et al. (2009). First steps to NetViz Nirvana: Evaluating social network analysis with NodeXL. In *Proceedings. IEEE international symposium on social intelligence and networking (SIN-09)* August, Vancouver B. C., Canada.

Hansen, D.L., Rotman, D., Bonsignore, E.M., Milic-Frayling, N., Mendes-Rodrigues, E., Smith, M., et al. (2009). Do you know the way to SNA?: A process model for analyzing and visualizing social media data, Technical Report, University of Maryland.

Smith, M., Shneiderman, B., Milic-Frayling, N., Mendes-Rodrigues, E., Barash, V., Dunne, C., et al. (2009). Analyzing (social media) networks with NodeXL. In *Proceedings Communities & Technologies Conference,* June 2009 State College, PA.

CHAPTER

5

Calculating and Visualizing Network Metrics

OUTLINE

5.1 Introduction 69

5.2 Kite Network Example 70
5.2.1 Opening an Existing NodeXL File 70

5.3 Computing Graph Metrics 70
5.3.1 Vertex-Specific Metrics 71
5.3.2 Eigenvector Centrality 72
5.3.3 Overall Graph Metrics 73

5.4 *Les Misérables* Co-Appearance Network 74
5.4.1 Sorting the Edge Weight Column 74
5.4.2 Visualizing Edge Weights 75
5.4.3 Calculating and Visualizing Vertex Metrics to Find Important Individuals 75
5.4.4 Mapping Graph Metrics to X and Y Coordinates 77
5.4.5 Viewing and Hiding Graph Elements Such as Axes 78

5.5 Practitioner's Summary 78

5.6 Researcher's Agenda 78

5.1 INTRODUCTION

When trying to understand networks, analysts often want to identify important vertices, locate subgroups, or get a sense of how interconnected a network is compared to other networks. Although visualization itself can help do this, it is often helpful to use the rich set of quantitative network metrics, also called network graph metrics, which have been developed by social network analysis researchers (Chapter 3).

Network graph metrics can describe an entire network or characterize subgroups or specific actors within a single network. Aggregate graph metrics such as network density can be used to systematically compare communities, helping analysts decide which communities are highly connected and which are sparse. Tracking aggregate graph metrics over time can determine the effectiveness of interventions on the network as a whole. For example, you would expect the total number of edges to grow, increasing the "density" of the graph, after a photo sharing activity designed to introduce people to those they don't know.

Individual person-level metrics provide insights about a person's position within the network, helping to identify important or "central" people. For example, network graph metrics can be used to identify those people in a network who are bridge spanners or who are popular. Once identified, analysts and managers can better know who to contact or influence or bring to the table when trying to implement new programs or gain broader understanding. Metrics can also be used to identify cliques or persistent social roles that show up in many communities. Understanding the mix of social roles that exist within a particular network can help analysts determine if they have a healthy mix of social types or who may be a good candidate to replace an outgoing leader.

NodeXL calculates several network graph metrics. Once calculated, you can use these metrics to change the visual display of your network graphs in powerful ways

as shown in this chapter. You can also filter out vertices or edges based on network metrics as discussed later (Chapter 6).

5.2 KITE NETWORK EXAMPLE

To better understand the meaning of each graph metric, you can explore the widely used Kite network, created by David Krackhardt.[1] You can download the Kite_Example.xlsx file from the book web site or you can manually reproduce the undirected edge list and graph shown in Figure 5.1 in a new NodeXL template. The download version fixes the vertex positions to match those found in Figure 5.1.

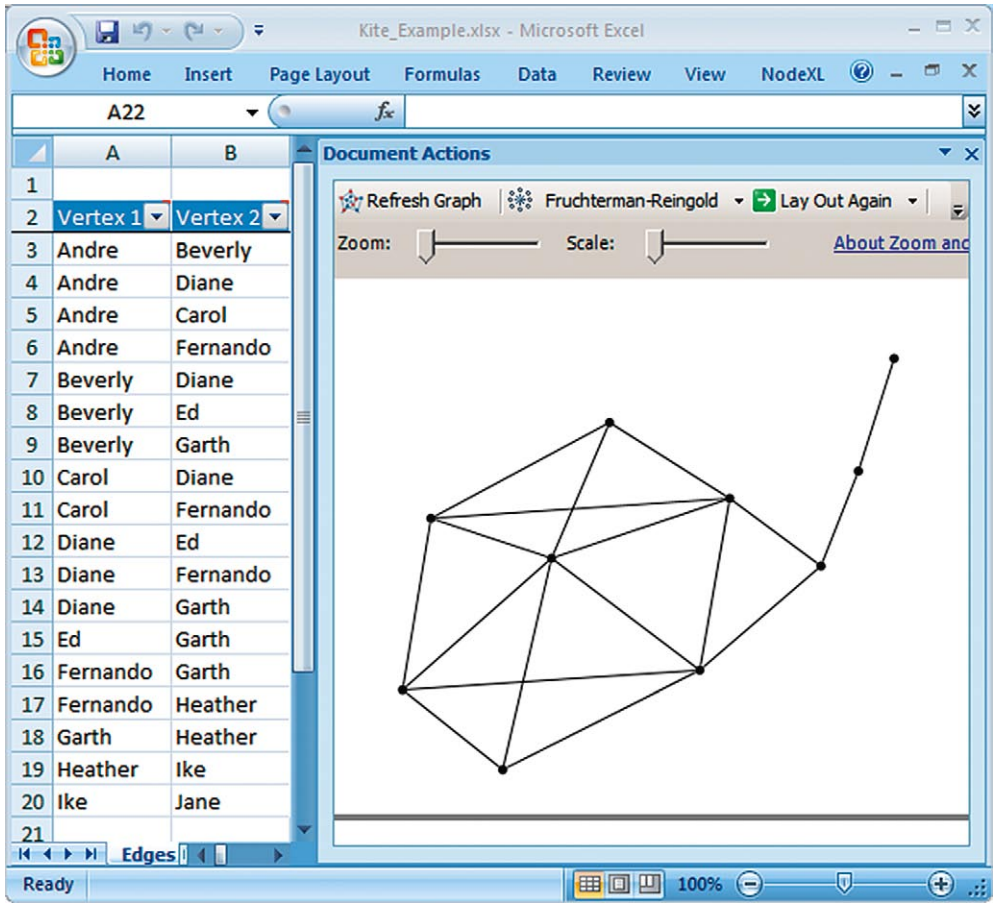
5.2.1 Opening an Existing NodeXL File

You can open a NodeXL file just like any other Excel file. If NodeXL is installed on a machine, Excel will recognize any file created using NodeXL even though it has the standard, xlsx extension. Because of the frequent NodeXL updates and differences in local computer configurations, you may need to choose the option "From NodeXL Workbook Created on Another Computer…" from the Import menu on the NodeXL ribbon.

Opening the Kite_Example.xlsx file will automatically launch NodeXL. Once you have opened the file, select Show Graph. Because the X and Y coordinates of each vertex is locked in place (see Advanced Topic: Fixing Vertex Placement in Chapter 4), the layout algorithm won't affect placement of the vertices.

5.3 COMPUTING GRAPH METRICS

To calculate graph metrics, first click on the Graph Metrics button on the Analysis section of the NodeXL ribbon. This opens the Graph Metrics dialog (Fig. 5.2). Select the metrics you want to calculate by checking in the boxes next to them. Clicking on the Details link next to a metric provides a brief explanation of that metric. Click on the Select All button, and then choose Compute Metrics. Some of the graph metrics can take a while to calculate when working with large networks,



	A	B
1		
2	Vertex 1	Vertex 2
3	Andre	Beverly
4	Andre	Diane
5	Andre	Carol
6	Andre	Fernando
7	Beverly	Diane
8	Beverly	Ed
9	Beverly	Garth
10	Carol	Diane
11	Carol	Fernando
12	Diane	Ed
13	Diane	Fernando
14	Diane	Garth
15	Ed	Garth
16	Fernando	Garth
17	Fernando	Heather
18	Garth	Heather
19	Heather	Ike
20	Ike	Jane
21		

FIGURE 5.1 The Kite network shown with an undirected edge list and manually created layout.

[1] www.orgnet.com/sna.html

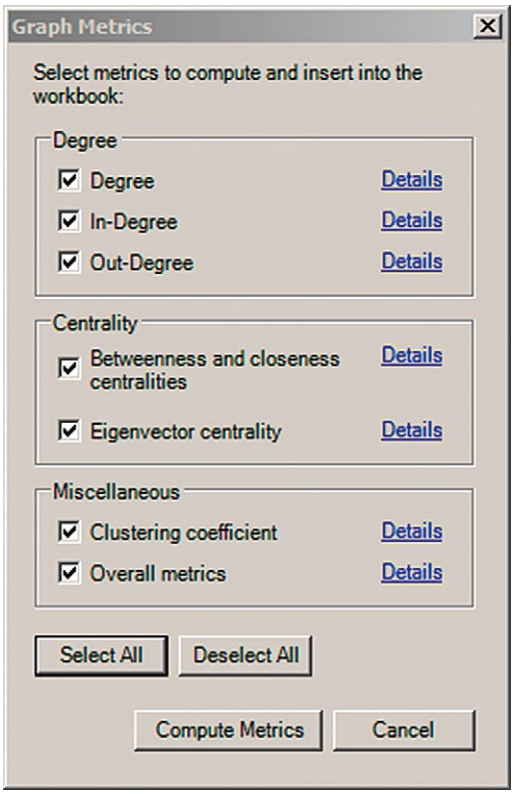


FIGURE 5.2 TheNodeXL Graph Metrics dialog with all metrics selected.

so a status bar is used to show progress. Newer versions of NodeXL have dramatically improved the calculation time. Once completed, NodeXL displays each vertex-specific metric in a set of Graph Metrics columns in the Vertices worksheet. In this case, because the graph is undirected, the In-Degree and Out-Degree metrics will not be calculated even though checked. NodeXL also populates the Overall Metrics worksheet that shows summary information for the entire network.

5.3.1 Vertex-Specific Metrics

To see the vertex-specific metrics such as centrality measures and clustering coefficients, start by going to the Vertices worksheet. You will see the new Graph Metrics columns, which can be hidden later if desired by unchecking Graph Metrics from the Workbook Columns button on the NodeXL ribbon (Fig. 4.17). Each value relates directly to one of the vertices. For example, row 3 shows the graph metrics that are specific to Andre (Fig. 5.3).

Vertex metrics can be mapped onto visual attributes (Fig. 5.3), which you can recreate by using the Autofill Columns feature found in the NodeXL Visual Properties menu ribbon. The graph legend shows that degree is mapped to Size (1.5 to 6) and betweenness centrality is mapped to Opacity (50 to 100). In addition, closeness centrality is mapped to the tooltip and the labels are set

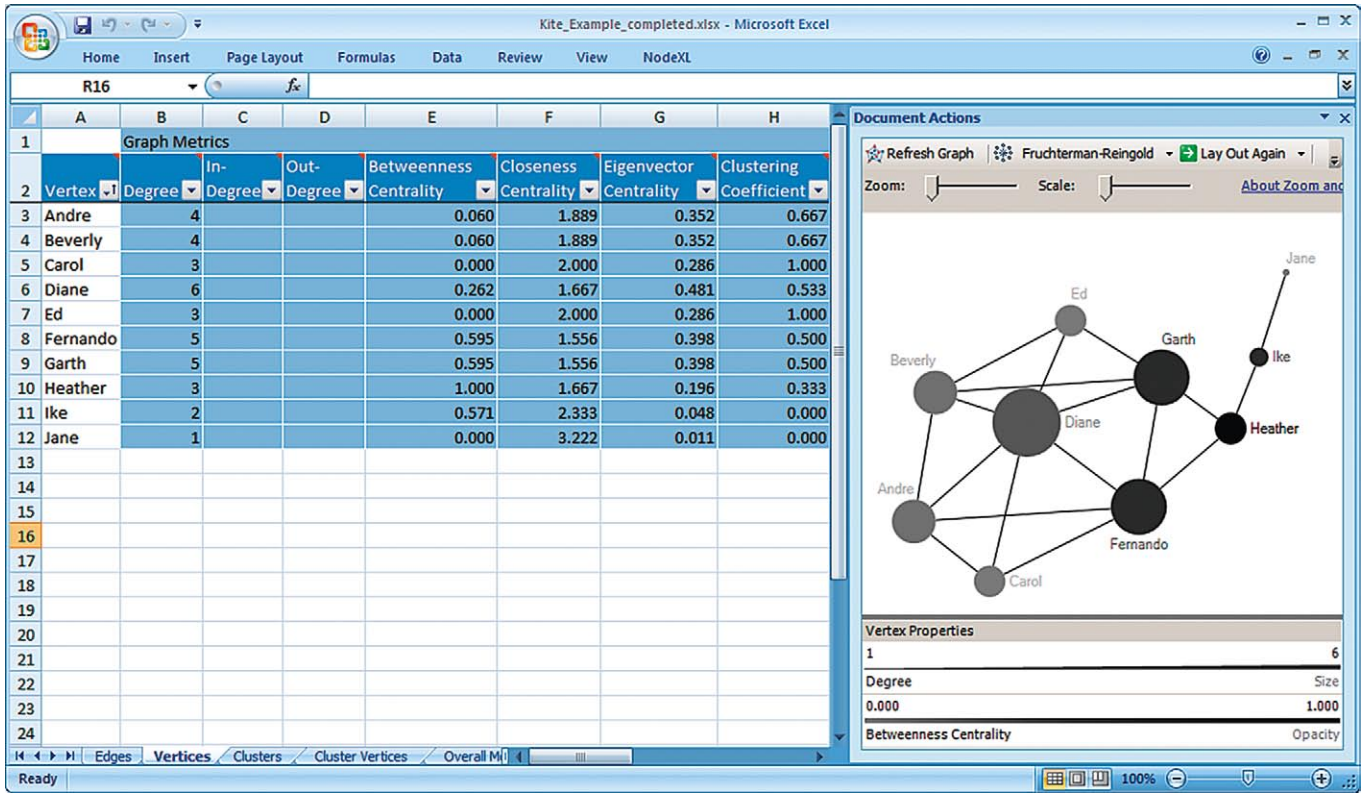


FIGURE 5.3 The Kite network showing graph metrics for each vertex. Degree is mapped to size (1.5 to 6), Betweenness Centrality (50 to 100) is mapped to opacity, and Closeness Centrality is set as the tooltip.

to Vertex and positioned so they don't cross edges. A description of each metric and how it relates to the Kite network follows.

Degree

The Degree of a vertex (sometimes called Degree Centrality) is a count of the number of unique edges that are connected to it. Diane has a Degree of 6 because she is directly connected to six other individuals. In comparison, Jane has a Degree of only 1 because she is connected to only one other person. If the edges represented the strong friendship ties of individuals in a class, we might say that Diane is the most popular person in the class and Jane is the least popular. The legend in Figure 5.3 shows the range of the Degree (1 to 6) mapped onto size. The size of the vertices has been set using the Autofill Columns feature to a range of 1.5 to 6 so the vertices are clearly visible, but not too large (see Chapter 4). If you were analyzing a directed graph (such as the Party Invitation network), the single Degree metric would be split into two metrics: (1) In-Degree, which measures the number of edges that point toward the vertex of interest (i.e., number of people that have invited the person to the party), and (2) Out-Degree, which measures the number of edges that the vertex of interest points toward (i.e., the number of people the person has invited to the party). In this case, NodeXL only calculated Degree since the network was specified as containing undirected ties.

Betweenness Centrality

Although popularity is important, it is not everything. Consider Heather in the Kite network. She is directly related to only three other people (i.e., she has a Degree of 3). Despite her relatively low Degree, her position as a "bridge" between Ike (and indirectly Jane) to the rest of the group may be of utmost importance. If, for example, information were passed from one person to another, Heather would be vital for assuring that Ike and Jane could communicate with the rest of the group. In fact, if she were removed from the network, Ike and Jane would be disconnected from the other class members. Thus, Heather has high Betweenness Centrality. In contrast, Ed has a Betweenness Centrality of 0. Notice that if he were removed from the graph, everyone would still be connected to everyone else, and their shortest communication paths would not even be altered. More generally, vertices that are included in many of the shortest paths between other vertices (called geodesic distances) have a higher Betweenness Centrality than those that are not included on such paths. In Figure 5.3 the legend shows that the Autofill Columns feature has set the opacity of each vertex to the Betweenness Centrality metric, which ranges from 0 (Ed and Carol who show up lighter) to 1 (Heather who shows up darkest). To make sure each vertex is visible, the minimum opacity was set to 50 and maximum was kept at 100.

Closeness Centrality

Another characteristic you may care about is how close each person is to the other people in the network. If information needed to flow through the network, some people would be able to get a message to all the other people relatively quickly (i.e., in few steps), whereas others may require many steps. Closeness centrality is a measure of the average shortest distance from each vertex to each other vertex. Unlike other centrality metrics, a lower Closeness Centrality score indicates a more central (i.e., important) position in the network in version 1.0.1.113 of NodeXL. In more recent versions of NodeXL the inverse is used so that higher values indicate more central people. The lowest possible Closeness Centrality score is equal to 1, which would indicate a person is directly connected to everyone in the network. In the Kite network, Fernando and Garth have the lowest Closeness Centrality measure, suggesting that they may be in a good position to initiate the spread of information through the network. In Figure 5.3 the Autofill Columns feature was used to set the tooltip to the Closeness Centrality metric so if you were to hover your mouse pointer over Ed, for example, a 2 will appear. Stanley Milgram (see Chapter 3) explored the idea that the average Closeness Centrality between any two people on Earth is equal to six; his work is the motivation for the idea of "six degrees of separation."

5.3.2 Eigenvector Centrality

In many cases, a connection to a popular individual is more important than a connection to a loner. The Eigenvector Centrality network metric takes into consideration not only how many connections a vertex has (i.e., its Degree), but also the Degree of the vertices that it is connected to. Both Heather and Ed have a Degree of 3. However, Ed is directly connected to Diane, the most popular person in the class, whereas Heather is connected to Ike, who is among the least popular. This explains why the Eigenvector Centrality metric for Heather is lower than it is for Ed (Fig. 5.3).

Clustering Coefficient

In some cases, a person's friends may be friends with each other. For example, Ed's three friends Beverly, Diane, and Garth are all directly connected to one another, creating a clique. More generally, a *clique* or *complete graph* occurs when all vertices in a group are directly connected to each other. In other cases, a person's friends may not be friends with one another. For example, Ike's two friends, Heather and Jane, are not friends with each other. The clustering coefficient measures how connected a vertex's neighbors are to one another. More specifically, it is the number of edges connecting a vertex's neighbors divided by the total number of possible edges between

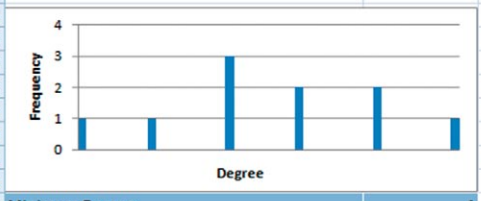
	A	B	C
1	Metric	Value	Comments
2	Graph Type	Undirected	
3			
4	Vertices	10	
5			
6	Unique Edges	18	
7	Edges With Duplicates	0	
8	Total Edges	18	
9			
10	Self-Loops	0	
11			
12	Connected Components	1	
13	Single-Vertex Connected Components	0	
14	Maximum Vertices in a Connected Component	10	
15	Maximum Edges in a Connected Component	18	
16			
17	Maximum Geodesic Distance (Diameter)	4	
18	Average Geodesic Distance	1.98	
19			
20	Graph Density	0.4	
21			
22	NodeXL Version	1.0.1.113	
23			
24			
25			
26			
27			
28			
29			
30			
31			
32			
33	Minimum Degree	1	
34	Maximum Degree	6	
35	Average Degree	3.60	
36	Median Degree	3.50	

FIGURE 5.4 The NodeXL Overall Metrics worksheet showing aggregate graph metrics for the Kite network and the frequency distribution of the vertex-specific metric degree.

the vertex's neighbors. For example, Heather's three neighbors are Fernando, Garth, and Ike. Only one connection exists between any of them (the connection between Fernando and Garth). There are three possible connections (Fernando-Garth; Fernando-Ike; Garth-Ike). Thus, the clustering coefficient for Heather is 1/3.

5.3.3 Overall Graph Metrics

Go to the Overall Metrics worksheet, which summarizes some of the key properties of the entire network (Fig. 5.4). These metrics include the following:

- *Graph type*. Undirected or directed.
- *Vertices*. The number of total vertices (i.e., rows on the Vertices worksheet).
- *Unique edges*. The number of unique edges found in the Edges worksheet.
- *Edges with duplicates*. The number of repeated vertex pairs on the Edges worksheet. Duplicate vertex pairs may occur, as, for example in a discussion forum network when person A replies to person B on multiple occasions. Duplicate vertex pairs can cause some metrics such as degree to be inaccurate, unless they are combined into a weighted edge as described later (Chapter 6).
- *Total edges*. The number of total edges (i.e., rows on the Edges worksheet).
- *Self-loops*. The number of edges that connect a vertex with itself. A self-loop occurs when the edge list includes the same exact name in the vertex 1 and vertex 2 columns on the Edges tab (i.e., a person is connected to themselves). This may happen when, for example, in an email list edge list a person replies to his or her own email. Self-loops are represented visually in the graph pane by a circular edge that comes out of a vertex and returns to that same vertex.
- *Connected components*. The number of connected components (i.e., clusters of vertices that are connected to each other but separate from other vertices in the graph). In the Kite network there is only one connected component because you can get from one vertex to all other vertices. In contrast, the Invitation network discussed in Chapter 4 included two connected components: the large group containing Carol, Ed, Dave, Bob, Ann, and Frank, and the smaller component containing Gary and Helen (see Fig. 4.17).
- *Single vertex connected components*. The number of isolated vertices that are not connected to any other vertices in the graph. There are no isolated vertices in the Kite network or Invitation network.
- *Maximum vertices in a connected component*. The number of vertices in the connected component with the most vertices. This is equal to the number of vertices in the Kite network, because they are all part of the only connected component. In the Invitation network, the largest component includes six people, so this value would be 6.
- *Maximum edges in a connected component*. The number of edges in the connected component with the most edges. This is equal to the number of edges in the Kite network, because they are all part of the only connected component. In the Invitation network, the component with the most edges has six connections.
- *Maximum geodesic distance (diameter)*. The geodesic distance is the length of the shortest path between two people. If you think of the edges as roads and the vertices as houses, the geodesic distance would be the number of roads someone must take to get from one house to another, assuming that the person is traveling on the shortest path possible. The maximum geodesic distance, or diameter of a network, is the largest geodesic distance of all, or the distance between the two vertices that are farthest

ADVANCED TOPIC
Calculating and Importing Additional Graph Metrics

Numerous network metrics exist in addition to those calculated by NodeXL. In addition, new metrics are constantly being developed. Additional aggregate metrics can be calculated using Excel's built-in functions. For example, some analysts like to look at the variance of the degree, which can be calculated by using the function:

```
=VARP(Vertices[Degree])
```

Other aggregate metrics capture aspects of the degree distribution. For example, network Centralization measures how much the network depends on key people for its connectivity. Many of these aggregate metrics can be calculated in NodeXL using formulas.

Some specialized graph metrics are not currently calculated in NodeXL, such as centrality metrics that use edge weights (see Newman's thoughtful review [1] for a comprehensive discussion of network metrics). Researchers who need such metrics may use other network analysis tools such as Pajek or UCINET to calculate them and import them into NodeXL as additional columns. This allows all of the advanced visualization features of NodeXL while still providing more network metrics.

from each other. In the Kite network, this value is 4. For example, the shortest path between Jane and Diane is 4; similarly the shortest path from Jane to Andre, Beverly, Carol, and Ed is also 4. All other geodesic distances are smaller. For example, the shortest path between Jane and Ike is 1.

- *Average geodesic distance*. The average of all geodesic distances. This value gives a sense of how "close" community members are from one another. If it is high, many individuals in the social network do not directly know each other. People may be connected through a friend of a friend of a friend of a friend, but not through short paths. If it is low, most people know one another either directly or through a mutual friend.
- *Graph density*. The number between 0 and 1 indicating how interconnected the vertices are in the network. For an undirected graph where all vertices are connected to all others through at least one edge, the graph density is calculated by dividing the number of total edges by the maximum number of possible edges. For the Kite network, there are 18 edges and 45 possible edges, resulting in a graph density of 0.4. A more dense graph (e.g., 0.6) would include more total edges for the same number of vertices.
- *NodeXL version*. Indicates the version of NodeXL in use when metrics were calculated.

In addition, a frequency chart is created for each of the possible vertex-specific graph metrics. For example, the chart shown in Figure 5.3 shows the frequency of the different Degree metrics. Only one person, Jane, has a degree of 1 (shown on the far left, but not labeled). Likewise, only one person has a degree of 2 (Ike) and one person has a degree of 6 (Diane). In contrast, three people have a degree of 3 (Carol, Ed, and Heather). These frequency charts are particularly helpful when analyzing large networks, as we will see in future chapters. Some basic statistics about the metric distributions are shown under the charts (minimum degree, maximum degree, average degree, and median degree). These help characterize the entire networks and allow for comparisons over time or across networks.

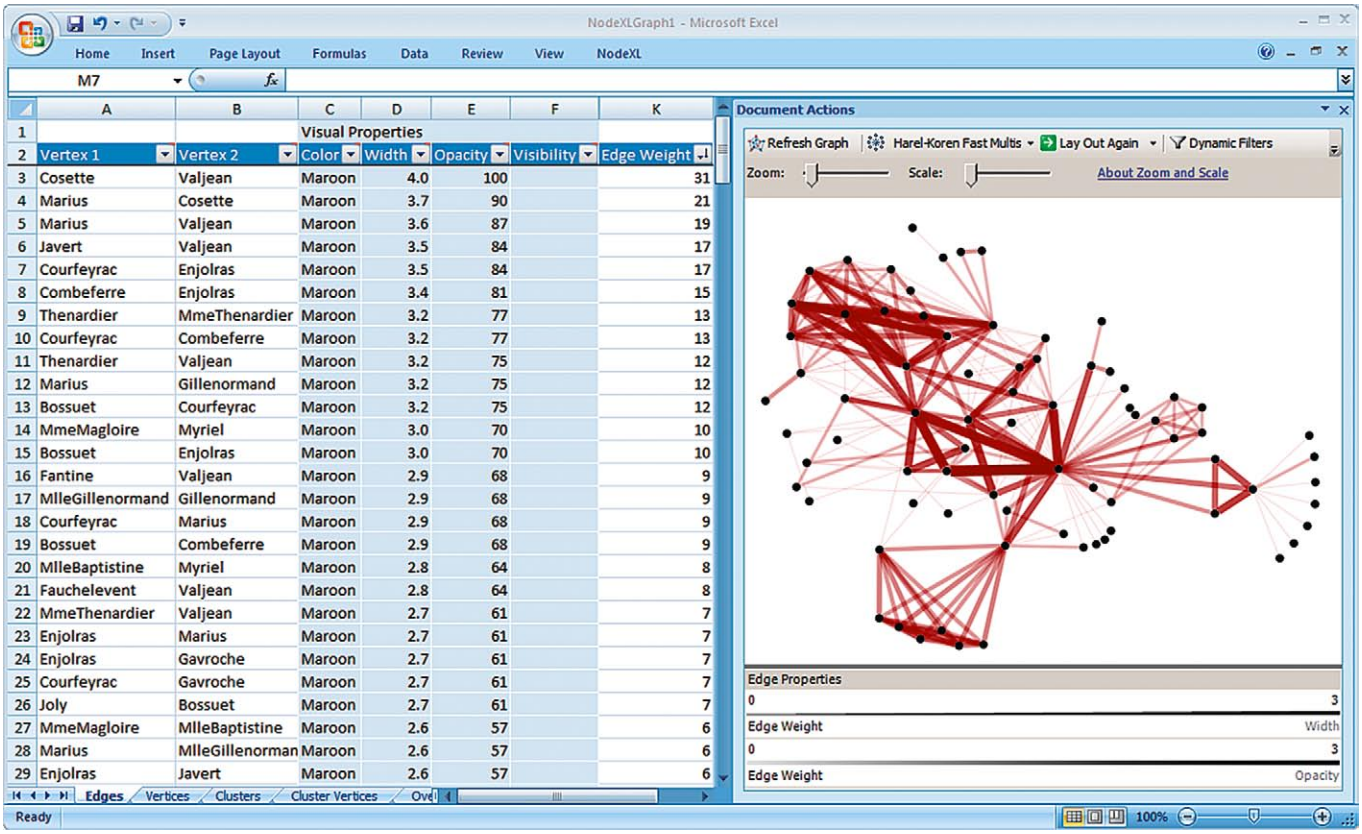
5.4 *LES MISÉRABLES* CO-APPEARANCE NETWORK

Most networks are larger and messier than the Kite example network. Vertex graph metrics are particularly helpful in cases of large networks because they help identify important vertices that may be buried visually in a tangle of vertices and edges. In this section you will look at a network of characters from Victor Hugo's epic novel *Les Misérables*. Data on character appearances by scene were originally compiled by Kohler as part of the Stanford GraphBase book by Knuth [2] and made into an edge list by Newman and Girvan [3]. The network connects characters in the book (vertices) based on the number of times they appear together in a scene (edges). This is an example of a weighted graph because characters may show up together in more than one scene. It is also undirected.

Download the *Les Misérables* example network (Les_Miserables_Example.xlsx) from the book web site and open it in NodeXL (see Section 5.2.1). The network should have 254 edges and 77 vertices (i.e., book characters).

5.4.1 Sorting the Edge Weight Column

Because this is a weighted network, the Edges worksheet contains a column titled Edge Weight. This column represents the number of scenes a character pair showed up in together. For example, Cosette and Valjean are in 31 scenes together, more than any other pair (Fig. 5.5).



	A	B	C	D	E	F	K
1			Visual Properties				
2	Vertex 1	Vertex 2	Color	Width	Opacity	Visibility	Edge Weight
3	Cosette	Valjean	Maroon	4.0	100		31
4	Marius	Cosette	Maroon	3.7	90		21
5	Marius	Valjean	Maroon	3.6	87		19
6	Javert	Valjean	Maroon	3.5	84		17
7	Courfeyrac	Enjolras	Maroon	3.5	84		17
8	Combeferre	Enjolras	Maroon	3.4	81		15
9	Thenardier	MmeThenardier	Maroon	3.2	77		13
10	Courfeyrac	Combeferre	Maroon	3.2	77		13
11	Thenardier	Valjean	Maroon	3.2	75		12
12	Marius	Gillenormand	Maroon	3.2	75		12
13	Bossuet	Courfeyrac	Maroon	3.2	75		12
14	MmeMagloire	Myriel	Maroon	3.0	70		10
15	Bossuet	Enjolras	Maroon	3.0	70		10
16	Fantine	Valjean	Maroon	2.9	68		9
17	MlleGillenormand	Gillenormand	Maroon	2.9	68		9
18	Courfeyrac	Marius	Maroon	2.9	68		9
19	Bossuet	Combeferre	Maroon	2.9	68		9
20	MlleBaptistine	Myriel	Maroon	2.8	64		8
21	Fauchelevent	Valjean	Maroon	2.8	64		8
22	MmeThenardier	Valjean	Maroon	2.7	61		7
23	Enjolras	Marius	Maroon	2.7	61		7
24	Enjolras	Gavroche	Maroon	2.7	61		7
25	Courfeyrac	Gavroche	Maroon	2.7	61		7
26	Joly	Bossuet	Maroon	2.7	61		7
27	MmeMagloire	MlleBaptistine	Maroon	2.6	57		6
28	Marius	MlleGillenorman	Maroon	2.6	57		6
29	Enjolras	Javert	Maroon	2.6	57		6

FIGURE 5.5 *Les Misérables* character co-appearance network data sorted by edge weight from largest to smallest and visualized with the Harel-Koren Fast Multiscale layout and some hand tuning. Edge width (1 to 4) and opacity (10 to 100) both use data in the Edge Weight column using the logarithmic mapping option. Edges are a different color (maroon) to keep the vertices identifiable.

Using Excel features, you can Sort the Edge Weight column from largest to smallest to identify the characters that show up the most often together in the book. To do so, choose the drop-down menu triangle in the label of the Edge Weight column. Select Sort Largest to Smallest from the menu to show the highest edge weights at the top and lowest at the bottom (Fig. 5.5). Scrolling down the edges after sorting is a good way to get a quick idea of the edge weight distribution. In this case you can see that few characters have many connections and most have very few connections, suggesting that it follows some type of logarithmic or power law distribution.

5.4.2 Visualizing Edge Weights

The Edge Weight column can be used to set visual properties of the edges in a similar way as you have done for vertices (Chapter 4). To recreate Figure 5.5, first make the edges maroon by typing the word "Maroon" into the Color column top empty cell and copying it down. This makes the edges more easily distinguishable from the black vertices. Next, use the Autofill Columns feature to set the edge width and edge opacity to edge weight. The combination provides a clear visual distinction between characters that have a strong connection and those with a weaker connection. Go into the Edge Width Options to set the maximum width to 4. Check the Use a Logarithmic Mapping box for both the edge width and the opacity, as the underlying data follow a similar distribution. Choose the Harel-Koren Fast Multiscale layout. It may not match Figure 5.5 exactly because the layout changes each time it is run. Try clicking Lay Out Again to see different layouts, some of which may resemble Figure 5.5 more closely. Notice that the values in the legend show a maximum of 3 instead of the maximum edge weight value of 31. This is because the natural logarithm of 31 is 3.434, which rounds down to 3.

5.4.3 Calculating and Visualizing Vertex Metrics to Find Important Individuals

Next, compute all of the graph metrics as described earlier in this chapter. You can sort on different network metrics to identify important individuals. For example, sorting based on Degree or Betweenness Centrality shows Valjean at the top of the list, indicating his central

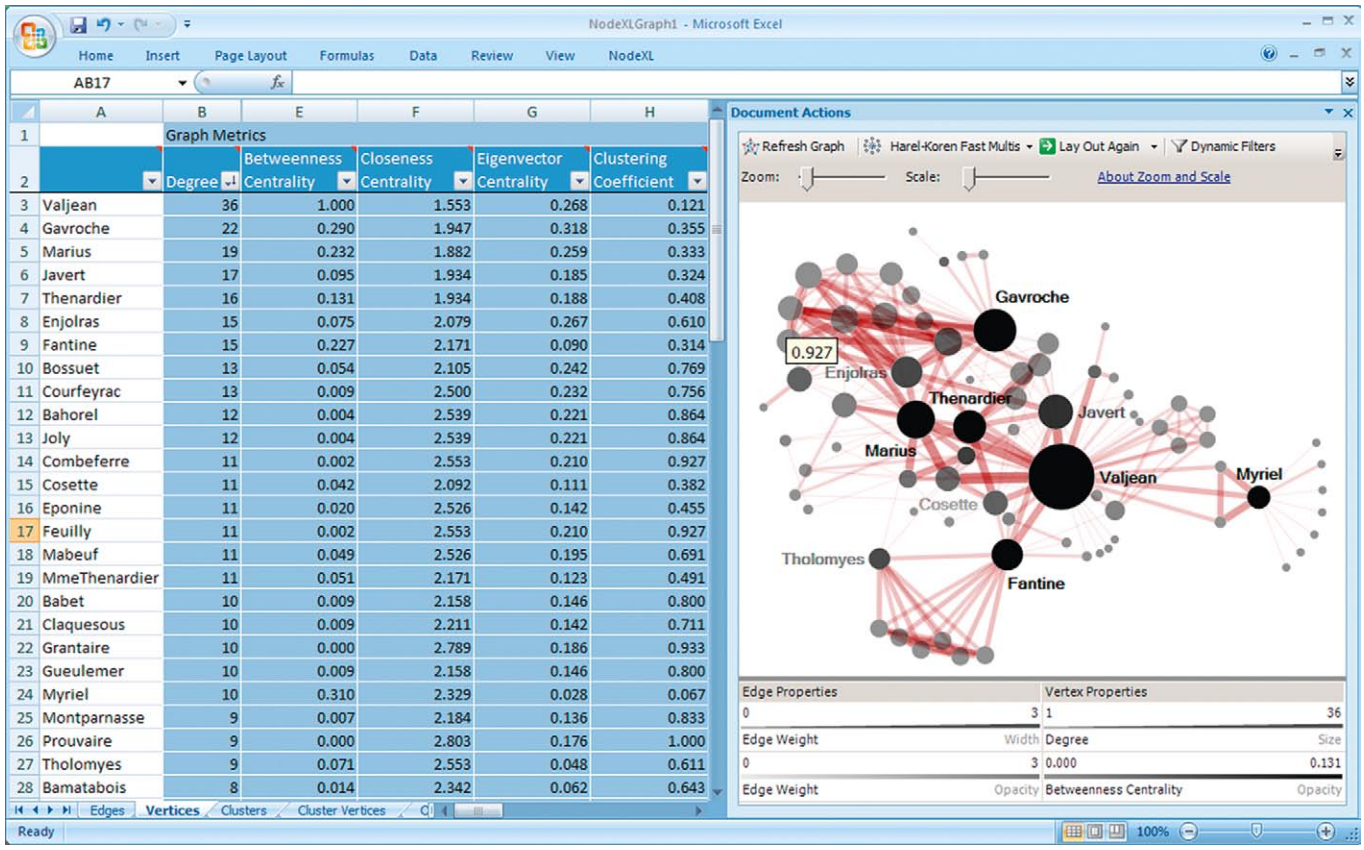
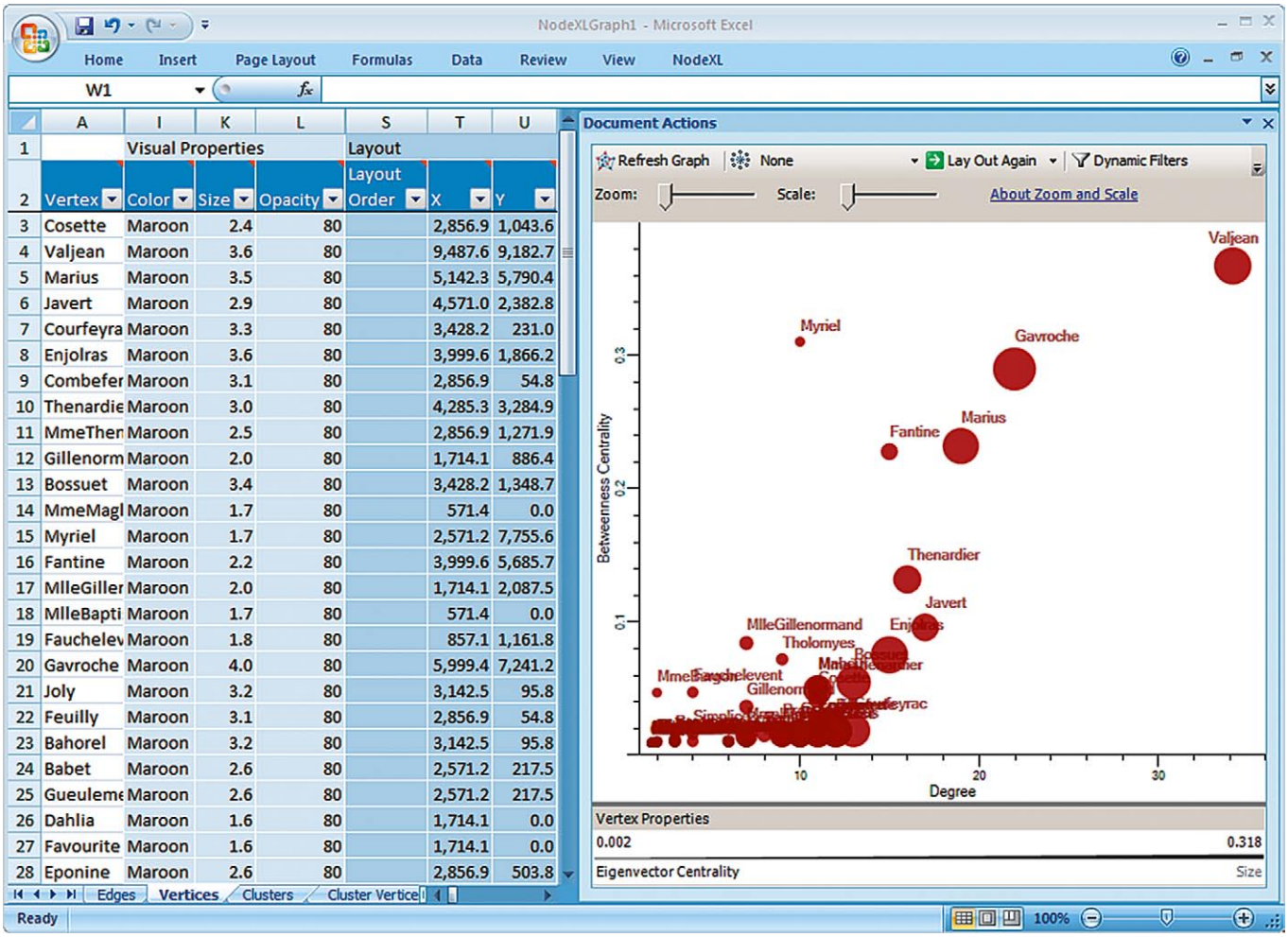


FIGURE 5.6 The *Les Misérables* network mapping Degree to vertex size (1.5 to 5), Betweenness Centrality to vertex opacity (50 to 100), and Clustering Coefficient to tooltip. Characters with significant metrics are labeled. Edge width (1 to 4) and edge opacity (10 to 60) are based on a logarithmic mapping of edge weight.

role in the book as the main character and his role as broker between different characters (Fig. 5.6). The character with the highest Eigenvector Centrality is Gavroche (0.318), a young street urchin who acts as an unofficial courier linking many other characters together.

Mapping the network metrics onto visual properties of the graph can help you to better understand what the metrics mean, as well as the relationships between characters. Figure 5.6 maps Degree to vertex size, Betweenness Centrality to opacity, and Clustering Coefficient to the tooltip using the Autofill Columns feature. In large graphs, it is often useful to use the advanced options within the Autofill Columns to ignore outliers and/or use a logarithmic mapping. The frequency distribution graphs on the Overall Metrics worksheet can help you identify metrics that have outliers or a logarithmic distribution. For example, the Betweenness Centrality graph shows a clear outlier far to the right-hand side from the other values. This corresponds to Jean Valjean's score. To account for this, the Ignore Outliers box was selected in the Vertex Opacity options, lest Valjean's high Betweenness score obscure all other differences. The legend shows that the maximum Betweenness Centrality has been limited to a value 0.131 because of the effects of the "Ignore Outliers" feature. All characters with higher "outlier" Betweenness values (including Valjean = 1 and Myriel = 0.310) are set to the highest nonoutlier Betweenness value (0.131). The edge opacity was set to a maximum of 60 so that edges would not obscure the labels and vertices too much.

Visualizing the *Les Misérables* character network data in this way helps identify individuals with unique roles. For example, the priest Myriel has a relatively small number of connections (i.e., Degree), but a high Betweenness Centrality. As shown in the graph, this is driven by the fact that he is the only link to seven other characters. A cluster of densely connected vertices in the upper-left-hand portion of the graph represents the student rebels. Hovering your cursor over one of the student rebels shows the high Clustering Coefficient of 0.927, suggesting that this character and his friends form a dense cluster. In this context, this means that student rebels rarely show up with characters other than themselves. Marius is the clear exception, which is what drives his high Degree and Betweenness Centrality. Gavroche's direct connection to the student group



	A	I	K	L	S	T	U
1		Visual Properties			Layout		
2	Vertex	Color	Size	Opacity	Layout Order	X	Y
3	Cosette	Maroon	2.4	80		2,856.9	1,043.6
4	Valjean	Maroon	3.6	80		9,487.6	9,182.7
5	Marius	Maroon	3.5	80		5,142.3	5,790.4
6	Javert	Maroon	2.9	80		4,571.0	2,382.8
7	Courfeyra	Maroon	3.3	80		3,428.2	231.0
8	Enjolras	Maroon	3.6	80		3,999.6	1,866.2
9	Combefer	Maroon	3.1	80		2,856.9	54.8
10	Thenardie	Maroon	3.0	80		4,285.3	3,284.9
11	MmeThen	Maroon	2.5	80		2,856.9	1,271.9
12	Gillenorm	Maroon	2.0	80		1,714.1	886.4
13	Bossuet	Maroon	3.4	80		3,428.2	1,348.7
14	MmeMagl	Maroon	1.7	80		571.4	0.0
15	Myriel	Maroon	1.7	80		2,571.2	7,755.6
16	Fantine	Maroon	2.2	80		3,999.6	5,685.7
17	MlleGiller	Maroon	2.0	80		1,714.1	2,087.5
18	MlleBapti	Maroon	1.7	80		571.4	0.0
19	Fauchelev	Maroon	1.8	80		857.1	1,161.8
20	Gavroche	Maroon	4.0	80		5,999.4	7,241.2
21	Joly	Maroon	3.2	80		3,142.5	95.8
22	Feuilly	Maroon	3.1	80		2,856.9	54.8
23	Bahorel	Maroon	3.2	80		3,142.5	95.8
24	Babet	Maroon	2.6	80		2,571.2	217.5
25	Gueuleme	Maroon	2.6	80		2,571.2	217.5
26	Dahlia	Maroon	1.6	80		1,714.1	0.0
27	Favourite	Maroon	1.6	80		1,714.1	0.0
28	Eponine	Maroon	2.6	80		2,856.9	503.8

FIGURE 5.7 *Les Misérables* network mapping Degree to the X axis (and size) and Betweenness Centrality to the Y axis (and opacity). Axes are shown. Edges are hidden. The scale is modified to account for Valjean's position as an outlier.

and other key characters is also what drives his high Eigenvector Centrality.

5.4.4 Mapping Graph Metrics to X and Y Coordinates

In most layouts, the exact location of the vertices is not meaningful, only their positions relative to one another. However, you may want to map network graph metrics, or other attribute data, to X and Y coordinates to visualize how two metrics interact with one another. Other metrics can be used to adjust visual properties, making it possible to display additional dimensions. For example, in Figure 5.7 the size of each vertex is based on its Eigenvector Centrality score, whereas its position on the X and Y axes is determined by its Degree and Betweenness Centrality score, respectively. The graph makes outliers like Myriel more apparent.

To recreate Figure 5.7, use the Autofill Columns feature. First, set the vertex label to Vertex so that the name of each character will be shown. Next set Vertex X to degree, Vertex Y to Betweenness Centrality, and vertex size to Eigenvector Centrality. If there is a significant outlier like Valjean, you can manually set the maximum value for a particular metric. Figure 5.8 shows how the Options dialog box can be used to set an appropriate maximum value for Betweenness Centrality. In this example, 0.4 was chosen because it was a bit higher than the value of Myriel, the character with the second highest value. The result is that the scale goes from 0 to 0.4 (Fig. 5.7). This approach makes the vertices more readable, but the position of Valjean can be misleading since his Betweenness Centrality is actually far higher than his apparent position on the graph.

Data for some columns, such as the Vertex Opacity column, can be erased by using the Autofill Options arrow on the appropriate row and choosing Clear Edge Sheet Worksheet Column Now. This will clear the data in the worksheet. You can then use Autofill Options to Reset Edge Opacity Source Column Name. This way

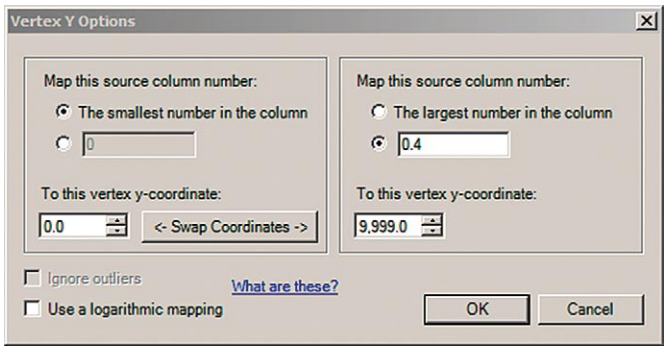


FIGURE 5.8 Vertex Y Options set to a maximum of 0.4 to remove the outlier Valjean and scale the Y axis appropriately.

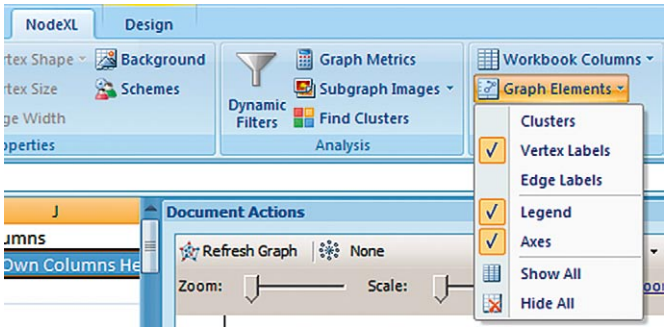


FIGURE 5.9 Selecting Graph Elements to display from the NodeXL ribbon including axes.

that information will not be shown in the legend or repopulated if deleted. Edges are removed from this graph by setting the Visibility column in the Edges worksheet to Hide for all edges. Vertex color was manually set to maroon and opacity to 80 percent in the worksheet.

5.4.5 Viewing and Hiding Graph Elements Such as Axes

To view the axes as shown in Figure 5.7, you must find the Graph Elements drop-down menu on the NodeXL ribbon (Fig. 5.9) and check the Axes box. This is also the place you can decide to show or hide other graph elements such as the legend.

5.5 PRACTITIONER'S SUMMARY

Social network analysis provides a set of powerful quantitative graph metrics for understanding networks and the individuals and groups within them. These include aggregate network metrics such as graph density and number of connected components, which characterize the network as a whole. They also include vertex metrics including Degree, Betweenness Centrality, Eigenvector Centrality, Closeness Centrality, and Clustering Coefficient that can be used to identify unique or important people within a network. These metrics can be mapped onto visual properties such as size and opacity to help more easily make sense of the data. This technique was helpful in identifying the prominent characters in the *Les Misérables* co-occurrence network. The *Les Misérables* example also illustrated how a weighted network can be meaningfully visualized using edge width and edge opacity. Metrics and affiliation data can also be mapped onto the X and Y coordinates, creating scatterplots with axis labels. The combination of metrics and visualization provides a powerful means of understanding and exploring the relationships between vertices and edges.

5.6 RESEARCHER'S AGENDA

The network metrics in NodeXL are widely used because they reveal important properties of individuals in a network [4–8]. However, their computation can be slow, so research efforts on improved algorithms (e.g., [9]), parallelization of execution using multiple processors, and the use of specialized graphic co-processors to speed computation are important. Improved centrality metrics for directed or bipartite graphs and graphs with varying edge weights are also needed (see [8] for initial attempts at some of these). Other metrics are regularly being created to help discover important vertices, edges, motifs, cycles, and other structural features, such as triangles, cliques, near-cliques, chains, holes, and more.

References

[1] M.E.J. Newman, Mathematics of networks, in: L.E. Blume, S.N. Durlauf (Eds.) The New Palgrave Encyclopedia of Economics, second ed., Palgrave Macmillan, Basingstoke, 2008.
[2] D.E. Knuth, The Stanford GraphBase: A Platform for Combinatorial Computing, Addison-Wesley, Reading, MA, 1993.
[3] M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2004) 026113.
[4] P. Bonacich, Power and centrality: A family of measures, The American Journal of Sociology 92 (5) (1987) 1170–1182.
[5] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry 40 (1977) 35–41.
[6] L.C. Freeman, Centrality in social networks: conceptual clarification, Social Networks 1 (3) (1979) 215–239.
[7] D. Koschützki, K.A. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, O. Zlotowski, Centrality indices, in: U. Brandes, T. Erlebach (Eds.) Network Analysis: Methodological Foundations, Springer-Verlag, LNCS 3418, 2005, pp. 16–61.
[8] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences), Chapter 5, Cambridge University Press, Cambridge, UK, 1994.
[9] U. Brandes, A faster algorithm for betweenness centrality, J Math Sociol 25 (2001) 163–177.

CHAPTER

6

Preparing Data and Filtering

OUTLINE

6.1 Introduction 79

6.2 Serious Eats Network Example 79
- *6.2.1 Working with Multimodal Network Data* 80
- *6.2.2 Merging Duplicate Edges* 81
- *6.2.3 Using Shapes and Colors to Identify Vertices of Different Types* 81
- *6.2.4 Sorting Vertices from A to Z* 81
- *6.2.5 Auto-Filling Data Columns* 81
- *6.2.6 Customizing Layout Options* 82

6.3 Filtering to Reduce Clutter and Reveal Important Features 82
- *6.3.1 Dynamic Filters* 84
- *6.3.2 Filtering by Controlling Visibility with Autofill Columns* 87
- *6.3.3 Creating Subgraph Images with NodeXL* 88

6.4 Putting It All Together 90

6.5 Practitioner's Summary 90

6.6 Researcher's Agenda 91

6.1 INTRODUCTION

Most real-world social media networks are much larger than those we have looked at thus far. Visualizing and making sense of large networks can be challenging, particularly if they are densely connected. In this and the following chapter, you will learn several different strategies for analyzing large network data sets.

One strategy is to roll relationship data up into a more summarized form before the analysis and visualization. The *Les Misérables* data set used previously (Chapter 5) is an example of this approach. Rather than reporting each scene that two actors appeared in together (e.g., Valjean and Cosette in the hotel scene), it indicates the two actors and the total number of scenes they were in together (e.g., Jean Valjean and Cosette appeared in 31 scenes together). Summarizing the data in this manner loses some information, as you no longer know in which scenes characters co-appear. However, knowing the total number of connections is sufficient to answer many questions (e.g., How well connected are two people? Who are the most connected people?), and is even better than detailed information for providing an overview of all relationships. In this chapter we discuss how to create weighted graphs like the *Les Misérables* network from a raw, unweighted network using NodeXL's Merge Duplicate edges feature.

Another strategy for dealing with large data sets is to filter the data by removing selected vertices or edges. Filtering can help to identify vertices or edges with extreme values and their relationship to one another. For example, you can look at the relationships between the 20 most active posters to a Listserv who account for the majority of messages. Filtering can also be used to compare relationships between different subsets of people, such as those in different regions or age ranges. It is often necessary to filter large networks to create understandable network visualizations without large numbers of edge crossings or vertex overlaps [1]. Filtering can also be an excellent method for exploring networks, particularly when using dynamic filtering tools such as those provided in NodeXL.

6.2 SERIOUS EATS NETWORK EXAMPLE

This section analyzes a network generated from discussion forum posts and blog comments made to the SeriousEats online community by food enthusiasts (www.seriouseats.com). Data were manually collected from

publicly accessible content taken from the SeriousEats web site on March 7 and 8, 2009, by Emily Mason. You will need to download the data from the file titled "SeriousEats_Example.xlsx" found at the book web site.

6.2.1 Working with Multimodal Network Data

The file includes only an edge list. Vertex 1 includes the usernames of community members who have contributed to the site. Vertex 2 includes abbreviated names of discussion forums or blog posts that the community members posted to. Blog posts begin with "B_" and discussion forum posts begin with "F_." For example, the first row shows that user gastronomeg posted to the blog entry with the abbreviated title Misosoup (Fig. 6.1). This type of network is an example of a "multimodal" network, which means that vertices are of different types (i.e., modes) including people, blogs, or forums. This is in contrast to standard, single-mode networks you've seen so far where all vertices are of the same type (e.g., people). A common type of multimodal network is a bimodal or two-mode network, which connects people (Vertex 1) with events, activities, or organizations with which they are affiliated (Vertex 2). These are called "affiliation networks" by network scientists [2, 3].

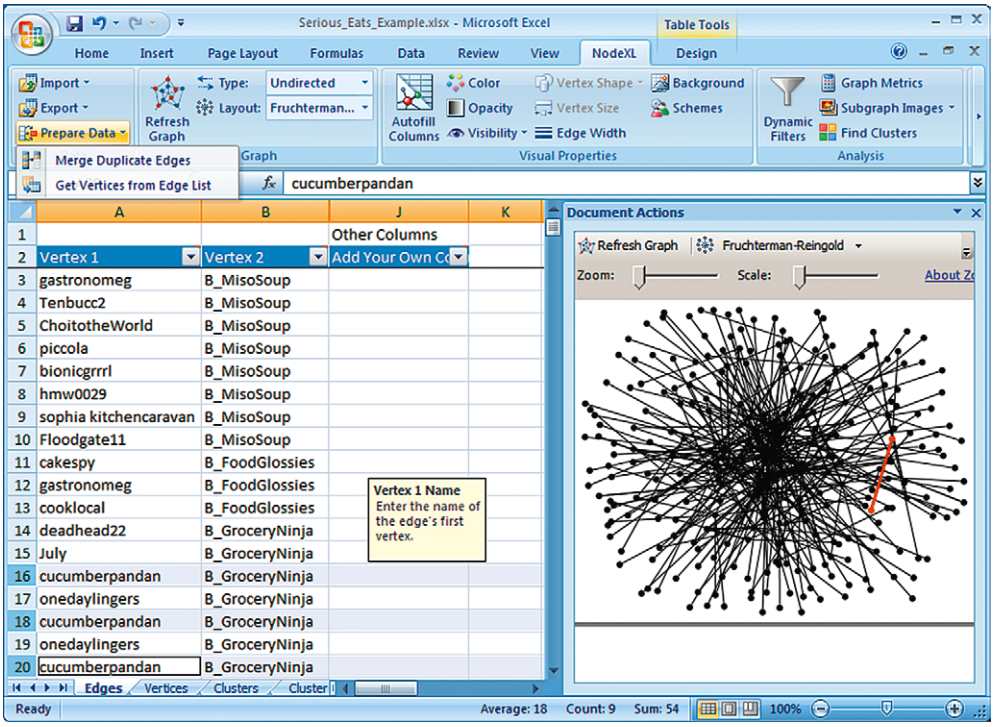


FIGURE 6.1 SeriousEats unmerged data with duplicate edges. For example, user cucumberpandan is connected to blog post B_GroceryNinja three separate times (rows 16, 18, and 20), but only one edge (in red) connects them. The Prepare Data drop-down list on the NodeXL ribbon is selected in preparation to "Merge Duplicate Edges."

ADVANCED TOPIC

Transforming Multimodal Affiliation Networks into Unimodal Networks

Bimodal affiliation networks like the SeriousEats Network can be transformed into two single-mode networks: a person-to-person network and an affiliation-to-affiliation network (where affiliation in this case stands for a blog post or forum topic). The size of the new network will depend on the number of people or affiliations. If you have many people associated with a relatively small number of affiliations (e.g., discussion topics), then the affiliation-by-affiliation network will be small, while the person-to-person network will be large.

A person-to-person network created from affiliation data represents an indirect relationship between people. For example, SeriousEats user gastronomeg is connected to user Tenbucc2 because they have posted comments in reply to the same blog; in other words, they have hung out in the same place. They may not have replied directly

to each other or even have read each other's comments. The *Les Misérables* network (Chapter 5) is an example of a person-to-person network created from affiliation data. It links people together because they show up in the same scene(s) together, not because they talk to one another, although they may. Chapter 15 includes another example based on individuals who have co-edited many of the same wiki pages.

An affiliation-to-affiliation network, in contrast, shows relationships between things that people are affiliated (e.g., events, groups, places) based on the number of people that share the affiliation (e.g., participate in the same event(s), are part of the same group(s), or visit the same place(s)). For example, the SeriousEats blog B_Quote is connected to the forum topic F_DoubleDevon because user NotAmerican posted to both of them. The topics of the discussions may be completely different, but the participants are the same. This type of inferred relationship is what serves as the basis for recommender systems such as Amazon's "Customers Who Bought This Item Also Bought" feature that relates books to other books based on the number of people who were "affiliated" (i.e., purchased) both books together. Chapter 15 includes a wiki page-to-page network of this type based on the number of co-editors of each wiki page.

Transforming a bimodal network into a person-to-person or affiliation-to-affiliation network typically requires the use of matrices. Some network packages, such as UCINET, will do this conversion for you [4]. Alternately, it can be done for reasonably sized networks using Excel's built-in Pivot Table feature and SumProduct function. An example of how to create the affiliation-to-affiliation data from the original bimodal SeriousEats Network edge list is shown in the file SeriousEats_Affiliation_Matrix_Example.xlsx found on the book web site.

6.2.2 Merging Duplicate Edges

You may notice that some rows are duplicates (rows 16, 18, and 20 in Fig. 6.1). This is not an error because some community members posted multiple times to the same forum or blog. For example, user cucumberpandan posted to the blog GroceryNinja on three separate occasions. However, only one edge is visually displayed in the graph pane to represent these three rows (Fig. 6.1). NodeXL allows you to remove the duplicate edges while retaining information about how many times an edge was duplicated. Click on the Merge Duplicate Edges button in the Prepare Data drop-down menu on the NodeXL ribbon as shown in Figure 6.1 and refresh the graph.

You will now see a new column called Edge Weight that indicates the number of edges that were rolled up (i.e., merged). As shown in Figure 6.2, there is now only one row connecting cucumberpandan with B_GroceryNinja showing an edge weight of 3, because three original rows were merged into one. In total, the original 417 unmerged edges are now condensed into 362 merged edges. You can verify that no data were lost by summing up the Edge Weight column, which should equal 417, the original number of rows.

6.2.3 Using Shapes and Colors to Identify Vertices of Different Types

The graph shown in Figure 6.2 is not easy to interpret, largely because it includes so many vertices and edges. It also doesn't indicate that vertices represent different things: users, blog posts, and forums topics. To resolve this issue, you can set unique shapes and colors to each type of vertex. This can be done manually with the aid of sorting or automatically with the use of formulas (see the Advanced Topic: Naming Vertices in Multimodal Networks).

6.2.4 Sorting Vertices from A to Z

Use Excel's built-in sorting feature by choosing the drop-down menu triangle next to the Vertex column title on the Vertices worksheet (Fig. 6.3). Select "Sort A to Z" from the menu (Fig. 6.3). This will sort all of the vertices alphabetically, which groups all of the blog posts (beginning with "B_") and discussion forum posts (beginning with "F_") next to each other making it easy to set unique color and shape attributes for each group.

6.2.5 Auto-Filling Data Columns

Set the color to black and shape to disk for the people (rows 1 through 15 as shown in Fig. 6.4). To efficiently fill in the column cells, you can enter the desired color and shape in the first row, highlight both cells, and move the cursor to the bottom-right corner of the Shape cell until it becomes a + symbol. Drag this symbol down and the cells will fill in with the same content. When you get to the first row beginning with a "B_" (B_FoodGlossies), change the color to blue and the shape to solid diamond. Use the same fill shortcut to populate all rows beginning with a "B_" as shown in Figure 6.4.

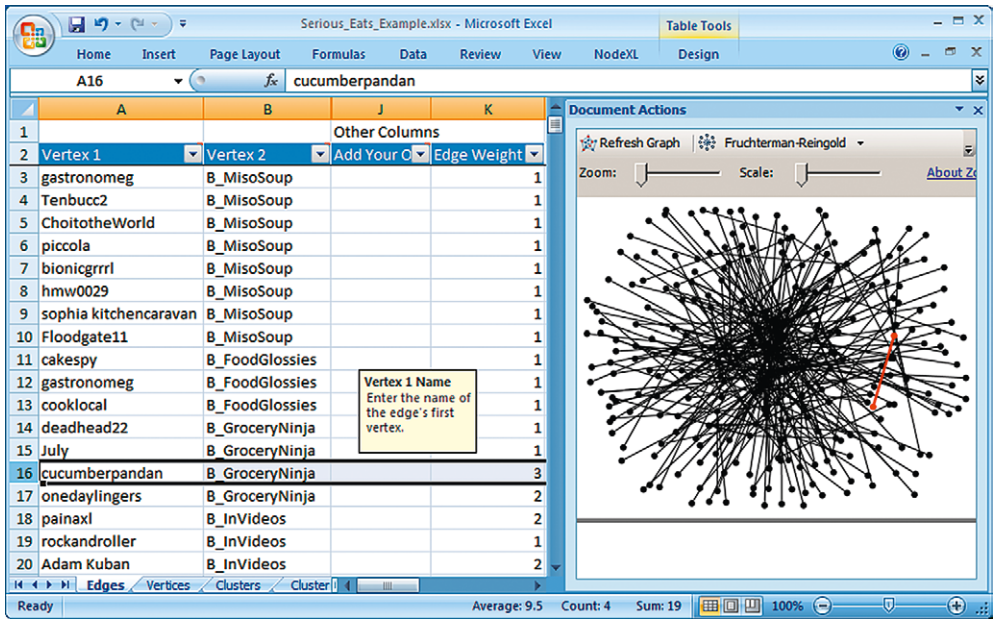


FIGURE 6.2 SeriousEats merged data showing only one row connecting user cucumberpandan with blog post GroceryNinja and a new Edge Weight column.

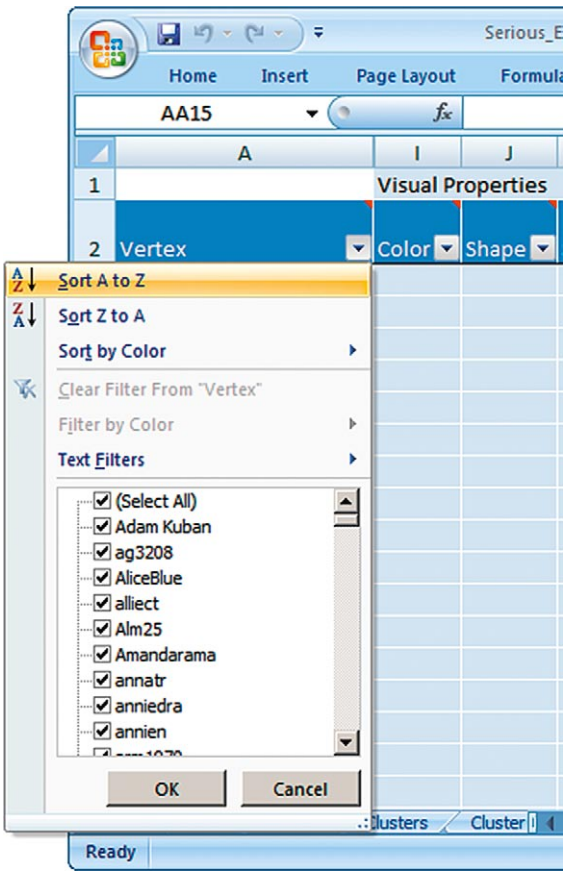


FIGURE 6.3 Sorting the Vertex column in alphabetical order (sort A to Z).

Populate the remaining usernames as black disks and the forum posts beginning with "F_" to orange solid squares. Refresh the graph to see the results (Fig. 6.5).

6.2.6 Customizing Layout Options

Clicking on the Layout Options button (see Fig. 6.5) opens the dialog shown in Figure 6.6. Check off the "Put the graph's smaller components at the bottom of the graph" box to ensure that any small components are separated out from the larger component and arranged in rows on the bottom of the graph.

The Layout Options dialog allows you to decide on a margin, which indicates how close to the graph pane edge the vertices can be. It also provides options specific to the Fruchterman-Reingold layout, which are discussed in Chapters 7 and 11.

6.3 FILTERING TO REDUCE CLUTTER AND REVEAL IMPORTANT FEATURES

There are a number of different reasons to filter out edges or vertices from a graph. Filtering can help you iteratively explore your data, identify important vertices and edges, create more readable and persuasive graphs, or exclude spurious data. The following types of filtering can be performed on edges and vertices, depending on your specific needs:

- *Value-based filtering.* This type of filtering removes edges or vertices that have associated data above or

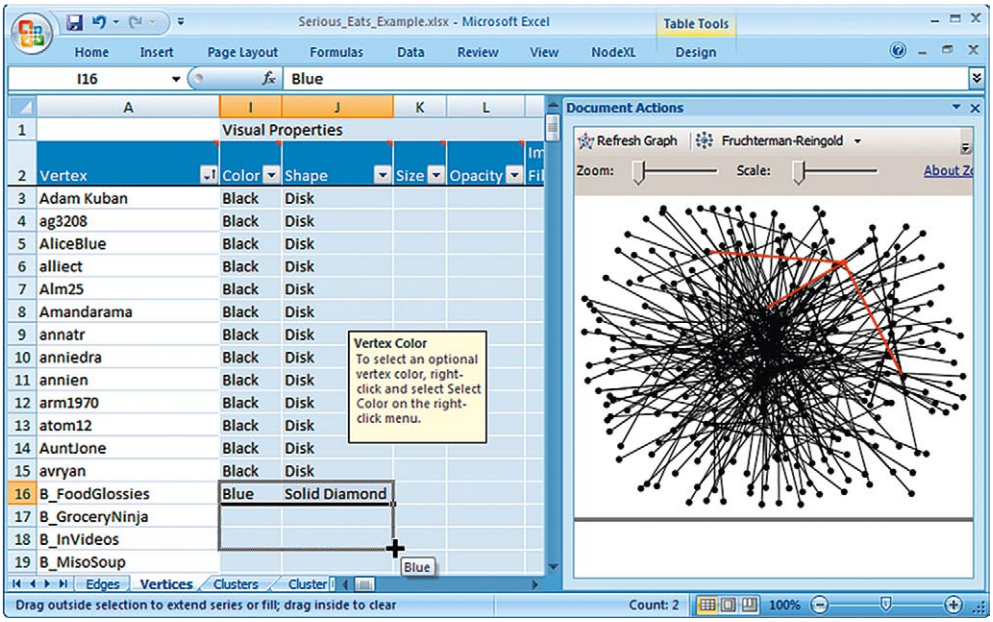


FIGURE 6.4 Populating the Color and Shape column data for blogs (beginning with "B_") by using Excel's built-in automatic fill tool.

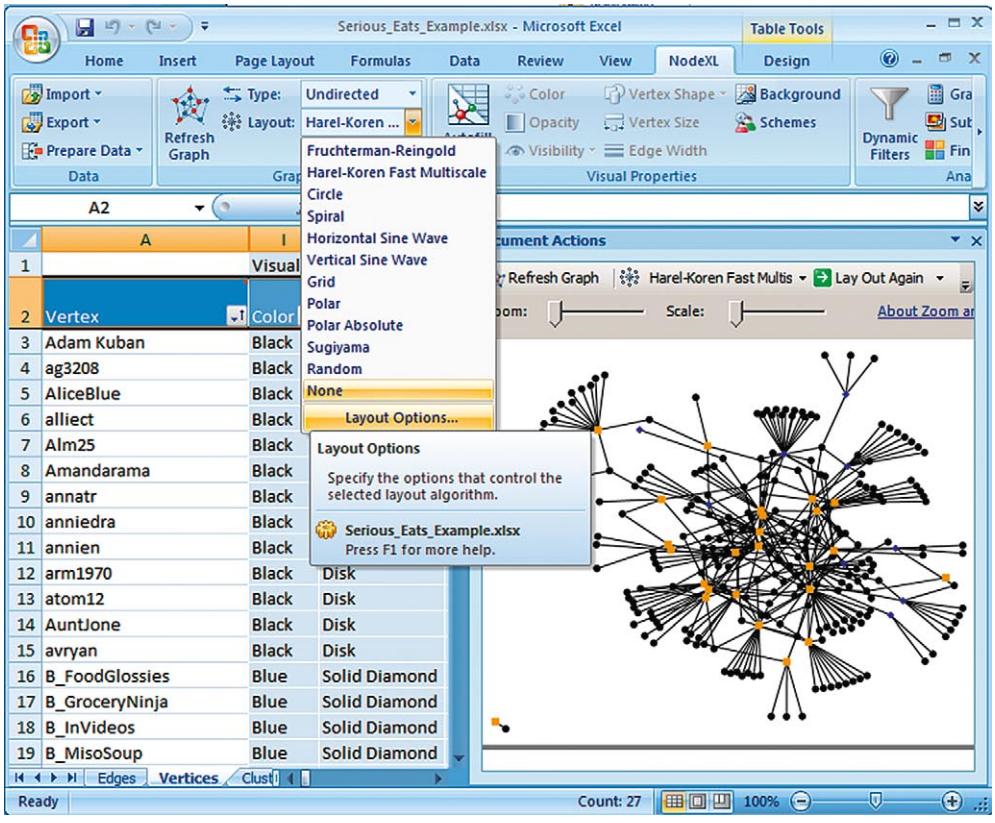


FIGURE 6.5 SeriousEats updated graph showing users as black disks, forum topics as orange solid squares, and blog topics as blue solid triangles. The Harel-Koren Fast Multiscale layout is used. The Layout options are selected so that the graph's smaller components can be separated from the larger one (notice the single forum-username pair in the bottom-left corner of the graph).

below a specified value. For example, a researcher may filter out vertices that have an Age value of Under 18 to protect children. Edges with an edge weight of less than 5 could be removed to focus in on the relationship between those with stronger or more frequent ties. Valued data may be attribute data that describe the edges (e.g., edge weight) or vertices (e.g., age), or they may be network metric data (e.g., vertex degree).

ADVANCED TOPIC
Naming Vertices in Multimodal Networks

When working with multimodal networks, you will need to keep track of the type of vertex. For example, in the SeriousEats data set you want to know if a particular vertex is a blog, forum topic, or person. One useful trick is to embed the information about the type of vertex into the vertex name itself. That way when NodeXL automatically creates the list of unique vertices, you will be able to differentiate between the types of vertices without needing additional columns describing what type each one is. The SeriousEats data set does this by including a "B_" at the beginning of each blog post and an "F_" at the beginning of each forum topic. Starting the vertex name with an identifier is helpful because sorting can be used to group similar items together as described in Section 6.2.3.

Excel's built in Text Formulas can use this information to automatically populate visual properties like shape and color. For example, the following formula could be used in the Color column to replicate the desired colors found in Figure 6.5:

```
=IF(LEFT(Vertices[[#This Row],[Vertex]],2)="B_",
"Blue",IF(LEFT(Vertices[[#This Row],[Vertex]],2)=
"F_","Orange","Black"))
```

This is called a nested If statement. It looks at the first two letters of each vertex name (i.e., the Left 2) and uses the If logic to assign a specified color. If you have more than three types of vertices, you would want to use a VLOOKUP table instead of If statements (see Chapter 11 for an example).

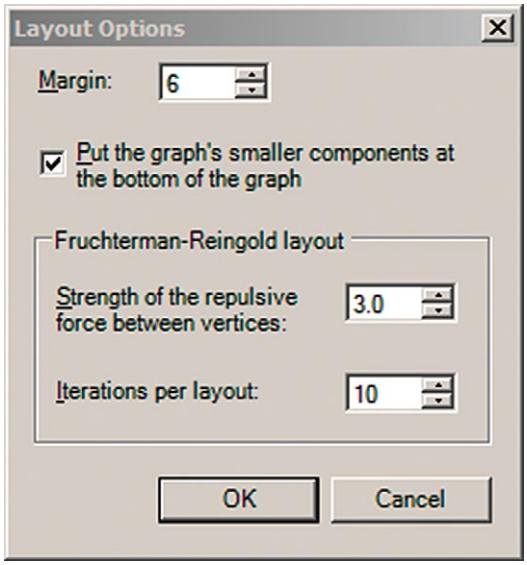


FIGURE 6.6 Layout Options dialog with the "Put the graph's smaller components at the bottom of the graph" selected.

- *Categorical filtering.* This type of filtering removes (or retains) edges or vertices that are associated with a particular category. For example, the network of male vertices may be viewed by filtering out the female vertices. Categorical filtering can be used to create subnetworks for easy comparison. For example, you may use categorical filtering to create four subnetworks representing four major regions of the United States.
- *Ordinal filtering.* This type of filtering removes edges or vertices that are ranked lower (or higher) than other edges and vertices. For example, you may want to show the network of the top 10 most popular vertices, where degree is used to measure popularity. Alternatively, you may want to look at the connections between the 20 least popular vertices.

These different types of filtering can be combined to form highly customized views into network data. Analysts often iteratively filter network data to reduce the complexity of graphs, not knowing beforehand which type of filtering or which values or categories to use. Filtering is often necessary to create understandable graphs without large numbers of edge crossings or vertex overlaps [1]. You will know you've succeeded when you see a comprehensible graph that emphasizes the important relationships and entities without including irrelevant content. NodeXL offers a variety of tools to filter edges and vertices that will be presented in this section using the SeriousEats data set.

6.3.1 Dynamic Filters

The Dynamic Filters feature allows you to remove vertices and edges from the graph pane in real time using sliders that represent attribute or metric data [5–8]. You can open the Dynamic Filters dialog box (Fig. 6.7) using the button in the NodeXL ribbon's Analysis section or just above the graph pane (you may have to click on the downward pointing arrow on the upper-right hand side of the graph to access the Dynamic Filters button). The box offers a number of double box range sliders to help you filter. The number on the left-hand side is the minimum value found in the workbook, whereas the number on the right-hand side is the maximum value. Above each slider is a frequency distribution showing the number of edges or vertices at each value (Fig. 6.7). For example, the Edge Weight frequency distribution graphic shows that

ADVANCED TOPIC
Exporting Images

NodeXL allows you to export images of the graph pane in a variety of formats and sizes. You can copy the image to the clipboard by right-clicking on the graph pane and selecting Copy Image to Clipboard. For more options you can right-click and choose Save Image to File. By default, an image the same size as the graph pane will be saved, but you can specify any size you desire using the Set Image Size feature. You can then choose Save Image, which offers you a variety of formats to save as including .jpg, .bmp, .gif, .tif, .png, and .xps. The .xps file type is vector based so that with the correct software it can be scaled to larger image sizes and retain its quality.

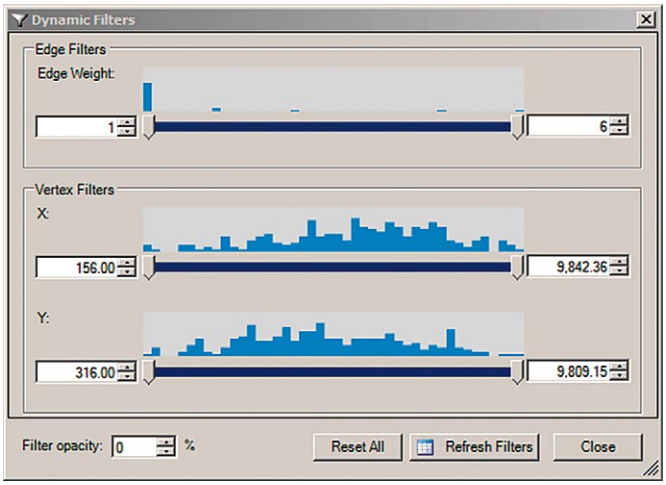


FIGURE 6.7 NodeXL Dynamic Filters dialog box with double boxed sliders, which allow you to set minimum and maximum values to filter out edges or vertices. Frequency distributions are shown visually above each slider.

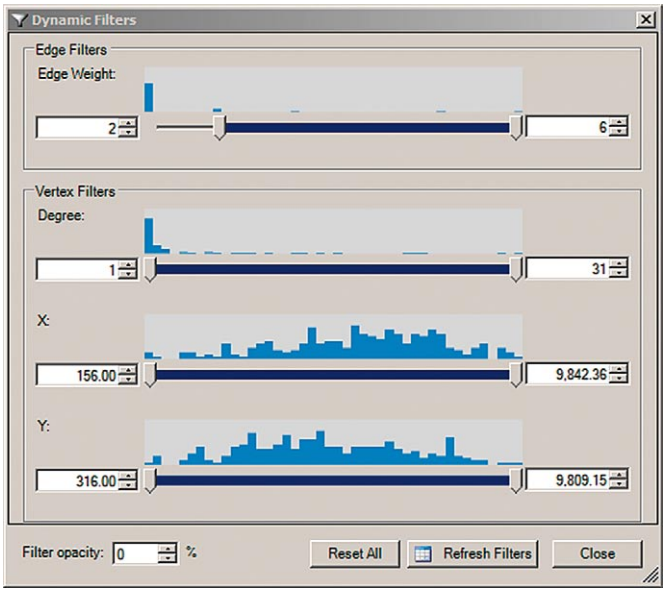


FIGURE 6.8 NodeXL Dynamic Filters dialog box after calculating the metric Degree,' choosing Refresh Filters, and increasing the edge weight from 1 to 2.

almost all of the edges have an edge weight of 1. The top set of sliders filter out edges, leaving in the vertices. The second set of sliders filter out the vertices and all edges associated with those vertices. Currently, the X and Y positions are shown, allowing you to filter out vertices based on their position in the graph, which is not particularly useful in this case but can be in other cases.

New filters appear when additional metrics are calculated or new columns are added to the Edges and Vertices worksheets and populated with numerical data. Calculate the metric Degree' as described earlier (Chapter 5). Then click on the Read Workbook button in the Dynamic Filters dialog box (Fig. 6.7). You will now see a new slider titled "Degree:" in the Vertex Filter's area as shown in Figure 6.8.

Try sliding the Edge Weight slider on the left-hand side to the right so that the number changes from 1 to 2. The graph will dynamically update so only edges that have an edge weight of 2 or higher will be displayed. The resulting graph (Fig. 6.9) shows only ties where a person has posted to a forum topic (or blog post) two or more times. When dynamic filters are used, the legend at the bottom of the graph pane is updated to reflect the filter settings (Fig. 6.9).

When items are filtered out, they are still stored in the graph's memory. They are simply hidden from view. They will show up if you click on the corresponding vertex or edge in the data portion of the spreadsheet. This is demonstrated in Figure 6.9 where the edge connecting gastronomeg and the blog post titled MisoSoup are shown in red even though their edge weight is less than 2.

Click on the Reset All button in the dynamic filters dialog box (Fig. 6.8) to show all of the edges and vertices. Next, click on the upward pointing arrow on the left-hand side of the Degree slider. This will incrementally remove vertices with a Degree smaller than the number in the left-hand box. Figure 6.10 shows a series of graphs starting with all vertices and continuing to remove vertices with a degree of 1, then 2, then 3, and so forth.

Browsing through the network lets you get a visual sense of the data relatively quickly. For example, you can see that most users are only connected to a single discussion or blog (i.e., most disappear when filtering

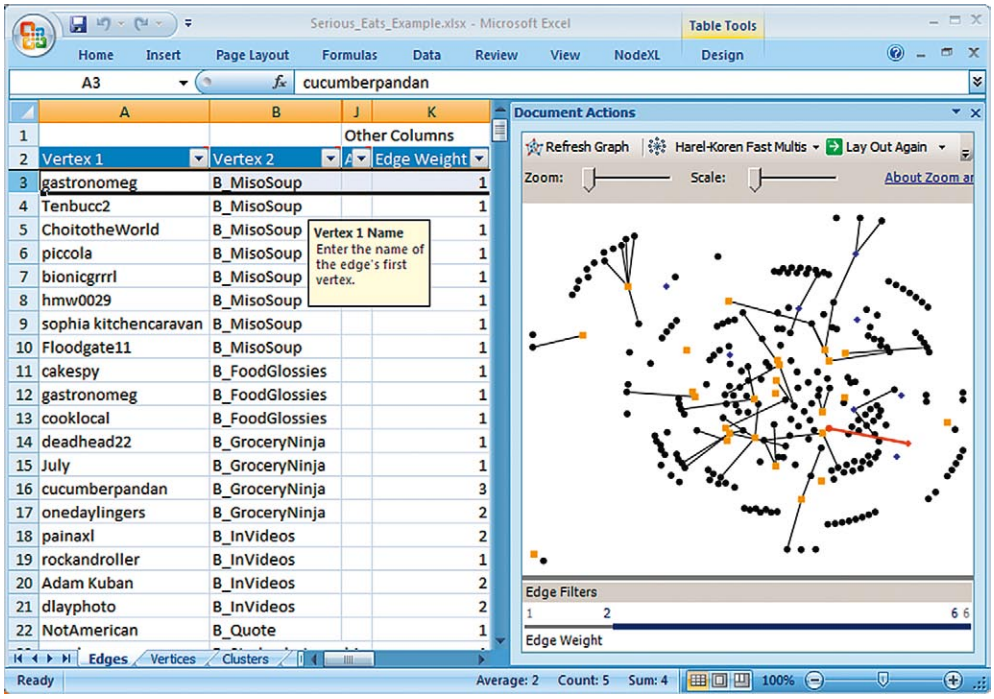


FIGURE 6.9 A dynamically filtered graph showing only edges with edge weight of 2 or higher, except for selected edges such as the one in red connecting user gastronomeg to blog post B_MisoSoup.

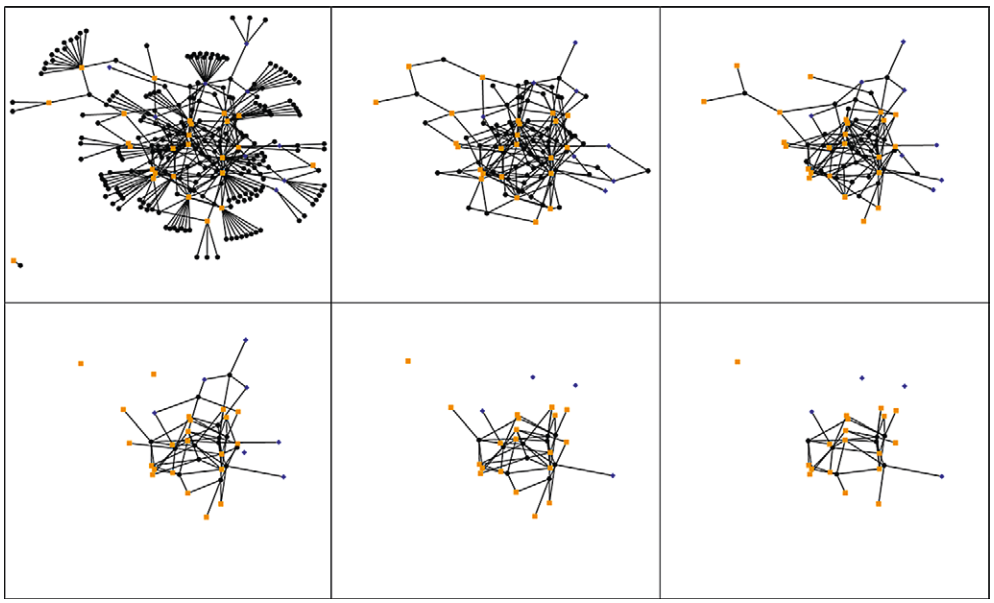


FIGURE 6.10 Six images of the SeriousEats social media network of blogs, discussions, and people created in NodeXL by incrementally increasing the minimum Degree slider beginning with a minimum Degree of 1 (upper-left image) and ending with a minimum Degree of 6 (lower-right image).

out those with a degree of 1); most forum topics have at least six contributors; a core group of around 10 members edit many forum posts; and not many of that core group edit blogs.

You can set the filter opacity to show the filtered out edges while still making them less prominent. Enter 10 into the Filter Opacity box on the Dynamic Filters dialog box (Fig. 6.8) to re-create Figure 6.11.

As mentioned before, even when the filter opacity is 0, the vertices and edges are retained in the graph's memory, they are just hidden from view. One result of this is that if you try to lay out the graph when filtering,

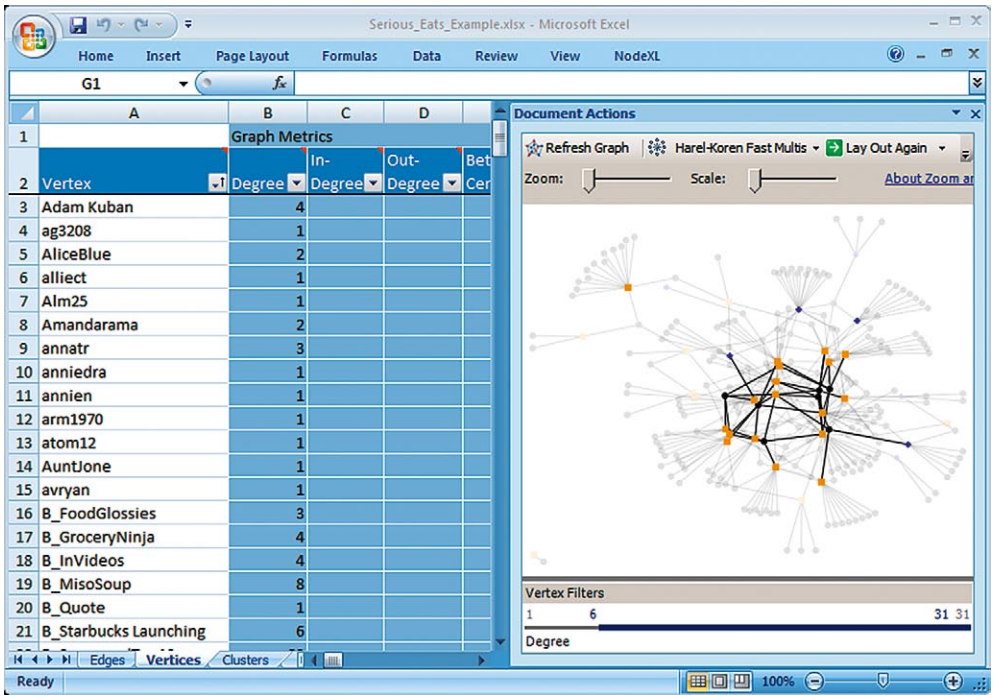


FIGURE 6.11 Dynamic filters set to a minimum degree of 6 with filter opacity at 10 percent.

the visible vertices will be arranged as if all of the vertices were included in the graph. If you want to lay out the visible vertices as if they were the only ones in the graph, you can do so by choosing the drop-down arrow next to the Lay Out Again button and selecting Lay Out Visible Vertices Again.

6.3.2 Filtering by Controlling Visibility with Autofill Columns

Another method of filtering is to use the Autofill Columns feature (Chapter 4) to automatically populate the Visibility Column. Before trying this, choose Reset All on the Dynamic Filters dialog box so that all vertices are shown (Fig. 6.8) and then close the dialog. Next, open the Autofill Columns dialog box, select Degree in the drop-down menu for Vertex Visibility, and choose the arrow to the right that opens the Vertex Visibility Options dialog box shown in Figure 6.12.

A number of options are available. Select the "Greater than or equal to" option, replace the value 0 with 6, and click OK. Unlike dynamic filters, when you use this method, only the unfiltered vertices (or edges) are read into the graph. As a result, using the regular Lay Out Again option will reposition the visible vertices and edges as if they were the only ones present in the network. For example, you can use the Sugiyama layout to reproduce the network shown in Figure 6.13. Additionally, if you select a vertex or edge in a network filtered with this method, only those that already appear in the graph will

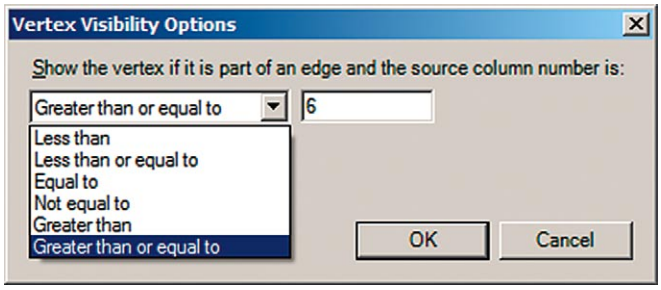


FIGURE 6.12 Vertex Visibility Options dialog box set to show only vertices that are greater than or equal to 6.

be turned red, as no vertices or edges are hidden by the Dynamic Filters setting (see the Advanced Topic: Visibility Column Options). For example, in Figure 6.13 the vertex representing the Forum BreadBaking is selected, which has a degree of 13 indicating that 13 unique people have contributed to that forum. However, only one person is connected to that forum in the current version of the network graph. The other 12 people were not read into the graph since they had a degree of less than 6.

To get back to the original graph, open the Autofill Columns dialog box. Delete the word "Degree" from the Vertex Visibility row. Next, click on the Options arrow on the Vertex Visibility row and choose Clear Vertex Visibility Worksheet Column Now. This will erase any content in the Visibility column, restoring the visibility to the default setting (see Advanced Topic: Visibility Column Options). This same technique can be

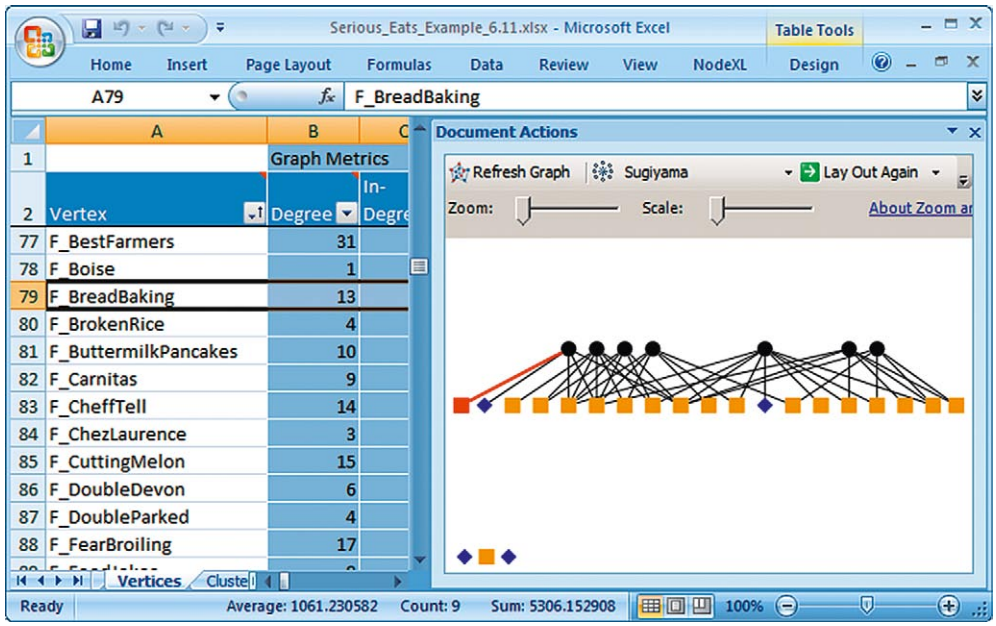


FIGURE 6.13 NodeXL Autofill Columns filtered view of the SeriousEats social media network, limited by Degree of 6 or more. The Sugiyama layout is used.

ADVANCED TOPIC
Visibility Column Options

Like other visual attribute columns, the Visibility column can be filled using Autofill Columns, populated using formulas, or manually by typing in the desired option. The following options are available:

- *Show if in an edge.* If the vertex is part of an edge, show it. Otherwise, ignore the vertex row. This is the default.
- *Skip.* Skip the vertex row and any edge rows that use the vertex. Do not read them into the graph. This is what the Autofill Columns Visibility filter feature controls. If you choose to calculate Graph Metrics after some vertices are skipped, metrics will be calculated as if those vertices do not exist. This is true for automatically calculating clusters as well (see Chapter 7).
- *Hide.* If the vertex is part of an edge, hide the vertex and its edges. Otherwise, ignore the vertex row. This is what the dynamic filters do. Hidden vertices and edges are included when network graph metrics are calculated; they just do not show up in the graph visualization (although their effect can be seen in the way they displace other vertices).
- *Show.* Show the vertex regardless of whether it is part of an edge.

used to delete information in the Size, Opacity, or other columns. Select the Harel-Koren Fast Multiscale layout, and choose Refresh Graph to create a graph similar to the one you saw earlier (Fig. 6.5).

6.3.3 Creating Subgraph Images with NodeXL

Another useful way to understand complex networks is to view individual sections, or subgraphs, of the larger graph. NodeXL allows you to create subgraph images for each vertex in the network. Network scientists call these egocentric networks (see Chapters 3 and 11). They provide a personalized view of the network from the perspective of an individual vertex and are useful when comparing vertices to one another. For example, vertices with a similar structure (e.g., a hub and spoke) may play similar social roles such as a question answerer in a discussion forum (see Chapter 9).

To create network subgraph images, click on the Subgraph Images button on the Analysis section of the NodeXL ribbon. The Subgraph Images dialog box (Fig. 6.14) will appear. The first option allows you to choose the levels of adjacent vertices to include in each subgraph. For example, the default of 1.5 will show edges connecting the source vertex with its direct neighbors, as well as any edges that connect the neighbors to one

another. Choosing 2.0 will show all of those edges, plus edges connecting the source vertex's neighbors with all of their neighbors. If the data were from a social networking site such as Facebook, a 2.0 setting would show your friends, which of your friends know one another, and all of your friends' friends (FOAF). For now, replicate Figure 6.14 by choosing 2.0, checking the boxes to select the vertex and vertex's incident edges, and clicking Create. This will generate a new column called Subgraph as shown in Figure 6.15. Subgraph images are positioned relative to each other based on the currently selected layout algorithm, so make sure you use an appropriate one (e.g., Harel-Koren Fast Multiscale) for your data.

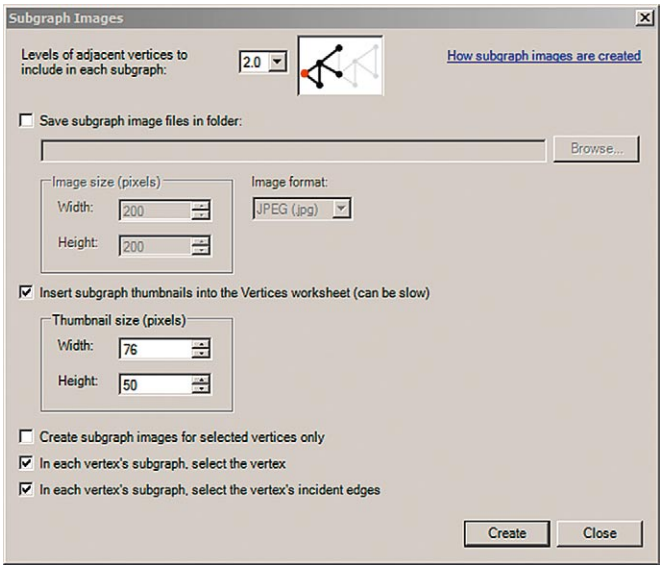


FIGURE 6.14 Subgraph Images dialog box with the number of adjacent vertices to include set to 2.0.

Subgraphs highlight important differences between vertices. To illustrate this point, sort on the Vertex column on the Vertices worksheet (from A to Z) and scroll down to those vertices beginning with "F_." Compare the subgraphs for F_Vietnamese and F_PerfectFood (Fig. 6.15). The F_Vietnamese image makes clear that F_Vietnamese discussion occurs between people who don't frequent other discussion forums or posts. In contrast, the F_PerfectFood forum includes many people who have posted to other forums and blog posts. Similar comparisons can be made for blogs (beginning with "B_") and people. Thus, subgraphs can show important structural differences even when some basic metrics such as degree are very similar.

Other options on the Subgraph Images dialog box (Fig. 6.14) allow you to change the size of the images, export them as new image files that are stored in your

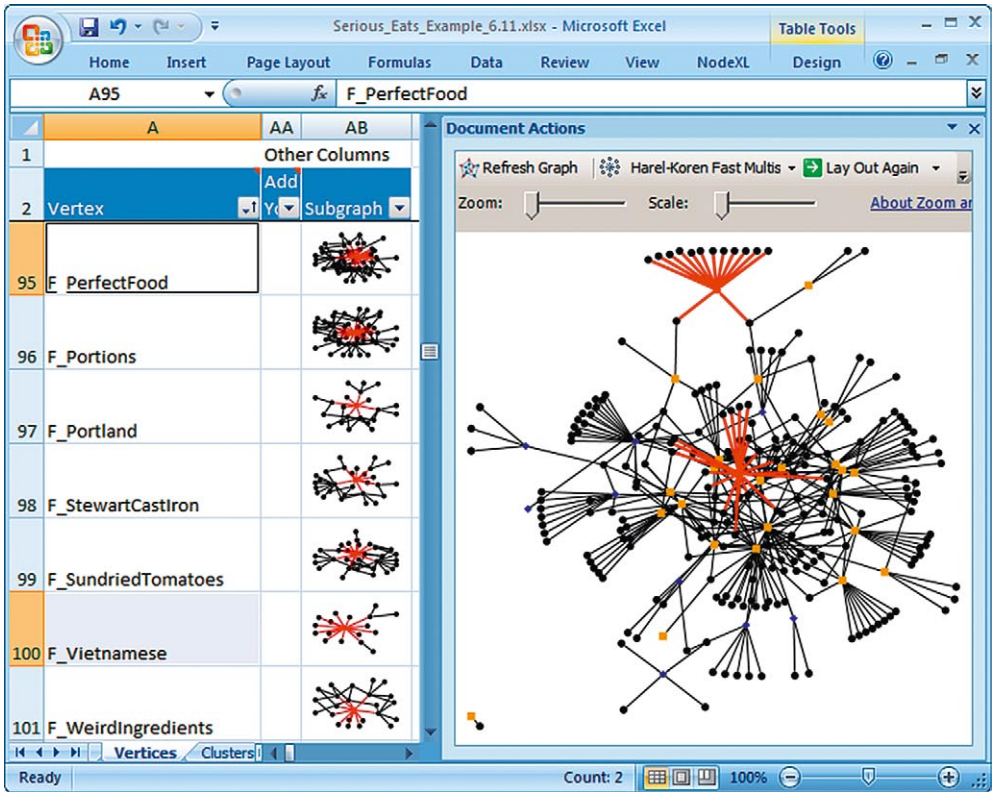


FIGURE 6.15 Subgraph Images on the Vertices worksheet showing differences between forums such as Vietnamese (top of graph) and PerfectFood (middle of graph), both of which are selected.

computer's file system, and limit the creation of subgraph images to vertices that are highlighted on the Vertices worksheet.

6.4 PUTTING IT ALL TOGETHER

By combining the various approaches in this and prior sections you can re-create Figure 6.16, which presents a much more readable graph than our original graph shown in Figure 6.1. Autofill Columns was used to set Vertex Visibility to "Greater than or equal to 4" (after using dynamic filters to find a reasonable threshold), vertex size (1.5 to 4) was mapped to Degree, edge width (1 to 3) was set to Edge Weight, and edge opacity (50 to 100) was also set to Edge Weight. Vertices were manually positioned to more easily make boundary spanners (i.e., those who post to both blogs and discussion forums) more obvious. Labels were manually entered for vertices with the highest degree. Figure 6.16 makes clear that few people (represented by black circles) post to multiple blogs (blue diamonds), while many people post to multiple forums (orange squares) or a blog and a forum. It also helps identify important blogs and forums that solicit significantly more participation than do others, such as I've Never Tasted and Best Farmers.

6.5 PRACTITIONER'S SUMMARY

Making sense of large networks can be challenging, particularly if they are densely connected. One strategy for making them more understandable is to roll up the data into a more summarized form, for example, by merging duplicate edges to create a weighted graph from an unweighted graph. Another strategy is to filter out edges or vertices. Filtering can be based on specific values (e.g., under age 18), categories (e.g., males), or ordinal relationships (e.g., top 10 most connected vertices). One reason to filter is to explore hidden structures that may be obscured by networks with too many connections. NodeXL's Dynamic Filters feature allows you

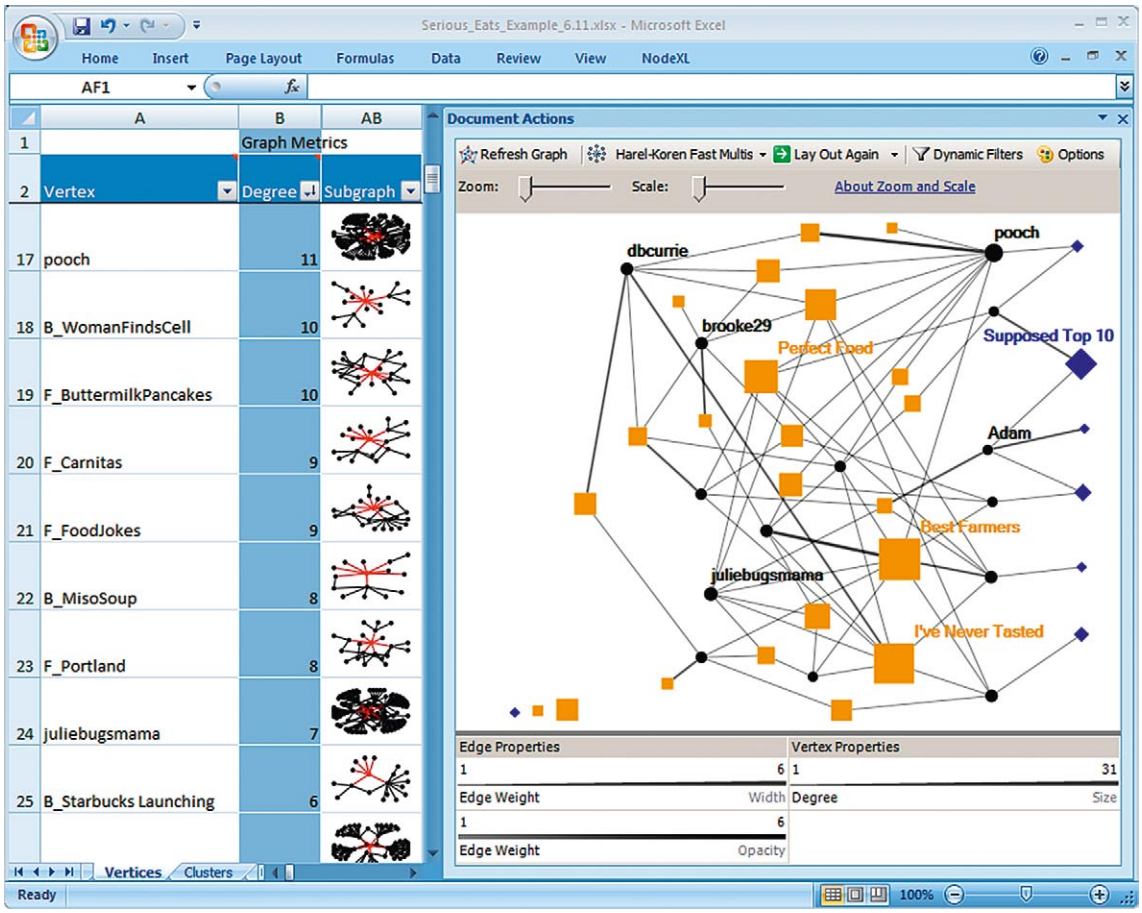


FIGURE 6.16 SeriousEats visualization emphasizing the most important people (black circles), forums (orange squares), and blogs (blue diamonds).

to interactively determine what is displayed on the graph by setting the position of sliders representing starting and ending ranges. These ranges can use data from graph metrics or other attribute data associated with the network. Another reason to filter the network graph is to create more readable and persuasive images. Using the Autofill Columns feature to selectively set the Visibility column based on predetermined criteria is an effective method to completely filter out vertices or edges so that the remaining ones can be manipulated effectively. Subgraph images provide a snapshot of each vertex's ego network, allowing you to identify common social roles and structures. This chapter also introduced multimodal networks, where vertices represent more than one type of entity (e.g., people and discussion forums) in the same network.

6.6 RESEARCHER'S AGENDA

This chapter demonstrated how value-based, categorical, and ordinal filtering of complex network graphs can produce insights and compelling displays, but the strategies for dynamic queries and filter design still allow room for improvement [9–11]. Applying filters in an orderly process that arrives at a successful outcome requires skill and creative problem solving. As network researchers gain experience, they may develop more systematic approaches to choosing and setting filters so as to emphasize important features and remove distractions [12]. Being able to save a sequence of actions and then replay it on a fresh set of data would be a useful improvement. Developing standard process models (sequences of actions) to ensure complete exploration could dramatically advance the state of the art for social media network analysis. Such a process model would be systematic yet flexible, smoothly integrating statistics and visualization [13, 14] and guiding users effectively while enabling them to explore interesting possibilities.

References

[1] C. Dunne, B. Shneiderman, Improving graph drawing readability by incorporating readability metrics: a software tool for network analysts, Technical Report, University of Maryland Technical Report HCIL-2009-13, 2009.

[2] J.P. Scott, Social Network Analysis: A Handbook, Sage, Newbury Park, CA, 2000.

[3] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences), Cambridge University Press, Cambridge, UK, 1994.

[4] R. Hanneman, M. Riddle, Chapter 17: Two Mode Networks. Introduction to Social Network Methods, University of California, Riverside, CA, 2005 Riverside (published in digital form at http://faculty.ucr.edu/~hanneman.

[5] M. Derthick, J. Harrison, A. Moore, S.F. Roth, Efficient Multi-Object Dynamic Query Histograms, Proceedings, IEEE Symposium on Information Visualization 1999, San Francisco, California, (October, 1999), pp. 84–91.

[6] S.G. Eick, Data Visualization Sliders, Proceedings 7th Annual ACM Symposium on User Interface Software and Technology, ACM Press, New York, 1994, pp. 119–120.

[7] K. Fishkin, M.C. Stone, Enhanced Dynamic Queries via Movable Filters, Proceedings. ACM CHI Human Factors in Computing Systems, ACM Press, New York, 1995, pp. 415–420.

[8] B. Shneiderman, Dynamic queries for visual information seeking, IEEE Software 11 (6) (1994) 70–77.

[9] Q. Li, C. North, Empirical comparison of dynamic query sliders and brushing histograms, Proceedings. IEEE Symposium on Information Visualization 2003, (October 2003), pp. 147–154.

[10] L. Tweedie, B. Spence, D. Williams, R. Bhogal, The Attribute Explorer, Proceedings of the CHI '94 Conference Companion on Human Factors in Computing Systems, ACM Press, New York, 1994, pp. 435–436.

[11] K. Wittenburg, T. Lanning, M. Heinrichs, M. Stanton, Parallel Bargrams for Consumer-Based Information Exploration and Choice, Proceedings 14th Annual ACM Symposium on User Interface Software and Technology, ACM Press, New York, 2001, pp. 51–60.

[12] A. Perer, B. Shneiderman, Systematic yet flexible discovery: Guiding Domain Experts through Exploratory Data Analysis, Proceedings ACM 13th International Conference on Intelligent User Interfaces, New York, NY, 2008, pp. 109–118.

[13] A. Perer, B. Shneiderman, Integrating Statistics and Visualization: Case Studies of Gaining Clarity during Exploratory Data Analysis. CHI '08: Proceedings SIGCHI Conference on Human Factors in Computing Systems, ACM, New York, NY, 2008, pp. 265–274.

[14] B. Shneiderman, Inventing discovery tools: combining information visualization with data mining, Inf. Vis. 1, (1) (2002) 5–12.

CHAPTER

7

Clustering and Grouping

OUTLINE

7.1 Introduction 93

7.2 The 2007 Senate Voting Analysis 94
- 7.2.1 *Filtering Edges to Identify Groups within a Network* 94
- 7.2.2 *Automatically Identifying Clusters* 95
- 7.2.3 *Creating Clusters Manually* 97

7.3 *Les Misérables* Character Clusters 98

7.4 Federal Communications Commission (FCC) Lobbying Coalition Network 100

7.5 Practitioner's Summary 101

7.6 Researcher's Agenda 101

7.1 INTRODUCTION

Many large networks are a complex combination of smaller groups or subgraphs. High school networks in the United States consist of subgroups of jocks, nerds, goths, and the like. Facebook networks are made up of clusters of family members, schoolmates, work colleagues, and other forms of association (Chapter 11). Legislative bodies like the United States Congress contain two main political parties and numerous smaller coalitions. Identifying groups within a network and mapping their relationship to one another can be essential to making intelligent strategic decisions. Network analysis can help identify competing or complementary groups, potential allies to form a powerful group, and individuals who can connect you to a new group. Identifying and visualizing groups is also a powerful strategy for making sense of large networks, helping one to avoid getting lost in the trees.

Social network analysis provides a set of tools for identifying and understanding groups, also called *clusters* or *communities* by network researchers. In the language of network analysis, clusters are pockets of densely connected vertices that are only sparsely connected to other pockets [1] (Fig. 7.1). For example, Figure 7.1 shows a network consisting of three densely connected pockets (within the dotted circles) that are loosely connected to each other by only a few ties.

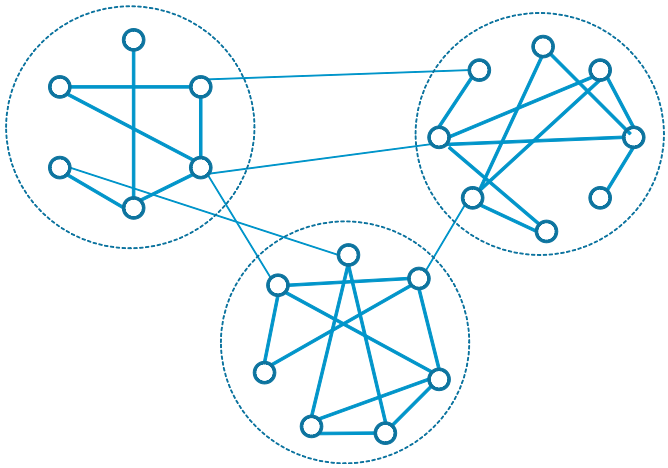


FIGURE 7.1 A network of three densely connected clusters, each shown inside a dashed circle. Ties between clusters are rare and less dense.

One of the powers of social network analysis is that it can be used to automatically identify clusters that you may not have known to look for ahead of time. These naturally occurring clusters are based on patterns of social ties rather than formal group memberships. For example, an analysis of a corporate email network (Chapters 8) can provide an authentic grouping of individuals based on communication patterns rather than

formal reporting hierarchies. In this chapter you will learn to identify and visualize clusters in NodeXL.

7.2 THE 2007 SENATE VOTING ANALYSIS

In this section you will analyze the voting patterns of United States senators in the year 2007, identifying clusters of senators connected together based on similar voting patterns. Chris Wilson of *Slate* magazine provided the original voting network data, which we have imported into NodeXL and made available at the book web site.

The Senate network is created from data that connect senators to one another based on the number of times they vote the same way (i.e., both voted yea or both voted nay to a bill). For example, Senators Akaka (D, Hawaii) and Allard (R, Colorado) voted the same only 84 times (35.4 percent of the time), whereas Senator Akaka (D, Hawaii) and Baucus (D, Montana) voted the same 208 times (84.9 percent of the time) (Fig. 7.2). Clearly the two Democrats have a stronger connection to one another than the Democrat and Republican senators. The network is undirected and is weighted based on the number of similar votes (see the Voted Same column in Fig. 7.2). Using the raw number of similar votes can be problematic in cases where senators were frequently absent for votes (i.e., were campaigning). For this reason, the data set also includes the total number of votes cast by each senator (Vertex1_Total & Vertex2_Total) and the percentage agreement (Percent_Agreement), which is calculated using the lowest of the two senators' total votes as the denominator. The Vertices worksheet includes data about each senator such as the senator's party affiliation, the state the senator represents, and the total number of votes the senator cast in 2007.

7.2.1 Filtering Edges to Identify Groups within a Network

When working with weighted networks such as the 2007 Senate network, it is often necessary to filter out some of the edges to identify subgroups. Because every senator voted the same as every other senator at least once, choosing Show Graph results in an uninformative dark mass of connections (Fig. 7.2). To make the graph more meaningful, you will want to only show edges between senators that have a high level of agreement.

Begin by selecting the fields shown in Figure 7.3 on the Autofill Columns dialog. Set the Option for Edge Visibility to "Greater Than 0.65," which is the average agreement percentage between all pairs of senators. The result is that pairs who voted the same less than 65 percent of the time will not be connected in the

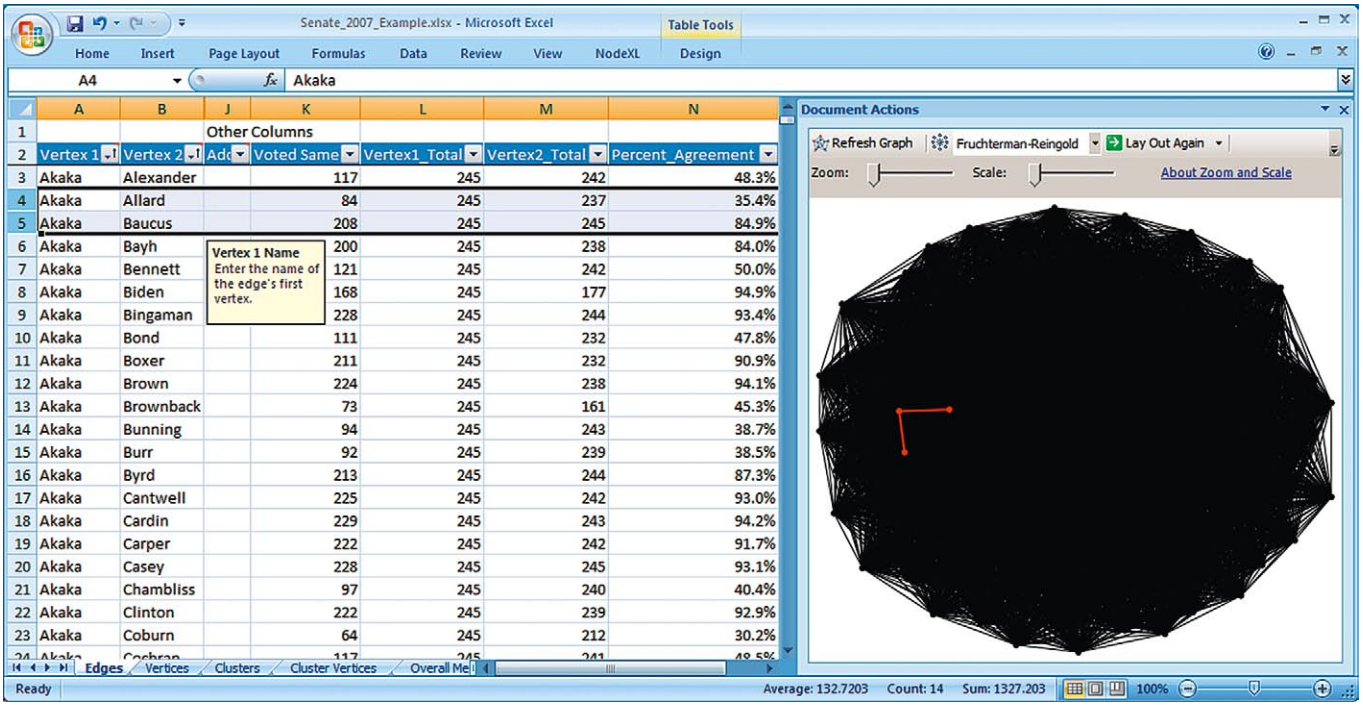


FIGURE 7.2 Unfiltered 2007 U.S. Senate co-voting network showing all 48 senators connected to each other. Other columns in the NodeXL Edges worksheet show the number of times each pair of senators voted the same and their percentage agreement. A weak tie (Akaka and Allard) and a strong tie (Akaka and Baucus) are highlighted. "Raw" visualizations like this require refinement to display useful insights.

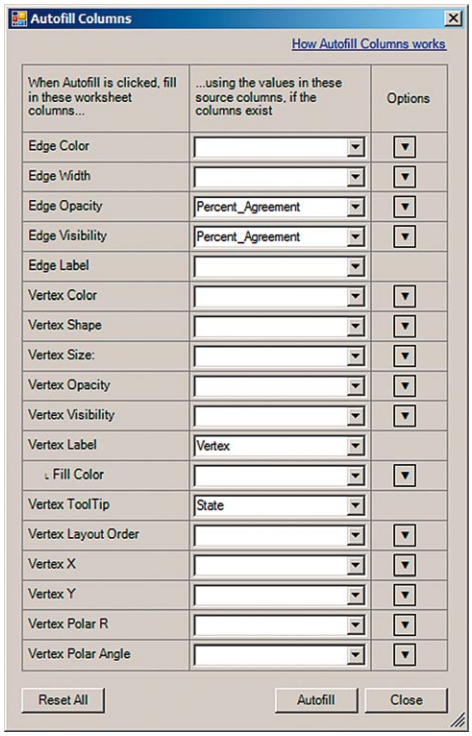


FIGURE 7.3 The NodeXL Autofill Columns window with Edge Visibility Options set to "Greater Than 0.65" and Edge Opacity Options set to a range of 0.65 (edge opacity 10) to "The largest number in the column" (edge opacity 100).

graph. As discussed previously (Chapter 6), edges that are filtered out in this way will be "skipped" instead of "hidden." As a result, they will not be shown in the graph or used in any calculations of graph metrics or clusters, which is desirable in this case. Set the Option for Edge Opacity so the minimum value is 0.65 instead of "The smallest number in the column." This ensures that there is enough difference between all of the edges that are shown since they start at .65 Percent_Agreement. After clicking the Autofill button, change the color of the edges to something lighter and set the Shape on the Vertices worksheet to Label. Refresh the graph using the Fruchterman-Reingold layout and continue to select Lay Out Again until you see two distinct collections of vertices (Fig. 7.4).

7.2.2 Automatically Identifying Clusters

NodeXL can automatically identify clusters based on the network structure. NodeXL version 1.0.1.113 uses an algorithm that looks for groups of densely clustered vertices that are only loosely connected to vertices in another cluster (see Wakita & Tsurumi [2] for the specific algorithm and Newman [3] for background and other community identifying algorithms). The number of clusters is not predetermined; instead the algorithm dynamically determines the number it thinks is best.

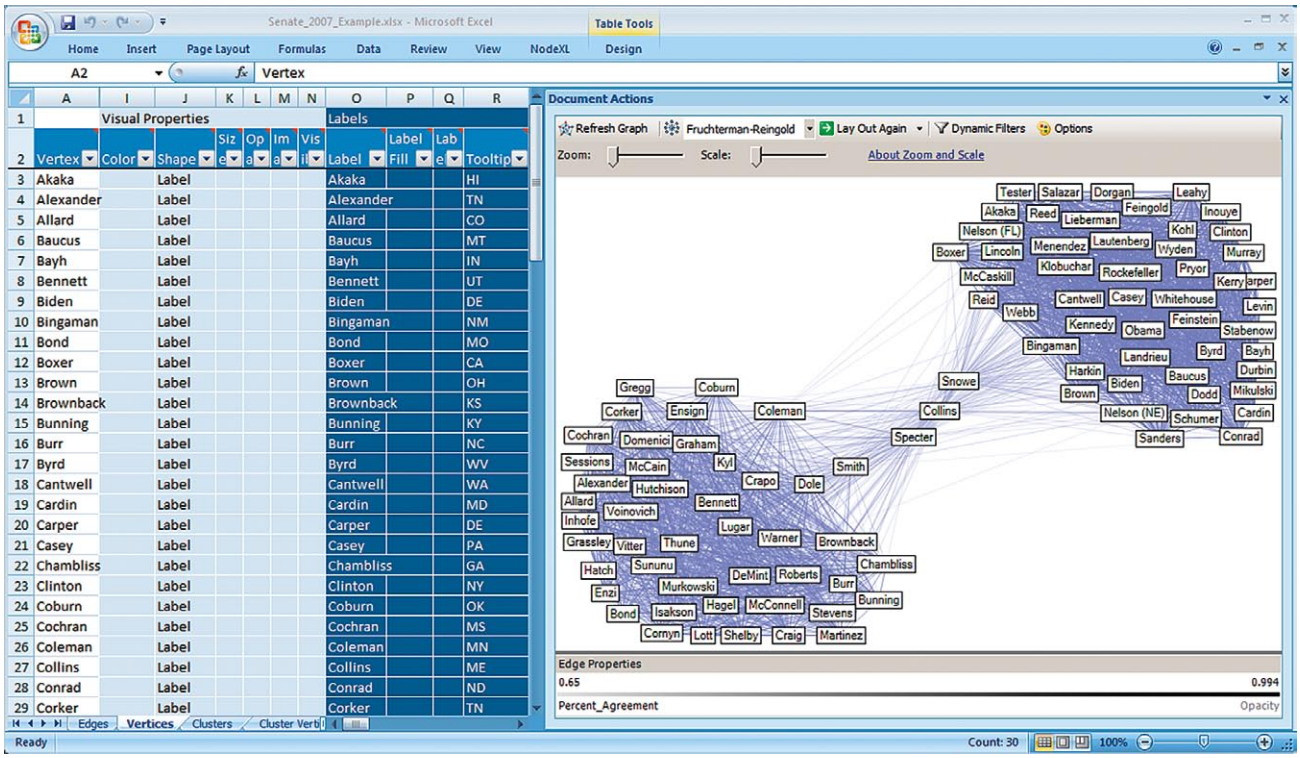


FIGURE 7.4 The 2007 U.S. Senate co-voting network graph after using Autofill Columns (see Fig. 7.3), setting Vertex Shape to Label, and Edge Color to (128, 128, 192). The Fruchterman-Reingold Layout visually creates two clear groups with a few boundary spanners in the middle after enough iterations.

ADVANCED TOPIC

Changing Advanced Layout Options for the Fruchterman-Reingold Layout

The Fruchterman-Reingold layout is a force-directed layout algorithm, which treats edges like springs that move vertices closer or further from each other in an attempt to find an equilibrium that minimizes the "energy" of the system (see https://nwb.slis.indiana.edu/community/?n=VisualizeData.Fruchterman-Reingold for details). NodeXL allows you to change settings for this layout by opening the Layout Options dialog available in the drop-down menu for the layout types (see the Chapter 6 description of Fig. 6.5 and Fig. 6.6). The following parameters are available for the Fruchterman-Reingold layout:

- *Strength of the repulsive force between vertices.* Higher numbers make vertices push away from each other more. Increase this to 7.0 to reduce Vertex overlap when working on the 2007 Senate network data (Fig. 7.5).
- *Iterations per layout.* Determines the number of times the Fruchterman-Reingold layout will be performed when the graph is refreshed (or updated with Lay Out Again). Each time the layout is run, it begins from the current position of the vertices. It is often necessary to run the layout several times to get to a desirable layout; however, doing so can be slow and computationally intensive. When working with large networks, you can lower the iterations per layout and, if needed, use the Lay Out Again button multiple times until the graph settles in on a relatively stable position. You will likely need to click on Lay Out Again at least once after you choose Refresh Graph when working on the Senate network data to achieve the desired layout. Alternatively you can increase the iterations per layout.

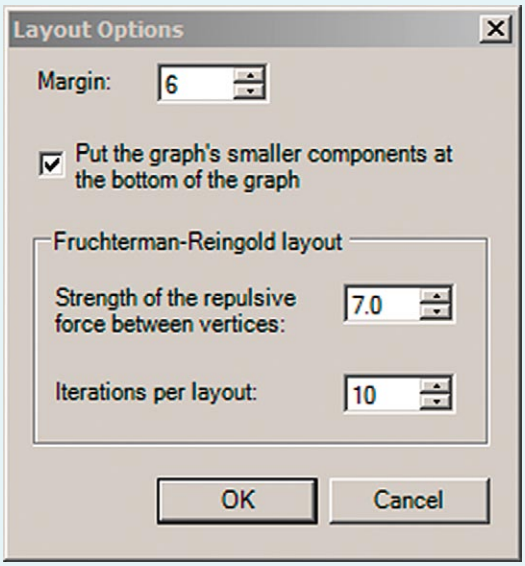


FIGURE 7.5 NodeXL Layout Options window used to increase the repulsive force between vertices, helping spread out vertices to reduce overlap.

Each vertex is assigned to exactly one cluster, meaning that clusters do not overlap. The number of vertices in each cluster can vary significantly. In some cases, a single cluster can encompass all vertices, whereas in other cases, a cluster can consist of a single vertex.

To run the clustering algorithm, click on the Find Clusters button in the Analysis section of the NodeXL menu ribbon. This will automatically populate data on the Clusters worksheet and the Cluster Vertices worksheet. The Clusters worksheet (Fig. 7.6) creates a name for each cluster (e.g., clusters C14, C143, and C193) and assigns a distinct vertex color and vertex shape combination to each cluster. For example, cluster C14 was automatically given the color red (255, 0, 0) and the shape disk. The cluster vertex color and vertex shape will not be shown on the graph until you go to the NodeXL ribbon, select Graph Elements, check Clusters, and click on Refresh Graph. Information on the Vertices worksheet in the Color and Shape columns will be ignored but can be shown later by un-checking the same Clusters Graph Element and choosing Refresh Graph.

The Cluster Vertices worksheet (Fig. 7.7) maps each vertex into a single cluster. The algorithm identified two large clusters that roughly correspond to the Republican and Democratic parties, as well as a single person cluster (Collins) who plays a unique boundary spanning role. The algorithm grouped Snowe and the two independent senators (Lieberman and Sanders) in the Democratic cluster even though they are not technically Democrats.

Unfortunately, the automatically assigned colors are not what we would expect based on convention in the United States where red is associated with the Republicans and blue is associated with the Democrats. The clustering algorithm lacked this political and cultural insight, as blue was assigned to the cluster of mostly Republicans and red was assigned to the Collins cluster. You can correct this color coding by changing the colors assigned to each cluster in the Cluster Vertices worksheet (see Fig. 7.6). You can also change the shape to Label in order to more clearly see the senators' names.

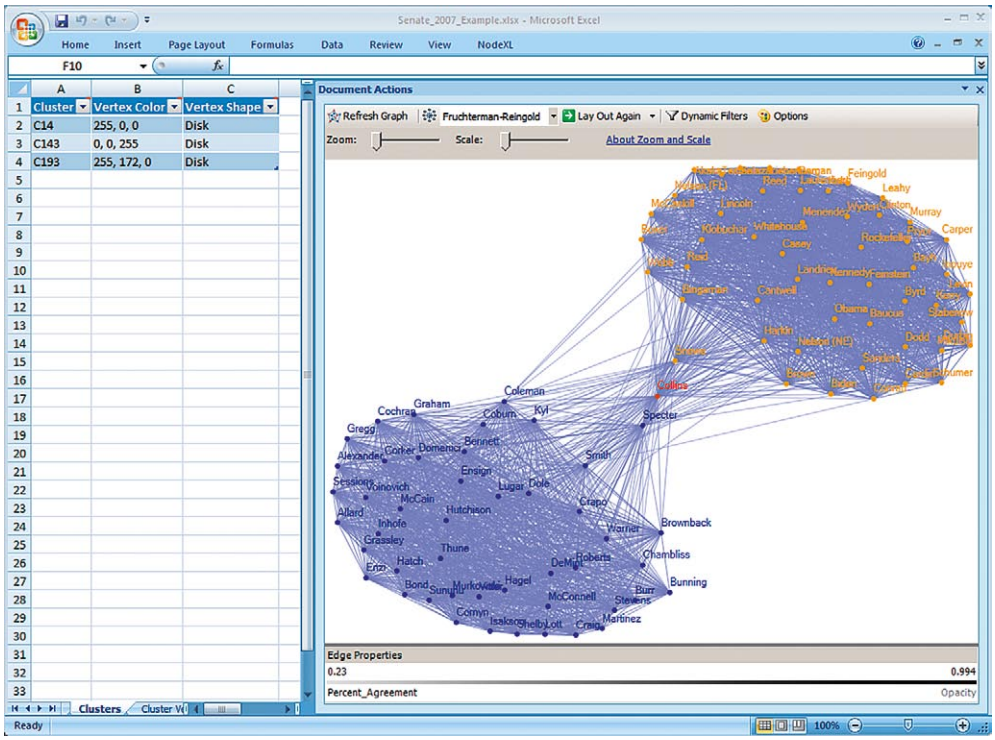


FIGURE 7.6 The 2007 U.S. Senate co-voting network after applying the Find Clusters feature and choosing to view the Graph Element–Clusters from the NodeXL ribbon. Each of the three automatically identified clusters is given a name, color, and shape on the Clusters worksheet.

	A	B	C
1	Cluster	Vertex	
2	C14	Collins	
3	C143	Specter	
4	C143	Smith	
5	C143	Coleman	
6	C143	McCain	
7	C143	Voinovich	
8	C143	Murkowski	
9	C143	Warner	
10	C143	Thune	
11	C143	Sununu	
12	C143	Stevens	
13	C143	Shelby	
14	C143	Roberts	

Clusters | Cluster Vertices

FIGURE 7.7 The NodeXL Cluster Vertices worksheet that maps each vertex into exactly one cluster. For example, Collins is the only member of Cluster C14, whereas Specter, Smith, and the others are members of Cluster C143.

7.2.3 Creating Clusters Manually

There are times when you already know what clusters each vertex belongs to and are interested in seeing the connections between and within the clusters. When group affiliations are already known, you can manually enter data into the Clusters and Cluster Vertices worksheets. In our example, you may want to visually represent the official party affiliation of each senator, which differs slightly from the automatically identified clusters found in the previous section. Try this by copying and pasting the Vertex column from the Vertices worksheet into Column B (vertex) of the Cluster Vertices worksheet, replacing the content already there. Then copy and paste the Party column from the Vertices worksheet into Column A (clusters) of the same worksheet. Each of the senators (i.e., vertices) is now assigned to a party (i.e., cluster) as shown in Figure 7.8.

Next, create a cluster representing each party type on the Clusters worksheet. Make sure you don't leave out any clusters that were included on the Cluster Vertices worksheet. In this case there are four unique clusters as shown in Figure 7.9. Assign an appropriate color to each, such as blue for Democrat, red for Republican, and dark orange for the two Independent clusters. Then choose Label for the vertex shape to make the names more readable.

Choose Refresh Graph to see a network graph that looks like Figure 7.9. You'll notice a clear clustering between the Republican and Democratic senators, as well as the tendency of the independent senators to vote Democratic. The image also identifies Senators Snowe, Collins, and Specter as clear bridge spanners between the parties, voting equally often with each party. Senator Specter's decision to switch from the Republican to the Democratic party in 2009 does not seem as surprising when seeing the 2007 U.S. Senate co-voting network.

You may want to further explore this data set using techniques introduced in earlier chapters. For example, you can calculate metrics to find senators with the highest betweenness centrality or other metrics. You can dynamically explore the network by hovering over a senator's name to see what state he or she is from, clicking on the senator's vertex to see red edges showing whom he or she is connected to, or using dynamic filters to change the edge-filtering threshold from 65 percent to higher or lower values. For example, increasing the edge-filtering threshold reveals that the Democratic party voted more as a block than the Republican party (i.e., they had many more edges connecting their members at a 90 percent threshold).

	A	B	C
1	Cluster	Vertex	
2	D	Akaka	
3	R	Alexander	
4	R	Allard	
5	D	Baucus	
6	D	Bayh	
7	R	Bennett	
8	D	Biden	
9	D	Bingaman	
10	R	Bond	
11	D	Boxer	
12	D	Brown	
13	R	Brownback	
14	R	Bunning	

Clusters | Cluster Vertices

FIGURE 7.8 The Cluster Vertices worksheet used to manually map each vertex (senators) to a cluster (political party).

7.3 *LES MISÉRABLES* CHARACTER CLUSTERS

To see how the automatic clustering algorithm works on a different type of network, open up the *Les Misérables* character co-appearance network introduced earlier (Chapter 5). Use the Find Clusters feature and visualize the network using the same settings as before (see Fig 7.10). Many of the clusters are visually grouped

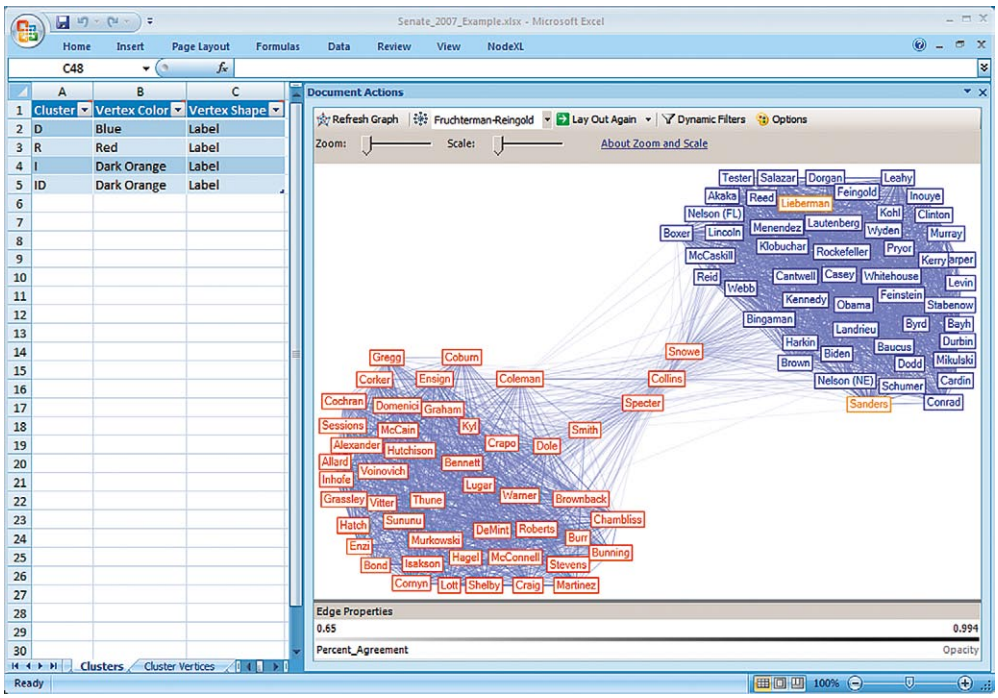


FIGURE 7.9 The NodeXL Clusters worksheet showing the four unique clusters and their associated colors and shapes: Democrats (D) in blue, Republicans (R) in red, and Independents (I) and Independent Democrats (ID) in dark orange.

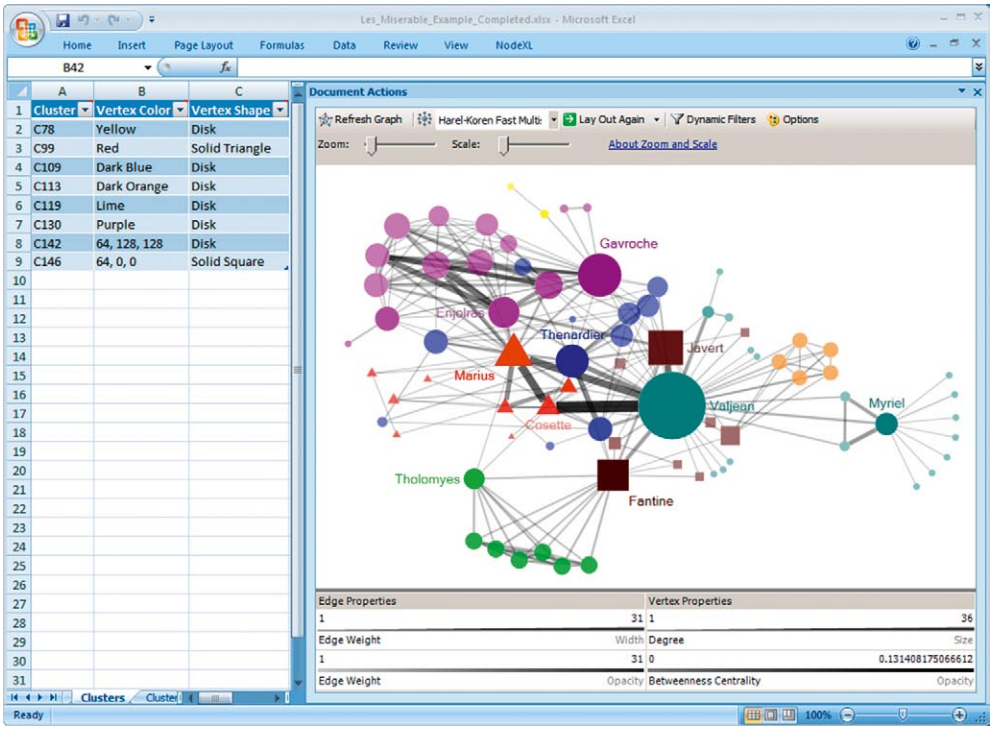


FIGURE 7.10 *Les Misérables* character co-appearance network with automatically identified clusters represented by unique color and shape combinations.

ADVANCED TOPIC
Additional Clustering Algorithms

There are a large and growing number of network clustering algorithms, also called community structure algorithms (see Newman & Girvan [1] for an overview including the results from a different algorithm used on the same *Les Misérables* network data). Many community and cluster discovery algorithms don't scale well for large networks, forcing a tradeoff between quality and speed. Most algorithms, including the one used by NodeXL, are based on undirected, unweighted networks. Although you can apply them to more complex networks and often get reasonable results, you may need to use a more specialized community detection tool that includes a range of algorithms that take into consideration the specific properties of your network data. Most tools will output data into a format that can easily be pasted into the Clusters and Cluster Vertices worksheet, allowing you to take advantage of NodeXL's rich visual features. Future versions of NodeXL will include additional clustering algorithms.

In addition, there are many non-network clustering algorithms that can operate on collections of vertex attributes. These include k-means, hierarchical agglomerative, hierarchical divisive, and many more. For example, a non-network clustering algorithm could be used to cluster people into groups that have similar participation patterns (e.g., those who use a certain collection of features similarly). This could then be represented on the network graph to see how those who use the system in a similar way are connected to one another.

together by the Harel-Koren Fast Multiplex layout. For example, the student revolutionaries shown in purple are clustered together, as are the collections of orange and green vertices. However, some vertices that are not adjacent to one another are identified as being in the same cluster, suggesting a closer analysis of their relationship. For example, Javert and Fantine are grouped in the same cluster (Brown squares), which makes sense given their shared ties with one another, Valjean, the Thenardiers, and other minor characters that are not apparent in the current visual arrangement. This demonstrates the value of visualizing clusters based on color and shape, particularly in complex networks with multiple clusters. The graph also shows the limitations of clustering algorithms like the one used in NodeXL that can only classify a vertex into a single cluster.

For example, Valjean and Marius are boundary spanners who could fit well within several clusters but by necessity are forced into just one.

7.4 FEDERAL COMMUNICATIONS COMMISSION (FCC) LOBBYING COALITION NETWORK

The power of social network analysis and visualization is best realized when combining the approaches discussed so far (Chapters 4 through 7). In this section you will see a network visualization created in NodeXL by Pierre de Vries, a Research Fellow at the University of Washington's Economic Policy Research Center. This example is based on his submission to the first annual Journal of Social Structure Visualization Symposium held in 2010.

The social network shown in Figure 7.11 shows the relationship between organizations that lobbied the FCC on just one of the hundreds of issues that are before it: Docket 01-92 on intercarrier compensation. This proceeding is a battle over the fees that telephone companies pay each other when phone calls move between them. Legislation requires that most interactions between organizations and the FCC are publicly documented. This network data were extracted from metadata about these filings reported via the FCC's Electronic Comment Filing System.

The network captures all links between lobbying organizations during the duration of the proceeding (2001–2008). Vertices represent organizations that filed. Edges connect organizations that filed jointly, with edge weight representing the number of joint filings. Darker edges reflect higher edge weights.

Vertex size is proportional to the total number of filings that an organization made and is a proxy for lobbying investment. Well-funded companies and trade associations are prominent, although they are not necessarily well connected because they don't need to be.

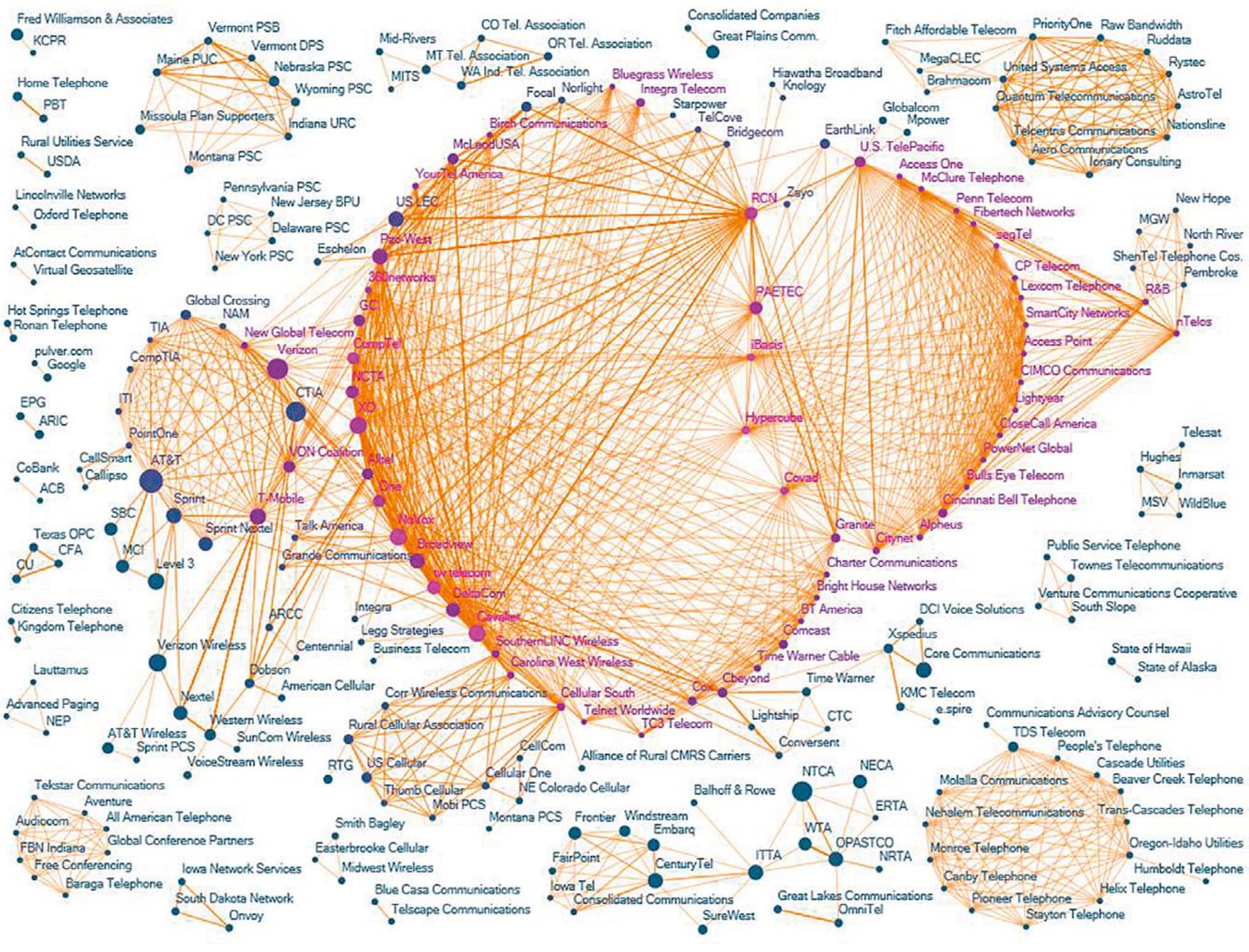


FIGURE 7.11 Lobbying Coalition network connecting organizations (vertices) that have jointly filed comments on U.S. Federal Communications Commission policies (edges). Vertex size represents number of filings and color represents eigenvector centrality (pink = higher). Darker edges connect organizations with many joint filings. Vertices were originally positioned using Fruchterman-Reingold and hand-positioned to respect clusters identified by NodeXL's Find Clusters algorithm.

Influence, measured by eigenvector centrality, is represented by the color of the nodes: the pinker the node, the better connected it is. Small companies can gain influence by linking different coalitions of local exchange carriers. Some organizations that hardly filed at all may be influential (i.e., small and pink), thanks to their many, straddling connections.

The Fruchterman-Reingold algorithm was used to prepare a preliminary network layout for these data. Next, the Find Clusters feature was used to identify distinct clusters, which were used to guide the manual placement of vertices into visually intelligible positions. Once clusters were represented by their placement, the cluster colors were shut off so that color could be used to represent eigenvector centrality. The file was exported as a high-resolution image, making it possible to zoom in on different sections of the graph and still read the vertex labels.

Interviews with policy practitioners confirm that the graph clusters correspond to real-world coalitions and alliances. For example, the clusters in the top right and bottom right are rural telephone companies; a loose coalition of rural trade associations and rural competitive local exchange carriers can be seen at the bottom middle of the illustration. The heavily connected cloud in the center of the chart shows competitive local exchange carriers who band together at various times in various permutations to make up strength in numbers.

Because graph clusters and evolution represent real-world behavior, they can be used to improve public understanding and lobbying effectiveness. Insiders can use graphs to identify potential collaborators or defectors (e.g., by looking for coalition members who are bridges between groups). They can also use changes in connectedness to track the emergence or breakdown of consensus in a proceeding. Outsiders can grasp the overall structure and evolution of the proceeding without having to understand the entire record.

7.5 PRACTITIONER'S SUMMARY

Many large networks are a complex combination of smaller groups, also called clusters, communities, or subgraphs. Clusters are pockets of densely connected vertices that are only sparsely connected to other pockets. Identifying clusters and their relationships to one another is essential to making sense of large networks. Clusters can be automatically identified using community detection algorithms such as the one behind NodeXL's Find Clusters feature [2]. If group memberships are known (e.g., Republican and Democrat party affiliations), vertices can be manually assigned to clusters. Whether clusters are automatically or manually created, they can be visualized in NodeXL with unique vertex color and shape combinations that indicate membership in different clusters. Clusters can also be identified using automatic layout algorithms such as Fruchterman-Reingold, which tend to group densely connected vertices close together. However, it may be necessary to filter out some edges before distinct groups are identifiable, as was demonstrated with the Senate network data set. Combining the strategies discussed so far in Part II can facilitate the creation of insightful and visually appealing graphs that can be the basis for understanding, explanation, decision making, and persuasion.

This taste of social network analysis is just the beginning. You have now used the main features of NodeXL. However, knowing how to apply them effectively to solve real-world problems can take a significant amount of practice. As you tackle increasingly complex social media data sets and challenges, you'll be contributing to a lively and growing field.

7.6 RESEARCHER'S AGENDA

Rapid progress in the past decade has turned the esoteric topic of clustering algorithms into a hot research area. Newman's work (see [1], [3], and Additional Resources) produced substantially improved strategies for organizing and presenting complex networks in meaningful ways, which stimulated further work on weighted, directed, bipartite, and multimodal networks. Because most algorithms run slowly, the research community has sought algorithms that can be adapted to run on multicore and specialized graphics processors that are increasingly embedded in modern computers [4]. The next step is to compute network clusters efficiently using parallel computers and cloud computing techniques. Most techniques create clusters based on edge connections, but an alternate strategy is to cluster nodes by attribute values of the nodes—for example, all people who graduated from the same university are in the same cluster [5]. This introduces the visual challenge of dealing with multiple memberships.

References

[1] M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2004) 026113.

[2] K. Wakita, T. Tsurumi, Finding community structure in mega-scale social networks: [extended abstract]. in: Proceedings of the 16th International Conference on World Wide Web (Banff, Alberta, Canada, May 08–12, 2007). WWW '07. ACM, New York, 2007, pp. 1275–1276.

[3] M.E.J. Newman, Detecting community structure in networks, Eur. Phys. J. B 38 (2004) 321–330.

[4] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebre, "Fast unfolding of community hierarchies in large networks" *arXiv* (2008): 0803.0476. Available at: http://works.bepress.com/lambiotte/4

[5] A. Aris, B. Shneiderman, Designing semantic substrates for visual network exploration, Inf. Visual. J. 6 (4) (2007) 1–20.

Additional Resources

Barnes, J., & Hut, P. (Dec. 1986). A hierarchical o(n log n) force-calculation algorithm. *Nature*, *324*(6096), 446–449.

Brandes, U., Delling, D., Gaertler, M., Görke, R., Hoefer, M., Nikoloski, Z., et al. (2008). On Modularity Clustering. *IEEE Transactions on Knowledge and Data Engineering*, *20*(2), 172–188.

Girvan, M., & Newman, M.E.J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, *99*, 7821–7826.

Newman, M. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America*, *103*, 8577–8582.

Newman, M.E.J. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, *98*, 404–409.

Newman, M.E.J. (2003). The structure and function of complex networks. *SIAM Review*, *45*, 167–256.

PART III

SOCIAL MEDIA NETWORK ANALYSIS CASE STUDIES

This section (Chapters 8–15) includes a number of contributed chapters that analyze networks created by a particular social media platform. The chapters need not be read in order, but they do assume you are familiar with social media and social network analysis fundamentals (Part I) and NodeXL (Part II). Each chapter in this section discusses how to extract network data, what questions can be addressed, and how to analyze and visualize the networks to reach actionable insights. Specific chapters focus on email (Chapter 8), threaded conversation (Chapter 9), Twitter (Chapter 10), Facebook (Chapter 11), the World Wide Web (Chapter 12), Flickr (Chapter 13), YouTube (Chapter 14), and wikis (Chapter 15).

CHAPTER

8

Email: The Lifeblood of Modern Communication

OUTLINE

8.1 Introduction 105

8.2 History and Definition of Email 106

8.3 Email Networks 107

8.4 What Questions Can Be Answered by Analyzing Email Networks? 108
8.4.1 Personal Email Network Questions 109
8.4.2 Organizational Email Network Questions 109

8.5 Working with Email Data 110
8.5.1 Preparing Email 110
8.5.2 Importing Email Networks into NodeXL 111

8.6 Cleaning Email Data in NodeXL 113
8.6.1 Remove Duplicate Email Addresses for the Same Individual 113
8.6.2 Merge Duplicate Edges 114

8.7 Analyzing Personal Email Networks 115
8.7.1 Creating an Email Overview Visualization 115
8.7.2 Creating an Expertise Network Email Graph 118

8.8 Creating a Living Org-Chart with an Organizational Email Network 119
8.8.1 TechABC's Organizational Unit Email Network 119
8.8.2 Normalizing and Filtering TechABC's Data 119
8.8.3 Creating an Overview of TechABC's Communication Patterns 120
8.8.4 Examining TechABC's Research Division 121

8.9 Historical and Legal Analysis of Enron Email 122
8.9.1 Identifying Key Individuals Using Content Networks 122

8.10 Practitioner's Summary 124

8.11 Researcher's Agenda 124

8.1 INTRODUCTION

Email has permeated society more than any other form of social media. It is hard to remember a time when inboxes sat on desks and spam was only a processed meat. Today, an estimated 1.4 billion worldwide email users send nearly 50 billion nonspam emails each day.[1] Numbers continue to increase annually with no sign of slowing down in the near future. Email is the de facto form of communication for many corporations, nonprofits, and government agencies. Email and email lists are used to keep extended families in touch, coordinate neighborhood activities, support medical patients, share cutting-edge research, solve technical problems, and perform a host of other activities.

In April 2009, 90 percent of U.S. Internet users sent or received an email message, making it the most common Internet activity of all.[2] More than half of them did

[1] www.radicati.com/?p=3237

[2] www.pewinternet.org/Trend-Data/Online-Activites-Total.aspx

so at least once a day. Unlike many social media tools, email is widely used among nearly every demographic group.[3] Groups like the Association of Cancer Online Resources (www.acor.org) demonstrate that email lists can reach many age groups including older adults whose Internet use is minimal. The steady flow of email arrives nearly continuously to personal computers and, increasingly, to mobile devices throughout the globe.

The integration of email into everyday life makes email networks the most accessible and in many cases most accurate source of data for mapping actual social and work relationships. Analyzing one's personal email collection is a lot like looking in a mirror. Systems like PostHistory demonstrate the interest people have in seeing representations of their social media and show how maps of social connections and activity can promote sustained engagement and storytelling around important events [1]. Network visualizations provide an objective representation of one's social ties, encouraging self-reflection and providing a guide to social hygiene. These maps and reports may help you realize unappreciated or forgotten relationships, or identify a past working group that could be rekindled for a current project. They can help us overcome some of our memory biases such as weighing recent events more or remembering things we've initiated more than those initiated by others. A visualization of your personal email network can be shared with other people to explain your social world. For example, new employees could benefit from a summary of important social ties and collaborative groupings related to a job role or position. Personal email collections are of increasing importance to historians, researchers, archivists, and lawyers who are engaged in the discovery and preservation of electronic records, much of which are in the form of email.

Analyzing organizational email networks and email lists can provide a wealth of social information that can inform important decisions and support novel interventions. Organizations can identify unique social roles, individuals who span the gaps between organizational silos, internal influencers, and employees in need of creating more connections. This information can be used as one input to help inform personnel hiring and promotion, improve retention, and spread important messages through a company. Analysis of organic employee clusters, rather than formal organizational charts, can be used to inform the community of practice formation and organizational restructuring, and it can help integrate relationships after mergers. Expertise networks can be identified by the use of keywords to infer topics, leading to more intelligent workgroup formation and information sharing. The analysis of an internal company or public email list can help identify experts on a topic, monitor the health of the community over time, and identify potential candidates for leadership roles in the list. Because it is based on actual behavior instead of self-reports that can be biased in organizational contexts [2], its validity is high.

Working with email poses several ethical challenges. Although company email is far from private and unless encrypted it is far from secure, many users don't realize just how public their email is. A 2007 survey found that nearly half of the 304 U.S. companies surveyed monitored email use.[4] More than a quarter of these companies had fired workers for email misuse. A related survey from 2006 found that 24 percent of employers had email subpoenaed by courts and regulators. Employers must walk a fine line between controlling the risks of litigation and security breaches by employees and not coming across as Big Brother. In such an environment, they must carefully consider the risks and rewards associated with analyzing company email collections. Researchers must also be careful to receive proper approval from list owners, managers, and members; using pseudonyms when advisable;[5] and protecting members' privacy. For corporations and researchers, transparency is needed when articulating the goals of the analysis, the procedures for assuring confidentiality of message content, and the decisions that will be informed by the analysis. Options for employees or research subjects to opt out or prefilter email may be desirable. Although an analysis of email collections poses some risks, social network analysis can be less intrusive than many other methods for understanding social interaction. It provides an interesting midway point for those willing to share who they talk to but not what they say.

8.2 HISTORY AND DEFINITION OF EMAIL

Electronic mail, or email, is an electronic message transmitted over a communications network, typically as a text file with optional attachments. Email is older than the Internet itself. Initially in the 1960s, email-like messages were sent between users of the same mainframe computer. Those who accessed the same mainframe or host computer through terminals could exchange messages.

[3] www.pewinternet.org/Infographics/Generational-differences-in-online-activities.aspx

[4] http://press.amanet.org/press-releases/177/2007-electronic-monitoring-surveillance-survey

[5] Changing the user names and personal identifiers included in a network data set is a common form of anonymization. However, it is ineffective in some situations in which a version of the original graph is available for analysis. The often-unique patterns found around each vertex in a network can be used to reidentify some anonymized entities.

In the 1960s and 1970s, many companies used this approach to allow employees to contact other employees located throughout the world in different branch offices or subsidiaries. Email became the "killer app" of ARPANET, the computer network developed by the United States Department of Defense that evolved into the Internet. In 1971, Ray Tomlinson sent the first network email, using an "@" symbol to separate the user's name and the host computer's name. By 1973, approximately three-fourths of all ARPANET data traffic was email. Over time, email became more standardized and increasingly interoperable between different computer and network systems. Applications for interacting with email became more feature rich and usable. Email services became essentially free at the point of incremental use from an end-user perspective with popular web services such as Hotmail, Yahoo mail, and Gmail. Although email services may have access fees, each additional email sent or received does not typically impose an additional cost to the user. Today users can access and compose email messages via a variety of mobile devices including laptops, netbooks, and smart phones and their less intelligent but more common "feature" phones. The widespread use of email has made it mandatory in many professional settings.

Most readers are familiar with email as everyday users. Some important technical characteristics make email particularly powerful:

- *Flexible form.* Email can be a simple plain-text message, a richly formatted newsletter, or even an interactive survey. With attachments, nearly any type of digital content can be sent, provided it is not too large a file. This flexibility allows email to support informal banter, semiformal memos, and formal letters. This flexibility also leads to email overload in which the same channel is used to host conversations, store files, track tasks, and manage transactions.
- *Asynchronous.* The asynchronous nature of email allows people to send and receive messages on their own time, without interrupting others. The lack of immediate feedback in a text-only medium can lead to misunderstandings, but it can also encourage more careful and thorough contributions. The standard reverse-chronological order, where the newest messages are shown at the top of the inbox, makes the asynchronous nature of email more manageable by helping people distinguish new messages from old.
- *Broadcast.* Emails can be sent to any number of people simultaneously. Ad hoc groups can be created on the fly by sending to multiple email addresses and using the common Reply to All feature. Listserv and other email list software tools allow thousands of users to communicate and listen in on the conversations of others.
- *Push technology.* Email is considered a push technology; the sender decides what shows up in the receiver's inbox without any action on the receiver's end. This is great when trying to get someone's attention, but is also the reason so much unwanted email spam gets sent around.
- *Threaded conversation.* Email messages are often organized into a threaded pattern consisting of messages, replies to messages, replies to replies, and so forth. This pattern mimics the natural turn-taking of spoken conversations, albeit with less frequent turnover. Threads also extend the structure of spoken interaction by enabling the creation of multiple parallel lines of conversation. If threaded properly, related messages are grouped together into a single linked collection of related messages within their context.

Services such as Usenet and discussion forums, also known as bulletin boards or web boards, share many of these characteristics making them a close cousin to email lists often called Listservs. We discuss these in detail in Chapter 9.

8.3 EMAIL NETWORKS

In a standard email network, vertices represent email addresses or corresponding people and edges or ties are created when a message is sent from one email address to another. Edges are directed because messages are transferred from a sender to a receiver. These ties are weighted by the number of messages sent between two individuals. Table 8.1 shows a summary of the information found in seven messages pulled from Derek's personal email collection. These data are converted into an "edge list" and visually represented in Figure 8.1.

Notice that the six rows in the Edges tab are a re-representation of the seven individual email messages shown in Table 8.1. The new edges are helpful for understanding sender-receiver relationships that are not as

TABLE 8.1* An Email Network Edge List

From	To	Cc	Subject
Derek	Ben		HCIL Brownbag
Derek	Marc, Ben		Travel Plans
Derek	Marc	Anna	Registration
Marc	Derek		Re: Travel Plans
Carol	Derek		Tuesday meeting
Marc	Derek	Anna	Re: Registration
Marc	Derek		Next Steps

**This email network edge list contains seven messages from Derek's personal email collection that includes nine unique edges (including both To and Cc) and five vertices.*

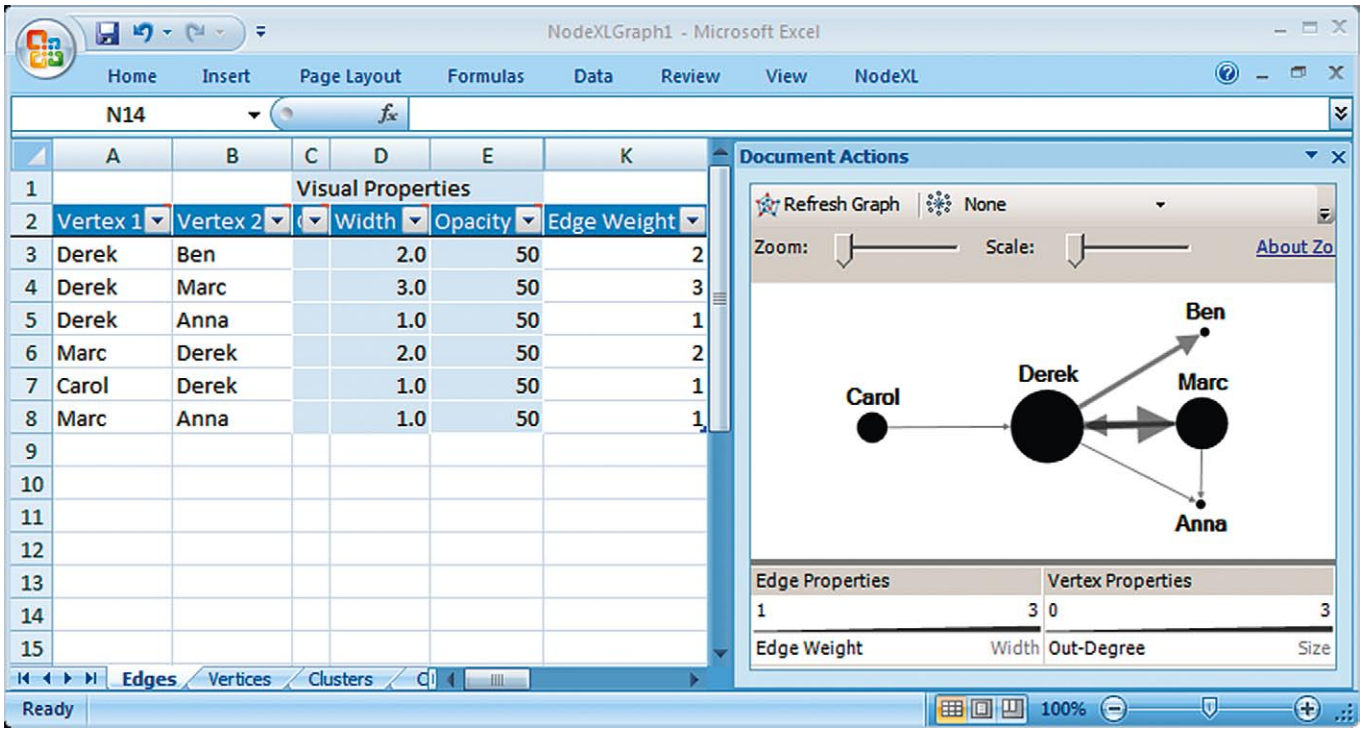


FIGURE 8.1 A simple email network visualized in NodeXL. Arrows point from the sender to the receiver(s). Edge thickness (1 to 3) is based on number of messages exchanged. Edge opacity is set to 50. Vertex size (1.5 to 5) is based on out-degree or number of messages sent.

obvious when seen in the form of a standard list of email messages. All people in the To and the Cc email fields are counted as receivers when tallying up the edge weight. For example, two messages were sent from Derek to Ben; the first was sent only to Ben and the second was also sent to Marc. Derek's message to Marc copied in Anne, so an edge is created between Derek and Anne. The power of this representation is that tens of thousands of email messages among a group of people can be captured in just a few hundred rows. Alternate ways of handling the data are discussed next.

A standard email network can be aggregated to create networks that show the connections between different social groupings. For example, vertices can represent company work groups, organizational departments, local branches, or entire organizations. Edges can represent the aggregate number of messages sent between people associated with different groups (i.e., vertices). For example, a directed edge pointing from the marketing department to the development department with a weight of 100 would suggest that marketing employees sent 100 messages to development employees. The use of organization elements in host names (e.g., @umd.edu versus @cs.umd.edu) as part of email addresses can facilitate this type of analysis by identifying people from different departments, although the frequent use of web mail (e.g., @yahoo.com) makes this technique problematic for studying broader populations. Alternatively, edges may represent the number of unique individuals who have sent emails from one department to another. For example, in our prior scenario there may have only been five people that sent the 100 messages, resulting in an edge weight of 5. A graph based on these networks provides an overview of the departmental relationships within an organization, highlighting the most connected departments and the most socially isolated ones. Section 8.8 provides an example of a summarized network showing connections between workgroups within a large technology company.

8.4 WHAT QUESTIONS CAN BE ANSWERED BY ANALYZING EMAIL NETWORKS?

Email messages can be analyzed as part of a larger corpus. Table 8.2 shows the three main types of email collections (personal, organizational, and community), each of which may be analyzed by a current participant or an outside observer.[6] Personal email collections include messages sent or received by an individual. Organizational

[6]Table 8.2 is loosely based on a similar figure provided by Perer, Shneiderman, and Oard who characterized the types of interactions people have with current and archival email collections [3].

TABLE 8.2 Types of Analysis for Email Collections with Different Scales and Observers

	Personal	Organizational	Community
Current Participant	Region A: Analyzing your own email	Region B: Analyzing your organization's email	Region C: Analyzing ongoing conversations in a community email list in which you participate
Outside Observer	Region D: Analyzing another person's email	Region E: Analyzing another organization's email	Region F: Analyzing a community email list archive in which you do not participate

email collections include messages sent and received by members of an organization. More generally, they are the aggregate of several individuals' personal email collections. Community email collections include messages sent to an email list address that get forwarded to a group of subscribed members. Email lists may be public, where anyone can participate and view prior messages, semipublic, where anyone who registers can participate and see the archive, or private, where only invited or approved members can participate and view prior messages.

The goals and process of the analysis are different for each of the regions specified in Table 8.2. Outside observers such as lawyers, historians, and researchers analyze email collections for historical, research, or legal reasons. In contrast, current participants such as managers, community administrators, list owners, and members analyze email collections to help inform decisions. Outside observers can benefit considerably from overviews that provide context before delving into specifics. In contrast, current participants typically understand the overall context and can delve into specifics quickly, although they may be biased in their perceptions. There are fewer privacy concerns when analyzing one's own email (Region A) or public community email lists (many communities in Regions C and F) than when analyzing organizational email (Regions B and E) or another person's email archive (Region D).

We examine personal and organizational email collections in this chapter and discuss community collections in Chapter 9 since they are similar to other community-based threaded conversation tools like discussion forums and Usenet. We discuss preparing, cleaning, and importing email data in this chapter.

8.4.1 Personal Email Network Questions

Several questions can be asked about personal email network data sets:

- *Individuals.* Who are important individuals within the network? For example, who are boundary spanners who link across clusters of contacts? Who is contacted most often? Who are the most active discussants of a particular topic? Who are unwanted or troublesome correspondents? Who are unresponsive recipients?
- *Groups.* What natural subgroups exist? What collaborative activities are individuals engaged in? What are the relationships between subgroups?
- *Temporal comparisons.* How have relationships changed over time? How did an event (e.g., move to a new location) affect the network? What inactive groups exist with whom I may benefit from reestablishing contact? What projects or people have I neglected?
- *Structural patterns.* Are there common social roles that reoccur among contacts (e.g., informant, decision maker, boundary spanner)? Are there types of subgroups that occur (e.g., cliques, fans)?

8.4.2 Organizational Email Network Questions

Several different questions can be asked about organizational email network data sets:

- *Individuals.* Who are the important individuals within an organization? For example, who are the boundary spanners who link across organizational silos? Who are the influencers or topical experts? Who is not well connected and could benefit from more social ties? Who would be a good replacement for an individual? Who fills a unique niche? Who was in-the-know about an important decision?
- *Groups.* How do email-based groupings differ from organizational structures? How is the "org-chart" different from the chart of the flow of email? How are groups interconnected? Which groups should be better connected? Is there a core competency that is not discussed in a particular branch or office?
- *Temporal comparisons.* How does information flow through the organization? How do connections among individuals and subgroups evolve over time? How are social relations affected by a major event such as a merger or opening of a new office?
- *Structural patterns.* What network properties are related to success? Can we identify up-and-coming stars or unique social roles based on their network structure? How is information on a particular topic distributed throughout the organization?

8.5 WORKING WITH EMAIL DATA

From a user's perspective the components of an email message are relatively simple. The email header includes the From, To, Cc, Bcc, Date, and Subject fields. The email body includes the message content and any attachments. Despite this apparent simplicity, there can be a great deal of hidden complexity. A full treatment of email protocols and formats is beyond the scope of this book. Instead, we list a few key terms and facts that can serve as starting points for those needing to learn more before accessing and analyzing email collections:

- Email is transmitted through the Internet via the Simple Mail Transfer Protocol (SMTP).
- Email uses the Multipurpose Internet Mail Extensions (MIME) format to allow character sets other than ASCII and nontext attachments to be included and transported via email.
- Email client applications such as Microsoft Outlook or Apple Mail retrieve or cache messages from a mail server using Post Office Protocol (POP) or Internet Message Access Protocol (IMAP). Corporate email is typically retrieved through proprietary protocols specific to Microsoft Exchange Servers or Lotus Notes instead.
- Email messages are stored in a variety of formats for different email clients. Some email clients store each message as a separate file; others save them in a database format. Some common formats include .eml (Microsoft Outlook Express, Mozilla Thunderbird), .emlx (Apple Mail), .msg and .pst (Microsoft Office Outlook), .mbx (Opera Mail, KMail), and .mbox (Pine, Eudora, and many email list archives).

Working with email poses technical challenges that often require preprocessing data to create useful results. The large potential size of email networks can be problematic and may require specialized programs to manage large data volumes. In practice, email will likely need to be filtered before analysis to reduce the data set to times, people, and topics of interest. Another major challenge is the use of multiple email addresses for the same individual. In most cases, analysts are interested in social relationships between individuals, not the relationships between email accounts. The problem of combining different aliases (email addresses) for the same entity (person) is called "entity resolution," "identity resolution," "de-duplication," or "record linkage." Creating automated tools to perform this matching is an active area of research that is still early enough in its development that full-function, off-the-shelf tools are not yet readily available. Research prototypes such as D-Dupe help support entity resolution but require expertise to make use of.[7] Another open source research prototype, Internet Community Text Analyzer, allows the creation of "Name Networks" that consider the personal names mentioned in the email text, rather than the sender and receiver of a message.[8]

8.5.1 Preparing Email

Most email clients do not export data in a format amenable to network analysis. Furthermore, the email you'd like to analyze may be stored in different formats and reside on different computers or web mail servers. As a result, you may need to prepare your email before it is ready to import into network analysis tools such as NodeXL.

The easiest way to transform email messages into network relationships (i.e., an edge list) is to use NodeXL's Import from Email Network feature. This feature relies on the Windows Search utility[9] that comes preloaded on recent versions of Windows and can be downloaded for free for older versions. Windows Search indexes files including email messages that are stored, for example, in Thunderbird, Outlook Express, or Office Outlook 2007. It can also be used to access files in shared directories on other computers.

You may not have the email you want to analyze on a local or shared machine. For example, you may exclusively rely on a web mail service such as Gmail or Hotmail. Nearly all web mail services allow you to download local copies of your messages via POP or IMAP to an email client such as Thunderbird, Outlook Express, or Office Outlook 2007. If you are using IMAP, make sure to download the complete email message files, not just the header information. Otherwise Microsoft Search will not index the content of the messages and allow you to import them using NodeXL (as described later). You can typically choose not to download attachments if there are space limitations. If you use IMAP you can also restrict the download by folder. For example, you may want to only download recent messages (i.e., those sent in 2009) rather than years of data. After downloading messages, it may take Microsoft Search some time to index all of the files. If you have subscribed to an email list and retained all of the messages you want to analyze, you can place them in a folder and use IMAP to download just those messages. Figure 8.2 shows an example of Windows Search results (running on Windows XP) after a Gmail folder called 2009 has been downloaded to Outlook Express via IMAP and indexed.

[7] www.cs.umd.edu/projects/linqs/ddupe
[8] http://textanalytics.net
[9] www.microsoft.com/windows/products/winfamily/desktopsearch

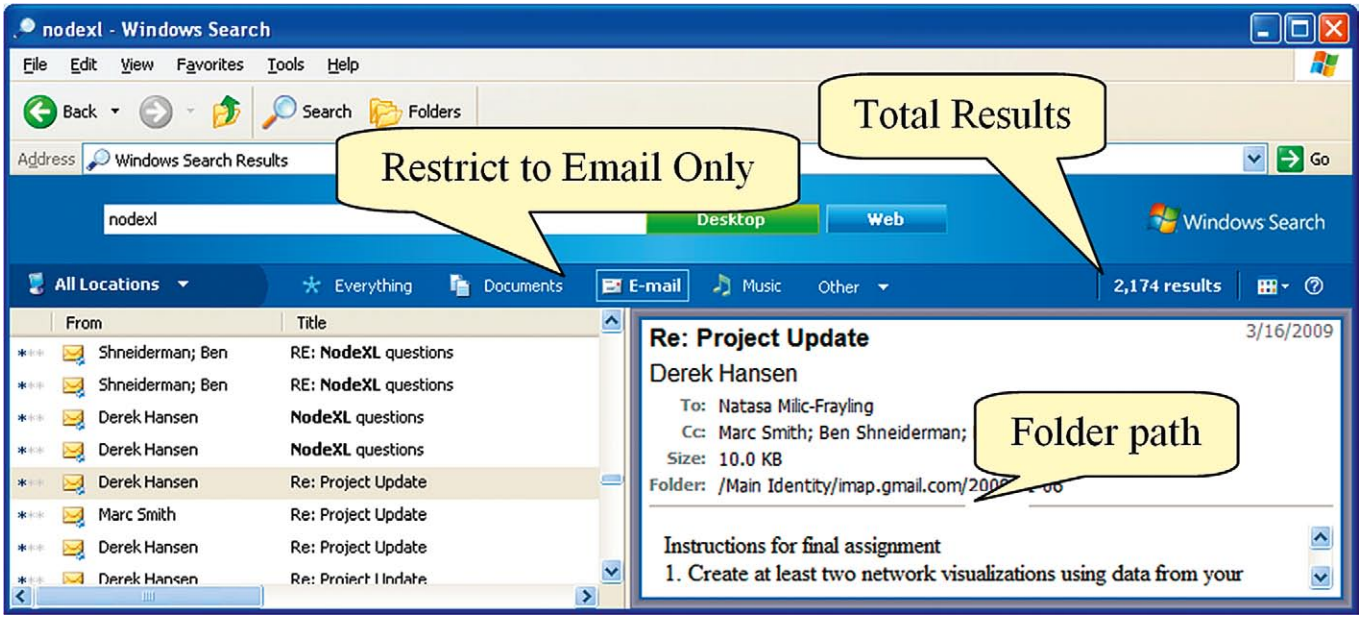


FIGURE 8.2 Window's search results show all 2174 email messages and the folder they are contained in on the desktop containing the search term "NodeXL."

ADVANCED TOPIC
Working with Large Email Collections

You may want to create networks based on email archives that are not in a format that Windows understands. For example, mbox and maildir are common formats found in Linux and Apple system mail clients. Maildir stores 100 text file per message in a directory hierarchy that matches the user's mail client, whereas mbox stores all messages in a single file.

One strategy for dealing with this issue is to use specialized programs like Mailbag Assistant and Aid4Mail[10] that can aggregate email stored in multiple devices or formats, perform and store advanced searches, and export emails into a range of formats. For example, you can use these programs to open email list archive files and convert them to a format such as "eml" files that can be indexed by Microsoft Search. One of the authors has successfully used Mailbag Assistant to work with email collections of approximately 100,000 messages with reasonable performance on a standard machine.

Another strategy is to create a database of the email messages that can be queried in multiple ways. This allows you to apply language processing and text mining approaches not available in the NodeXL import wizard. Some email programs like Mailbag Assistant will create a database for you. Alternatively, you can convert emails into XML[11] and then use Excel's built-in XML maps feature to populate the Excel fields based on the XML database content.[12]

[10] www.fookes.com/products.php
[11] www.osvisnet.co.cc
[12] www.osvisnet.co.cc/index.php?option=com_content&view=article&id=26&Itemid=24

8.5.2 Importing Email Networks into NodeXL

Once Microsoft Search has indexed the email you want to analyze, you are ready to import the data directly into NodeXL. Selecting the Import from Email Network option from the NodeXL ribbon will open a window that looks like Figure 8.3.

The enormous size of many email collections often requires filtering out messages before analysis. Even when email collections are of manageable size, filtering messages can hone in on a specific subset of messages ideally suited for addressing a question of interest. There are several ways of filtering:

- *Filter based on time*. Include all messages sent and received during a specific time period. In NodeXL, the Date Range fields allow you to specify a time window in which messages must appear in order

to be included in results. Filtering based on time can be used to slice data into networks to facilitate comparison over time or measure the impact of an important event. In Figure 8.3, only messages sent between 1/1/2009 and 5/31/2009 are included.

- *Filter based on sender and receiver(s)*. Include only messages sent or received by certain people. These people may be part of a group (e.g., department, workgroup), share some characteristic in common (e.g., senior managers, located in Maryland), or have a certain relationship to another person (e.g., all those who have received a Bcc message from John). In NodeXL it is possible to specify email addresses to be found in the From, To, Cc, and Bcc fields. The default setting is a Boolean OR relationship, so if you include multiple addresses it will pull all messages with any of the addresses included. It is also possible in NodeXL to restrict messages to those that include (or don't include) any addresses in the Cc or Bcc fields through the checkboxes available the right-hand side of Figure 8.3. Filtering based on the sender and receiver(s) is ideal for focusing in on a subgroup of important people for further analysis.
- *Filter based on content*. Include only messages that share some characteristic or content of interest. In NodeXL messages with or without attachments, messages within a certain size range, and messages or subjects with specified text can be filtered. The text search feature can be powerful when combined with standard naming conventions. For example, all messages from the Association of Internet Researcher's email list can be selected by searching for the text string "[air-l]" in the Subject search box. In the example shown in Figure 8.3, only messages with the word "NodeXL" in the message body are included. A preliminary search for a term can be performed in Window's Search to assure that the number of returned messages is reasonable for NodeXL to import (see Fig. 8.2). Results with fewer than tens of thousands of messages should fit within the scale limits of the NodeXL tool.
- *Filter based on folders and labels*. Include only messages that are found in a specified folder or messages with a certain label (e.g., in Gmail). People often organize or label (i.e., tag) email into meaningful collections based on subject matter or projects. In NodeXL, messages can be restricted to those found within a certain folder. Filtering based on folders is ideal for capturing interactions about a specific project or topic that may not be identifiable from a simple keyword. In the example shown in Figure 8.3, only messages in the "/Main Identity/imap.gmail.com/2009" folder are included. Once messages are downloaded and indexed, Window's Search can be used to identify the full path name of the folder of interest (see Fig. 8.2).
- *Filter based on a combination of features*. The various filtering options can be combined in intricate ways to find, for example, messages from a subset of key people sent during an important time period with a certain keyword. If more advanced filtering is needed, you can use an advanced email management tool like Mailbag Assistant to create a folder of desired messages and restrict the import to that folder. This enables the use of advanced search queries (e.g., regular expressions) for identifying messages.

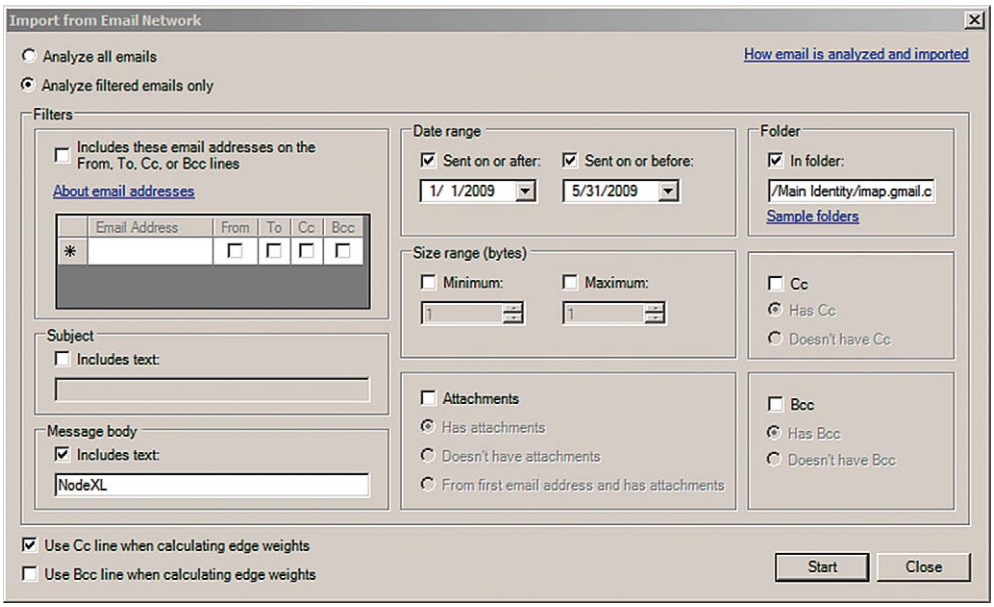


FIGURE 8.3 NodeXL Import from Email Network dialog filtered to only include messages with the term "NodeXL" between 1/1/2009 and 5/31/2009 in a specified folder.

In addition to filtering the messages included in the social network data set, NodeXL allows the way the edge weight is calculated to be specified either based only on addresses in the To field or including those in the Cc or Bcc fields as well. This is independent of filtering. By default only those addresses in the To field are counted. In the example displayed in Figure 8.3, the Cc field is included when calculating edge weights, but not the Bcc field.

8.6 CLEANING EMAIL DATA IN NODEXL

After importing email data into NodeXL, you will likely need to clean it to remove duplicate email addresses for the same individuals, as well as self-referring loops created when people reply to their own messages.

8.6.1 Remove Duplicate Email Addresses for the Same Individual

Importing email data into NodeXL populates the Edges worksheet only. Before selecting Show Graph or Get Vertices from Edge List, you should combine email addresses from the same individual. Unless you are using an advanced entity resolution software program to do this, this is likely to be a somewhat manual process.

The simplest approach is to use the *Find and Replace Strategy*, which uses the Find and Replace functionality familiar to most Excel and Word users. To do this effectively you may want to sort Vertex1 alphabetically so that addresses with the same username but different domains will be side by side (e.g., shakmatt@umich.edu and shakmatt@gmail.com). Then do a Find and Replace to change addresses of the same person to be the same (Fig. 8.4 presents an example). Sorting on Vertex2 may identify some duplicates not identified when sorting on Vertex1. There will likely be duplicates that don't start with the same username (e.g., dlhansen@umd.edu, which corresponds to the same person as the shakmatt addresses). To find the important people based on the frequency of interaction you can sort columns by edge weight and make sure that those with a high edge weight are not duplicates.

The problem with the *Find and Replace Strategy* is that it must be repeated each time the data is reimported or updated, even if the email addresses are the same. There is also no trace of the changes once they are made, making it hard to audit mistakes. A more time intensive, but careful, approach is to use the *Lookup Table Strategy* as described in the Advanced Topic: Performing the Lookup Table Strategy to Merge Duplicate Email Addresses.

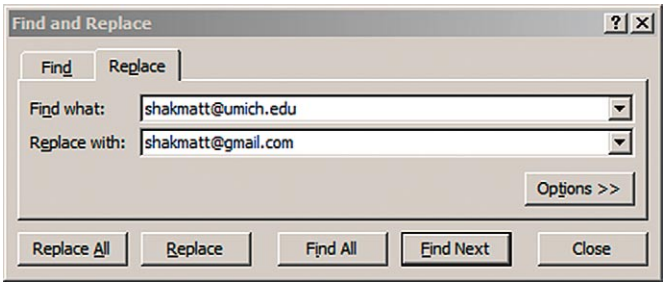


FIGURE 8.4 Excel 2007 Find and Replace function used to merge different email addresses in a NodeXL Edges worksheet.

ADVANCED TOPIC

Performing the Lookup Table Strategy to Merge Duplicate Email Addresses

Once you have imported email data into the Edges worksheet, you can create a list of all of the unique email addresses in the Vertices worksheet by choosing Prepare Data and then Get Vertices from Edge List. Copy that list to a new worksheet or workbook, title it Original_Addresses, and create a new column next to it called New_Addresses. When you find duplicate addresses for the same person in the Original_Addresses column, repeat the desired address in the New_Addresses column. An example is provided in the Lookup_Addresses table shown in Figure 8.5.

You can now use a VLOOKUP function to look up the New_Addresses that correspond with the Original_Addresses to create a new edge list with no duplicate email addresses. Copy the original edge list from the Edges worksheet (columns D and E) and create two new columns for a new edge list (columns F and G). The New Edge List columns will be automatically populated by the results of a VLOOKUP function. Figure 8.5 shows an example in Cell F9. The VLOOKUP function looks for the original Vertex1 address found in cell D9 (shakmatt@umich.edu shown in the blue rectangle) within the first column of the Lookup_Addresses table (cells $A$3:$B$8 shown in a green rectangle). In this example, it finds an exact match in cell A7. It then returns the value of the second column in the Lookup table (because the number 2 was entered into the VLOOKUP formula), which is shakmatt@gmail.com found in cell B7. The FALSE in the VLOOKUP formula specifies that the value that is

(*Continued*)

ADVANCED TOPIC (Continued)

SUM =VLOOKUP(D9,$A$3:$B$8,2,FALSE)

	A	B	C	D	E	F	G
1	Lookup_Addresses			Original Edge List		New Edge List	
2	Original_Addresses	New_Addresses		Vertex1	Vertex2	Vertex1	Vertex2
3	air-l@listserv.aoir.org	air-l@listserv.aoir.org		ben@cs.umd.edu	shakmatt@gmail.com	ben@cs.umd.edu	shakmatt@gmail.com
4	casci@umd.edu	casci@umd.edu		ben@cs.umd.edu	dlhansen@umd.edu	ben@cs.umd.edu	shakmatt@gmail.com
5	dlhansen@umd.edu	shakmatt@gmail.com		shakmatt@gmail.com	ben@cs.umd.edu	shakmatt@gmail.com	ben@cs.umd.edu
6	shakmatt@gmail.com	shakmatt@gmail.com		shakmatt@gmail.com	air-l@listserv.aoir.org	shakmatt@gmail.com	air-l@listserv.aoir.org
7	shakmatt@umich.edu	shakmatt@gmail.com		shakmatt@gmail.com	casci@umd.edu	shakmatt@gmail.com	casci@umd.edu
8	ben@cs.umd.edu	ben@cs.umd.edu		shakmatt@umich.edu	ben@cs.umd.edu	shakmatt@gmail.com	ben@cs.umd.edu
9				shakmatt@umich.edu	shakmatt@gmail.com	=VLOOKUP(D9,$A$3:$B$8,2,FALSE)	

FIGURE 8.5 A Lookup table and the Excel = VLOOKUP() formula is used to combine duplicate email addresses in NodeXL.

looked up must exactly match a value in the first column of the Lookup_Addresses table. An error is returned if an exact match is not found. Once the New Edge List is created, you can copy the new Vertex1 and Vertex2 columns (F and G) and use the Paste Special feature to paste their values into a new workbook. You should also copy and paste the Edge Weights column from the original file after assuring that the Vertex1 and Vertex2 columns are in the exact same order as the original data.

VLOOKUP functions can use considerable computational resources when working with large files, so once they are calculated you may want to copy them, choose Paste Special, and select Values so they do not need to be recalculated each time changes to the workbook are made. See Chapter 11 for a different use of the VLOOKUP function when using NodeXL.

	A	B	C	D	E	F	G
1	New Edge List (Before Merge Duplicate Edges)				New Edge List (After Merge Duplicate Edges)		
2	Vertex1	Vertex2	Edge Weight		Vertex1	Vertex2	Edge Weight
3	ben@cs.umd.edu	shakmatt@gmail.com	10		ben@cs.umd.edu	shakmatt@gmail.com	12
4	ben@cs.umd.edu	shakmatt@gmail.com	2		shakmatt@gmail.com	ben@cs.umd.edu	18
5	shakmatt@gmail.com	ben@cs.umd.edu	15		shakmatt@gmail.com	air-l@listserv.aoir.org	1
6	shakmatt@gmail.com	air-l@listserv.aoir.org	1		shakmatt@gmail.com	casci@umd.edu	4
7	shakmatt@gmail.com	casci@umd.edu	4		shakmatt@gmail.com	shakmatt@gmail.com	2
8	shakmatt@gmail.com	ben@cs.umd.edu	3			TOTAL:	37
9	shakmatt@gmail.com	shakmatt@gmail.com	2				
10		TOTAL:	37				

FIGURE 8.6 Effects of NodeXL's Merge Duplicate Edges feature after combining duplicate addresses. A self-loop is shown in red.

8.6.2 Merge Duplicate Edges

Once you have updated the email addresses in the Edges worksheet so that different addresses for the same person are replaced with a single address, you will likely have multiple edges that represent multiple messages sent from one person to another. It can be useful to "roll up" these duplicate edges, replacing multiple connections between a pair of email addresses with a single edge (see Chapter 6). The rolled up edge has a weight equal to the total number of message exchanges listed in the data. It is important to roll up the data so that network metrics can be accurately calculated, as some of them assume that edges connecting any pair of vertices are unique. To prepare your email network for analysis, you can roll up repeated email messages from the same pair of people using the Merge Duplicate Edges feature in the Prepare Data section of the NodeXL Ribbon. This will roll up the duplicate edges and sum up the Edge Weights so the total Edge Weight remains the same. Figure 8.6 shows the New Edge List from Figure 8.5 in Columns A and B and a manually merged version of it in Columns E and F. Notice that the total of the Edge Weight column is the same.

ADVANCED TOPIC
Automatically Identifying Self-Loops

To easily identify self-loops, you can create a new column called Self-Loop and populate it with the function "=Edges[[#ThisRow],[Vertex1]]=Edges[[#ThisRow],[Vertex2]]". If the Vertex1 and Vertex2 column are the same for a given row, a TRUE will be returned. Otherwise a FALSE will be returned. Once calculated, you can sort on the column to find the Self-Loops. If desired, you can delete them or choose to skip them using the Visibility column.

Sometimes people send email messages to themselves as a reminder, as a To Do list, or to share a file between computers. This results in a row with the same address in the Vertex1 and Vertex2 columns on the Edges worksheet and is called a self-loop. The use of multiple email addresses and the removal of duplicate addresses can also cause self-loops. Row 9 of Figure 8.5 is an example. For many analyses these self-loops are not important and can be distracting when visualizing data or calculating network metrics. You may want to remove self-loop edges such as the red pair in Figure 8.6 as an additional step after you have merged duplicate edges.

8.7 ANALYZING PERSONAL EMAIL NETWORKS

This section presents two projects that serve as examples of how to analyze personal email collections. They are both based on the following scenario.

Scenario: You have a new employee coming to work with you next week whom you will supervise. He doesn't know you well and is new to the organization. To help him smoothly transition into his new job, you want to provide him with an overview of the people you work with and their relationships to each other. You decide to create two network visualizations, one that provides an overview of all of your contacts and another that provides more detail on the workgroup that he will work with most closely.

In the following examples, the new employee is a doctoral student coming to work with Derek Hansen, assistant professor at the University of Maryland's iSchool. The student will be working with a group of researchers evaluating and testing NodeXL. For privacy reasons, the email networks analyzed in this section are not made public. You are encouraged to analyze your own email data with a similar scenario in mind. This section assumes that Microsoft Search already indexes your email as described in prior sections.

8.7.1 Creating an Email Overview Visualization

Step 1: Import Data into NodeXL

Import all of your email sent within the past month. Although some people you know may not have gotten in touch in more than a month, this time period lets you collect a broad set of your active email contacts. Use the Import From Email Network feature in the Data menu and filter based on date range (e.g., 5/1/2009 to 5/31/2009). Check the "Use Cc line when calculating edge weights" box to be more inclusive. For Derek's data set, a total of 2477 edges are created with a total edge weight of 6763 (that is, 6763 messages were exchanged among 2477 pairs of email addresses).

Step 2: Clean Data

Because this is an informal analysis, you may want to use the quick-and-dirty *Find and Replace Strategy* to combine email addresses associated with the same person (see the earlier section). Then merge duplicate edges. This rolls up the 2477 edges into 2416 (61 pairs were merged). To make sure no data were lost, check that the sum of the Edge Weight column is the same as it was before the merge.

Step 3: Filter Data

To more clearly focus on the key relationships, it is desirable to remove infrequent email exchanges. Using the Excel Data&Sort features, sort the Edge Weight column from largest to smallest (Z > A). It is likely that the values will have a skewed distribution with many connections with very low edge weights and relatively few connections with a high edge weight. Remove the least common connections by choosing a cutoff point for deletion. You may want to use the Dynamic Filters feature discussed in Chapter 6 to find an appropriate cutoff. For example, in Derek's data when all of the connections with an edge weight of less than 5 are removed, the key individuals are still retained and the total number of edges is reduced to 286, less than 12 percent

ADVANCED TOPIC
Calculating Total Messages Sent and Received

The following formulas will aggregate the count of all sent (or received) messages for each individual in the Vertices worksheet based on the data in the Edge Weight column of the Edges worksheet:

- **Sent:**[13] Calculates the total number of sent messages for each person:

```
=sumif(Edges[Vertex 1],Vertices[[#This
  Row],[Vertex]],Edges[Edge Weight])
```

- **Received:** Calculates the total number of received messages for each person:

```
=sumif(Edges[Vertex 2],Vertices[[#This
  Row],[Vertex]],Edges[Edge Weight])
```

Additional formulas can be used to calculate the total number of messages sent or received by an individual and the percentage of messages that are sent:

- **Total:** Calculates the total number of messages sent or received for each person:

```
=Vertices[[#This Row],[Sent]]+Vertices[[#This
  Row],[Received]]
```

- **%_Sent:** Calculates the percentage of all messages received:

```
=Vertices[[#This Row],[Sent]]/Vertices[[#This
  Row],[Total]]
```

[13]If the "Use cc: line when calculating edge weights" (or use bcc: line) is chosen when importing email data, then the Sent formula gives the number of messages you sent that were received by others. In other words, if you sent one message to two people (one of whom was cc'ed in), your total sent would be 2, not 1 even though you only authored 1 message.

of the original edges. You can manually delete the rows, in which case your data will be a more manageable size, but will lose some data that may be needed in your later analysis. For example, if you calculate the total number of messages a person sends (see Advanced Topic: Calculating Total Messages Sent and Received), the number would be incomplete if all of the infrequent connections were deleted. Alternatively, you can use the Autofill Columns feature to "Skip" connections with an edge weight that falls below the cutoff point. This will keep the data in the workbook, but not use it in the display of graphs or the calculation of network metrics.

Step 4: Compute Graph Metrics and Add New Columns

Next, select Show Graph, which will populate the Vertices worksheet with data about each vertex and display a preliminary email social network graph. The next step in the data analysis is to compute all of the graph metrics. You can insert additional columns indicating people's attributes such as their relationship to you, their location, or affiliation. You can also use formulas to calculate the total number of messages sent or received by an individual as is described in the Advanced Topic: Calculating Total Messages Sent and Received.

Step 5: Visualize the Email Social Network

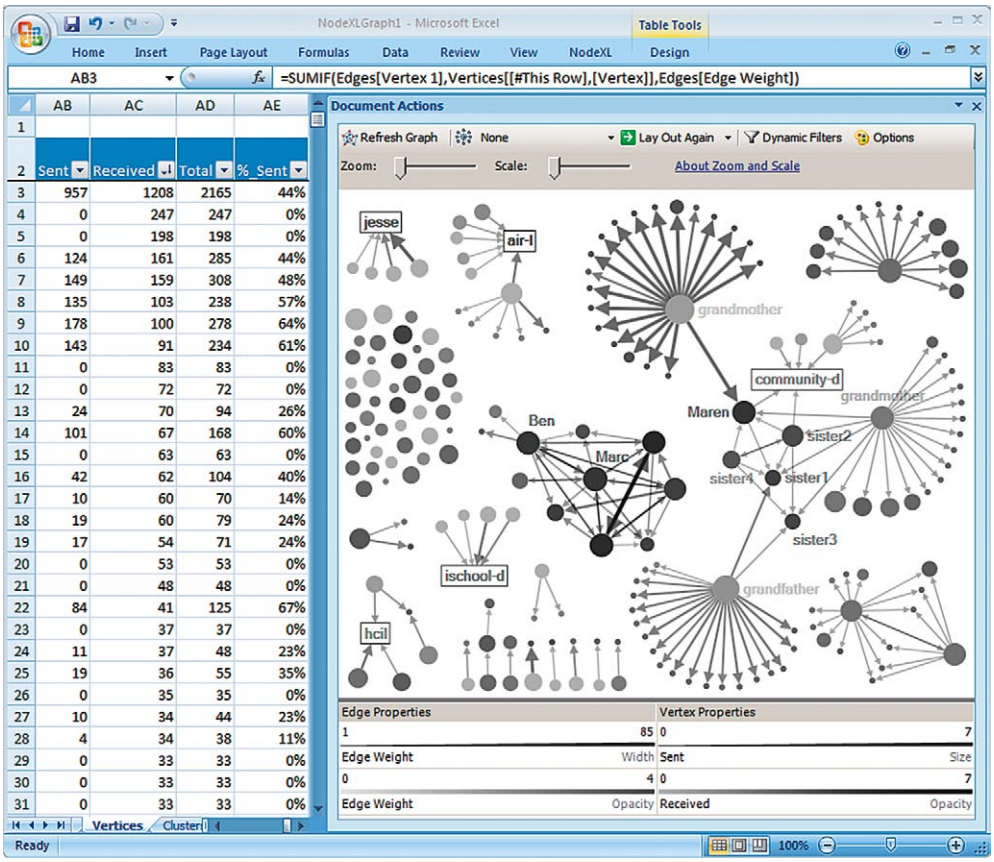
The next step is to map the metrics and new columns onto display attributes in the visualization (see Chapter 4). Many display attributes like color, transparency ("opacity"), edge width, and location can be mapped to data attributes about messages, relationships, and authors. Selecting the optimal mapping of data attributes to display attributes will inevitably require some trial and error. You may want to look at the social network graph with and without the vertex that represents your own email address by manually setting the Visibility column for the row with your email address on the Vertices worksheet to Skip. Figure 8.7 shows Derek's network after using the Harel-Koren Fast Multiplex layout and manually positioning vertices.

Derek's email address and its connections are not shown, which makes the graph less cluttered (because the "Derek" node had been connected to almost every other node) and highlights distinct groups. However, it hides information about whom he communicates with most often and the direction of his communications. Email addresses and names of most individuals have not been displayed for privacy reasons. Derek could set the tooltip to display email addresses and provide a file to his new doctoral student so that he could map vertices to email addresses. Alternatively, a printed version with selected individuals that the doctoral student is likely to work with could be created.

Step 6: Understand Social Network Visualizations and Metrics Data

Analysis of the graph and accompanying data helps answer many of the questions offered earlier in the chapter:

- *Individuals.* Who are important individuals within the network? The graph helps clearly identify



	AB	AC	AD	AE
1				
2	Sent	Received	Total	% Sent
3	957	1208	2165	44%
4	0	247	247	0%
5	0	198	198	0%
6	124	161	285	44%
7	149	159	308	48%
8	135	103	238	57%
9	178	100	278	64%
10	143	91	234	61%
11	0	83	83	0%
12	0	72	72	0%
13	24	70	94	26%
14	101	67	168	60%
15	0	63	63	0%
16	42	62	104	40%
17	10	60	70	14%
18	19	60	79	24%
19	17	54	71	24%
20	0	53	53	0%
21	0	48	48	0%
22	84	41	125	67%
23	0	37	37	0%
24	11	37	48	23%
25	19	36	55	35%
26	0	35	35	0%
27	10	34	44	23%
28	4	34	38	11%
29	0	33	33	0%
30	0	33	33	0%
31	0	33	33	0%

FIGURE 8.7 Derek Hansen's email network for May 2009, showing only connections with over five messages (Edge Visibility > 4). Size is based on the Log of total sent messages (1.5 to 3.5). Opacity is based on the Log of total received messages (30 to 100). Edge weight is mapped to Edge Width (1 to 4) and log of edge weight is mapped to Edge Opacity (10 to 70). Shape is set to Label for email lists; Ben Shneiderman and Marc Smith are labeled with the Shape kept as a Disk.

individuals that play unique roles in Derek's social network, including those who send a large amount of email (larger vertices), those who receive a considerable amount of email (darker vertices), those who span different groups, and those who are a part of a team. Using the Excel Data&Sort feature on the spreadsheet data is an effective way of identifying important individuals. Labeling was used to show email lists, which have the unique property that they receive email from numerous people but never send it out. There are also many individuals without any connections in this graph because they send or receive email from Derek directly. Those that are large and light colored are potential spammers because they send out many messages and don't receive any. In contrast, those that are small and dark may be unresponsive recipients since they receive many messages but rarely send them out. Large, dark circles represent those who are highly engaged with Derek or other colleagues shown in the graph.

- *Groups.* What natural subgroups exist? By far the most interconnected subgroup is the one with Ben and Marc in it, which represents the NodeXL team who was writing a paper that month. Sorting by clustering coefficient from largest to smallest helps identify these individuals because members of the team are all connected to one another through the exchange of email. The largest component includes a subgroup of connected individuals that include Derek's extended family and friends. The interconnected group represents Derek's four sisters and wife Maren, with grandparents from different sides of the family who send regular updates to numerous extended family members. Other separate components not labeled represent various workgroups.
- *Temporal comparisons.* This social network graph represents only a single month snapshot of email connection activity. Comparison with similar social network graphs from other time periods can show interesting patterns of change and stability. For example, new email addresses can show up linked to the NodeXL team as new members join the team, connections between individuals that are currently isolated can be formed, and the send/reply structure may shift depending on the nature of activities that are under way.
- *Structural patterns.* At a high level, this graph represents the life of a professor, one filled with numerous one-on-one relationships with colleagues and students. There is only one densely connected group in this month and ongoing connections with

family members. A similar graph of an analyst or accountant in a large firm may look very different, with messages exchanged with several densely connected distinct groups in a given month [3]. One clearly identifiable structural pattern that shows up often in this network graph is the fanlike arrangement around several vertices. The hubs at the center of these fan structures are large, light-colored vertices representing people who send many messages while receiving few. The hubs may receive replies, but not with Derek copied in on them, suggesting that if people are replying they are using the Reply and not Reply to All email features.

8.7.2 Creating an Expertise Network Email Graph

Step 1: Import Email Social Network Data into NodeXL

In this case, we want to include only those messages that mention a particular topic. This gives us some idea of who knows about this topic and how they are related to one another. For this example, we are interested in finding individuals with whom Derek exchanges emails that refer to "NodeXL." Figure 8.3 shows the Import From Email Network window set to include only messages with the text "NodeXL" exchanged during a 5-month period.

Step 2: Clean Data

Use one of the previously specified methods for joining duplicate addresses for the same person and merge duplicate edges. In Derek's data set, there are 388 unique edges and 181 unique vertices (i.e., email addresses after removing duplicates). One edge visibility was manually set to "Skip (0)" because it was a self-loop.

Step 3: Compute Graph Metrics and Add New Columns

When there are relatively few connections as in this example, it is feasible to calculate the metrics and add columns before filtering the data. Calculate all of the metrics and add the same new columns to the Vertices worksheet as in the prior example.

Step 4: Filter Data

It is reasonable to use the dynamic filters to determine a good cutoff point when there are few connections. To focus in on those who know the most about the topic (NodeXL), filter out those with a low edge weight or those who send few messages. To hone in on the cluster of tightly connected vertices, filter out those with a low in- and out-degree, since those who are densely clustered send and receive messages from others that are part of the group. Figure 8.8 shows the NodeXL email network before and after the dynamic filtering.

Step 5: Visualize Network

Figure 8.8 uses a similar mapping of data to visual properties as was used in Figure 8.7 with a few minor changes in minimum and maximum values. The key difference is the inclusion of Derek and his connections. This adds clutter to the graph, but also adds valuable information about whom he worked with most closely during the 5-month period. For example, the filtered graph in Figure 8.8 shows that he corresponded most often with Dana and Beth, two doctoral students with

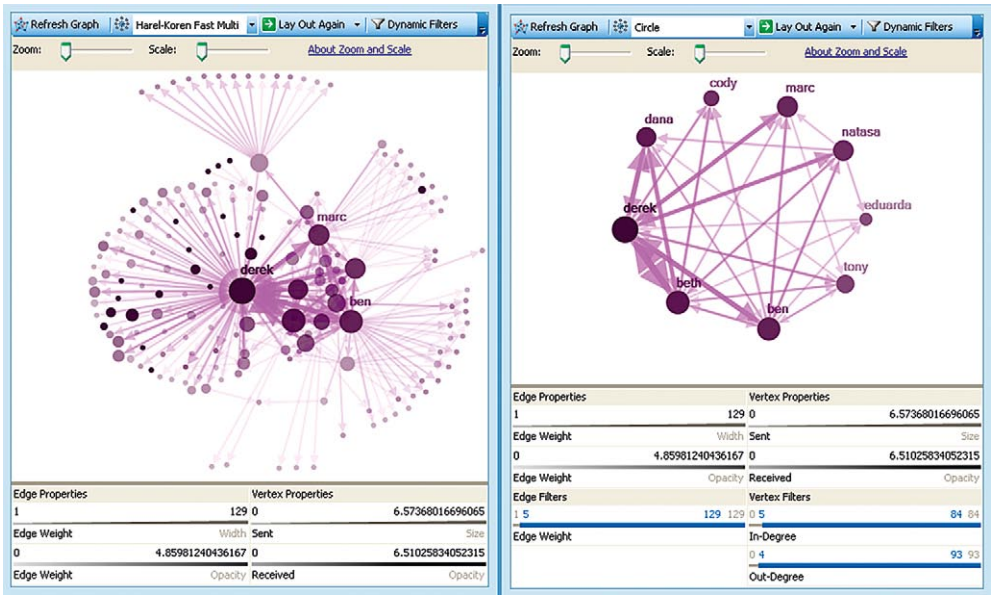


FIGURE 8.8 NodeXL visualization of Derek's personal email message network limited to messages with the term "NodeXL" sent or received over a 5-month period; shown before and after the application of dynamic filters. Size maps to the Log of total sent messages, Opacity maps to the Log of total received messages. Edge width and opacity map to edge weight.

whom he was working, and Ben who is co-located at Maryland and a close collaborator.

Step 6: Understanding the Network Visualization and Data

The analysis of the first graph is similar to that of Figure 8.7, except that connections are solely based on messages containing the text "NodeXL." Thus, many important individuals from the prior graph do not appear in this graph, whereas others become comparatively more important than in the prior graph. Even with this change, there are some apparent similarities. Hubs are still apparent, although they are more likely to receive messages than the hubs in Figure 8.7. The same group of individuals found in Figure 8.7 appear, these are the members of the NodeXL team and frequently use the term "NodeXL" in emails they exchange.

8.8 CREATING A LIVING ORG-CHART WITH AN ORGANIZATIONAL EMAIL NETWORK

Enterprises rely on their communication networks to function. A combination of phone, email, calendars, web boards, discussion lists, blogs, wikis, status updates, instant messaging, chat, and file sharing are often used in concert to share ideas, documents, schedules, and data. Analyzing the patterns of connection within these collections can reveal important insights into the structure and dynamics of an organization. When an employee, for example, emails another employee, a link is formed that connects the two individuals, but also their organizational groups and divisions. These connections often crosscut the branches commonly represented in an "org-chart." Most enterprises and institutions are organized hierarchically with people in a group reporting to a single manager who in turn, reports to a manager. These connections repeat until they connect to the single most senior part of the company, creating a tree or pyramid of branching, nested connections known as the traditional org-chart. But "leaf" groups at the ends of these branches often connect directly to other groups, without passing messages up and down the chain of command. A map of the network of connections among groups in an enterprise is an alternative vision to the org-chart that reveals information about the flows of information and connections through the organization.

The extraction of enterprise social media network data is not trivial and requires the coordination of several parts of a typical business. Support from the managers of enterprise email systems is essential to access records of email exchanges. Data about the organizational structure of the business are often stored in a separate corporate directory system that contains information about each employee, such as one's job title, physical location, level, and reporting structure (i.e., to whom the person reports). Coordinating the extraction of data from these two systems can be a challenge for organizations accustomed to managing these functions separately, although unique employee identifiers and email addresses can often be used to join the separate data sets that must be integrated. Privacy, security, and legal concerns arise and must be addressed, potentially for multiple jurisdictions. Although it may be nice to match performance data with network information, it is often not feasible because of potential privacy concerns. Data from multiple information systems is rarely available in a form that is immediately useful for network analysis such as an edge list, and it must be scrubbed to remove errors or inconsistencies. Despite these challenges, a number of companies have begun to create social network data that combine corporate email network data and corporate directory information, giving them a live window into their corporate communication patterns.

8.8.1 TechABC's Organizational Unit Email Network

In this section you will analyze a sample of email traffic from a large global technology company we'll call TechABC. The company has more than 100,000 employees in dozens of countries and hundreds of locations. Employees are aggregated into roughly 10,000 organizational units that have an average of 15 members. Organizational names in the visualizations have been anonymized. For privacy reasons we cannot provide the data set.

People in each organizational unit send and receive email from people within their own unit as well as to people in other units. These events were logged in the corporate email server and were extracted for a week-long period. An edge list of events in which an employee sent an email to another employee in the To, Cc, or Bcc fields was created. Data about each employee were then removed and replaced with the name of the organizational unit in which they were a member, helping address individual privacy concerns. Data were then aggregated (see earlier Merge Duplicate Edges section), creating an edge weight that represents the number of messages sent from one unit to another. This process rolls messages exchanged between members of the same unit into self-loops, where the sending unit and receiving unit are the same. The total number of internally exchanged messages can be useful, but is best treated as attribute data on the Vertices worksheet rather than captured in the edge list.

8.8.2 Normalizing and Filtering TechABC's Data

Whole graph maps of enterprise networks are likely to be too large and dense to be informative. For

example, TechABC's raw sent email network includes more than 1.3 million edges and around 10,000 vertices. A process of filtering and selective display is required to peel away parts of the network that obscure structures of interest (see Chapter 6). When working with large data sets such as TechABC's, you will likely want to perform the first round of filtering using a database program like Microsoft Access because of size limitations in Excel.

A common edge filtering technique is to remove all connections below a threshold, helping whittle away infrequent ties to reveal the strong core skeletal structures of the company. The easiest threshold to use is the raw number of messages sent between units. However, because organizational units differ in size, this approach disadvantages smaller units with fewer members contributing to the number of messages. To account for this discrepancy, you can normalize the data by creating a new edge variable based on the number of messages sent per employee (e.g., per fulltime equivalent or FTE). You'll need to decide if you want to use the number of FTEs from the sending unit, receiving unit, or some combination of the two. For the graph shown in Figure 8.9, we removed edges with fewer than 50 messages per FTE sent in a week, where we used the minimum of the sender and receiver FTE values as the denominator. This approach keeps an edge if it is important (i.e., a high number of emails per FTE) to either the sending or receiving unit (see the U.S. Senate co-voting example in Chapter 7 for another illustration of a similar technique). The resulting, filtered TechABC network includes 2303 edges and 2267 vertices. Figure 8.10 uses a similar approach, but because it focuses on a subset of units (only research units), the threshold was lowered to 10 messages per FTE sent in a week.

You could also normalize the data by calculating the number of messages sent from one unit to another unit as a percentage of all messages sent from the unit. This approach accounts for differences in a unit's overall email usage patterns, which can be desirable in some cases. For example, it would remove edges representing company announcements from a single unit (e.g., the human resources or information technology department) because the messages sent to any one unit would be a small percentage of the sending unit's overall sent messages. As with the prior example, you will need to decide if you want to use the sending or receiving unit's total message count as the denominator, or some combination of the two (e.g., maximum, minimum, average).

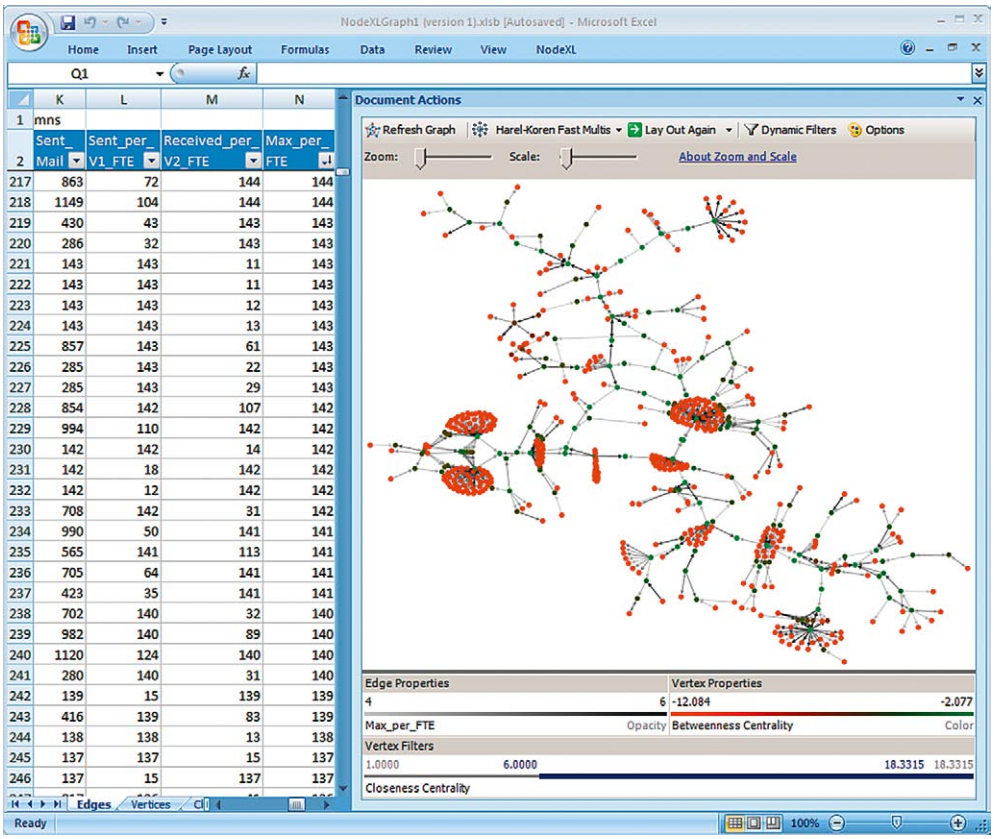
Other strategies for filtering data can lead to other insights. For example, showing only weak ties (edges with between 3 and 10 messages per FTE) can highlight lesser-known connections that might guide management efforts to improve connections across gaps in the company. Attributes of organizational units can be used to filter the network graph as well, helping to zoom into subsections of the larger graph. For example, you could remove all but the most "between" groups to reveal the network of core groups while hiding more peripheral groups. You can also focus on units within a particular department or with a similar mission. You will see this approach used in our second example (Fig. 8.10), which looks at connections between research units of TechABC.

8.8.3 Creating an Overview of TechABC's Communication Patterns

You may want to create an overview graph of an organization's email communication before moving into more detailed analyses of specific departments or groups. Overview graphs can be difficult to read because of their size. However, they are excellent for dynamically exploring by sorting on metric properties to identify important units, highlighting vertices of interest and seeing their connections on the graph, and using dynamic filters to further hone in on specific areas of interest. A highly filtered overview of TechABC is shown in Figure 8.9. Only edges with more than 50 messages per FTE are shown, with additional filtering to show only the main component. You can think of this as the backbone of the company.

This graph and the accompanying data tell interesting stories about the company. Overall, the graph is sparse, largely because of the high threshold we have chosen (10 messages a day for a week per FTE). The graph density is very low, suggesting that most units only communicate heavily with one or two other units. The average geodesic distance is 10.2 and maximum geodesic distance (i.e., diameter) is 29, both of which are quite high. If high numbers existed at a lower threshold, it would suggest that units may not be well connected with certain other units on the other "side" of the company. Increasing connections between otherwise disconnected groups may be a goal for an organization. For example, many organizations have created "communities of practice" consisting of people with similar skills who are scattered throughout different organizational units. An initiative to increase connections throughout the company could be evaluated by looking for increases in the network density and decreases in the diameter over time.

In addition to looking at global trends, it is possible to look at the role of individual units in the company. Even with the highly filtered graph shown in Figure 8.9, it is possible to identify several hubs (with high outdegree), some densely connected clusters, and units that act as bridges between other units. Some of these fill critical locations in the network, demonstrating their unique value that results from their network position. Organizational units that are connected to many other units (i.e., the hubs) perform services like IT



	K	L	M	N
1	mns			
2	Sent_ Mail	Sent_per_ V1_FTE	Received_per_ V2_FTE	Max_per_ FTE
217	863	72	144	144
218	1149	104	144	144
219	430	43	143	143
220	286	32	143	143
221	143	143	11	143
222	143	143	11	143
223	143	143	12	143
224	143	143	13	143
225	857	143	61	143
226	285	143	22	143
227	285	143	29	143
228	854	142	107	142
229	994	110	142	142
230	142	142	14	142
231	142	18	142	142
232	142	12	142	142
233	708	142	31	142
234	990	50	141	141
235	565	141	113	141
236	705	64	141	141
237	423	35	141	141
238	702	140	32	140
239	982	140	89	140
240	1120	124	140	140
241	280	140	31	140
242	139	15	139	139
243	416	139	83	139
244	138	138	13	138
245	137	137	15	137
246	137	15	137	137

FIGURE 8.9 TechABC's organizational unit email network "backbone," focusing on high-traffic connections between units (i.e., those who exchange more than 50 messages per FTE). Color is mapped to betweenness centrality with green vertices playing important roles as bridge spanners. Edge opacity is mapped to messages per FTE. Dynamic filters were used to exclude those with low closeness centrality, which is a trick for filtering out all vertices that are not part of the large component.

management or library services that touch many parts of the company. Groups that are less connected but have a high betweenness centrality likely have coordination functions, bridging information between multiple groups such as specific geographical units within a larger region. Isolated groups and clusters of groups are likely specialists that perform a function for one or a few other groups to consume. Analyzing networks like the one visualized in Figure 8.9 also allows you to compare units that serve a similar function to see how they compare on various metrics, helping to identify those that could benefit from additional connections.

8.8.4 Examining TechABC's Research Division

Although overview maps like Figure 8.9 can be helpful, they can also be cluttered and may filter out too much of the detail for large companies. To gain actionable insights, you will typically need to focus on subsections of the network, such as units that serve a similar purpose (e.g., IT, marketing, research). In this section you will explore the organizational units within TechABC that have a research mission. They were identified by looking for organizational unit names with the word "research" in them using Microsoft Excel's Search function (a non-case-sensitive function similar to the Find function).

Although you can restrict the network analysis to only the core units that meet your criteria (i.e., research units), it is often insightful to include all units connected to the core units. For example, Figure 8.10 includes all research units (maroon boxes), as well as all units they sent or received messages to (blue disks). A cutoff point of 10 messages per FTE was used in order to account for unit size differences. Because the focus is on the research units, connections between the nonresearch units are not shown. The result is a collection of all of the 1.0-degree networks of the research units. To create a similar graph, you can add a new column called Research to the Edges worksheet that is a 1 if either Vertex1 or Vertex2 is a research unit and a 0 otherwise. You can then set the Edge Visibility Equal to 1 (in the Research column) using the Autofill Columns feature to exclude all other edges.

The network highlights several important bridge-spanning units, as well as some disconnected units that may need to be connected. For example, research unit Specific 6 plays an important role in connecting several other research groups either directly or indirectly. The organizational unit Specific 10 is important because it is the only path connecting the large Specific 2 unit to the other research units (albeit indirectly). There are also

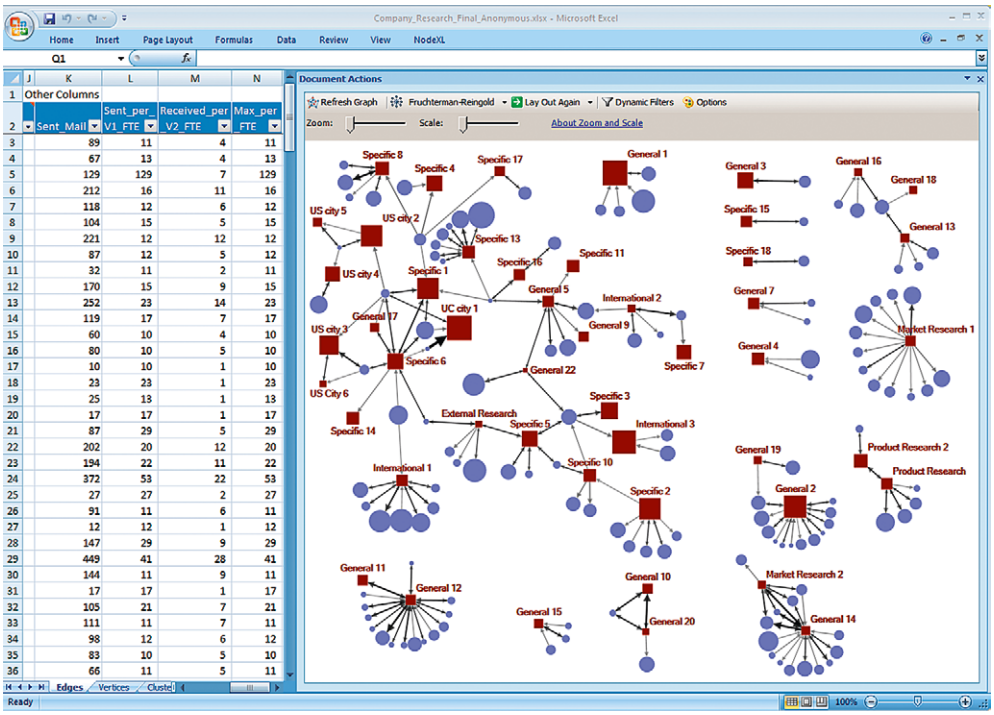


FIGURE 8.10 TechABC's organizational unit network including research units (maroon squares) and nonresearch units connected to them (blue disks) through the exchange of email. Only edges with 10 or more messages sent per FTE are included. Edge width (1-2) and opacity (40-100) are based on raw sent messages. Vertex size is based on the number of group members (FTEs). Notice some obvious disconnects (e.g., Market Research 1 and 2), as well as some of the important bridge spanning research groups (e.g., Specific 6) and nonresearch groups (e.g., Blue disk just above General 17).

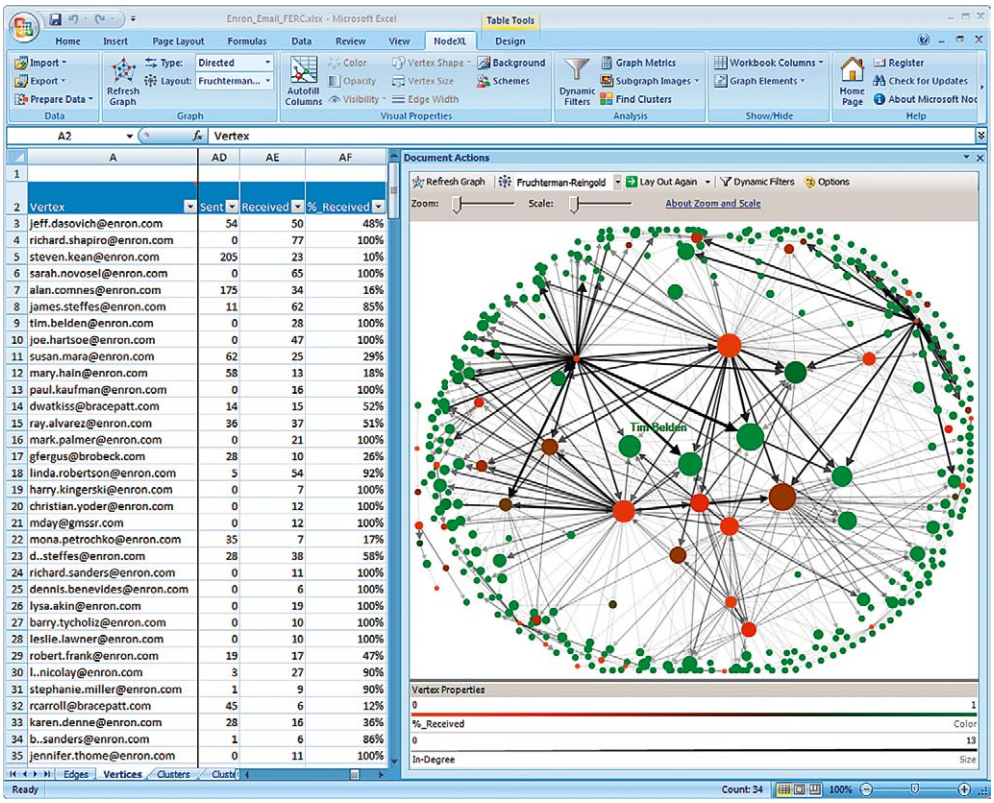
several nonresearch groups that play pivotal bridge spanning roles, such as the very small unit just above General 17 that is connected to six different research groups, none of which is directly connected to another. This small unit likely plays an important role and its small size may make it vulnerable to employee turnover, suggesting that the company may consider if additional resources are needed to support the group's function. In contrast, the network shows Market Research 1 and 2 in completely different components, not even connected indirectly. More generally, few research units are directly connected to each other, suggesting that there may be potential for increased exchanges through employee swaps, internships, or other shared projects. This assumes there would be benefits from interdisciplinary projects, which content experts would need to determine. Although many of the actionable insights require knowledge about the organization, Figure 8.10 gives you some idea of the potential benefits of this type of analysis.

8.9 HISTORICAL AND LEGAL ANALYSIS OF ENRON EMAIL

In the prior section you explored an organizational network from the perspective of an insider who knows the company. In this section you will explore an organizational network from the perspective of an outsider trying to make sense of an email corpus collected as part of a lawsuit. Specifically, you will explore a subset of email messages sent and received by Enron employees. The original, publicly available data set included approximately a half-million messages and was made public by the Federal Energy Regulatory Commission (FERC) during the investigation of Enron. It was later cleaned and made permanently accessible by researchers at MIT, CMU, and SRI (see www-2.cs.cmu.edu/~enron for details). The analysis in this chapter is based on a subset of 1700 messages coded by students and researchers at the University of California at Berkeley, filtered to only include messages that are work related. It focuses on business-related messages occurring later in the collection and includes discussions of the California Energy Crisis (see http://bailando.sims.berkeley.edu/enron_email.html for a complete description and compressed file of the individual messages). Messages were downloaded, indexed, and imported into NodeXL using the process described earlier in this chapter. You can download the NodeXL files that correspond to the images shown in this section from the book web site. The analysis is inspired by Jeffrey Heer's work [4].

8.9.1 Identifying Key Individuals Using Content Networks

One problem historians and lawyers face is identifying individuals who played key roles in important events. For employees that use email frequently, email networks provide a quick sense of who communicates



Vertex	Sent	Received	%_Received
jeff.dasovich@enron.com	54	50	48%
richard.shapiro@enron.com	0	77	100%
steven.kean@enron.com	205	23	10%
sarah.novosel@enron.com	0	65	100%
alan.comnes@enron.com	175	34	16%
james.steffes@enron.com	11	62	85%
tim.belden@enron.com	0	28	100%
joe.hartsoe@enron.com	0	47	100%
susan.mara@enron.com	62	25	29%
mary.hain@enron.com	58	13	18%
paul.kaufman@enron.com	0	16	100%
dwatkiss@bracepatt.com	14	15	52%
ray.alvarez@enron.com	36	37	51%
mark.palmer@enron.com	0	21	100%
gfergus@brobeck.com	28	10	26%
linda.robertson@enron.com	5	54	92%
harry.kingerski@enron.com	0	7	100%
christian.yoder@enron.com	0	12	100%
mday@gmssr.com	0	12	100%
mona.petrochko@enron.com	35	7	17%
d..steffes@enron.com	28	38	58%
richard.sanders@enron.com	0	11	100%
dennis.benevides@enron.com	0	6	100%
lysa.akin@enron.com	0	19	100%
barry.tycholiz@enron.com	0	10	100%
leslie.lawner@enron.com	0	10	100%
robert.frank@enron.com	19	17	47%
l..nicolay@enron.com	3	27	90%
stephanie.miller@enron.com	1	9	90%
rcarroll@bracepatt.com	45	6	12%
karen.denne@enron.com	28	16	36%
b..sanders@enron.com	1	6	86%
jennifer.thome@enron.com	0	11	100%

FIGURE 8.11 Enron Corporation's network of email including messages with the word "FERC" exchanged between employees. Vertex size (1.5 to 4) is based on in-degree. Greener vertices received many FERC messages but did not send many out. The Harel-Karen layout was initially used, followed by Fruchterman-Reingold, to push the more peripheral vertices to the edges. Tim Belden, who pleaded guilty and witnessed against other top Enron executives, is labeled.

with whom. Filtering email collections to include only those that include a particular keyword or set of words is a particularly useful method for finding people related to some event.

You can see an example of this by analyzing the Enron email network of messages that include the term "FERC," the commonly used acronym for the Federal Energy Regulatory Commission, an "independent agency that regulates the interstate transmission of natural gas, oil, and electricity" (see www.ferc.gov). To create this FERC network, you can use the NodeXL import tools, making sure to filter messages to include those that have "FERC" in the body of the message. The resulting network includes 370 vertices representing employee email addresses and 672 weighted edges. This is a smaller subset of the Enron message network tagged by UC Berkeley students that includes 1803 edges and 1102 vertices. The total sent and received FERC messages are included on the Vertices worksheet (see Advanced Topic: Calculating Total Messages Sent and Received), along with a column called "%_Received," which equals Received/(Sent + Received).

Once you calculate the graph metrics, you can use them to create a graph such as Figure 8.11 designed to highlight important individuals. The graph sets the size of each vertex based on in-degree, because those receiving FERC messages from many different individuals are likely "go to" people. Vertex color is based on the %_Received data, with greener vertices indicating that the individual received many messages but did not send out many. Individuals with something to hide may not send out messages, suggesting that focusing on the large green vertices may lead to potential violators. Indeed, one of these vertices represents Tim Belden, the head of trading in Enron Energy Services considered by many to be the mastermind of Enron's scheme to drive up energy prices in California. Belden pleaded guilty to one count of conspiracy to commit wire fraud as part of a plea bargain and ended up serving as a key witness against many top Enron executives.

Although visualizations like Figure 8.11 can help identify individuals worth following up on, they should be used cautiously. In this particular example, there are no messages sent from Tim Belden in the data set, making it unclear if his high received ratio is due to his actual email usage patterns, purposefully deleted messages, or limitations with the original data set. Even if the data accurately reflect actual email patterns, Figure 8.11 is imperfect in that it emphasizes many individuals aside from Tim Belden who were not accused of illegal activities. Furthermore, many of those found guilty of crimes were not included in this graph at all, perhaps because they recognized the liability of using email for sensitive communication or perhaps because of limitations in the data set. Clearly, reading the content of the messages is of utmost importance. However, viewing the network can help identify individuals and messages

of interest. Once an individual is known to be involved, mining email is an effective way to identify people with whom the suspect frequently interacts. For example, Figure 8.11 shows a strong connection from John Shelk to Tim Belden (and many other recipients), which is explained by the fact that John Shelk often reported on congressional meetings but rarely received replies to his reports. Integrating the content with network visualization tools can provide a powerful exploratory platform, as has been done with the Enron network data set [4, www.enronexplorer.com].

8.10 PRACTITIONER'S SUMMARY

Email networks provide an intimate look into individuals' social and work relationships making them of interest to managers, community analysts, historians, researchers, and legal professionals. Because email is frequently and widely used in professional contexts, it reliably captures important aspects of many professional relationships. There are three main types of email collections: personal, organizational, and community. An analyst's existing experience with a collection is also important and impacts the types of questions asked and amount of detail needed.

Working with email networks can be challenging. Large collections must often be filtered to a manageable size. Filtering can be based on time, sender/receiver, messages' content, folders or labels, or any combination. Combining duplicate email addresses for the same individual can be time intensive but is often necessary. Integrating email networks with corporate personnel data can be challenging and poses ethical considerations, but when done responsibly can provide new insights.

Personal and organizational email networks can be explored for insights or shared with others to provide an overview. These networks may be based on individuals and their connections or on organizational units and their connections. Analysis can uncover important individuals and relationships such as boundary spanners, central members, broadcasters, and unresponsive recipients. Tightly connected subgroups can be identified and their relationship to one another can be mapped. The impact of interventions or external shocks on the network can be tracked over time, and common structural patterns such as recurring social roles or types of subgroups (e.g., cliques, fans) can be identified. These analyses can lead to actionable insights by identifying people or departments that need more cross-fertilization, helping newcomers get an overview of the social structure they are entering into, evaluating the effectiveness of a new community of practice initiative, and much more.

8.11 RESEARCHER'S AGENDA

The widespread use of email has fostered a growing community of researchers whose goals are to understand usage patterns so as to improve user interfaces and management tools. Researchers have focused largely on individual usage of email [5, 6], but they increasingly work on forensic tools to analyze other person's email or a group's email [3, 7]. A popular theme has been to improve the strategies for finding relevant documents in a large email collection [8, 9]. Exploration tools have built on the traditional keyword or key phrase search strategies, but increased attention to visualization tools has enabled users to get an overview of temporal patterns, relationships with individuals, or the social structure within groups [10–13].

The many opportunities to improve on email analysis systems is generating increased research on these issues and an increasing demand for such tools from corporate human resources staff, forensic investigators, legal analysts, and social scientists. The ability to detect temporal changes, such as sharp increases/decreases in communication among certain people or about certain topics, is a valuable guide to analysts. Temporal changes might be visualized by simple timelines or by animated changes to network diagrams, assuming stable layouts are used. The formation and dissolution of subgroups signal important changes that are useful in applications as diverse as detecting rumor spreading (gossip), corporate reorganizations, or antecedents of important events. Tying email to geographical position or even location in office buildings could help us to understand social processes in organizations.

References

[1] F.B. Viégas, D. Boyd, D.H. Nguyen, J. Potter, J. Donath, Digital artifacts for remembering and storytelling: PostHistory and Social Network Fragments, in: Proceedings of Hawaii International Conference on System Sciences (HICCSS), 2004, pp. 105–111.
[2] S.I. Donaldson, E.J. Grant-Vallone, Understanding self-report bias in organizational behavior research, J. Bus. Psychol. 17 (Dec. 2002) 245–260.
[3] A. Leuski, Email is a Stage: Discovering People Roles from Email Archives. Proc SIGIR 2004, ACM Press, New York, 2004, pp. 502–503.
[4] Heer, Exploring Enron: Visualizing ANLP Results, Available online at: http://hci.stanford.edu/jheer/projects/enron/v1
[5] N. Ducheneaut, V. Bellotti, Email as habitat: an exploration of embedded personal information management, Interactions 8 (5) (2001) 30–38.
[6] S. Whittaker, V. Bellotti, J. Gwizdka, Email in personal information management, Commun. ACM 49 (1) (2006) 68–73.
[7] J.R. Tyler, D.M. Wilkinson, B.A. Huberman, E-mail as spectroscopy: automated discovery of community structure within organizations, Informat. Soc.: Int. J. 21 (2005) 143.
[8] V.R. Carvalho, W.W. Cohen, On the Collective Classification of Email Speech Acts. Proc. SIGIR 2005, ACM Press, New York, 2005, pp. 345–352.

[9] D. Elsweiler, M. Baillie, I. Ruthven, Exploring memory in email refinding, ACM Trans. Information Systems 26 (4) (2008) 1–36.
[10] A. Perer, B. Shneiderman, D.W. Oard, Using rhythms of relationships to understand e-mail archives, J. Am. Soc. Inf. Sci. Technol. 57 (14) (2006) 1936–1948.
[11] A. Perer, M.A. Smith, Contrasting portraits of email practices: Visual Approaches to Reflection and Analysis, Proceedings International Conference on Advanced Visual Interfaces (AVI 2006), 2006, pp. 389–395.
[12] G. Venolia, C. Neustaedter, Understanding Sequence and Reply Relationships within Email Conversations: A Mixed-Model Visualization, Proc. CHI 2003, ACM Press, New York, 2003, pp. 361–368.
[13] F.B. Viegas, S. Golder, J. Donath, Visualizing Email Content: Portraying Relationships from Conversational Histories, Proceedings SIGCHI 2006, ACM Press, New York, 2006, pp. 979–988.

CHAPTER

9

Thread Networks

Mapping Message Boards and Email Lists

OUTLINE

9.1 Introduction 127

9.2 History and Definition of Threaded Conversation 128

9.3 What Questions Can Be Asked 129

9.4 Threaded Conversation Networks 130

9.5 Analyzing a Technical Support Email List: CCS-D 132

9.5.1 Preparing Email List Network Data 132

9.5.2 Identifying Important People and Social Roles at CSS-D 133

9.6 Finding a New Community Administrator for the ABC-D Email List 136

9.7 Understanding Groups at Ravelry 138

9.8 Practitioner's Summary 140

9.9 Researcher's Agenda 140

9.1 INTRODUCTION

Threads are the things that hold the net together. Since the inception of the Internet, most virtual communities have relied on asynchronous threaded conversation platforms as a main channel of communication. Usenet newsgroups, email lists, web boards, and discussion forums all contain collections of messages in reply to one another. The natural conversation style supported by the basic post-and-reply threaded message structure has proven enormously versatile, serving communities ranging widely in focus and goals. Cancer survivors and those seeking technical support or religious guidance are as likely to use a threaded discussion as a corporate workgroup. The new crop of search engines that mine these conversations, such as Boardreader, Omgili, and BoardTracker, underscore the popularity and importance of threaded discussions. Although the basic structure of threaded conversation has remained surprisingly similar since the early days of Usenet and email lists, designers have found new ways to integrate conversations with other content and ratings. Many threaded discussion forums now include profile pages showing a member photo and bio, participation statistics (e.g., number of posts, most recent activity), reputation scores based on user ratings of each member's content, and private messaging. Communities have also found ways to attach threaded message conversations to different entities such as people (e.g., a public wall on a person's profile page), items (e.g., movies; actors), groups (e.g., UMD alumni), and events. Not only have traditional forums become more feature laden, but threaded conversations have worked themselves into many other social media platforms such as Facebook (Chapter 11), YouTube (Chapter 14), Flickr (Chapter 13), and Twitter (Chapter 10). In this chapter we focus primarily on more traditional forums and email lists, but many of the analyses apply to other contexts.

The threaded conversation structure lends itself well to network analysis, because a directed link between individuals is created each time someone replies to another person's message. Unfortunately, most threaded conversation systems do not make this networked data easily accessible. The majority of threaded message content is not easily accessible because of the number of software platforms used and the fact that many groups only make content accessible to subscribed members.

Many threaded message systems do report participation statistics and ratings (e.g., top 10 contributors), which are important metrics. However, they fail to capture the social connections between members, a critical component of virtual communities and internal communities of practice.

9.2 HISTORY AND DEFINITION OF THREADED CONVERSATION

Threaded conversation is a commonly used design theme that enables online discussion between multiple participants using the ubiquitous post-reply-reply structure. It shows up in many forms from email lists to web discussion forums to photo sharing and customer review sites. The key properties of threaded conversation were enumerated in Resnick et al. [1] and are listed here with some modification:

- *Topics.* A set of topics, groups, or spaces, sometimes hierarchically organized to aid users in discovering interesting groups to "join." Topics or groups are persistent, though their contents may change over time. Figure 9.1 includes two topics: TOPIC 1: Social Media and TOPIC 2: NodeXL.
- *Threads.* Within each topic or group, there are top-level messages and responses to those messages. Sometimes further nesting (responses to responses) is permitted. The top-level message and the entire tree of responses to it is called a thread. In Figure 9.1, there are five unique threads. Thread A includes only two messages, whereas Thread B includes six messages. Thread D includes only a single message.
- *Single authored.* Each message contributed to a thread is authored by a single user. Typically, the person's username or email address is shown alongside the post so people know who is talking. In Figure 9.1, the author of each message and the time of its post are indicated. Users may post to multiple threads (e.g., Beth) or multiple times within a thread (e.g., Cathy).
- *Permanence.* In many threaded conversations including email lists and Usenet, once a message has been posted it cannot be rewritten or edited. A new message may be posted, but no matter how much someone may wish it, an original post often cannot be retracted. In some discussion boards and newer systems like Google Wave, original posts can be modified after initial contribution.
- *Homogeneous view.* The partitioning of messages into topics is a feature shared by many discussion interfaces. Moreover, in most systems users all see the same view of the messages in a topic, either in chronological or reverse chronological order.

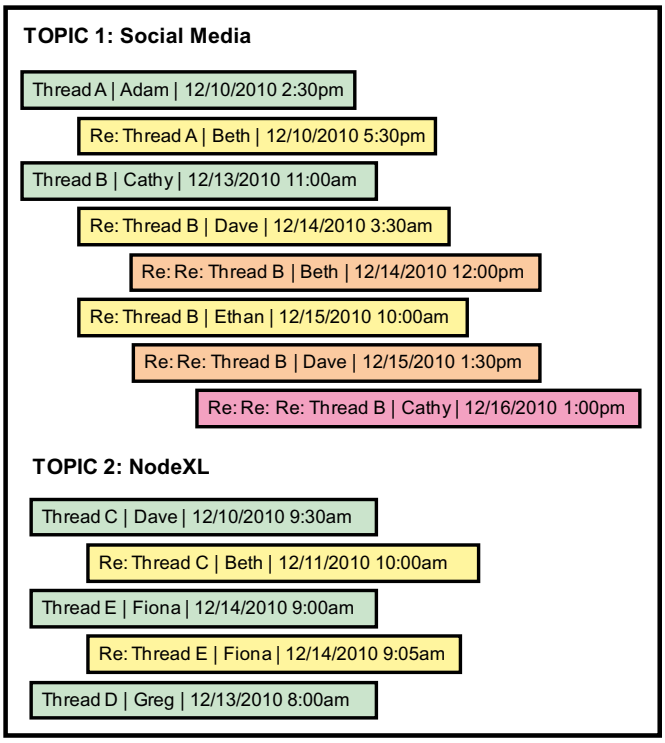


FIGURE 9.1 Threaded conversation diagram showing five threads that are part of two different topics. Each post includes a subject (e.g., Thread A), a single author (e.g., Adam), and a time stamp (e.g., 12/10/2010 2:30 pm). Indenting indicates placement in the reply structure. Green posts initiate new threads (i.e., they are top-level threads), yellow posts reply directly to green posts, orange posts reply to yellow posts, and the pink post replies to the orange post.

> Messages are often sorted into threads (e.g., Fig. 9.1). In some cases, the system will keep track of which messages a user has previously viewed, so that it can highlight unread messages, but that is the only personalization of how people view the messages.

In addition to this basic structure, there are a few important ways that threaded conversation platforms differ. Some, such as email lists, are push technologies that send updates to all subscribers. Others, such as discussion forums, are pull technologies that require individuals to visit a web site in order to view messages. However, email list archives and discussion forum email notifications blur these distinctions considerably. Also, we focus in this chapter on asynchronous threaded conversations, but many synchronous conversations such as Instant Messenger and chat follow a similar reply structure even if the pace of interaction and nature of messages is different.

Another important distinction is who can access content. Public conversations allow anyone who visits a web site (or email list archive) to read the content, even if they are not part of the community. These are often indexed by search engines, helping their content rise

out of obscurity. Semipublic conversations require users to create a username and log in before accessing content, but allow anyone to join, or at least anyone willing to abide by the community policies. Once you have logged in, you have access to all prior content that has been retained. Private conversations are only open to those who receive invitations via some existing member or are a member of some organization. Many corporate forums or email lists fall into this category.

The history of public online, threaded conversation communities began in the late 1970s with significant developments in the 1980s as bulletin board systems (BBS), email lists, and Usenet gained traction. The earliest threaded conversation systems relied on dialup connections, which encouraged groups within local telephone calling distance to form. Users of these early community systems demonstrated that text-only communication was sufficient to develop surprisingly meaningful relationships and rich cultures (e.g., see Jason Scott's "BBS: The Documentary," www.bbsdocumentary.com). Early communities covered topics ranging from dentistry to gaming to the occult. Access to public communities was often free. Others charged subscription fees, such as the WELL, a community of writers primarily from the San Francisco Bay Area described in Howard Rheingold's classic book "The Virtual Community: Homesteading on the Electronic Frontier" [2]. Technologies such as Listserv began to develop in the mid-1980s, allowing interested groups to create their own community email lists with increasing ease.[1]

As the Internet and the World Wide Web became ubiquitous in the 1990s, many of the original BBS services became or were replaced by Internet service providers (ISPs). Although they provided many services, asynchronous threaded conversation became one of the mainstays. Tools like Usenet that had relatively few users in the 1980s experienced exponential growth in the 1990s, growing from approximately 2000 newsgroups in 1991 to close to 11,000 newsgroups in 1996. Today, email lists and discussion boards are still the dominant technology underlying most online communities. They have proven surprisingly robust, enabling user groups to adapt them to a wide range of usage scenarios.

Threaded conversations have also worked their way into social networking sites, corporate intranets, customer review sites, and groupware tools like Yahoo! Groups. Over time, many threaded conversation tools have added complementary features including personal profile pages, message ratings, and activity statistics such as number of posts, number of times read, and membership join dates. Communities like Slashdot have developed distributed rating schemes that allow members to collaboratively identify messages with certain characteristics (e.g., high quality, humorous), as well as provide incentives for individuals to develop reputation scores. Google has introduced Wave, a new interaction platform that uses a threaded conversation structure at its core, although it allows posts to be edited and to include many types of content (e.g., search results, images, video, widgets, maps).

Research on communities that use threaded conversation began in the early days of BBS and Usenet. Many of the same themes continue to be explored today. For example, Kollock and Smith's 1999 book "Communities in Cyberspace" included chapters on identity online, deviant behavior and conflict management, social order and control, community structure and dynamics, visualization, and collective action [3]. All of these topics are still being explored in new contexts and with new technologies such as social networking sites, blogs, microblogging, and wikis. Early books by Preece [4], Kim [5], and Powasek [6] provided some enduring, practical advice and inspiration for those managing online communities.

Researchers from a variety of disciplines analyze threaded conversation communities and publish results in communication, business, information science, health, sociology, and computer science journals and conferences. Several have emerged around the Internet, such as the Association for Internet Research (AoIR), the Journal of Computer Mediated Communication (JCMC), the Association for Computing Machinery's Computer-Human Interaction (ACM-CHI), and Computer-Supported Cooperative Work conferences. Findings show that there is a consistent pattern of participation with few core members contributing the majority of content, many peripheral members contributing infrequently, and a large number of lurkers [7] who benefit by overhearing the conversations of others [8]. The nature of computer-mediated conversations depends largely on the type of community that engage in them. For example, technical and medical support communities differ in the level of empathy expressed [9] and the reusability of their content [10].

9.3 WHAT QUESTIONS CAN BE ASKED

There are many reasons to explore networks that form within large collections of conversations. New employees or community members need to rapidly catch up with the "story so far" to get to a point that they can make useful contributions. Community managers need tools to help them serve as metaphorical

[1]See www.lsoft.com/products/listserv-history.asp for a brief history of Listserv.

fire rangers and game wardens for huge populations of discussion contributors and the mass of content they produce. When outsiders such as researchers or competitors peer into a set of relationships, social network analysis can point out people, documents, and events that are most notable. A few of the specific questions that can be addressed with network analysis of community conversations are described next:

- *Individuals.* Who are important individuals within the community? For example, who are the question answerers, discussion starters, and administrators? Who are the topic experts? Who would be a good replacement for an outgoing administrator? Who fills a unique niche?
- *Groups.* Who makes up the core members of the community? How interconnected are the core group members? Are there subgroups within the larger community? If so, how are the subgroups interconnected? How do they differ?
- *Temporal comparisons*. How have participation patterns and overall structural characteristics of the community changed over time? What does the progression of an individual from peripheral participant to core participant look like and who has made that transition well? How is the community structure affected by a major event like a new administrative team, the leaving of a prominent member, or an initiative to bring in new members?
- *Structural patterns.* What network properties are related to community sustainability? What are the common social roles that reoccur among community members (e.g., answer person, discussion starter, questioner, administrator)?

9.4 THREADED CONVERSATION NETWORKS

The network most commonly used to analyze threaded conversations is the Reply network. Each time someone replies to another person's message, she creates a directed tie to that other person. If she replies to the same person multiple times, a stronger weighted tie is created. To understand the nuances of how a Reply network gets created, you can compare the original data in Figure 9.1 to the Reply network derived from that data and shown in Figure 9.2.

A related, but different network is the Top-Level Reply network, which connects all repliers to the person who started each thread (Fig. 9.3) instead of the person they are replying to directly. This network emphasizes those who start threads (i.e., post the top-level message), while de-emphasizing conversations that occur midway through a thread. In some communities with short threads where all replies are typically directed at the original poster, such as Question and Answer (Q&A) communities, this network can better reflect the underlying dynamics. However, in discussion communities or forums with longer threads, the standard Reply network is typically preferred because people later in the thread are often replying to each other.

Affiliation data connecting posters to threads (or forums) can also be used to create bimodal networks.

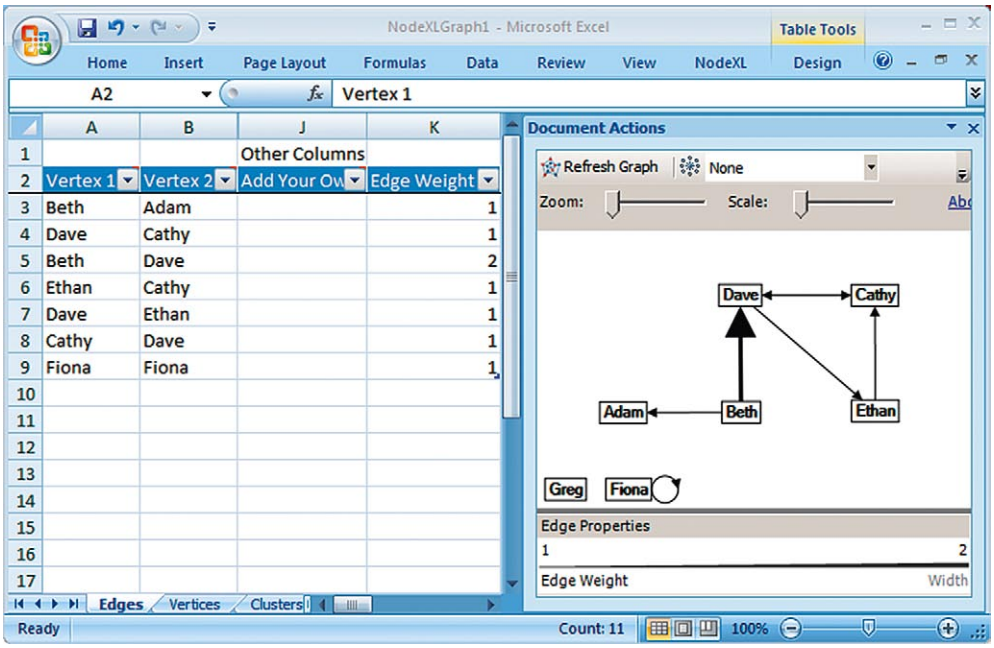


FIGURE 9.2 An example discussion Reply network graph displayed in NodeXL, based on the data found in Figure 9.1. The network is constructed by creating an edge pointing from each replier to the person he or she replied to and then merging duplicate edges. Notice that Beth has replied directly to Dave twice, so the edge connecting them is thicker. Fiona replied to her own message, so there is a self-loop. Greg started a thread but was not replied to. He would normally not show up on the graph because he is not in the edge list; however, he was manually added to the Vertices tab and his visibility was set to "Show," so he would appear.

These are undirected, weighted networks that connect posters (i.e., users) to threads. For example, an edge would connect Cathy to Thread B with a weight of 2 because she posted to that thread twice. Beth would be connected with a weight of 1 to Thread A, Thread B, and Thread C because she posted to each of them once. This network is ideal for identifying boundary spanners, as you will see in Section 9.7. As discussed in Chapter 6, affiliation networks can be transformed into two additional undirected, weighted networks: a user-to-user network connecting people based on the number of threads (or forums) they both contribute to, and a thread-to-thread (or forum-to-forum) network connecting threads together based on the number of contributors they share. These networks are good for creating overview graphs of large communities with many threads or forums.

Preparing data needed to create threaded conversation networks can be challenging because they rely on such a wide range of technologies. Email lists are the easiest conversations to capture in NodeXL, because you can use the email import wizard as discussed in Section 9.5.1. Data from discussion forums, Usenet, and related systems must be generated by using screen scrapers, manually entering data, or performing queries on the forum's database or, in an increasing number of cases, through a web Application Programming Interface (API). Whatever approach you use, your data set will likely have header information that includes some of the following information for each message: a time stamp, a message author, an identifier for the message this message is a reply to (if any), a subject line (or thread ID), a set of tags, an attachment, a link to the author's profile, a group or forum the thread is a part of, and a rating. A separate file of information on each user is also often useful. It may include aggregate participation statistics on other community activities (see Section 9.7 for an example). All of these data can be useful in creating maps of conversation networks, but at the core a simple edge list is the minimum necessary to start a social network analysis of a conversation.

The type of discussion platform in use will also influence the potential data problems you are likely to run into. For example, email lists often have people registered with multiple email addresses, making it necessary to combine duplicate addresses for the same person (see Chapter 8). Email lists also have problems when the reply structure isn't clear, because people reply to the email list address rather than to one another as described in Section 9.5.1. Corporate email lists and discussion forums typically have the cleanest set of unique identifiers for individuals, but even then, name and title changes can cause problems.

It is also important to realize that the reply network from an email list only captures messages sent or posted to the list. Many personal messages sent directly to and from individuals on the email list are not captured. Depending on the type of community and the default settings (e.g., email list Reply To settings), these private messages may account for the majority of all messages exchanged among a population. In addition, other types of communications such as corporate meetings, phone calls, and instant messenger exchanges are invisible to discussion forum networks. People who communicate

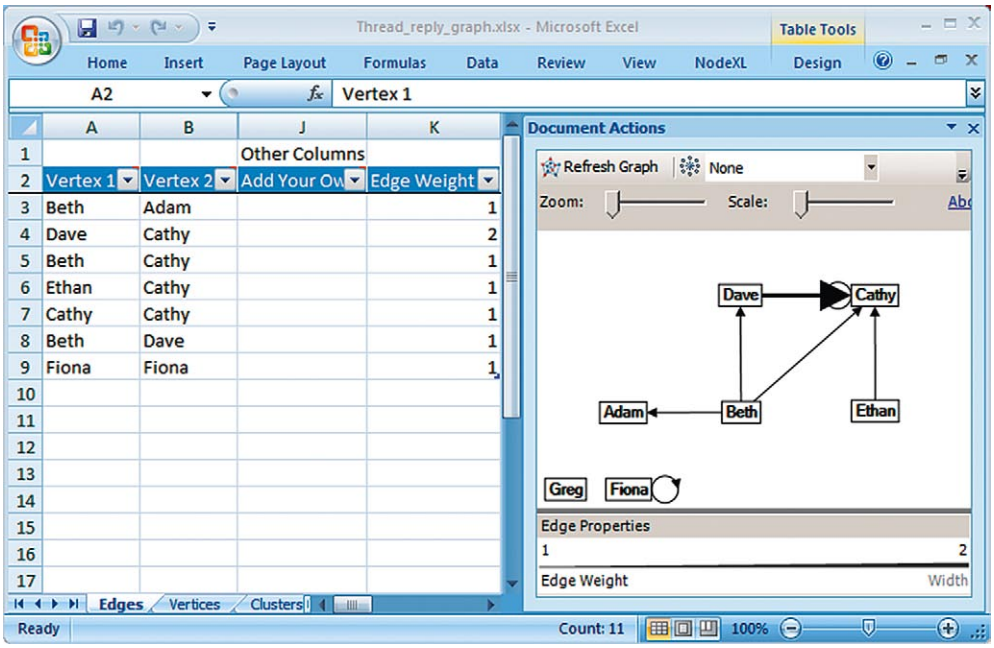


FIGURE 9.3 A Top-Level Reply network graph displayed in NodeXL based on the data found in Figure 9.1. The network is constructed by creating an edge pointing from each replier to the person who started the thread (i.e., posted the top-level message) and then merging duplicate edges. Notice how Cathy plays a more prominent role (i.e., has a higher in-degree) than in the standard Reply network graph (Fig. 9.2) because she started the longest thread and all subsequent repliers link to her. Self-loops are more frequent in this type of network because people like Cathy may respond to those who replied to her initially, leading to a self-loop.

and contribute more effectively through other channels may show up only marginally in discussion forum data sets. However, even given these limitations, analysis of threaded conversation networks can provide vital information about community dynamics and help identify important individuals and groups.

9.5 ANALYZING A TECHNICAL SUPPORT EMAIL LIST: CCS-D

There are a host of technical support groups that use email lists, Usenet newsgroups, or web discussion forums to help individuals solve problems and make sense of a specific technology like Java, a product such as the iPhone, or a topic such as web design. Many companies host these forums to learn about problems with existing products, resolve customer concerns, generate new ideas on future improvements, and build a loyal customer community. To meet these goals, it is often important to understand which individuals play important roles within the community, something that can be challenging when managing multiple, active communities. In this section you will learn how to identify key members of the CSS-D email list devoted to the effective use of cascading style sheets (CSS) in web design. It is a highly active list with around 50 messages sent each day. There are a handful of administrators who keep the conversation friendly and encourage contributors to follow the guidelines. See [10] for a complete description of the community and some of the strategies used by the community administrators and members to make it so effective.

9.5.1 Preparing Email List Network Data

The easiest way to collect data from an email list is to subscribe to it, import the emails into Outlook (or Outlook Express), and use NodeXL's Import feature to create an edge list using the From Email Network option (see Chapter 8 for details). If you are not subscribed to the email list, you may be able to download an archive of the past messages and use a specialized email program (see Chapter 8) to import, convert, and export the data you need. Here we assume that you are subscribed to the list.

You will need to include only emails sent to the list. If they are in a separate folder, you can use the Import from Email Network dialog filter to only look in that folder. If list emails are mixed in with your personal emails, you can use the Subject line filter (e.g., Subject Includes Text: [css-d]), assuming the name of the email list is included in the Subject line as is common (Fig. 9.4). Alternatively, you can include only those messages sent to the list (in the To, Cc, or Bcc fields) (Fig. 9.4). You will typically want to restrict the messages to a certain time. In this case, data for January and February of 2007 are included.

Email list data are unique in that all messages are sent to the same address (in this case css-d@lists.css-discuss.org). The result is a star network connecting all contributors' email addresses to the list email address. Messages that begin a new thread (i.e., initial posts) will be sent to the list address and rarely will Cc other individuals. The way that replies to initial posts are handled depends largely on the email list configuration. Some lists, like css-d, set the default Reply

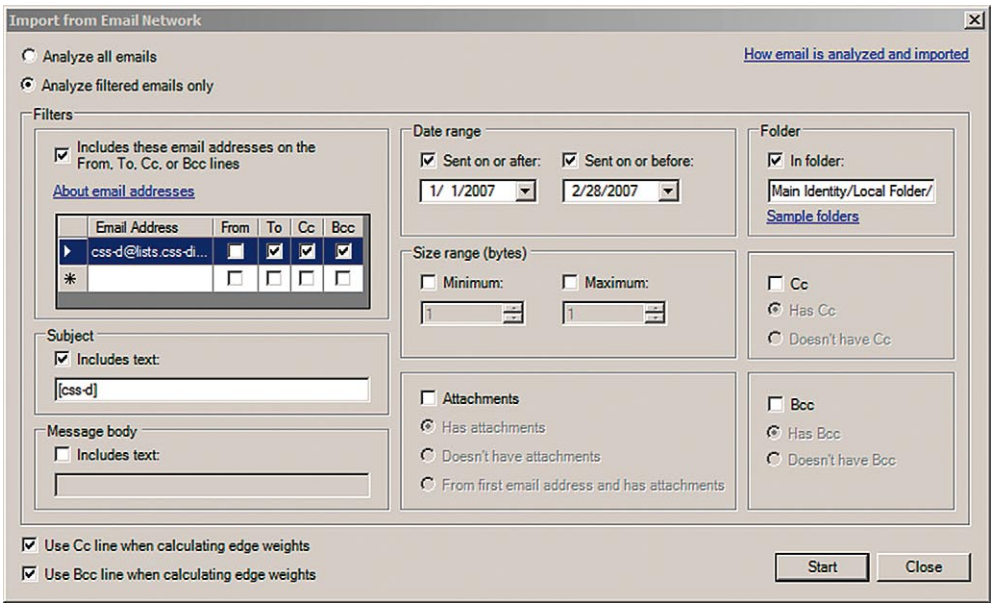


FIGURE 9.4 NodeXL's Import from Email Network dialog showing two methods to restrict the network to messages sent to the email list. One method is to include messages with "[css-d]" in the Subject line. The other method is to include messages sent to the list address (css-d@lists.css-discuss.org). Messages are also filtered to only include 2 months of data: January and February of 2007.

To address to that of the original sender. If you click "Reply" to the initial email in your email client, it will send directly to the person, whereas if you hit "Reply to All," it will send to the initial person in the To field and the email list in the Cc field. This configuration is good for network analysis using the NodeXL email import in that most of the replies include the email address of the person that is being replied to. It does encourage more private messages, however, which are missed by the email list and are thus absent in the network analysis. Other email lists set the default so that when you click on "Reply To" it sends to the list and you must choose "Reply to All" to explicitly Cc the initial sender. This configuration can cause problems when using the NodeXL email import, because it cannot detect who is replying to whom. To analyze these lists, you would need to use an email program that can identify threads based on subject lines and header information, making it possible to reconstruct who is replying to whom.

To remove the overwhelming effects of the email list address, you will need to remove it from the network. Choose Show Graph to create the list of vertices. Next, find the email list address row on the Vertices worksheet and type "Skip" into the Visibility column so that it will not be included in future analysis and visualizations. For example, compute the metrics and you will see that no metrics are calculated for the email list address, which is desirable in this case. Choose Refresh Graph and you will see that the email list address is not included in the graph.

9.5.2 Identifying Important People and Social Roles at CSS-D

In an online community, users contribute in different patterns and styles. In other words, community members fill different social roles. Understanding the composition of social roles within your community can provide many insights that can help you be a more effective community manager. Unfortunately, simple activity and participation metrics are unable to capture the different types of contributions in discussion forums. In contrast, social network analysis provides metrics that can be used to automatically identify those who fill unique social roles and track their prevalence over time. This can help community managers:

- *Identify high-value contributors of different types.* Which community members are the most important question answerers or question starters? Who connects many other users together? Answering these questions can help you know who to thank (and for what) and how to support individuals' needs.
- *Determine if your community has the right mix of people.* Is your community attracting enough question answerers? Are there enough connectors to hold the community together? Is discussion crowding out Q&A? Is a discussion space dissolving into Q&A? Knowing the answers to these questions can help you know who to recruit or encourage more, as well as what policies may be needed.
- *Recognize changes and vulnerabilities in the social space.* How has the community composition changed as it has grown? What is the effect of a certain prominent member leaving the community going to have? Which members are currently irreplaceable in the type of work they do? What is the effect of a policy change or change in settings on the community dynamics (e.g., changing the default Reply To behavior to send to the sender versus the entire list)? Answering these questions can help you prepare for change, understand the effects of prior decisions and events, and cultivate important relationships.

In this section you will learn how to identify important individuals and social roles within the CSS-D community. You will do this by using subgraph images and creating a composite metric that helps identify the two most important social roles within Q&A communities like CSS-D: answer people and discussion people. You will then use this metric to develop visualizations that show the relationships between these individuals.

The easiest way to get a sense of the key individuals within a network is to create the 1.5 subgraph images for each vertex using a layout like the Harel-Koren Fast Multiscale layout. Once these subgraph images of each email contributor's local networks are created, you can sort on the graph metrics in the Vertices worksheet associated with each email contributor such as in-degree (who receives messages from the most people) and out-degree (who sends messages to the most people) to bring differently connected individuals to the top. You can also sort by centrality measures like eigenvector to get a sense of who is a core member of the community because this member is an active participant and talks to other active participants.

Scanning through the Subgraph Images of CSS-D contributors helps you get a sense of the different social roles that exist within the email list community. Figure 9.5 shows examples of three types of contributors (question people, answer people, and discussion starters) along with some of the metrics that could be used to identify them (see Advanced Topic: Social Role Measures for more). Question people post a question and receive a reply by one or two individuals who are likely to be answer people. Answer people mostly send messages (arrows point toward other vertices) to individuals who are not well connected themselves [11]. Discussion starters mostly receive messages (arrows point toward them), often from people who are well connected to each other.

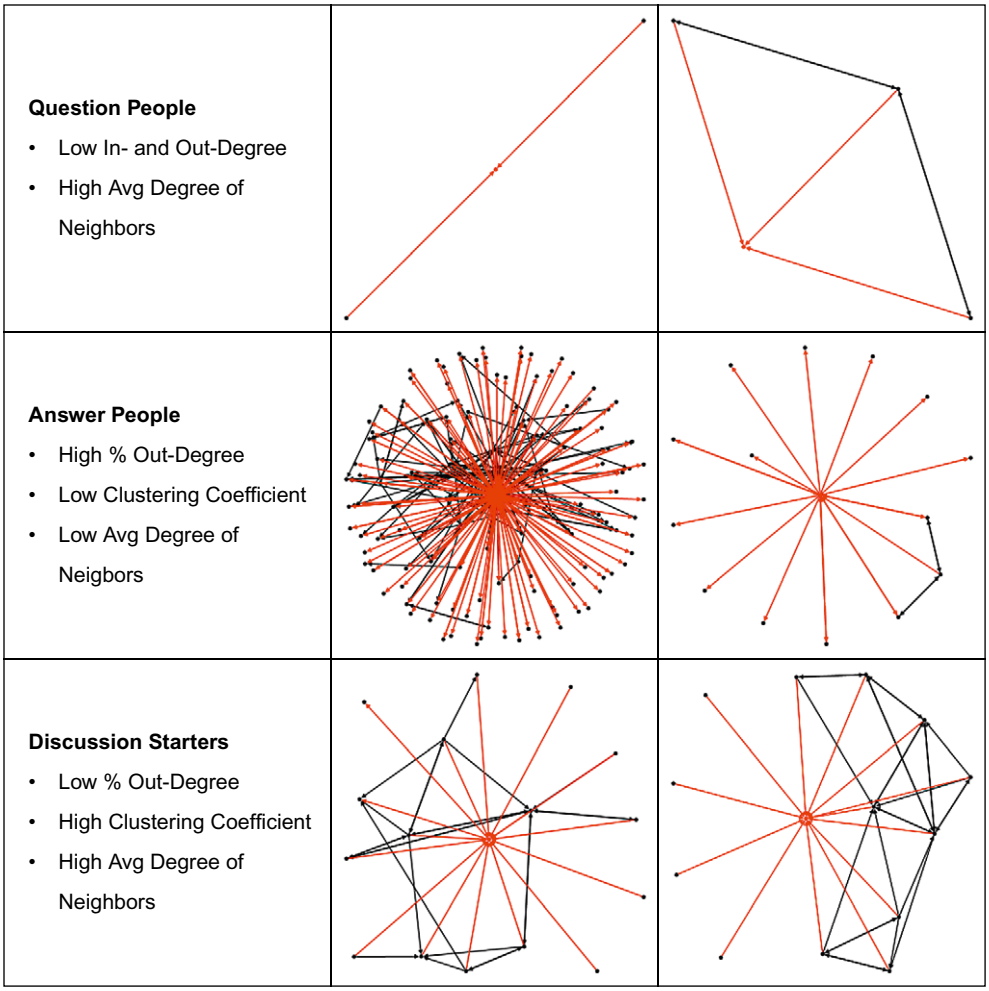


FIGURE 9.5 NodeXL Subgraph images (1.5 degree; vertex and incident edges highlighted red) for six CSS-D contributors that fill three different social roles within the CSS-D community. Answer people predominantly reply to questions from isolates (i.e., those who are not connected to others). Question people typically have a low degree themselves, but they receive messages from those with high degree (i.e., answer people). Discussion starters initiate long threads and receive many replies, often from people who know each other.

ADVANCED TOPIC
Social Role Measures

You can typically identify a person's social role by looking at his or her subgraph image (Fig. 9.5), but doing so for many individuals becomes problematic. Instead, it is possible to create aggregate metrics that automatically identify those who play certain social roles. These metrics consist of a combination of network metrics and participation metrics. Automatically identifying social roles within a community using metrics facilitates their tracking over time, which allows you to keep your pulse on the health of your community. It can also be used in combination with visualizations as shown in Figures 9.6 and 9.7 to more easily understand individuals' social roles and how they relate to one another. The specific metrics used to identify social roles will depend on the metrics that are available (i.e., those that are tracked) and will be tied to some extent to the underlying type of social media being analyzed. Some metrics, such as the core social network metrics, can be created using NodeXL. Other metrics, such as the average posts per thread or the days active, must be tracked via some other means. Table 9.1 shows several metrics that can be used to identify answer people in threaded conversations.

All of these metrics are devised so that higher values correspond with typical answer person behaviors. Depending on what data you have available, you can combine different metrics into an aggregate metric by averaging them, multiplying by them, or taking a weighted average. Later in this section we calculate a simple, aggregate answer person score by multiplying the percent out-degree and the clustering coefficient inverse, both of which are easy to calculate in NodeXL. We also create a cutoff total degree (out-degree + in-degree), so

TABLE 9.1 Social Role Metrics

Metric	Description
(User's Thread Count) ÷ (User's Post Count)	Brevity is preferred. Larger values = fewer messages per thread.
(User's Reply Posts) ÷ (User's Total Posts)	Initiation is avoided. Larger values = avoids starting threads.
(User's Degree) ÷ (Total Users)	Talks to many people. Larger values = replies to a significant fraction of community members.
(1 − Clustering Coefficient)	Talks to people who aren't well connected to each other. Larger values = lower clustering coefficient (i.e., less well-connected neighbors)
1 ÷ Avg of Neighbor's Degree	Talks to people who connect to few others. Larger values = talks to more isolates
(User's Days Active) ÷ (User's Possible Active Days)	Posts on most days. Larger values = posts on multiple days more often.
(User's Out-Degree) ÷ (User's Out-Degree + User's In-Degree)	Percent out-degree. Larger values = is connected to more people because of replying to them than because of receiving from them.

that only vertices with a total degree of 15 or higher are candidates. High values of the answer person score identify answer people as expected. In addition, low values identify discussion starters, as they have a high percent in-degree (they solicit replies from many people while not sending messages to many people) and high clustering coefficient (those who they are connected to know each other).

The specific social roles and their prevalence within a particular community will depend on the nature of that community. Because the CSS-D community is primarily a Q&A community, it consists of mostly question askers, a handful of prominent answer people, and a small number of discussion starters. Other more discussion-based communities would have many more discussion starters as well as other social roles such as flame warriors, commentators, and connectors. Tracking the ratio of people that play different social roles can be a good way to assure that a community is healthy. For example, if the CSS-D community had too few answer people or an influx of many question people, it could not function effectively.

Viewing the entire reply network for the CSS-D email list (Fig. 9.6) can provide some general insights about the composition of its population, although the size of the network can make it challenging to interpret without filtering. Figure 9.6 maps the answer person score (see Advanced Topic: Social Role Measures) to color: green-colored nodes represent answer people, red-colored nodes represent discussion starters, and blue nodes have a total degree of less than 15. Larger nodes have a higher eigenvector centrality suggesting they are connected to many people and others who are well connected. The binned layout is used to identify isolates, of which there are many because the email list address itself was removed. Isolates represent those who posted to the list and didn't receive a response (e.g., they posted an announcement) or in some cases those who replied to the list without copying in the address of the person who they were replying to. Overall the composite network shows many individuals connected primarily through a handful of central question answerers and a small but stable core group of members that interact with one another regularly.

To better focus in on the core members of the community you can filter out vertices with a total degree of less than 15. Figure 9.7 shows the resulting network after manually positioning the vertices. Subgraph images for the top three answer people are shown. The edge weights, represented in the edge width and opacity, provide a good sense of who interacted with whom during the 2-month time period and is thus likely to know each other and perhaps have similar interests. Note that even among these core members, discussion starters rarely reply to other discussion starters. Also notice that the largest vertex (i.e., the one with the highest eigenvector centrality), while categorized as an answer person, receives many messages from the core members. This suggests that he plays multiple important roles within the community. In fact, if he were removed from the network, there would be considerably fewer connections between the core members. This suggests that community administrators should make sure this individual is adequately appreciated and encouraged to remain in the community.

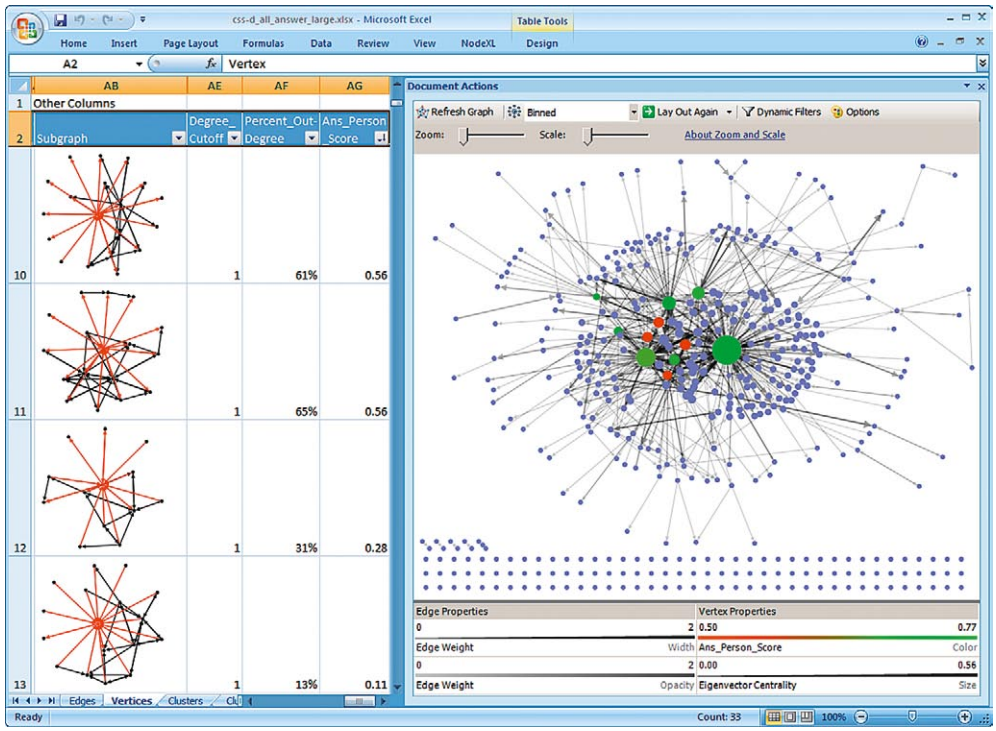


FIGURE 9.6 NodeXL map of the CSS-D email list network for January through February of 2007 after removing the vertex for the email list address itself. Answer people (greener) and discussion starters (redder) are identified by the calculated answer person score (see Advanced Topic: Social Role Measures). Blue vertices have a total degree of fewer than 15. Subgraph images (1.5) of the top four discussion starters are shown. Vertex size is mapped to eigenvector centrality. Edge weight is mapped to both edge size (1.5–4) and opacity (20–80), applying the logarithmic scale and ignoring outliers. Like many help-based communities, CSS-D consists of mostly question askers with a handful of answer people and discussion starters.

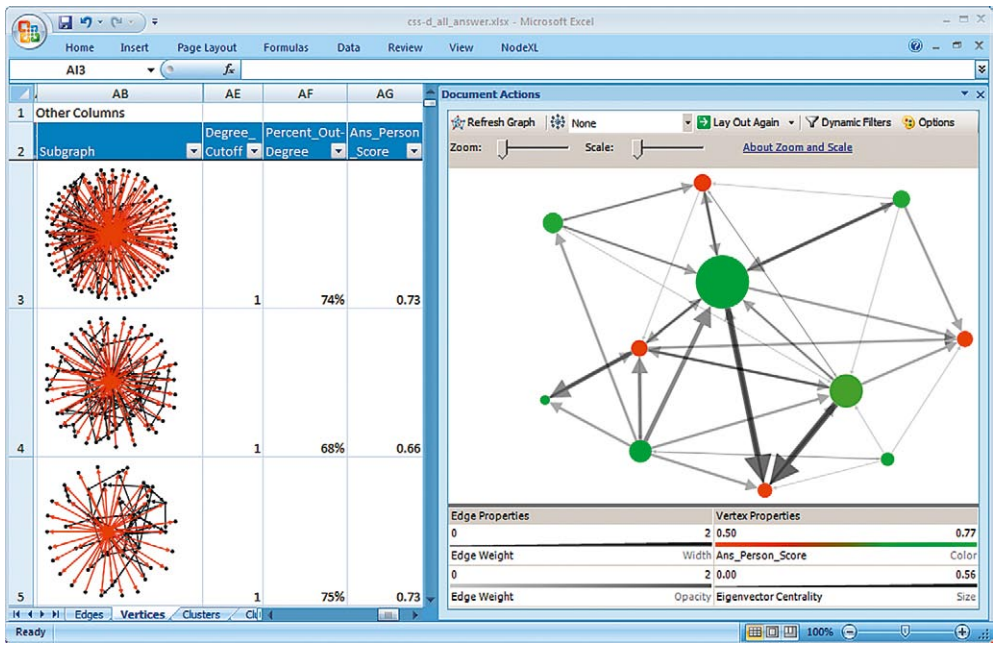


FIGURE 9.7 NodeXL map of the filtered version of the CSS-D email list seen in Figure 9.6 showing only the most central members. The maximum size of vertices and edges has been increased to more clearly draw comparisons.

9.6 FINDING A NEW COMMUNITY ADMINISTRATOR FOR THE ABC-D EMAIL LIST

Administrators of online conversations play pivotal roles in maintaining social order, encouraging participation, and making communities feel like home [12]. They are typically among the most active members of a community and can function better when they are known and respected by the members of the community. Because of the importance of administrators, when one leaves or steps down it has the potential to disrupt

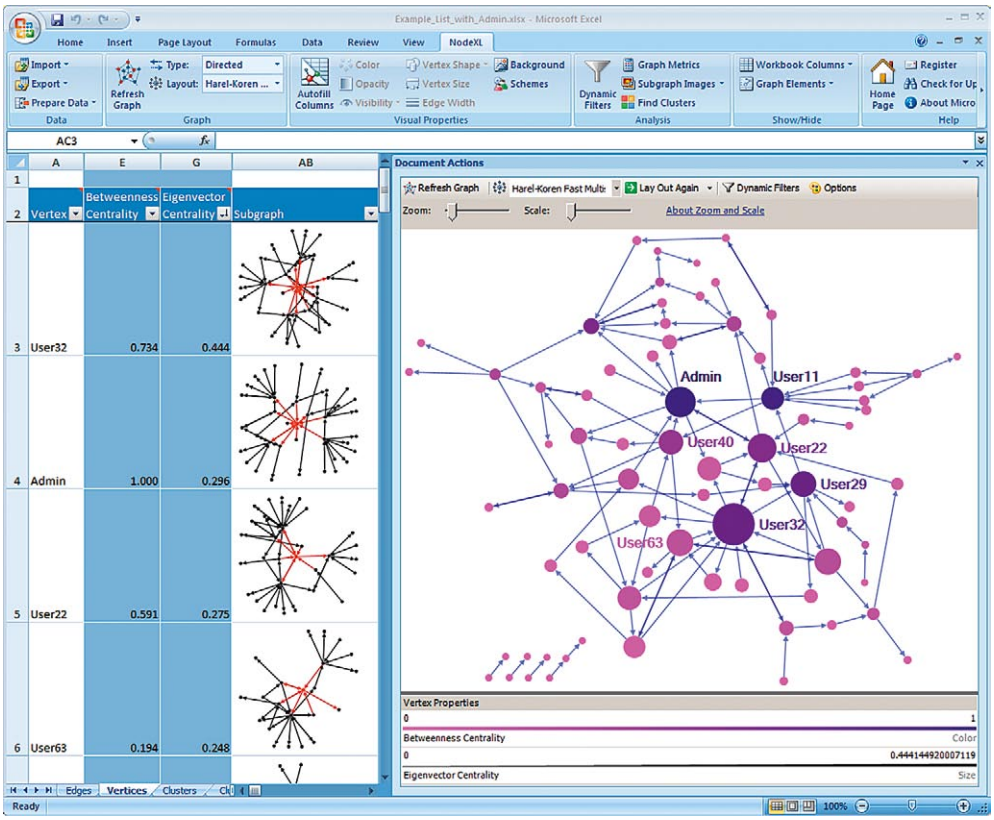


FIGURE 9.8 NodeXL map of ABC-D's email list reply network. The current administrator (Admin) and other members that are most central to the network are labeled. Larger vertices have a higher eigenvector centrality and darker purple vertices have a higher betweenness centrality. Subgraphs are shown in the worksheet to indicate each member's local 2-degree network neighborhood.

the community. In this section we look at how network analysis can help in identifying a potential replacement for an administrator that is going to step down.

Data for this analysis comes from an email list we will call ABC-D, based around a specific profession. It is a classic example of a community of practice [13] that spans multiple institutions. Unlike CSS-D, ABC-D encourages in-depth discussion about the community's domain and is not primarily about questions and answers.

The network is directed: an arrow pointing from person A to person B indicates that person A replied to a message of person B. The threaded nature of the conversation (i.e., who started the threads) is not reflected in the data. Data from ABC-D was collected for a 2-week period by Chad Doran, a graduate student at the University of Maryland College of Information Studies, who also came up with the administrator replacement scenario. A more complete analysis would include a longer time period (e.g., 2 months) and include edge weights, but the current data set is sufficient to illustrate the key idea. All data, including the name of the community, has been anonymized to respect the privacy of the group members.

The first graph, Figure 9.8 shows the entire reply network with a few key individuals (as identified by graph metrics) labeled. The network is interesting in that individuals are almost all connected in one large component, but the degree of any one individual is relatively low (e.g., the maximum total degree is 14 and the average total degree for an individual is about 3). The result is a fairly spread out network. Graph metrics were calculated and used to identify the most central individuals, who presumably are in the best position to serve as an administrator replacement. Dark purple vertices have a high betweenness centrality, suggesting that they are important at connecting different vertices and integrate the network as a whole. Larger vertices have a higher eigenvector centrality, which in this case suggests that the person is well connected to others who are themselves well connected.

As expected, the current administrator (labeled Admin) has the highest betweenness centrality and a high eigenvector centrality. Interestingly, the individual with the highest eigenvector centrality (User32) is not directly connected to the admin, in the time period of our data collection neither of them replied to the other. Another important individual is User11 who has a high betweenness centrality because he was the only link to several vertices, but he has a relatively low eigenvector centrality because most of his connections were with individuals who rarely posted. All of the labeled individuals scored high on the metrics and may be good candidates to replace the administrator. Of course, other characteristics not captured in the network structure,

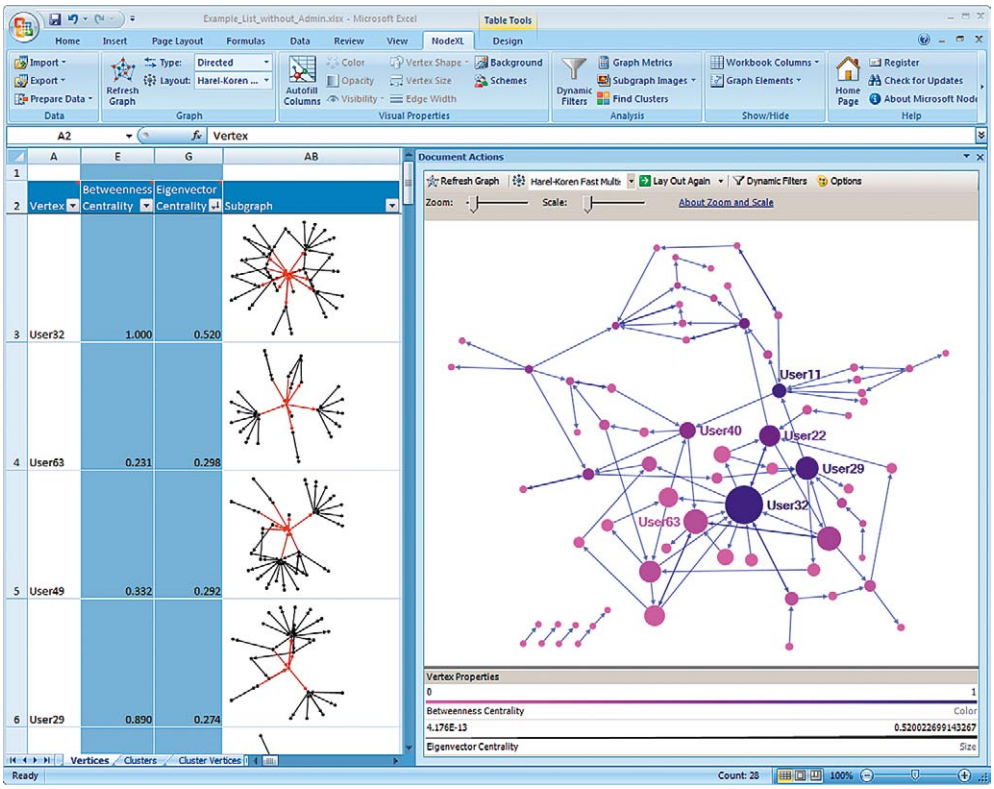


FIGURE 9.9 NodeXL map of ABC-D's email list reply network with the current administrator removed from the network. Graph metrics are recalculated without the administrator. Size and color mean the same thing as the prior graph (Fig. 9.5), and vertex locations have been fixed to facilitate comparison.

such as their willingness to serve, their friendliness, and their experience, would also be key determinants.

You may want to know how things would change if we removed the administrator from the network (Fig. 9.9). The key network metrics, betweenness centrality and eigenvector centrality, will change for the remaining individuals because they are dependent on the network properties of other vertices (see Chapter 5). Thus, looking at the graph without the admin can help more accurately assess an individual's potential as a replacement. It also helps you to see how the network as a whole may be impacted. For example, removing the admin increases the average closeness centrality from 3.2 to 3.5, suggesting that people will not be as directly connected with others once the admin is gone. You may also notice certain subgroups within the network that lose an important connection to other subgroups, such as the large group at the top of the graph. These differences can be more easily noticed when the location of the vertices has been fixed (see Chapter 4).

Figure 9.9 confirms that the initial individuals identified as possible replacements are good candidates. It also suggests that if certain candidates were chosen, such as User32, there may be subgroups of the community that would not be as well connected (e.g., the group of nodes at the top of the graph). The fear is that these individuals may feel alienated by a new administrator they either don't know or don't converse with often. The graph also points out individuals who may be able to keep them involved: User11 and User22. Armed with this information, the outgoing administrator may be wise to recommend that User32 and User22 jointly serve the role of administrator or that whoever is chosen should foster a relationship with these individuals to link to those who may feel alienated.

9.7 UNDERSTANDING GROUPS AT RAVELRY

Ravelry (www.ravelry.com) is a thriving online community for anyone passionate about yarn. As of January 2010, more than 600,000 knitters and crocheters were registered on the site. Users organize their projects, yarn stashes, and needles; share and discover designs, ideas, and techniques; and form friendships through discussions and exploration of shared interests. In this section, you play the role of a fictional Ravelry community administrator. You will work with data on the top 20 posters to three discussion forums created for different groups. The data and initial network analysis for this section were developed by Rachel Collins, a graduate student at University of Maryland's iSchool. Special

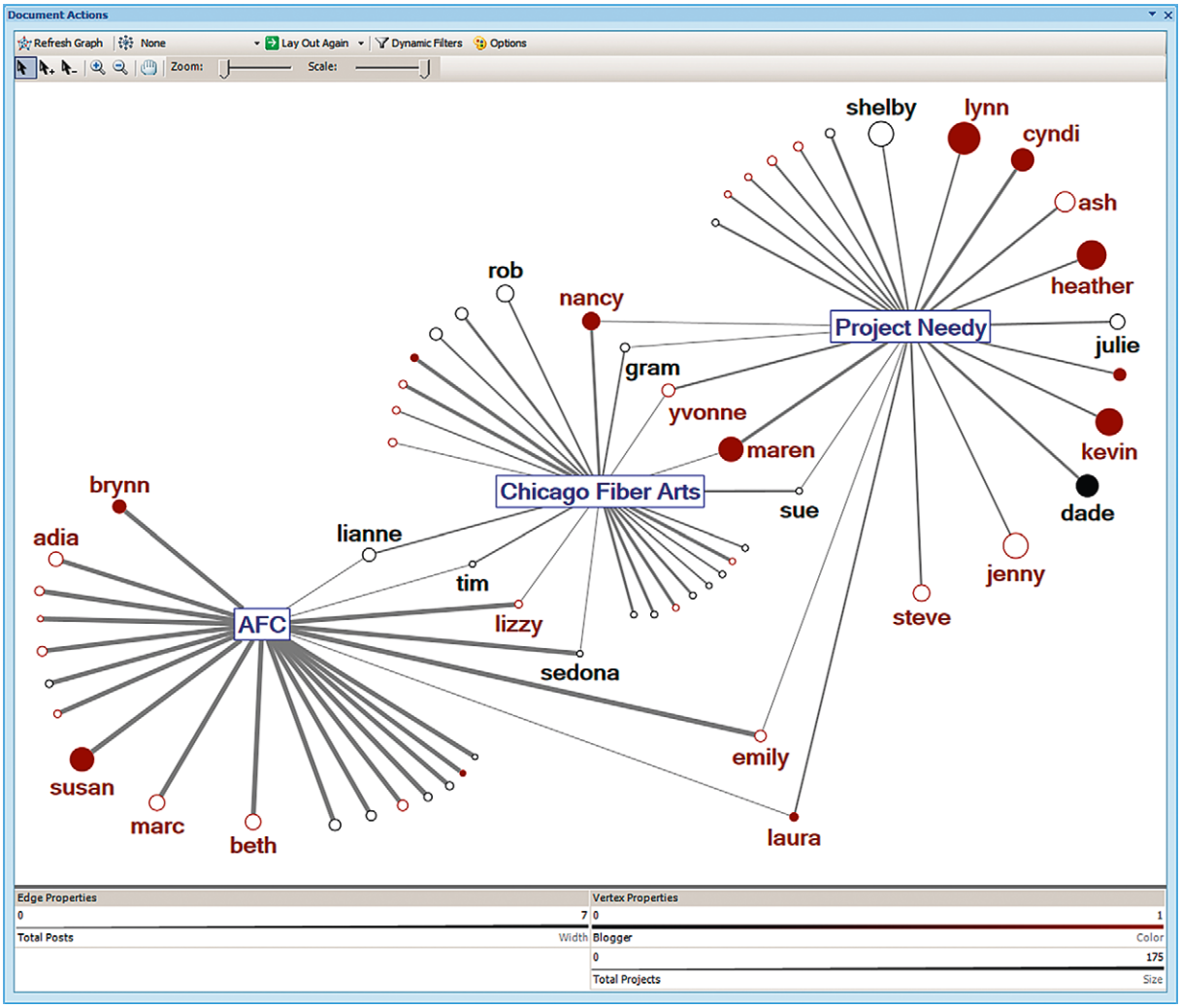


FIGURE 9.10 Bimodal network connecting three Ravelry groups (i.e., forums) represented as blue text boxes to contributors represented as circles. Edge width is based on number of posts (with logarithmic mapping). Vertex size is based on number of completed Ravelry projects. Maroon vertices have a blog and solid circles are either community moderators or volunteer editors. The network helps identify important boundary spanners (e.g., those connected to multiple groups) as well as compare groups.

thanks to the Ravelry community for allowing us to analyze it and discuss their fascinating community in the book. All group and individual usernames have been modified for privacy reasons.

Imagine you are assigned three group discussion forums to monitor and help develop. They are highly active groups, making it hard to keep up with all the messages and see the forest from the trees. You'd like to get a better sense of how the most important community members relate to one another, as well as how the groups differ. This understanding will help you recommend the best group for a newcomer to join (which could mean you link to the forum more prominently in your web site), as well as identify individuals with certain expertise or social relations that you can call upon if needed. You also hope to share the visualization with the groups themselves to encourage self-reflection.

The three groups you are in charge of include one common-interest group (Apathetic Funloving Crafters [AFC]), one meetup (Chicago Fiber Arts), and one knit-along (Project Needy). They are three of hundreds of similar groups. Discussion forums for each group serve as their central hubs. Individuals can participate in as many forum groups as they desire. You have collected data on project output, discussion board usage, blog activity, community roles, and total friends for the top 20 posters in each group. This lets you relate many different activities together in a single analysis, focusing attention on the most active members who are typically the most important.

Figure 9.10 shows a bimodal graph of the three forums/groups (shown in blue) connected to

individuals who have posted to them. Edge thickness is based on the number of forum posts (using a logarithmic mapping). The thinnest lines connect users to groups that they are members of but have not yet posted to. Other visual properties are used to convey individuals' levels of activity in other parts of the community. Maroon vertices also maintain a community blog, whereas solid disks are community moderators or volunteer editors. The graph helps you identify important individuals, such as those who post to multiple groups or have certain color/size/shape combinations. It also helps you compare the three groups. For example, the graph makes clear that the Apathetic Funloving Crafters (AFC) forum is very active, includes many bloggers, and includes relatively few people who complete a large number of projects (perhaps explaining the "Apathetic" in the title). In contrast, the Project Needy group includes many highly productive members, many of whom are both administrators and bloggers. In contrast, the Chicago Fiber Arts group has fewer bloggers and less project activity.

A newcomer to the Ravelry community could use a visualization like the one displayed in Figure 9.10 to quickly get a sense of which group(s) he or she may want to join, as well as identify some of the prominent members. Administrators can use similar graphs to identify potential candidates for volunteer editors or identify clusters of boundary spanners with which to form new groups because of shared interests. Providing graphs like this one to the groups themselves can also prompt self-reflection and potentially foster new connections. They can also be used to better understand how the activities on the site relate to one another, although use of statistics may be needed to more systematically validate initial claims. For example, Figure 9.10 shows that location-based groups have a lower percentage of active members who blog and people who complete many projects seem to cluster into project groups. Understanding trends such as these can help you better target your community and groups around the different user types involved.

9.8 PRACTITIONER'S SUMMARY

Many online communities use threaded conversations in the form of email lists, discussion forums, and Usenet. Although they make use of a wide range of technologies that differ in their delivery infrastructure, all threaded conversations share similar characteristics: they are composed of single-authored messages organized into threads (i.e., a top-level message, replies to that message, and possibly replies to those replies), threads are often found within topics or groups, messages are often permanent, and users often have a shared view of the conversation. These conversations lend themselves to the creation of several networks including the directed, weighted Reply network and Top-Level Reply network; the undirected, weighted affiliation network connecting threads (or forums) to the individuals that posted to them; and the undirected, weighted networks derived from the affiliation network including the user-to-user network and thread-to-thread network. The analysis of the CSS-D technical support community showed how to identify important social roles and individuals who fill those roles including answer people, discussion starters, and questioners. The analysis of ABC-D discussion-based email list showed how to identify good candidates to replace a community administrator based on network metrics such as betweenness and eigenvector centrality. And the analysis of Ravelry showed how to use a bimodal affiliation network to understand how forum-based groups are connected, identify important boundary spanners, and relate nondiscussion network metrics (e.g., blog activity, project activity) to group discussion activity.

9.9 RESEARCHER'S AGENDA

Research on threaded conversation communities has a long history as outlined in Section 9.2, yet there remain many interesting research questions to explore. As threaded conversations become embedded within more complex social spaces with multiple interaction technologies, it is increasingly important to understand how they all interact. For example, Hansen has found that technical and patient support groups benefit from combining a threaded conversation (i.e., email list) with a more permanent wiki repository [10]. The Ravelry example showed strategies that have not yet been widely used by the research community to understand how network position relates to use of other tools (i.e., blogs) or activities (i.e., projects). Network-based research is also needed to better understand the determinants of successful online communities. For example, we don't know what proportion of mixtures of answer people, discussion starters, and questioners lead to better outcomes or what overall network statistics (e.g., clustering coefficient) are correlated to success. From a design perspective, there are many fascinating opportunities to enhance the threaded conversation model [1] as evidenced by Google Wave and other prototype systems. One particularly promising approach is to use visualization to help people make sense of community interaction as was done in the Netscan project and related efforts [14–15].

References

[1] P. Resnick, D. Hansen, J. Riedl, L. Terveen, M. Ackerman, Beyond threaded conversation, in: CHI '05 Extended Abstracts on Human Factors in Computing Systems (Portland, OR, USA, April 02–07, 2005). CHI '05. ACM, New York, NY, 2005, pp. 2138–2139.

[2] H. Rheingold, The Virtual Community: Homesteading on the Electronic Frontier, Adison-Wesley Pub. Co, Reading, MA, 1993.

[3] M. Smith, P. Kollock (Eds.), Communities in Cyberspace, Routeledge, London, UK, 1999.

[4] J. Preece, Online Communities: Designing Usability and Supporting Sociability, John Wiley & Sons, Inc., New York, NY, 2000.

[5] A.J. Kim, Community Building on the Web : Secret Strategies for Successful Online Communities, first ed., Peachpit Press, 2000.

[6] D. Powazek, Design for Community, illustrated ed., Waite Group Press, 2001.

[7] B. Nonnecke, J. Preece, Lurker demographics: counting the silent, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (The Hague, The Netherlands, April 01–06, 2000). CHI '00. ACM, New York, NY, 2000, pp. 73–80.

[8] D.L. Hansen, Overhearing the crowd: an empirical examination of conversation reuse in a technical support community, in: Proceedings of the Fourth international Conference on Communities and Technologies (University Park, PA, USA, June 25–27, 2009). C&T '09. ACM, New York, NY, 2009, pp. 155–164.

[9] J. Preece, K. Ghozati, In search of empathy online: a review of 100 online communities, Proceedings of the 1998 Association for Information Systems Americas Conference, 1998, pp. 92–94.

[10] D. Hansen, Knowledge Sharing, Maintenance, and Use in Online Support Communities, Unpublished Dissertation, University of Michigan, http://hdl.handle.net/2027.42/57608

[11] H.T. Welser, E. Gleave, D. Fisher, M. Smith, Visualizing the signatures of social roles in online discussion groups, J. Social Struct. 8 (2) (2007).

[12] B. Butler, L. Sproull, S. Kiesler, R.E. Kraut, Community effort in online groups: who does the work and why, in: S. Weisband, L. Atwater (Eds.), Leadership at a Distance, Lawrence Erlbaum Associates Inc, Mahwah, NJ, 2005.

[13] E. Wenger, Communities of Practice: Learning, Meaning and Identity, Cambridge University Press, Cambridge, 1998.

[14] T. Turner, M. Smith, D. Fisher, H.T. Welser, Picturing usenet: mapping computer-mediated collective action, J. Comput. Mediat. Commun. (2005).

[15] F.B. Viégas, M. Smith, Newsgroup Crowds and AuthorLines: Visualizing the Activity of Individuals in Conversational Cyberspaces, Proceedings of Hawaii International Conference on Software and Systems (HICSS) 2004. [Best Paper: Persistent Conversation Minitrack]

CHAPTER

10

Twitter

Conversation, Entertainment, and Information, All in One Network!

Vladimir Barash[*] *and Scott Golder*[†]

[*]**Information Science, Cornell University, Ithaca, NY**

[†]**Department of Sociology, Cornell University, Ithaca, NY**

OUTLINE

10.1 Introduction **143**

10.2 The Nuts and Bolts of Twitter **144**
10.2.1 @replies and @mentions *146*
10.2.2 #hashtags *147*
10.2.3 Retweeting *148*

10.3 Networks in Twitter **149**
10.3.1 Friends, Followers, Information, and Attention *149*
10.3.2 Attention, Importance, and Eigenvector Centrality *150*
10.3.3 Information, Advantage, and Betweenness Centrality *150*
10.3.4 @replies and Symmetric Connections *151*
10.3.5 Retweets, #hashtags, and the Diffusion of Information *151*

10.4 Acquiring Data **152**

10.5 Discovery with Twitter **154**
10.5.1 The Ego Network *154*
10.5.2 A Trending Topic *158*

10.6 Practitioner's Summary **163**

10.7 Researcher's Agenda **164**

10.1 INTRODUCTION

It is almost too basic: a stream of messages from a list of people, businesses, and celebrities you find interesting, and an empty box where *you* can write very short messages to those who find *you* interesting. Twitter's brass-tacks construction has been one of the keys to its success, however. Through customization and extension by a large base of users and third-party developers, Twitter has grown to be one of the most popular, talked-about, and versatile social media platforms of the past several years. It has quickly garnered millions of users and has been employed by political candidates to reach out to supporters during campaigns and while in office [1]; by actors, musicians, and comedians to give their fans a peek into their personal lives and build their fan base; and by businesses to share product information with and provide more personalized customer service for consumers. Twitter has played a role in coordinating information sharing among victims of natural disasters like hurricanes [2] and wildfires, and some even suggest that Twitter helped raise global consciousness about terrorist attacks in India in 2008 and political turmoil in Iran in summer 2009.

Not bad, considering Twitter's humble beginnings. Twitter began life as a side project inside of podcasting startup Odeo in San Francisco in 2006.

Twitter first built real "buzz" in the tech-savvy community when it was publicly demonstrated at the South-by-Southwest (SXSW) film and music festival in

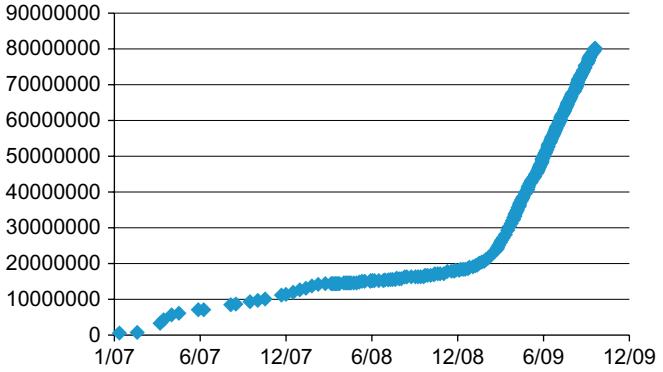


FIGURE 10.1 Estimated growth of Twitter.[1] The growth curve shows sharp increases in March 2007 and April 2009, because of the SXSW festival and The Oprah Winfrey Show.

Austin, Texas, in 2007 [3]. It won the Web Award and garnered a raft of new users, who used it as a "back channel" during the festival to share information about hot new products and gatherings.

Flash forward to late 2008/early 2009. Basketball superstar Shaquille O'Neal (@THE_REAL_SHAQ) became a Twitter user in November 2008. Early on, he addressed his Twitter fans this way: "Let me give you all a hint on how to relate to me, Shaquille O'Neal I have a sense of humor I am very quotatious ..."

Soon after, actor/producer Ashton Kutcher (@aplusk) and talk show host Oprah Winfrey (@oprah) joined Twitter, and Kutcher appeared on Winfrey's talk show to publicize his friendly competition with news network CNN to see who could reach one million followers first, Kutcher, or CNN's "breaking news" Twitter stream (@cnnbrk). Kutcher won. But the ensuing media attention brought Twitter into the mainstream as curious new users learned about Twitter from television.

The growth chart in Figure 10.1 shows large increases in user accounts in March 2007 (corresponding to the SXSW festival) and April 2009 (showing the mass-media effects of Ashton, Oprah, and friends). Look at the right-hand side of the chart. The growth since the fall of 2009 is dramatic—and now it's almost as though everyone's twittering.

The idea of brief status message posting is growing generally; as of fall 2009, nearly 20 percent of Internet users reported using some status-updating service [6], and Twitter is far and away the most popular.

As Twitter has gained popularity, it has itself grown from being a standalone platform to being the center of an entire communication ecosystem. Twitter itself sports a basic web-based interface (Fig. 10.2). However, other desktop-based and web-based tools are prevalent. One popular client, TweetDeck (Fig. 10.3), has multiple columns, so users can follow multiple feeds at once, such as all their friends or particular conversations, or topics of interest and search queries. Another, Twhirl (Fig. 10.4), is in single-column format and mimics the layout of an instant messaging client.

As more and more businesses are starting to use Twitter, tools to support them such as CoTweet (www.cotweet.com) and HootSuite (www.hootsuite.com) have been developed. In addition to allowing collaborative management and use of a single, shared Twitter account (e.g., one belonging to a corporate brand), services like these tabulate statistics, such as numbers of messages and replies.

This is only a sample of the many third-party Twitter programs out there, and new ones are being developed every day.

In this chapter, you will learn how you can tap into and map parts of the global Twitter conversation. Analyzing Twitter users and conversations is more than tabulating counts and trends—it's about connections and interactions between people. You'll see how social network analysis techniques can help you examine your Twitter network and see, for example, how all of your connections relate to each other. Finally, you will see how to explore Twitter as a mass media phenomenon and learn how to track the flow of the global conversation.

10.2 THE NUTS AND BOLTS OF TWITTER

Twitter can be thought of as a *conversational microblog*. Like bloggers, Twitter users post messages that show up in the streams of all the people who are subscribed to them. Twitter's infrastructure in this way mirrors the functionality of really simple syndication (RSS) feeds.

Twitter calls the people who subscribe to and receive your messages your *followers* or the people who *are following you*. Flipping that around, the people whom *you* are following are called your *friends*. These terms correspond with *subscribers* and *subscriptions*, respectively, in the context of blogs. Table 10.1 lists several features that may be familiar to users of blogs, and the Twitter equivalents.

One significant difference between blogs and Twitter is the size of the message—the characteristic that puts the *micro* in *microblog*. Twitter messages are limited to 140 characters. The reason for this is that Twitter is designed to be used via SMS (text messages) on mobile phones (SMS messages are limited to 160 characters).

[1]Estimate performed by the authors, through statistical sampling.

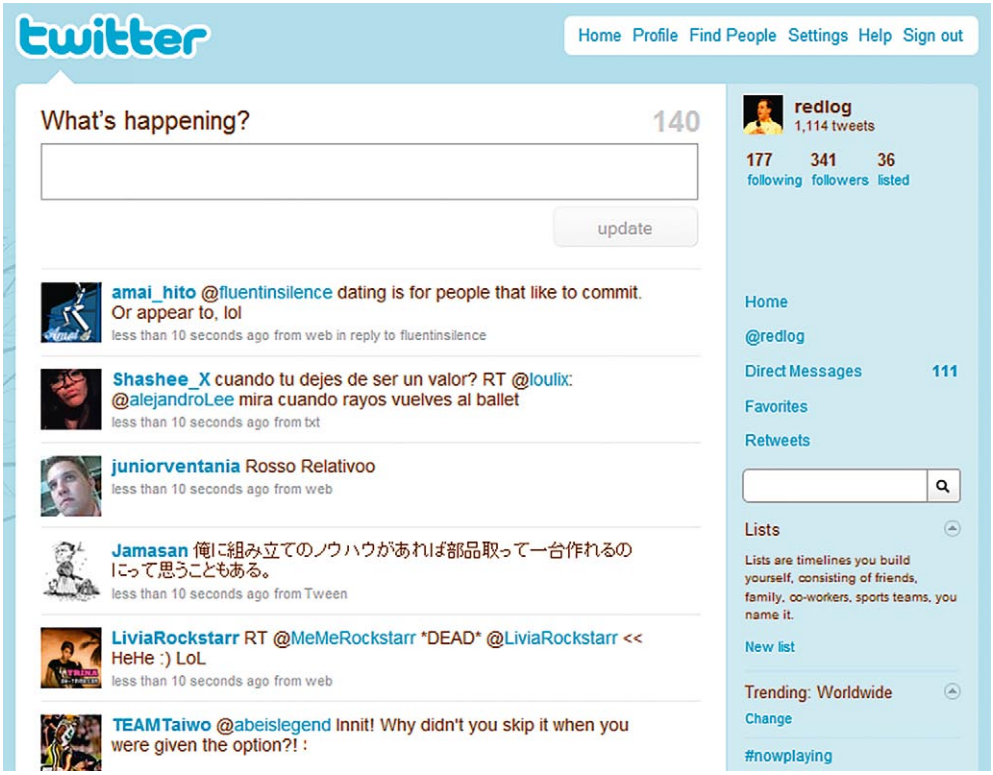


FIGURE 10.2 Twitter's web-based interface. Its most prominent contents are a stream of tweets (left) and a dashboard with user-specific information (right).

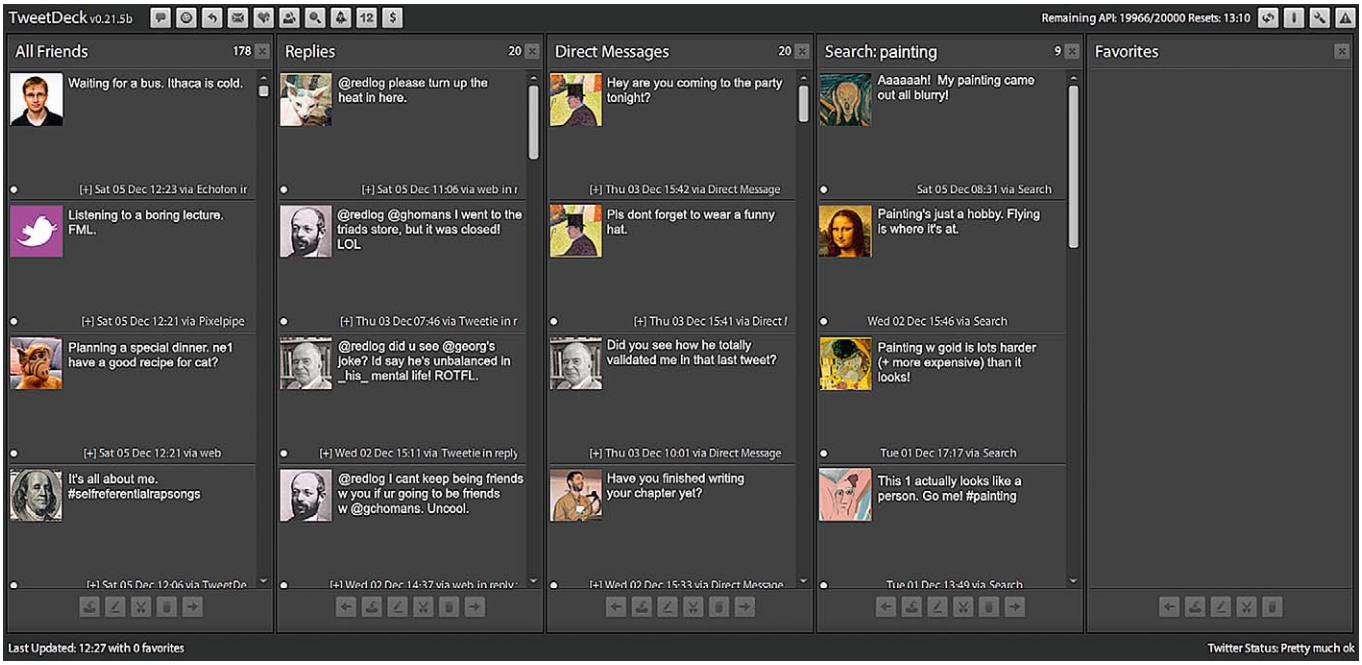


FIGURE 10.3 TweetDeck is a popular desktop-based Twitter client.

These short messages are called —and we'll call them from this point forward—*tweets*.

A lot of structured information can be packed into only 140 characters! We'll unpack some of it now.

Twitter did not start off with a great deal of structure. However, like other Internet social spaces before it, Twitter's user population has innovated beyond the core features. Twitter has, more than other services,

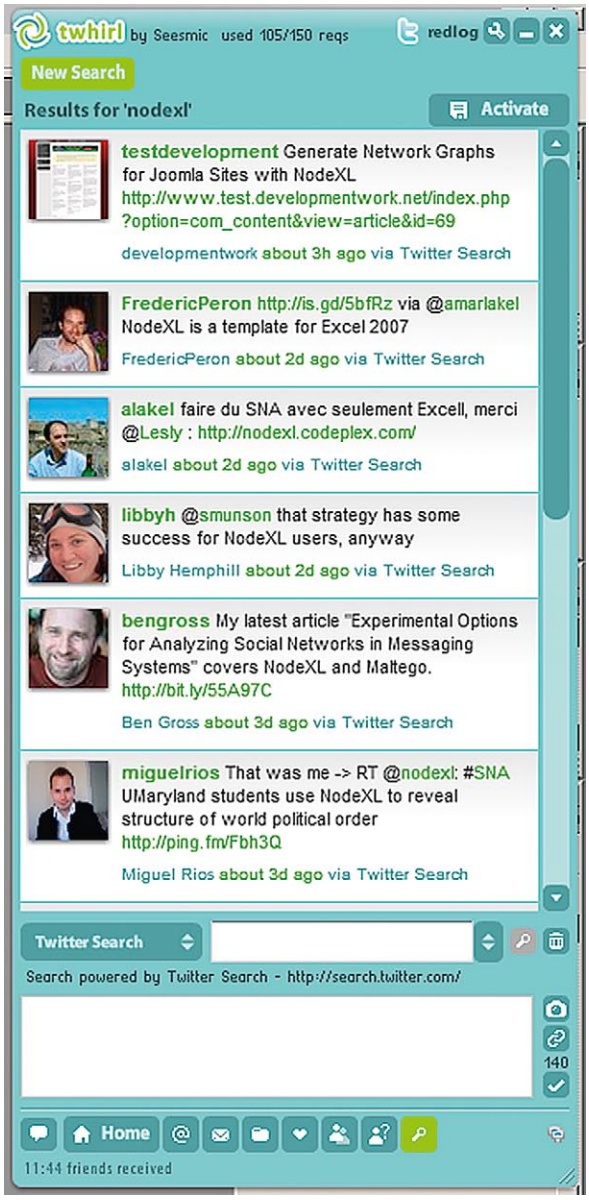


FIGURE 10.4 Twhirl is a popular desktop-based Twitter client.

TABLE 10.1 A Comparison between Blog and Twitter Functionality and Terminology

Blogs	Twitter
Subscribers	Followers
Subscriptions	Friends
Posts	Tweets

been willing to learn from and work with its users in adding useful capabilities by adopting their conventions and building them back into the technology.

10.2.1 @replies and @mentions

When you're writing in a space in which all messages are public, how do you have a conversation? In conversations, people don't just *say* things, they say things *specifically to each other*. How do you know which messages are intended especially for you? This is a hard enough problem in a two-person conversation—every conversation requires at least two!—but it gets challenging to follow along in a group discussion.

Twitter users solved that with something called the @reply (pronounced *at-reply*). Simply include a user's username, preceded by the @ sign to indicate that the message is specifically intended for them. The tweet is still public, but now the intended flow of conversation is clearer.[2] Note that @replies show up in recipients' feeds, even if the recipient is not following the sender.

Here's an analogy: If Scott were crossing the street unsafely, Vlad might say, "Hey Scott, watch out for that bus!" He wouldn't be announcing *in general* to watch out for the bus, but rather alerting Scott specifically, lest he become a road pancake. In a tweet, it would look like that shown in Figure 10.5.

Let's illustrate why @replies make it easier to follow conversations in Table 10.2.

In the case on the right of Table 10.2, it's clearer that *C* is using NodeXL, not making scrambled eggs.[3] Honeycutt and Herring, Internet scholars who study the linguistic structure of online conversations, have referred to the use of the @reply as a "marker of addressivity" and demonstrated that its use is one of the ways conversations in Twitter are able to be coherent, despite a "noisy" environment, in which many conversations are taking place simultaneously and there's a continual need to capture the attention of your intended conversation partners.

In some ways, @replies are better indicators of social ties than follower/friend relationships. One way of measuring how strong a social tie is, is the frequency of interaction between the two people involved—this is an insight that appears repeatedly in the academic social networks literature and has been capitalized upon by scholars of computer-mediated communication across domains, from email to online forums of all kinds.

In Twitter, as in other online social network services, it's entirely possible to have a long list of declared "friends," "contacts," "buddies," and so on—once someone's on

[2]It is possible in Twitter to send private messages to other users (called "direct messages"), but we're not talking about them here.
[3]Though there's no good reason not to do both.



FIGURE 10.5 A tweet with an @reply message. This message is sent from @vlad43210 to @redlog.

TABLE 10.2 A Conversation without (l) and with (r) @replies

Without @replies	With @replies
A. I'm using NodeXL to visualize my Twitter net.	A. I'm using NodeXL to visualize my Twitter net.
B. I'm making scrambled eggs for breakfast.	B. I'm making scrambled eggs for breakfast.
C. So am I!	C. @A So am I!

your contact list, it takes no effort to keep them there; mere acquaintances and best buddies look no different from one another. In contrast, to engage in a conversation with someone takes time, effort, and thought. It's no wonder that studies [9] have found that, even though people have large numbers of "friends" online, the proportion of those friends who they regularly exchange messages with is rather small in comparison.

Strictly speaking, not all @replies are replies, either in the technical or conversational sense; some are simply mentions of another person by name (let's call them @mentions). Now return to the preceding example. If the message wasn't received in time, Vlad might subsequently tweet, "I just saw @redlog get hit by a bus. Bummer." This tweet isn't directed *to* @redlog, but rather it just mentions him.

In addition to the natural-language distinction you might make between messages that are *for* someone and those that are *about* someone, Twitter makes a technical distinction between @replies and @mentions. Only when the tweet starts with the @ sign (i.e., the other user's name is at the beginning of the tweet) does it register as an @reply. The Twitter infrastructure only keeps track of replies, for conversation threading purposes, and not of @mentions—there's a field in the API for the former, but not the latter. A tweet that is an @reply will contain an additional line in the Extensible Mark-up Language XML response named in_reply_to_user_id:

```
<in_reply_to_user_id>28361132</in_reply_to_user_
  id>
```

It contains the user ID (in this case, Vlad's) of the person who is being replied to. Incidentally, if the reply was to a specific tweet, there would be another item named "in_reply_to_status_id," which contains the unique ID of that particular tweet.

Recall that @replies are intended to be *conversational*. Not everyone cares about your conversation. So Twitter made the (in our opinion) wise choice to show @replies only to third parties who are following both the other users. So returning to the example in Table 10.2 (r), C's response "@A so am I!" would be seen by B only if B is following *both* A and C.

Whether the distinction between @replies and @mentions matters for the purposes of your own analyses is up to you. As a matter of clarification, we'll think of @replies as a subset of @mentions—all @replies are @mentions, but not all @mentions are @replies. Got it? That's almost as confusing as "followers" and "friends."

10.2.2 #hashtags

If what you're interested in is not some*one*, but rather some*thing*; then keeping track of everyone who's talking about that thing can be a challenge, especially if most of those people don't know one another. You could use Twitter's search feature (http://search.twitter.com), but what would you search for? If it's something simple, like a one- or two-word product name, that might work. But if it's more complicated, a better solution is needed.

Fortunately, the Twitter community has come up with an answer—hashtags. Tags are short, descriptive keywords that are popular in many kinds of online communities, from social bookmarking and photo sharing services, to blogs and book reviews. In Twitter, these tags are identified by the # sign. The # sign is called by a variety of names—"pound," "number," and "sharp," for example—but in Twitter it's called "hash." And so the hashtag is a descriptive keyword with a # sign in front.

Hashtags originated not in Twitter, but in a similar service called Jaiku. Blogger and prominent tech community member Chris Messina was early in advocating the use of hashtags, calling them "channel tags" after the Jaiku "channel" feature.

Though it may seem counterintuitive, a group of loosely connected people choosing keywords for something—events, conferences, news stories, and so on—will quickly converge on a consensus choice.

Some hashtags are for events like the #chi2010 conference, or for major public events like #iranelection in summer 2009 or the attack in #mumbai.

Some hashtags have become games. In a hashtag game, users pick a funny topic and try to come up with the best examples. Two of our favorites were #robotpickuplines,

begun by Grant Imahara (@grantimahara) of the Discovery Channel's *MythBusters* show, and #internetbeatles., in which the goal was to make puns by combining Beatles song titles with Internet-related terms. Here are some of ours:

- #robotpickuplines "If I could rearrange the qwerty keyboard, I'd put u and i—oh, wait, nevermind"
- Yelp! #internetbeatles
- FFFF00 Submarine #internetbeatles

You'll notice that each of these examples contains the hashtag associated with the game in question. This way, all Twitter users who are following the game (via the hashtag) can find these tweets, even if they don't already follow the person who wrote them.

One particular hashtag has driven a major cultural event specific to Twitter, called "follow Friday." Each Friday, Twitterers tweet about people who they enjoy following and encourage others to follow them too. It's like a recommendation system, and is a nice way of recognizing people who you find interesting and helping them attract new followers. The "follow Friday" tradition—if it is old enough to be called that—combines a hashtag (#followfriday or #ff) with @mentions.

10.2.3 Retweeting

Sharing ideas with others is one of the most important parts of Twitter. You may want to not only share your own ideas, but pass along the things you've read, to the people who follow you. In this example, we'll show you a series of "retweets" by Scott and Vlad. Let's start with the interesting tweet shown in Figure 10.6.

This tweet, written by @ASAnews,[4] includes a link to a news article. Scott follows ASAnews but knows that many of his friends do not. Scott would like to share this information with others, so he "retweets" it. Retweeting is taking someone else's tweet and rebroadcasting it with attribution, so that your own followers can see it. In order to retweet this tweet, Scott would write the message shown in Figure 10.7.

Scott's tweet starts off with "RT @ASAnews." RT stands for "retweet" and is followed by an @mention of the ASAnews account and the text of the retweeted message. Now, Scott's own followers can see this, even if they are not following @ASAnews.

Vlad, for example, follows Scott, but does not follow @ASAnews. If Scott hadn't retweeted it, Vlad would never have seen this tweet. But he's glad he did, and he would like to pass it along to *his* followers, as shown in Figure 10.8.



FIGURE 10.6 An interesting tweet posted by @ASAnews. This tweet starts a retweet chain (see the next figures).



FIGURE 10.7 @redlog retweets the tweet posted by @ASAnews. The @ASANews message about David Brooks' article now spreads to @redlog's followers, even if they don't follow @ASANews directly.



FIGURE 10.8 @vlad43210 retweets @redlog's retweet of the tweet posted by @ASAnews. The message now spreads to @vlad43210's followers, even if they do not follow either @ASAnews or @redlog. In this way, messages can spread via retweets and reach a very large audience.

[4]This is the Twitter account for the American Sociological Association.

Note what Vlad did here; he took Scott's entire tweet and prepended "RT @redlog" to the beginning. Looking at Vlad's tweet, a follower of his can deduce that he learned about this idea from Scott, who in turn learned about it from @ASAnews.

Tracking retweets is one way of following the spread of an idea or piece of information throughout the Twitter social network. Note that, when Scott added *"RT @ASAnews"* to his retweet, the length of the tweet would have exceeded the 140-character limit. He had to abbreviate somehow and chose to shorten "New York Times" to "NYT." Other abbreviations are often used in retweeting, such as omitting articles and adjectives. Therefore, following a retweet through the network is not exactly straightforward because the text morphs as it travels, much in the way things change in the children's game "telephone" [4].

Retweeting is a surprisingly complex social activity. Research [4] shows that the "best practices" are still being worked out, especially when the "right thing to do" is ambiguous. This is often true when it comes to attribution. If a message traveled "far" enough, it would soon be nothing but a string of "RT @username" references, leaving no room for the message itself! The person doing the retweeting has to decide who he or she wants to give credit to; depending on the content, the person may choose to give credit to the person he or she learned the information from (e.g., if passing along a news article) or, alternatively, to the person who is the original writer (e.g., a clever joke or turn of phrase). As of yet, no convention has emerged in Twitter with regard to proper attribution of retweets.

Retweeting lets you show off the cool people you know. Social media researchers boyd, Golder, and Lotan [4] have suggested that people retweet both to "amplify or spread tweets to new audiences" and to "validate others' thoughts." This is important for social network researchers as well as marketers; if @A retweets one of @B's tweets, then it shows that @A thinks pretty highly of @B, and that @B is influential (at least, to @A).

As it has before with @replies and #hashtags, Twitter has recently recognized the value of retweeting and has built retweeting into its architecture. On the one hand, this is valuable because it allows retweets to be tracked and accessed via the API. However, if there is any constant about Twitter users, it is that they will keep molding their use of the technology to suit their ever-changing needs.

10.3 NETWORKS IN TWITTER

Structured content, in the form of friends and followers, @replies and @mentions, #hashtags and retweets makes the Twitter user population networked in multiple ways. You can think of each of these features as a kind of connection that can exist between two Twitter users. For instance, an @reply connects its author to its target (the username following the @ sign).

The sets of @replies, @mentions, and the like between all Twitter users are different sets of connections overlaying the same set of people. Together, a particular set of connections and all the individuals implicated in it form a network. There are at least as many networks in Twitter as there are features listed here, and each of these networks describes something different.

There are two major ways we will be talking about networks in the following sections. First, we will discuss the individuals (users or actors) involved in these networks and the relationships (ties) between them. Second, we will discuss the visual representation of the network in the NodeXL Graph canvas, where individuals are represented as nodes or vertices of the graph (most often in the shape of disks of varying sizes and colors) and ties are represented as edges of the graph (most often in the shape of lines between the vertices). We will use the terms "user" or "actor," and "tie" when talking about networks as sets of relationships between people, and the terms "node" or "vertex" and "edge" when talking about networks as representations of relationships between people via disks and lines.

10.3.1 Friends, Followers, Information, and Attention

Recall earlier we described people who are receiving your tweets are your *followers*. People whose tweets you receive are your *friends*. Figure 10.9 illustrates these relationships.

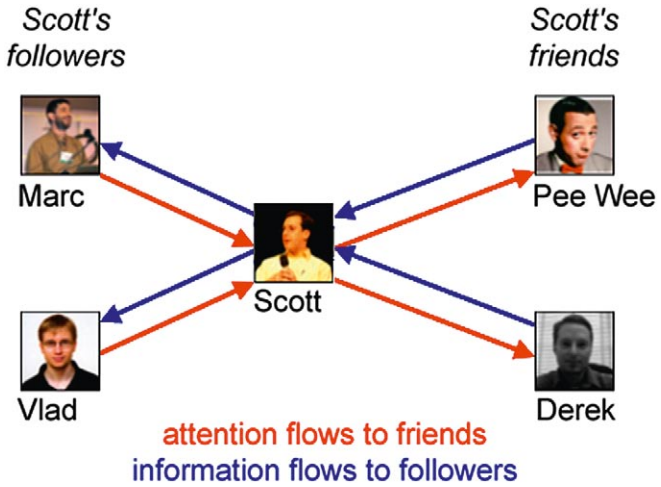


FIGURE 10.9 The friends and followers network is an *information/attention network*. This figure shows Scott with two friends (Pee-Wee and Derek) and two followers (Marc and Vlad).

In this figure, you can see that Scott (@redlog) has two followers: Marc (@marc_smith) and Vlad (@vlad43210). Scott has two friends, Derek (@shakmatt) and 1980s TV entertainer Pee-Wee Herman (@peeweeherman). This means that Scott pays attention to Derek and Pee-Wee and receives their tweets, and that Vlad and Marc are paying attention to Scott and receiving his tweets.

You can think of the friends/followers network as being two kinds of networks in one: an *information* network and an *attention* network. The follower tie, represented by the red edges in the preceding figure, shows how attention is flowing (e.g., from Scott to Pee-Wee). To think of friends/followers instead as an information network, simply change the direction of the edges. The blue arrow from Derek to Scott shows that the information Derek tweets flows to Scott, who is one of Derek's followers.

It is useful to think about the friends/followers network in these two different ways, because each is useful for thinking about different kinds of problems.

10.3.2 Attention, Importance, and Eigenvector Centrality

The attention network is similar conceptually to the web as a whole. Twitter users following one another are like web pages having hyperlinks to one another. Twitter users want lots of followers, the same way that owners of web pages want lots of links to them from other pages. Therefore, some of the same analytical techniques for the web translate directly to the follower network. A particularly relevant analytical technique from the web domain is *eigenvector centrality*, which is a measure of how important a particular individual is within a network. "Important" users who are paid attention to by lots of others, who are *themselves* paid attention to by lots of others, and so on. The measure of *eigenvector centrality* is a key part of Google's PageRank algorithm for measuring the "importance" of web pages.

Eigenvector centrality can have practical importance for identifying spammers. Being followed by a spammer is a kind of unwanted attention. Spammers want to have as large a number of followers as possible in order to spread their spammy messages. However, even if a spammer followed a large number of people and amassed many followers, you would want to be able to recognize that the spammer is not actually all that influential. To illustrate, consider a spammer who is following 10,000 people. In an undirected network—where it was not possible to distinguish the flow of information from the flow of attention—this person would look influential indeed. However, attention is flowing *from* the spammer *to* those he is following, not vice versa. And even if the spammer is able to gain a large number of followers—some people like following back all those who follow them—you would want to know whether that link is important, and so you'd ask whether the people who are following the spammer are themselves "important." For this reason, eigenvector centrality is a useful measure and, if the spammer were being followed by a large number of people who were relatively unimportant or had few followers, you could conclude that the spammer is not all that influential at all.

10.3.3 Information, Advantage, and Betweenness Centrality

In some kinds of networks, we're less concerned with *importance* and more concerned with how individuals' *position* in the network affect their access to information. Let's look at the flow of information; this is illustrated by the blue edges in Figure 10.9. Figure 10.10 shows two clusters (A–D on the left and F–J on the right) and one actor in the middle (E). The only connection between the two clusters is actor E; for any information to pass between the two clusters, it must pass through E. C has four neighbors: A, B, D, and E. However, A, B, and D provide the same information, as they all know the same people; only E knows different people and so can provide nonredundant information. In general, *betweenness centrality* measures how many of the shortest paths between two others some actor lies on. If an actor has high betweenness, this means that a lot of nonredundant information passes through him.

Actor E is a "bridge" between different parts of the network. Consider a large company—an employee with contacts in multiple divisions of the company is more likely to have timely access to diverse kinds of information. Knowledge is power, and being in the right

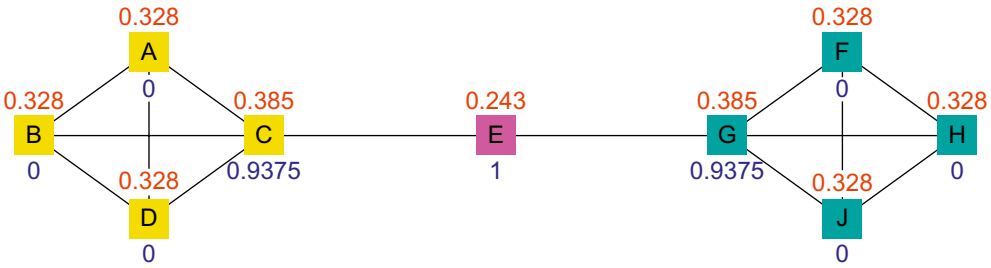


FIGURE 10.10 The middle actor is a bridge between two communities. The top, red numbers above the actors are their eigenvector centrality, and the bottom, blue numbers are their betweenness centrality.

network position can be socially as well as professionally advantageous (readers interested in the relative advantages of different network positions may find the work of [5] particularly relevant).

Figure 10.10 distinguishes between eigenvector centrality (shown above the vertices) and betweenness centrality (below). Actor E has the highest betweenness, for the reasons we described, followed by C and G, who are connected to E. However, E has the lowest eigenvector centrality, and C and G have the highest.

In the previous section, we described how eigenvector centrality can help identify spammers. Now, we'll show you what you can do with betweenness centrality. Recall that we described high betweenness as having access to nonredundant information; in contrast, people with low betweenness provide largely redundant information. If you find that many of the people you follow are tweeting the same things, sharing the same links, and so on, you might want to stop following some of them. One way of choosing is to rank them by betweenness and remove the people whose values are the lowest.

10.3.4 @replies and Symmetric Connections

Unlike in the friend/follower network, in the @reply network, the flows of information and attention move in the same direction, because messages convey information for the intended recipient and are an indication that the sender is paying attention to the receiver. Vlad's exhortation to Scott (Figure 10.5) to look out for the bus is an example: Vlad is paying attention to Scott, and also providing him with (important) information. This is an *asymmetric connection* in contrast to the symmetric exchange of attention and information in a friend/follower relationship.

Asymmetric attention is the key to another important concept, *celebrity*. Being famous means that a lot of people pay attention to you—after all, by definition it is famous people who appear on the cover of magazines, which are purchased because lots of people want to know what's happening with their favorite celebrities. But the celebrity doesn't, for the most part, pay any attention to the fans (at least, not individually). Asymmetric attention ties in Twitter allow people like Oprah Winfrey to have millions of fans, just like she does on television, without needing to pay attention to all or even any of them. In short, asymmetric attention allows Twitter to be an extension of other forms of *mass media* as well as a conversational medium.

Likewise, the existence of symmetry is informative as well. If two people follow each other, then they are each paying attention to one another, and each is receiving information from the other. This is powerful stuff, even more so if each has sent @replies to the other, because this is better evidence of a strong social tie between them. Note that the exchange of @replies makes a symmetric connection out of two asymmetric ties. Symmetry isn't a perfect proxy for the strength of a relationship—some people follow anyone who follows them as a kind of automatic reciprocity, though others feel this is untenable—but it's the most reasonable approximation we've got and, in the aggregate, is a pretty reliable indicator of relationships.

10.3.5 Retweets, #hashtags, and the Diffusion of Information

Both the friend/follower and the @reply networks are information networks in the sense that information flows through them. As we mentioned earlier, information tends to spread quickly through these networks, and individual messages tend to be lost in the sheer volume of content that is generated and circulated on Twitter every second. The #hashtag and retweet features described in Sections 10.2.2 and 10.2.3 help Twitter users organize this wave of information and follow specific, important messages. The networks associated with these features describe the flow of specific bits of information through Twitter.

It is important to note that #hashtags do not actually form a network of their own in Twitter.[5] Instead, they are often combined with friends/followers, @ replies, or retweets relationships. For example, let's say you are interested in how the #hashtag #BlackFriday (corresponding to the shopping event that follows the Thanksgiving holiday in the United States in November) spreads through the friend/follower network; you would look at all of the Twitter users who put #BlackFriday into at least one of their tweets and then find (and draw) all of the friend/follow links between these users. Alternatively, if you wanted to look how the same #hashtag spread through @replies or retweets, you would only draw @reply or retweet links between the same set of users.

The spread of #hashtags through any of these networks is of key interest for both academics and industry professionals. Twitter is an excellent platform for reaching out to potential customers, advertising your product, and so on. The #hashtags are a great way of giving your product a unique identifier so its spread through Twitter can be easily tracked. The overall volume of spread is a key metric of success, but the pattern of spread is important too. Different patterns of spread

[5]Technically, hashtags form a "bimodal" network in Twitter, where we can create a tie between everyone who uses the same hashtag. The study of bimodal networks is beyond the scope of this chapter.

suggest different marketing strategies: a "star" pattern, where one mention of a #hashtag by a particular user triggers its mentions by many of that user's followers, suggests that the key in product adoption lies at the message—and individual levels. For instance, the message may be very catchy, the adopters easily influenced, or the early adopters very influential. In contrast, a grid pattern, where users mention a #hashtag only after seeing it multiple times in the tweets of different friends, suggests that social influence is a big factor in adoption and that it is important for the message to spread in redundant ways.

Retweets are particularly relevant to the spread of #hashtags and other messages through Twitter. When a user retweets a message with a #hashtag rather than simply mentioning it in his or her tweets or @replying to someone with the same #hashtag, this action indicates that the user thinks the message (including the #hashtag) is of value to a significant fraction of his or her followers and so is interested in making the message popular. The retweet action by itself identifies the user as a potential promoter. The position of the user in the retweet network of the #hashtag identifies his or her importance to the promotion effort. You can measure this position with eigenvector centrality as noted earlier. If Vlad retweets someone else's message with #BlackFriday but nobody retweets Vlad's message, he is relatively unimportant to the promotion effort. If many people retweet Vlad's message, he becomes that much more important. Finally, if many people retweet Vlad's message, and many people go on to retweet Vlad's retweeters, Vlad is extremely important to the promotion effort and should be rewarded (perhaps with free merchandise!) by a Black Friday retailer for bringing the event to the attention of many potential customers.

10.4 ACQUIRING DATA

Collecting Twitter data with NodeXL is a simple and straightforward process. Click on the NodeXL ribbon on the main Excel menu. On the left side you'll see a menu group labeled Data and a drop-down list called Import. Clicking on the list reveals several options. The two that we'll be concerned with are "From Twitter User's Network" and "From Twitter Search Network" (see Fig. 10.3). The first option collects data from the personal network of a particular Twitter user—(i.e., the person's friends and followers). The second option, when given a search term, will collect all the individuals who have used that term recently in their tweets, as well as the links between those individuals (Fig. 10.11). We discuss both options in greater detail later in the chapter.

Collecting data from Twitter is generally a slow process. This is due to several factors. First, data transfer rates over the Internet can be slow, and Twitter has its own bandwidth limitations—after all, millions of people are sending data to and from Twitter all the time. Second, networks can grow unexpectedly large rather quickly. Even someone who is not an active Twitterer might have a large second-degree network, for example, if that person's few neighbors (friends and followers) themselves have a large number of neighbors.

Therefore, when collecting data from Twitter, one needs to be aware of "rate limits." Twitter limits the amount of data collected by any one user to 150 "requests" per hour. In practical terms, this means you can retrieve information on up to 150 people per hour. This may seem like a large number, but when analyzing search and large user networks, it is not uncommon to exceed this limit. If you have data-intensive needs,

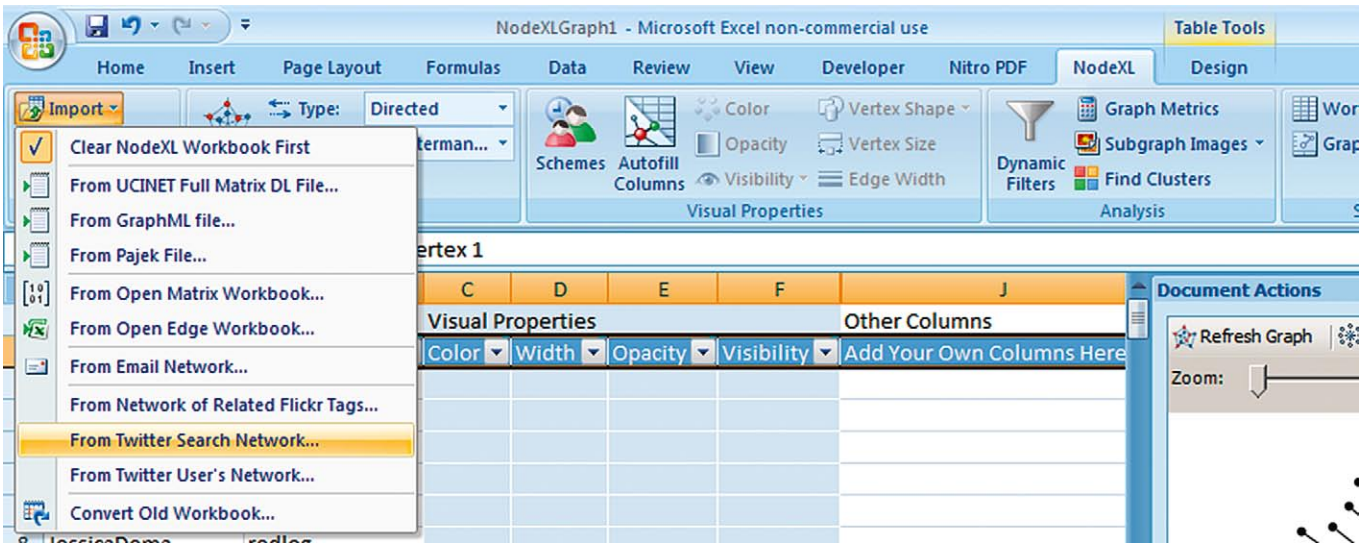


FIGURE 10.11 The NodeXL Import menu. Note there are two options that import Twitter data, from a user's network and from a search query.

see the "The REST API and Whitelisting your account" box for a workaround.

Large networks also mean lots of time. You can expect data collection to take roughly 10 to 30 seconds per user; if you are going to be collecting data about 1000 users who have mentioned a particular search term, the process may take several hours!

NodeXL provides several options for speeding up data collections, such as artificially limiting the number of users in the data set. In general, however, NodeXL is a data analysis and visualization tool and is less optimized for data collection. Please be aware of these limitations when collecting data with NodeXL.

ADVANCED TOPIC
The REST API and Whitelisting Your Account

Twitter uses a representational state transfer (REST) application programming interface (API) to provide data in XML or JavaScript Object Notation (JSON) to third-party clients. This API is what allows third-party clients like TweetDeck and Twhirl—and also NodeXL—to retrieve data from Twitter.

Though you can use NodeXL for even complex Twitter-related tasks without having to worry about the underlying API, if you are doing something especially data-intensive, you may want or need to consider whitelisting your account. (If that was too large a helping of alphabet soup, you may skip the rest of this box.)

Every time you want to retrieve data from Twitter, NodeXL constructs an API request in the form of a URL. If you wanted all the followers for one of the authors, you'd issue the following request:

```
http://www.twitter.com/followers/ids.xml?screen_
  name=redlog
```

Twitter's API recognizes this as a request for the IDs of all of the people following the user with the screen name @redlog. You can learn more about the API at http://apiwiki.twitter.com/.

Now, suppose you wanted to examine @redlog's *second-degree network*. That is all of @redlog's friends and followers, plus all *their* friends and followers, and requires API requests for each. For example, if @redlog has 400 friends and followers combined, then that'll take 802 requests—one request to get a list of @redlog's friends, and another for his followers, plus two requests per person to get *their* friends and followers. With a regular account, you will be limited to 150 queries per hour, meaning the process will take you over 6 hours.

Whitelisting your account is a way of being allowed to make far more than 150 queries per hour—up to 20,000 per hour.[6] Your account is added to a "white list" of accounts that are allowed the larger allotment.

To have your account whitelisted, visit http://twitter.com/help/request_whitelisting. Log in with your Twitter account credentials (username and password) and fill out the form.

If your account is whitelisted, then enter your username and password into the NodeXL Twitter import interface.

[6]These numbers, 150 and 20,000, are current as of the time of writing.

ADVANCED TOPIC
Using VBA to Manipulate Networks

Did you know? NodeXL spreadsheets can be manipulated just like normal spreadsheets. This means you can write scripts to manipulate your NodeXL networks using Visual Basic macros.

Say, for example, you're looking at a Twitter network and you want to see a link between two people only if they have a mutual following relationship—that is, each one follows the other—and not an asymmetric relationship. The way to accomplish this is, for every directed edge, make sure a directed edge exists in the opposite direction:

```
Sub Mutuality()

 ' change these if you have added or removed rows
  or columns in the Edges worksheet
 COL_V1 = 1
 COL_V2 = 2
 COL_VISIBILITY = 6
 ROW_START = 3
```

(*Continued*)

ADVANCED TOPIC *(Continued)*

```
Dim wksht As Worksheet
Set wksht = Sheets("Edges")

' Be sure to go to Tools > References and select
 Microsoft Scripting Runtime
Dim arr As Dictionary
Set arr = New Dictionary

' step 1: put all the edges into an array
current_row = ROW_START
While wksht.Cells(current_row, COL_V1).Text> ""
k = wksht.Cells(current_row, COL_V1) & "_" &
 wksht.Cells(current_row, COL_V2)
arr.Add k, 1
current_row = current_row + 1
Wend
' step 2: show edges if and only if their
 counterparts appear in the edge list.
current_row = ROW_START
While wksht.Cells(current_row, COL_V1).Text < ""
 k = wksht.Cells(current_row, COL_V2) & "_" &
  wksht.Cells(current_row, COL_V1)
 If arr.Exists(k) Then
  wksht.Cells(current_row, COL_VISIBILITY) = 1
  Else
wksht.Cells(current_row, COL_VISIBILITY) = 0

 End If

 current_row = current_row + 1

 Wend

End Sub
```

This macro has two steps. First, it makes a list of all of the directed edges. Then, going through the edges again, it makes sure the opposite-direction edge exists. If so, the edge is marked visible, otherwise invisible.

Once you have executed the macro, click "Refresh Graph" to reload the workbook and update the graph. If the vertices' visibility is set to "Show if in an Edge," then some vertices might no longer show up. You can get these vertices to appear again by clearing the Visibility column or by changing the setting to 1, which stands for "Show."

10.5 DISCOVERY WITH TWITTER

In the next several pages, we'll show you some examples of how to use NodeXL to discover interesting patterns and structures in Twitter. We will look at personal networks (egonets) and then explore how cohesion allows information to spread through a social network.

10.5.1 The Ego Network

One common use of Twitter is keeping in touch with your personal social network—family, colleagues, acquaintances, and so on. Some of your friends and followers on Twitter will most certainly belong to people you are close to, whereas others may be distant acquaintances or even complete strangers. We refer to the set of a Twitter user's friends and followers as his or her "egocentric network." When talking about the egocentric network of a Twitter user, said user is often referred to as "ego" and his or her friends and followers as "alters" (but still use the term "friend" or "follower" when we are talking about one of those relationships specifically). We will use the terms "ego" and "alter" in this section.

Some ties in the egocentric network are reciprocal—ego follows alter and alter follows ego, so both can remain aware of each other's status updates. Research on reciprocal ties in social science suggests that such ties more often exist between close friends and family than between distant acquaintances. Following this research, in this section we assume that a network with many mutual ties indicates a set of strong social relationships, where both people are interested in each other's updates, whereas a network with many one-way ties indicates weak social bonds, where the ego may not even know the follower, as in the case of Twitter celebrities.

Another indicator of strong ties in a social network is "closed triads," which are ties between two alters of an ego. As with reciprocal ties, research on closed triads shows that tight communities of strong social ties feature more closed triads than loose communities of weak social ties. Following this research, we assume that an ego with many closed triads among his or her alters is more likely to be embedded in a tight community, where everyone knows each other. In contrast, an ego with few closed triads between his or her alters is more likely to be solitary or not an individual at all. For example, Twitter accounts for business marketers may have many followers interested in their products, but those followers often do not know each other.

Analysis of egocentric networks can often lead to actionable results if you want to use Twitter as a development or advertising platform. Tight communities, for instance, might be much more likely to adopt a

product together or not at all, whereas sparse egocentric networks with many one-way connections might be heavily influenced by the behavior of the "hub" of the network—the Twitter user with many followers but few friends. In particular, analysis of the egocentric networks of celebrities on Twitter gives us a glimpse into how the rich and famous of today use social media to reach out to thousands (if not millions) of adoring fans.

With the help of NodeXL, you can investigate the relative presence or absence of reciprocal ties and closed triads in your egocentric Twitter network. Most egocentric networks are a mix of strong and weak social ties, so that summarizing reciprocal ties and triads in a single number often leads to incorrect conclusions about your social circle. Instead, visualizing your social network can give you insights about your close versus distant ties and the interactions between your close friends and acquaintances.

Many Twitter users broadcast events in their lives, from the trivial (eating habits) to the important, such as travel arrangements, major life events, and so on. These events are in the short-term visible to all of the user's followers, which make up part of her egocentric network. In very small egocentric networks (e.g., when a user's followers are limited to one's family), this form of broadcast is useful for staying in touch with relatives who have traveled abroad or moved away. However, many active users have rather large egocentric networks, and their tweets often go out to people completely uninterested in them. At the same time, the sheer number of followers a Twitter user amasses makes it hard to keep track of who is reading the tweets. NodeXL makes it relatively easy to identify different groups in your egocentric networks, such as your followers who are your colleagues versus those who are your family. You can take advantage of the new Twitter Lists feature to organize these groups into convenient lists for following and referencing in your tweets.

The first step in analyzing your egocentric network in NodeXL is to collect the data. One way to do so is to use the Import > From Twitter User's Network feature in NodeXL. Clicking on the NodeXL tab in the Excel menu, then on the Import drop-down list at the left of the NodeXL ribbon, you are presented with several import options (make sure the checkbox "Clear NodeXL Workbook First" is checked). Select the From Twitter User's Network option in the drop-down list, at which point you are presented with a dialog box (Fig. 10.12).

At the top of the box is a text field for the Twitter user whose egocentric network you want to collect. For this example, we will use one of our own Twitter accounts—@vlad43210. Below the text field lie three checkboxes: Add an edge for Follows, Replies-to, and Mentions relationships in tweets. Make sure only the Follows checkbox is checked, the other two are more

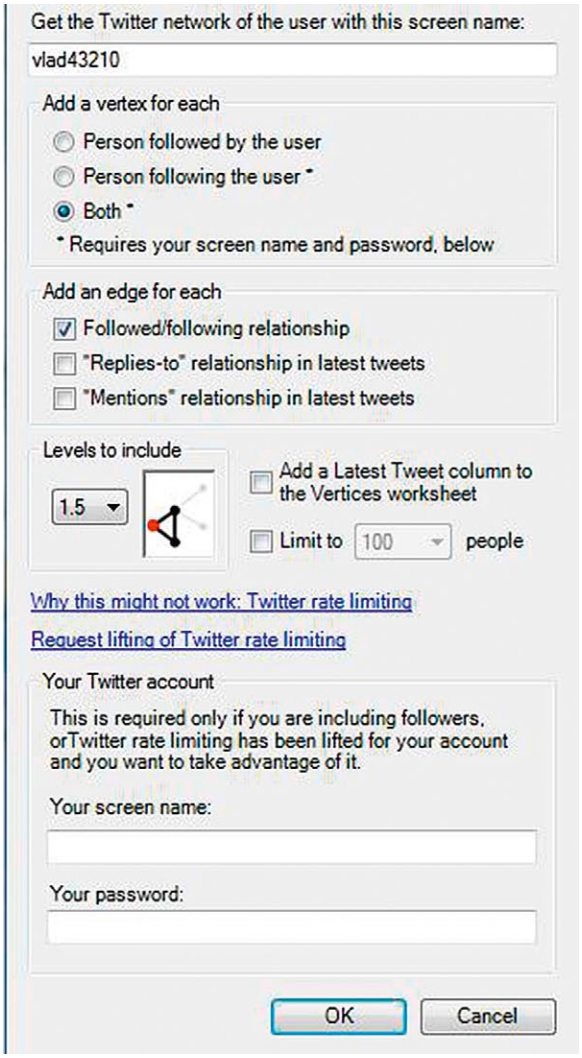


FIGURE 10.12 NodeXL's Twitter User's Network import screen. The levels options selected will import both @vlad43210's friends and followers, as well as any friend/follower connections between them.

relevant for information networks, which we will investigate later. Also make sure the "get both friends and followers" checkbox is checked so you can collect data about all alters, not just friends or followers.

Further down, you are presented with the "Get connections out to X degrees" drop-down list. Select "1.5" in the drop-down list to capture all of your alters, as well as all of the follow and friend relationships between them. The latter set of relationships is necessary for observing the presence of closed triads in your egocentric network. Next, make sure the "Add statistic columns to the Vertices worksheet (slower)" and the "Add a Tweet column to the Vertices worksheet" checkboxes are unchecked. These are highly useful for collecting data about Twitter activity in your egocentric network, but for the purposes of the current example we are focused on relationships, not activity.

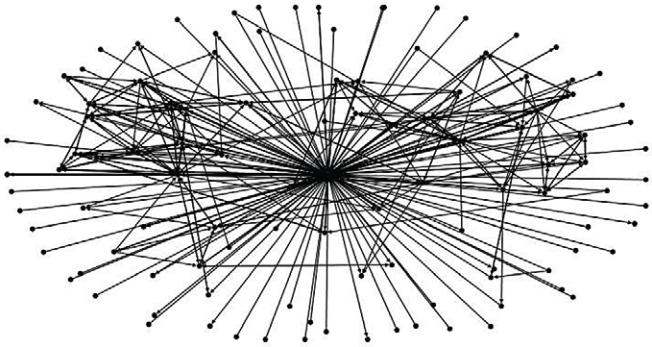


FIGURE 10.13 The @vlad43210's 1.5-degree egocentric Twitter network. The vertex at the center is @vlad43210, the many vertices at the edges represent @vlad43210's weak social ties, most of them complete strangers.

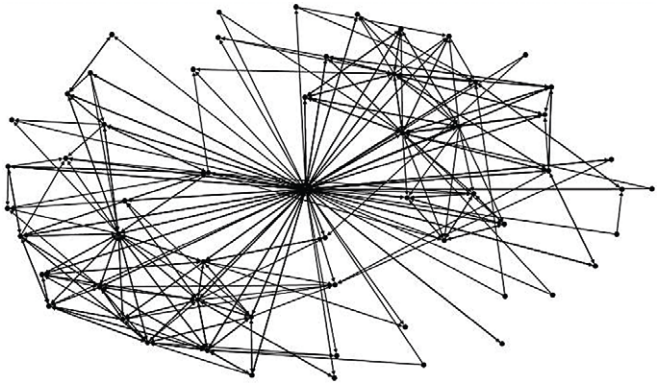


FIGURE 10.14 The @vlad43210's 1.5-degree egocentric Twitter network. We have filtered out the complete strangers to focus on @vlad43210's strong social ties—his friends and colleagues. Two clusters emerge (top right and bottom left).

The final two text fields allow you to specify a user name and password should you have a whitelisted Twitter account (discussed earlier). Whether or not you fill these out, you can next click OK and begin the data import process. Once the process is complete, you will see the spreadsheet again, filled with rows of pairs of Twitter users and the relationships between them. You may have to click on "Refresh Graph" to display these users and relationships on the graph canvas. Note that the ego is in the center—in our example, that is @vlad43210—surrounded by a multitude of alters. The picture on the canvas resembles a star, as @vlad43210 has many alters who do not follow each other. These are likely to be his weak social ties, most of them complete strangers (Fig. 10.13).

You can focus on stronger social ties by looking at alters in @vlad43210's egocentric network with at least two ties to other people (both ties may be to the same person) in either direction, friend or follower. Everyone we have collected data about will have at least one tie, to @vlad43210. Alters with at least two ties will have either a tie to another alter of @vlad43210 (indicating they are part of a closed triad), or they will have a reciprocal friend/follow tie to @vlad43210. As we discussed earlier, both of these factors may indicate the presence of a stronger social relationship between @vlad43210 and the individual in question. In the graph representation of the network, alters with two ties to other people will be represented by vertices with at least two edges (in either direction) between them and at least one other vertex. In order to focus only on these vertices, you must first calculate the in- and out-degree. Click on the "calculate graph metrics" button, check "in-degree" and "out-degree," and click "Compute Metrics."

Now let's set vertex visibility in the graph based on in-degree + out-degree $>$ 1. Vertices that fit this criterion will have at least two friends, two followers, or one friend and one follower (who may be the same person), which is precisely what we're looking for. To do this, first add a new column by clicking on the cell under the "Add your own Columns here" label in the Vertices worksheet, then setting the cell value as the sum of C3 and D3 (corresponding to in- and out-degree, respectively). The rest of the cells in the column will autofill with the same formula. Now label this new column as "Sum Degree" by editing the column label. Finally, click on Autofill Columns, click on the drop-down list next to "Vertex Visibility," click on "Sum Degree," then click on the button next to the drop-down list and type in 1 in the textbox to indicate that you want to show only vertices with in-degree + out-degree $>$ 1. Click OK, then Autofill, then Refresh Graph. Note that the number of vertices has become much smaller and the graph no longer looks like a star—instead we see @vlad43210 surrounded by two densely connected groups of vertices with a few connections between the groups.

Figure 10.14 suggests that @vlad43210's egocentric network has two separate tightly connected groups, or clusters, of alters. NodeXL can help confirm this intuition by performing automatic cluster detection. To detect clusters, simply click on "Find Clusters" in the Analysis section of the NodeXL menu ribbon. NodeXL will automatically search for clusters and assign different colors to vertices in different clusters (these assignments override any vertex colors defined via Autofill columns or via editing the worksheet). Now click "Refresh Graph" and the new vertex colors will show up on the canvas.

You can now click on vertices in different clusters to get a sense if the clusters represent meaningful partitions of @vlad43210's egocentric Twitter network.

Indeed, exploration of the worksheet reveals that the purple cluster corresponds to alters who are Vlad's academic colleagues, whereas the green cluster corresponds to alters who are his friends (in the common sense of the word, not in the Twitter sense of users @vlad43210 follows). The clusters can be used to create two Lists for Vlad's Twitter account: one for communicating with his friends and one for communicating with his colleagues.

You can also use the notions of *eigenvector centrality* and *betweenness centrality*, described previously, to look at the relative importance of actors in @vlad43210's egocentric Twitter network, from the attention and information perspectives. As we discussed in Section 10.3.1, actors with high eigenvector centrality in the friends/followers network are centers of attention, whereas actors with high betweenness centrality in the same network are information brokers. NodeXL allows us to visualize both of these quantities at the same time by mapping them to different properties of the vertices in the graph canvas.

Let's map vertex color to eigenvector centrality and vertex size to betweenness centrality.[7] To do this, first click on the Graph Metrics button in the Analysis section of the NodeXL menu ribbon, then check "Eigenvector centrality" and "Betweenness and closeness centralities" and click "Compute Metrics." Next click on the Autofill Columns button in the NodeXL menu, click on the drop-down list next to "Vertex Color" and select "Eigenvector Centrality," click on the drop-down list next to "Vertex Size" and select "Betweenness Centrality." Click on "Autofill" and then the Refresh Graph button on the graph display pane to update the vertex sizes and colors in the graph canvas. Make one more adjustment before displaying the graph—because @vlad43210 is at the center of his egocentric network by definition, his betweenness and eigenvector centralities will be high but will tell us nothing about @vlad43210's position in his network. Manually set @vlad43210's betweenness and eigenvector centralities to 0 in the worksheet and hit "Refresh Graph" again.

If you compare Figures 10.15 and 10.16, you can see that @vlad43210's colleagues are more centers of attention than his friends, but that the two clusters are pretty evenly matched in terms of information flow (with the exception of @redlog, who has both very high eigenvector and very high betweenness centralities and shows up as a large green node in the graph).

Finally, let's compare @vlad43210's egocentric network of friends and followers to the network of @replies and mentions of @vlad43210. Note that both of these networks are egocentric, but the alters are not necessarily the same in each. NodeXL does not import detailed @reply and mention data for users at the time of writing, but the Twitter API allows for a quick workaround. Using the statuses/mentions API, you can visit the URL www.twitter.com/statuses/mentions.xml and manually record the names of any of @vlad43210's friends or followers who @replied to him or mentioned his Twitter username in one of their tweets, as well as the number of @replies or mentions per name. Because @vlad43210 does not get many @replies or mentions, this is not a time-intensive task—for more time-intensive tasks, an alternative would be to write a script to record these names automatically. Next, for each such name, find the

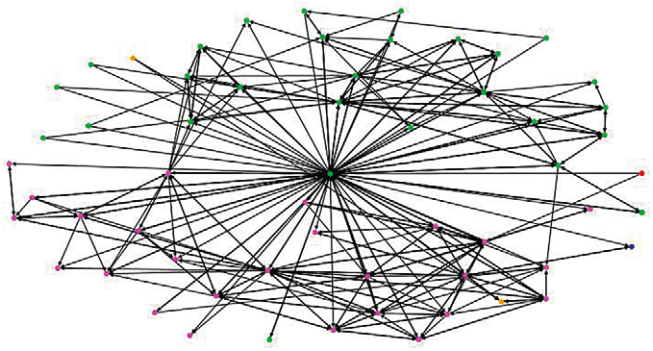


FIGURE 10.15 The @vlad43210 1.5-degree egocentric Twitter network (strong social ties only). NodeXL automatically identifies the clusters and colors them differently. Top right (green) corresponds to @vlad43210's friends, and bottom left (purple) corresponds to his academic colleagues.

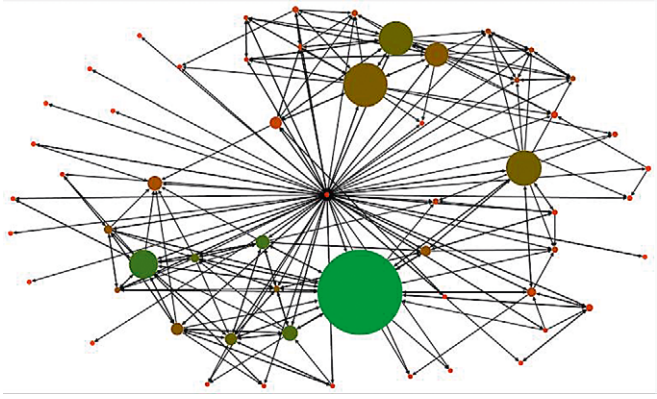


FIGURE 10.16 The @vlad43210 1.5-degree egocentric Twitter network. Greener vertices have higher eigenvector centrality, and larger vertices have higher betweenness centrality.

[7]If you have just computed clusters, please begin by clearing the Clusters and Cluster Vertices worksheets as well, so that cluster assignments do not override the vertex color and size properties in the canvas. You can also deselect "Clusters" from the Graph Elements menu item in the Show/Hide section of the NodeXL menu ribbon.

edge in the Edges worksheet connecting @vlad43210 to a user who @replied to or mentioned him. Manually set the width of these edges to the number of @replies or mentions of @vlad43210 by the corresponding user. Finally, hit "Refresh Graph" to reflect these updates (Fig. 10.17).

You can see that although @vlad43210's colleagues are more centers of attention than his friends, his friends @reply to @vlad43210 and mention him more often than his colleagues do. Also note that although some @reply and mention connections link @vlad43210 to high-centrality actors such as @redlog, others link @vlad43210 to low-centrality actors. Clearly, the information and attention relationships in the friends/followers egocentric network of @vlad43210 do not necessarily hold the same weight in his @replies/mentions egocentric network.

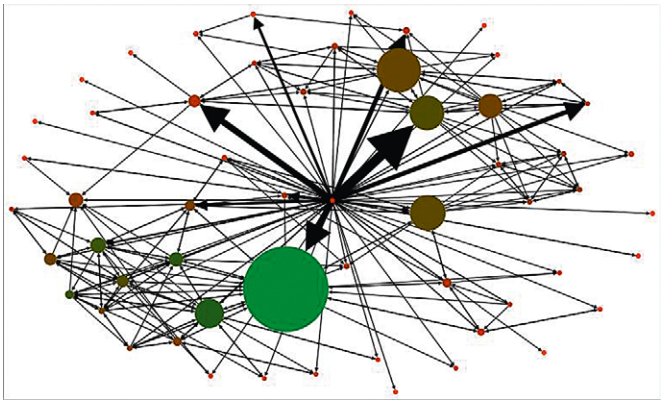


FIGURE 10.17 The @vlad43210 1.5-degree egocentric Twitter network. Greener vertices have higher eigenvector centrality, and larger vertices have higher betweenness centrality. Thicker edges correspond to users that @reply to @vlad43210 or mention his username in one of their tweets.

10.5.2 A Trending Topic

We conclude with a section on information flows in Twitter. Many of the principles discussed in Sections 10.2.2 and 10.2.3 (such as #hashtags and retweeting) are applicable here. For the convenience of gathering a sizable and illustrative sample at the time of writing, the example that follows uses a particular popular search phrase ("Black Friday") instead of a #hashtag, but the

ADVANCED TOPIC
Aligning the Vertices in a Graph

Let's look briefly at Figures 10.9 and 10.10 (the information/attention network and the centrality network examples). Suppose you want to line up a small number of vertices in a way that clearly shows groupings. It must be a lot of work to line those vertices up so sharply by hand, dragging them around one by one.

As we described in the "Advanced" box earlier, you can manipulate NodeXL networks just like regular spreadsheets. The Vertices tab contains columns called "X" and "Y," which hold the vertices' positions in the graph. There is also a column named "Locked?" which controls whether the vertices' positions are locked into place or whether they can be moved when the graph is redrawn.

After you've rearranged the vertices the way you like them, run the following macro to align them perfectly:

```
Sub Realign()

 ' the distance to round each location to
 RDIST = 500

 COL_VERTEX = 1
 COL_LOCK = 21
 COL_X = 19
 COL_Y = 20
 ROW_START = 3

 Dim wksht As Worksheet

 Set wksht = Sheets("Vertices")
 current_row = ROW_START
 While wksht.Cells(current_row, COL_VERTEX).Text
  > ""
 ' make sure the vertex's position is locked
 wksht.Cells(current_row, COL_LOCK) = "Yes (1)"
 ' round the x and y

 x = Round(wksht.Cells(current_row, COL_X) /
  RDIST) * RDIST

 wksht.Cells(current_row, COL_X) = x

 y = Round(wksht.Cells(current_row, COL_Y) /
  RDIST) * RDIST

 wksht.Cells(current_row, COL_Y) = y

 current_row = current_row + 1

 Wend

End Sub
```

This macro loops through each vertex in the graph. It first locks the vertex's position, then rounds the X, Y coordinates to the nearest 500 pixels. So if two vertices are next to each other and their Y coordinates are 1498 and 1502, each will be repositioned to y = 1500. This ensures that the vertices are properly aligned.

analysis could be replicated with a #hashtag for an alternative popular topic as well. The following techniques will be useful to anyone who wishes to use Twitter for advertising or promotions. Interested readers may also look at Hopkins [8] for another example of analyzing the spread of information in Twitter via NodeXL.

One of the most important features of Twitter is the ability to quickly relay information in a distributed manner. News centers and celebrities tweet messages about important events, promotions, and so forth. Their followers see the message and, if it is important and interesting enough, tweet about it as well (possibly retweeting the original message). At this point, the followers' followers (who may or may not be following the original information source) are also exposed to the message, and so on. In this manner, breaking news can spread throughout the Twitter follower network until thousands or even millions of people are aware of it, within hours. Twitter keeps a list of the fastest-growing messages in its search interface as "trending topics."

Breaking news is not the only thing that spreads on Twitter. Announcements, promotions, and events are often of interest to users who wish to advertise on Twitter. Tweeting about an upcoming promotion is a very low-cost advertising solution—but how many people will your tweet reach? It is possible that users will find the promotion relevant to their social circle and retweet it, beginning a virtuous cycle that could greatly increase your customer base. Most users, however, will ignore promotions or take advantage of them without passing them on. It is, therefore, important to identify those Twitter users who have a track record of having their messages repeated (and possibly reward them to ensure more prominent mentions in the future).

NodeXL allows you to search for any search phrase (including news, advertisements, and everything else that shows up in tweets) and observe its spread through Twitter. The first step is to identify an appropriate search phrase. There are a number of relevant criteria here: first, the search phrase has to concern a recent event. Though Twitter has been around for several years, the volume of information being produced every second is so huge that the search interface has limits on how many tweets it will return for a given query, or how old tweets can be. Searching for "2008 Election" may in theory produce a valuable set of tweets about the election cycle, but in practice those tweets are too far back in time for the search interface to collect them efficiently. The second criterion is that the search phrase has to relate to a piece of news, promotion, event, and so on that is "contagious" (i.e., Twitter users who see the message will, at least in principle, want to pass it on to their followers). A search phrase like "Thanksgiving" is a trending topic on Twitter (shortly before and on Thanksgiving) but lacks a contagious property—there is no need to pass on the message because a large fraction of the population already knows about it, so tweets about Thanksgiving are independent events rather than the sign of a "Thanksgiving meme" spreading throughout the Twitter population.

For the following example, you will observe the spread of the "Black Friday" search phrase on Twitter. While Black Friday is, like Thanksgiving, an annual event already familiar to the Twitter population, this search query is often associated with product deals that businesses advertise in the weeks prior to the event. It is important for businesses to spread awareness of their deals as widely as possible, because they have to compete for customer attention and time on the busiest shopping day of the year.

With this search term in mind, let's move on to the Twitter import interface in NodeXL. If you click on the NodeXL tab in the Excel menu, then on the Import drop-down list in the Data section at the left of the NodeXL ribbon, you are presented with several import options. At the top of the drop-down list is a checkbox that says, "Clear NodeXL Workbook First." For this example, you can uncheck this checkbox, which allows us to combine the results of several Twitter imports into the same worksheet. We explain the utility of this feature later. For now, move on to the From Twitter Search Network option in the drop-down list, at which point you are presented with a dialog box (Fig. 10.18).

At the top of the Twitter search network data import dialog box is a text field for our search term. You can enter "Black Friday" in quotation marks in order to capture only tweets with both of those words. Below the text field lie three checkboxes: Add an edge for Follows, Replies-to, and Mentions relationships in tweets. These checkboxes allow you to specify the kinds of networks you wish to capture the spread of the search phrase over. For instance, if you check only Follows, the Import procedure will only capture Follows relationships between users who mention "Black Friday" in their tweets, but not any Mentions relationships (where user A mentions user B's name in his or her tweet), nor Replies relationships (where user A directly replies to user B by beginning his or her tweet with "@user_B_name"). In most cases, it is advisable to check all three boxes to capture all possible relationships between users who mention the search phrase. It is relatively straightforward to focus only on one relationship (e.g., Follows) once the data are collected. It is important to note, however, that the Follows relationships are significantly slower to collect than the "Replies-to" or the "Mentions" relationships. For this example, you can check all three checkboxes. Note that even though all three checkboxes are checked, the resulting data set (see online materials) contains only Follows edges (no Mentions or Replies-to relationships between the individuals crawled were found).

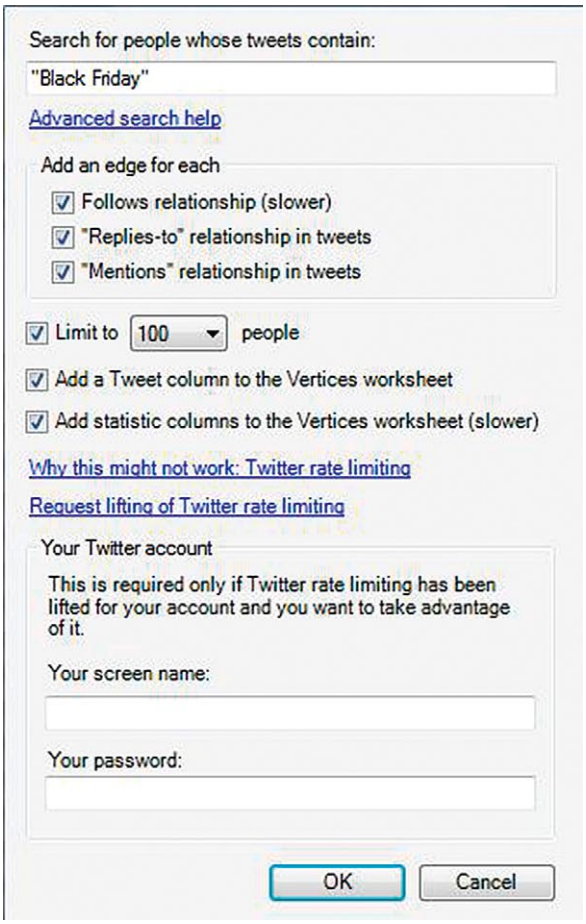


FIGURE 10.18 Twitter Search Network import dialog. The options selected import the first 100 people to mention "Black Friday" in their tweets, and all of the Follows, Replies-to, and Mentions relationships between them.

The next three checkboxes allow you to further specify the scope of your Twitter data collection crawl. The "Limit to" checkbox is perhaps the most important feature of the Import dialog box. Unchecking this box allows you to collect a lot of data but runs the danger of exceeding the Twitter rate limits (discussed earlier) and may take a very long time. It is important to remember that the search interface takes (with all boxes checked) roughly 10 to 30 seconds per Twitter user to collect data, so collecting mentions of a popular search phrase may take hours or even days! The Twitter collection process also frequently reaches corrupted or missing user information, which interrupts the collection process and introduces errors into the data set. Therefore, we highly suggest keeping this checkbox checked and setting the limit as low as possible (100 people) to start although larger data sets can contain information that is missing from smaller samples. The next two checkboxes, "Add a Tweet column to the Vertices worksheet" and "Add statistic columns to the Vertices worksheet (slower)" are highly useful for collecting accurate data about the spread of a search phrase, and you can leave them both checked as well.

The final two text fields allow you to specify a user name and password should you have a whitelisted Twitter account (discussed earlier). You can enter the whitelisted account information here, which allows you to not worry about exceeding Twitter rate limiting on our limited population of 100 users. At this point, you can click OK and begin the data import process. Once the process is complete, you will see the spreadsheet again, filled out with a set of Twitter users and the relationships between them.

It is important to be aware of the overall sparsity of Twitter relationships. The first two times we tried this search, we found absolutely no edges—just unconnected users mentioning "Black Friday." This sparsity does not, however, suggest that "Black Friday" is not contagious. As mentioned earlier, most users will ignore advertisements or take advantage of them without passing them on, but the rare user that passes ads on to his or her followers is worth discovering. On the third search, we found 66 edges and an overall pattern of connections that looks like that shown in Figure 10.19.

This pattern suggests that a small number of "stars"—users with many unconnected followers/friends—are passing along Black Friday ads to their social circle. But do the ads spread farther? Keep in mind that the links in this graph represent "follow" relationships. It could be that the followers of the stars simply happen to mention Black Friday around the same time the stars do, instead of passing a specific piece of information (e.g., a particular Black Friday advertisement) on to their followers. Fortunately, NodeXL allows us to test this hypothesis by looking at the tweets of the stars and their followers.

You can start by looking at some of the stars. From the Tweet column, you can see that one of these users, @ShopNPartyGirl, tweets, "*SneakPeek Kohl's Black Friday Ad - ...*" (link to the ad omitted). Let's see if any of @ShopNPartyGirl's followers mention the Black Friday ads she has posted.[8] First, let's find all of @ShopNPartyGirl's followers by going to the Edges worksheet, sorting all rows by Vertex 2 (the person being followed, in our case). Now find all the rows where Vertex 2 is "ShopNPartyGirl." The Vertex 1 values for those rows are @ShopNPartyGirl's followers. Let's make

[8]Because of the small size of this particular data set, you can do this by hand. In larger data sets, it would be easier to write a macro (see Advanced code examples in this chapter) to automatically identify followers who retweet their friends' posts.

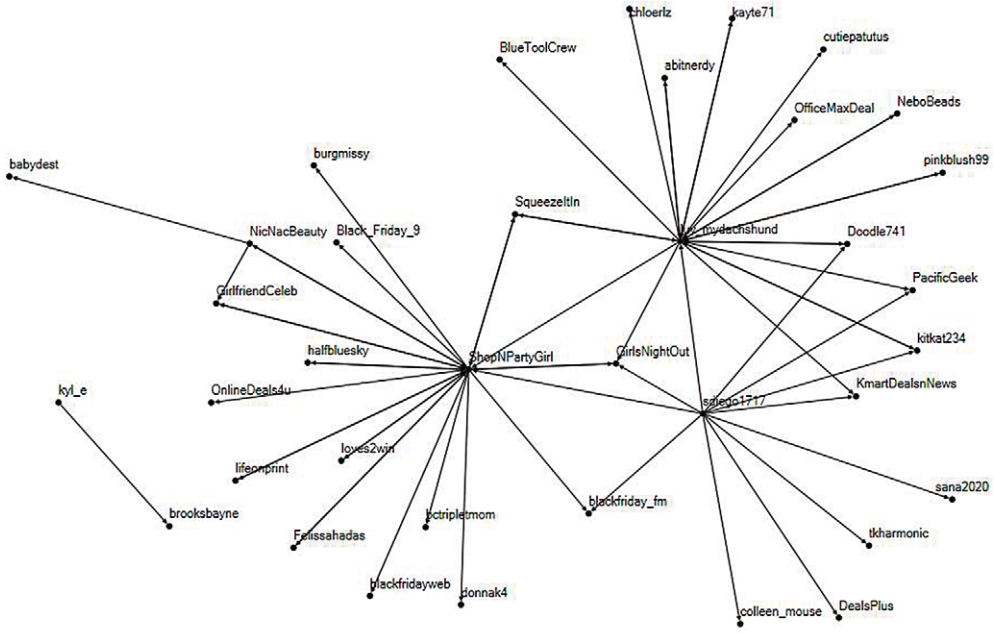


FIGURE 10.19 Twitter mentions network for "Black Friday." Note that the network resembles a set of connected stars: around @ShopNPartyGirl, @luv_mydaschund, and @sdiego1717, respectively.

a note of these followers by copy-pasting their Twitter names (from Vertex 1) to a scratch space—Notepad, for instance. Now return to NodeXL, switch to the Vertices worksheet, and sort all rows by Vertex. We can see, for instance, that user GirlfriendCeleb retweets one of @ShopNPartyGirl's ad tweets:

```
GirlfriendCeleb: RT @ShopNPartyGirl: Sneak Peek
  Meijer Black Friday Ad - ...
```

Although the particular tweet about Kohl's may not have been retweeted, it certainly seems that @ShopNPartyGirl's followers pass on advertisements she tweets about.

It would be nice to highlight all of the relationships through which @ShopNPartyGirl's tweets are retweeted, to see the full reach of ads she posts. As you go down the Vertex list in the Vertex worksheet, make a note of those followers of @ShopNPartyGirl (saved in Notepad) who have retweeted one of @ShopNPartyGirl's ads, by putting an "R" next to their name in Notepad. Now come back to the Edges worksheet and find all of the edges where Vertex 2 is "ShopNPartyGirl." Let's highlight those edges by setting their width (in the respective column of the Edges worksheet) to 2.0.

Let's look at the Vertices worksheet some more. The user @abitnerdy, connected to another star, @luv_mydaschund, has also retweeted @ShopNPartyGirl. Meanwhile, @luv_mydaschund does follow @ShopNPartyGirl, but his tweet is not a retweet of @ShopNPartyGirl. But because @abitnerdy is much more likely to have retweeted @ShopNPartyGirl if a tweet of hers showed up in his tweet-stream, it is likely that @luv_mydaschund *did* retweet @ShopNPartyGirl in some tweet that was not picked up by NodeXL, and then @abitnerdy retweeted that tweet. You can thicken both the edge between @luv_mydaschund and @ShopNPartyGirl, and between @abitnerdy and @luv_mydaschund, to indicate the reach of @ShopNPartyGirl's tweets about Black Friday even though you do not have the full record of retweets between all three parties. In the same manner, you can highlight the edges between @luv_mydaschund and his followers who have retweeted @ShopNPartyGirl's tweets (even if they do not follow @ShopNPartyGirl directly). At this point, let's refresh the graph (Fig. 10.20).

At the end of this process, you can see that @ShopNPartyGirl's tweets about Black Friday ads were retweeted by many of her followers, including @luv_mydaschund, many of whose followers retweeted @ShopNPartyGirl's tweets as well. Clearly, @ShopNPartyGirl is a good "seed" for spreading Black Friday ad information. Her tweets were picked up by 16 other people! Don't forget, each of those people's followers automatically saw the retweet, so the actual reach of @ShopNPartyGirl's tweets is much higher.

We can even better estimate the reach of @ShopNPartyGirl's tweets by incorporating some statistics about the Twitter users in our sample into the canvas visualization. We can assume that users with more followers are more popular, and users with more tweets are more active, on Twitter. Under these assumptions, we can visualize user popularity on the canvas by mapping vertex size to followers and vertex color to tweets. To do this, click on Autofill columns in the NodeXL Visual Properties menu section, click on the drop-down list next to "Vertex Size," click on "Followers," then click

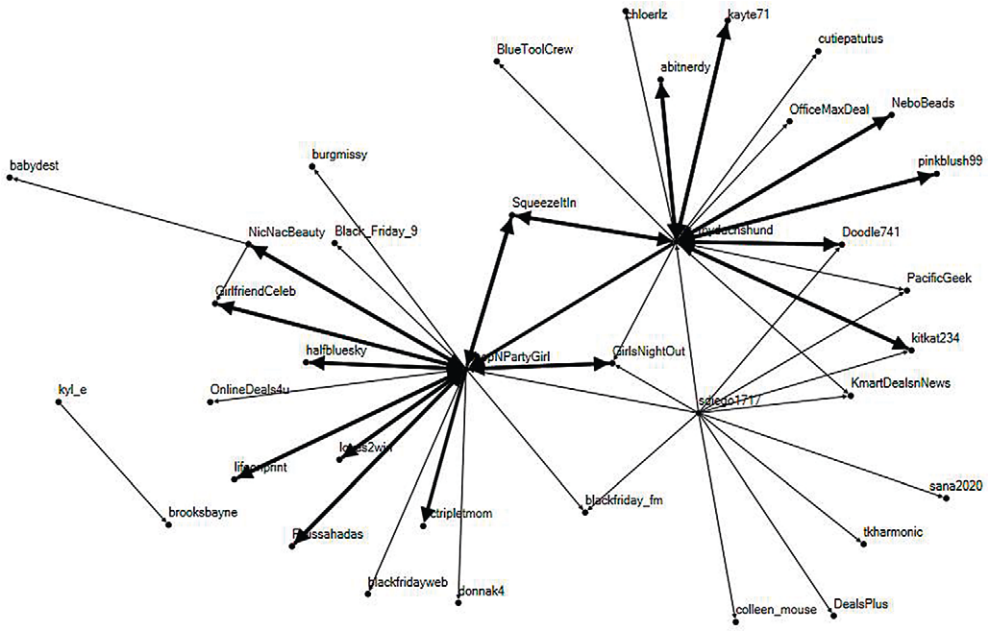


FIGURE 10.20 Twitter mentions network for "Black Friday." Thicker edges represent retweets of @ShopNPartyGirl's messages.

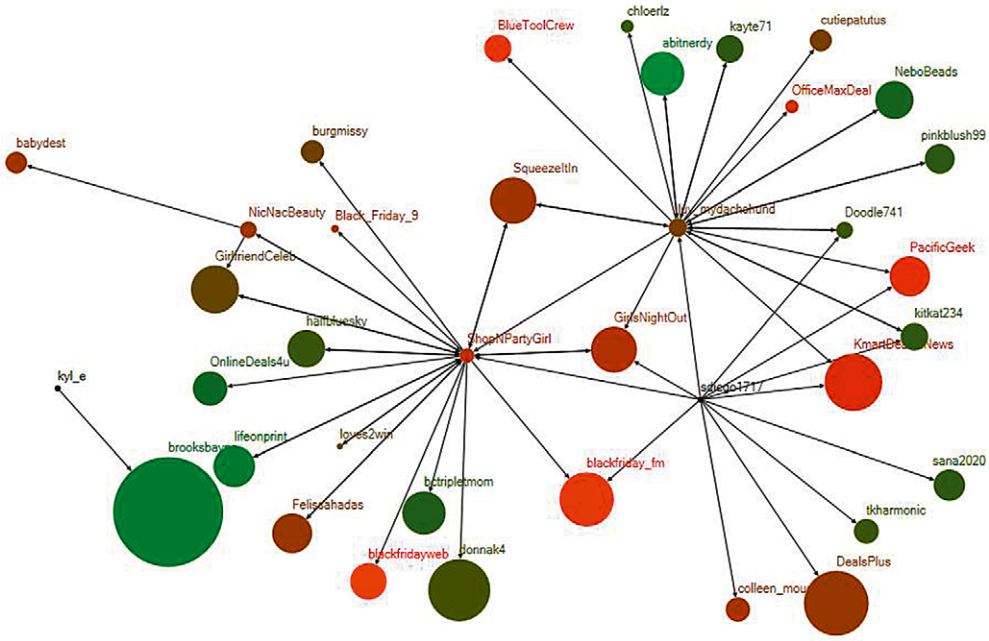


FIGURE 10.21 Twitter mentions network for "Black Friday." @ShopNPartyGirl and her friend/follow relationships are highlighted in red. Greener vertices have more tweets, larger vertices have more followers.

on the drop-down list next to "Vertex Color" and click on "Tweets" (don't click Autofill yet!). At this point, you have identified all of the retweeting information, so let's hide the tweets from the canvas and replace them with usernames. To do this, click on the drop-down list next to "Vertex Label," and select "Vertex." Now click "Autofill" and "Refresh Graph" (Fig. 10.21).

You can see that @ShopNPartyGirl, highlighted above in red, is a small vertex (indicating relatively few followers) that spreads her message to many large vertices (many followers), some of which are also green (many tweets). Perhaps @ShopNPartyGirl is just a huge fan of Black Friday, or even a specialized account for tweeting on shopping and party deals. You can click on @ShopNPartyGirl's Twitter Page (under the second "Custom Menu" column in her row of the Vertices worksheet) and confirm that, indeed, this is the case. A specialized account with a narrow focus,

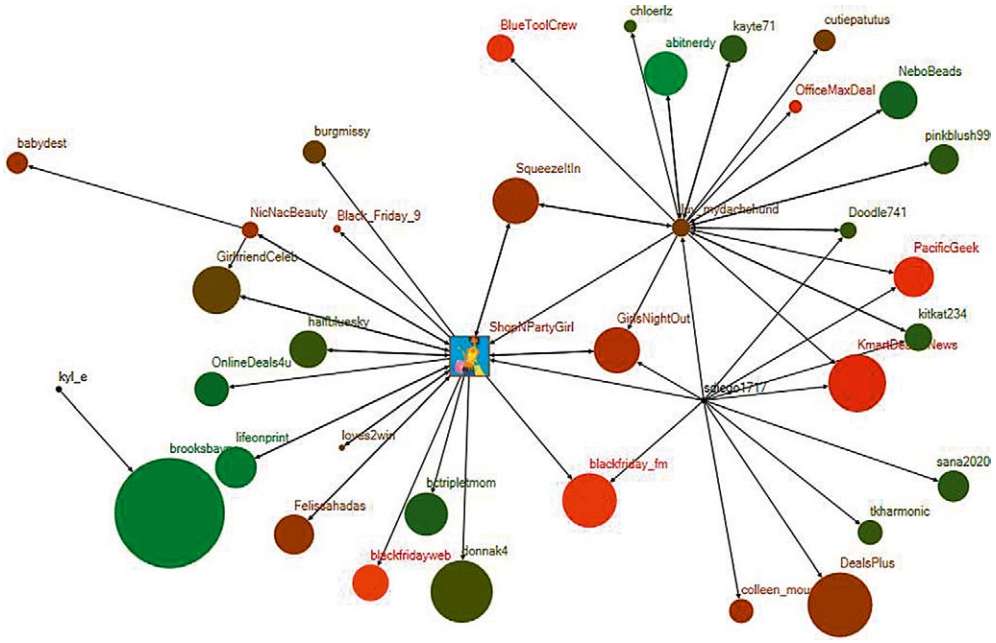


FIGURE 10.22 Twitter mentions network for "Black Friday." @ShopNPartyGirl and her friend/follow relationships are highlighted in red. Greener vertices have more tweets, larger vertices have more followers. @ShopNPartyGirl is represented in the canvas by her Twitter profile image.

@ShopNPartyGirl does not need a lot of followers or a lot of tweets to reach a wide audience. The owner of this account is clearly a competent promoter and a prime target for shops with Black Friday deals to send their advertisements to. Complete the exercise by highlighting @ShopNPartyGirl in the canvas, turning her vertex shape into her Twitter profile image. You can accomplish this goal by clicking on @ShopNPartyGirl in the canvas, right-clicking, selecting "Edit Selected Vertex Attributes," then clicking on the Shape drop-down list and selecting "Image." To save this image for your records, right-click on the canvas, then select "Save to Image file" (Fig. 10.22).

10.6 PRACTITIONER'S SUMMARY

Twitter's popularity as a social media platform comes from its multipurpose uses. Through a common framework of 140-character micro-updates, Twitterers can keep in touch with friends and keep up to date with important information like news announcements and promotions. Businesses can use Twitter to improve their brand, reach a broader audience and increase profits via advertising, and target others who are most likely to be interested in their messages and be most likely to act upon them.

No matter why you use Twitter, you need an appropriate strategy to get actionable results. For example, you may want a better awareness of your personal social network or want to build a stronger community in which to promote your product. Knowledge of the networks in Twitter, and of the relationships those networks embody, is key to formulating an appropriate strategy. For instance, knowing the influential individuals in a Twitter information network (such as @ShopNPartyGirl in the "Black Friday" network) helps you pick "seeds" to which you can send your ads and promotions. These seeds might tweet about your ads, and their tweets may be retweeted by some and (if the seed is like @ShopNPartyGirl) seen by many, introducing a multitude of potential shoppers to your Black Friday deals for free.

Understanding the network structure in a particular Twitter network is equally important to getting actionable results out of Twitter. For example, a "starlike" network structure is centrally focused, so if your personal network looks like a star (few connections between your friends and followers), your tweets receive a disproportionate amount of attention relative to the tweets of any of your followers. Without a good understanding of the kind of network you are working with in Twitter, and the structure of that network, you may be unable to formulate a good strategy for advertising your product or, worse yet, will formulate a bad strategy that will result in negative branding.

NodeXL helps you extract and analyze many different social networks from Twitter. The previous sections contain two examples of NodeXL at work: one for managing your personal contacts and another for tracking the spread of a phrase from one Twitter user to another. We hope that these examples will inspire you to take advantage of NodeXL to explore Twitter networks in more detail and discover more ways to use microblogs to help your business or career.

10.7 RESEARCHER'S AGENDA

Twitter's immense popularity and openness of access (via the API) has led to a boom in research based on its data. In computer and information science, this research draws on a long history of analyzing physical computer networks and information networks like the World Wide Web. Social network researchers also draw on a long research tradition; from attendance at social events to the exchange of emails, social network researchers have long recognized that interaction and communication patterns are the tangible proxies by which otherwise ineffable social relationships can be measured.

To date, research using Twitter has largely fallen into one of two categories: research at the micro level has documented the characteristics of individual activity within Twitter and the use of Twitter's features, such as @replies, retweets [4] and hashtags, and at the macro level has shown how memes, ideas, and such spread through the network, how like-minded people form structural clusters, and how aggregate network properties of Twitter compare to other large-scale collaborative networks [9]. However, as a general purpose conduit for sharing information across social ties, Twitter presents even greater opportunities for social network research.

Many social network researchers are concerned with how network structure influences individual behavior, how one's position in a social network affects one's access to resources [5], and how individuals' choices in these matters in turn affect the structure of the network. The effects of these things on one another play out over time and in subtle ways. To be able to answer these questions, new measures and new kinds of data will be needed. Data will be needed that describes how very large networks and their members change over time. The challenges for the future are immense; even in cutting-edge research, it is a challenge to obtain the kind of data necessary, much less analyze it or visualize it. Twitter is one of the few large-scale online social networks that can provide this data, by recording minute details of individuals' conversations (through their tweets), their position in the network (through friend/follower relationships), and the interaction between these two. It is also one of the few public networks that spans across institutional boundaries, providing opportunities for researchers to understand how organizations are connected via individual ties.

References

[1] J. Golbeck, J. Grimes, A. Rogers, Twitter Use by the U.S. Congress, Technical Report, College of Information Studies, College Park, MD, University of Maryland, 2009.
[2] A.L. Hughes, L. Palen, Twitter Adoption and Use in Mass Convergence and Emergency Events, Proceedings of the Sixth International ISCRAM Conference, Gothenburg, Sweden, May 10–13, 2009.
[3] D. Sagolla, 140 Characters, Wiley, Hoboken, NJ, 2009.
[4] d. boyd, S.A. Golder, G. Lotan, Tweet, tweet, retweet: conversational aspects of retweeting on twitter, in: Proceedings of the Forty-Third Hawaii International Conference on System Sciences (HICSS-43), Kauai, HI, 2010.
[5] R. Burt, Structural Holes: the Social Structure of Competition, Harvard University Press, Cambridge, MA, 1995.
[6] S. Fox, K. Zickuhr, A. Smith, Twitter and Status Updating, Fall 2009, Pew Internet and American Life Project, 2009, www.pewinternet.org/~/media/Files/Reports/2009/PIP_Twitter_Fall_2009web.pdf
[7] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, in: Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia, 1998.
[8] J. Hopkins, Visualizing a Monetized Twitter Network, http://julianhopkins.net/index.php?/archives/266-Visualising-a-monetised-Twitter-network.html
[9] B. Huberman, D. Romero, F. Wu, Social networks that matter: twitter under the microscope, First Monday 14 (1) (2009).

Additional Resources

Golder, S., & Huberman, B.A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*, *32*(2), 198–208.
Golder, S., Wilkinson, D., & Huberman, B.A. (2007, June). Rhythms of social interaction: Messaging in a massive online network. In C. Steinfield, B.T. Pentland, M. Ackerman, & N. Contractor (Eds.), *Communities and technologies 2007*. Springer.
Honeycutt, C., & Herring, S.C. (2009). Beyond microblogging: Conversation and collaboration via twitter. In: *Proceedings of the Forty-Second Hawaii International Conference on System Sciences (HICSS-42)*.
Messina, C. *Groups for Twitter, or a Proposal for Twitter Tag Channels*. http://factoryjoe.com/blog/2007/08/25/groups-for-twitter-or-a-proposal-for-twitter-tag-channels/
Owyang, J. *Coverage of the SXSW Web Awards. Blog post*. 3/11/07./www.web-strategist.com/blog/2007/03/11/coverage-of-the-sxsw-web-awards/

CHAPTER

11

Visualizing and Interpreting Facebook Networks

Bernie Hogan

Oxford Internet Institute, University of Oxford, Oxford, UK

OUTLINE

11.1 Introduction to Facebook: The World's Social Graph 165
11.2 The History of Facebook 166
11.3 Why Map a Facebook Network? 167
11.4 What Kind of Network is a Facebook Friendship Network? 167
11.5 Getting your Data into NodeXL 168
11.6 Creating a Basic Facebook Visualization 170
11.6.1 Hiding Yourself From Your Own Network 170
11.6.2 That Networky Look 170
11.7 Ordered and Nonordered Data and Attributes 171
11.7.1 Visualizing Nonordered Data: Clusters and Categories 171
11.7.2 Visualizing Ordered Data 174
11.8 Friend Wheel to Pinwheel: A Facebook Visualization the NodeXL Way 176
11.9 Practitioner's Summary 179
11.10 Researcher's Agenda 179

11.1 INTRODUCTION TO FACEBOOK: THE WORLD'S SOCIAL GRAPH

Do you have a single group of friends? Do people from high school know your buddies from college? Are your friends all a part of the same cluster or different clusters? Facebook may have the answer.

Facebook participation is based on a unique class of networks called "publicly articulated networks" [1]. That is, they are not networks of behavior, like email traffic, nor are they the networks we hold in our heads, like closeness networks [2]. Rather they are networks we show to others and use as a means for managing access to information. Twitter has such a network representing people followed by a user. YouTube has these networks, as well as friends and subscribers.[1] On Facebook, friends show up in a rolling list of information, called a news feed (see Figure 11.1). Friends can play games, chat, send messages, share photos, and invite each other to events. Basically, one's list of friends is a launching pad for a host of activities that people want to do online with *some* people, but not everyone. Facebook is likely the world's largest publicly articulated social network. As of this writing, Facebook has more than 400 million members, virtually all of whom are connected in some way to everyone else. Two hundred million of these Facebook users log in every month.[2]

This chapter has two purposes. First, it is an extended discussion on how to interpret a friendship network in Facebook (i.e., who is friends with whom). Of course, as

[1]For an extensive discussion of the history and definitions of social network sites, of which Facebook is one, see [11].
[2]www.facebook.com/press/info.php?statistics



FIGURE 11.1 Example Facebook page showing a user's news feed.

you have seen from other chapters in this volume, there are many ways to create a network from social media participation: for example, Facebook networks can be created from a list of who messages whom, who is featured in the same photo, or whom is a part of the same group [3]. Readers who have other interesting sources of network data captured from the Facebook site may still benefit from the techniques described here. Second, this chapter also has an extensive discussion on visualization that extends much of the earlier discussions in Part II (Chapters 4 through 7).

One of the drawbacks of the way Facebook is organized is that it is difficult to get a clear picture of the all of the friendships that make up a person's social network. Do your friends cluster into distinct groups, such as sports team and co-workers? Are you the only person that links these groups, or do you have friends that also know people in both groups? Do you want to bring your friends together, or perhaps try to keep some people separate? Do you want to find all of those people you added long ago and never speak to? To help you answer these questions, it may be helpful to examine your network as a whole and to use NodeXL to help.

11.2 THE HISTORY OF FACEBOOK

Facebook began in 2004 as just another fledgling social network site in the wake of Friendster (whose traffic was plummeting in the midst of service quality issues caused by the sudden rapid growth of the service). The other rising star at the time, MySpace, offered customization, links to musicians, and a healthy dose of trendy. By contrast, Facebook offered something seemingly counterintuitive to a successful online business: exclusivity. The site started among Harvard students and then took an incremental approach to growth, first adding users at select universities, followed by virtually any university, and only then the general public. This approach gave Facebook two advantages over the competition. The first is that because students at most universities are already well connected to each other, the site could benefit from "network effects" early on.[3] The second is that its origins at Harvard coupled with a cool blue and white aesthetic (in contrast to the anything goes aesthetic of the competition) signified the sort of legitimacy that invites older and more skeptical users into the fray.

One of the contradictions of Facebook is that the site provides tools for fine-tuning privacy but tends to want to push out as much content to as many people as publicly as possible. This is a contradiction because social network sites are typically premised on the idea that friends act as a sort of access control; friends can see each other's photos, read each other's updates, and get invited to each other's events in a way that strangers cannot. Facebook's first significant expression of its push toward more open information came in 2006.

[3]A network effect is when something becomes more useful as more people who use it, such as a telephone network, the DVD format, and so forth (compare with Metcalfe's law; [12])

In September of that year, Facebook added a "news feed" that consolidated recent activity by a user's friends on a user's homepage. Almost immediately, groups organized to protest the feature, saying it was an invasion of privacy and an intrusion. To this day, however, it remains a highly popular feature. In fact, this stream of information from one's friends has become a de facto reason for frequently checking the site.

Although Facebook does allow people access to a great deal of information about users and their friends, Facebook's information libertarianism has its limitations. For example, it does not alert users when others have severed the friendship. Nor does it allow users to see who has viewed the user's page. In fact, it has publicly banned applications designed to provide this information to users (such as Burger King's infamous "Delete 10 friends for a free Whopper" promotion in 2009).

Ultimately, however, the most important current limit on Facebook's information libertarianism is its data sharing policy. Facebook does not claim to own the data that it stores (rather it requires users to give it a nonexclusive license to redistribute content such as photographs). However, application developers who capture these data from Facebook are not supposed to keep these data without user consent. That said, numerous academic data sets drawn from Facebook are known to exist, typically through usage agreements with the site itself, or through direct collection of data and the pseudo-anonymization of network details.

Facebook's openness to information coupled with its significant rates of participation in many nations makes it an attractive source of data, especially if you have your own network in the service. However, note that you are capturing data from Facebook and that use of that data is governed by Facebook's privacy policy. Because this policy can change at any time, be sure to check the current terms of use to discover whether any data you download must be deleted or anonymized after you have finished using it.[4]

11.3 WHY MAP A FACEBOOK NETWORK?

Mapping and interpreting your network on Facebook has a number of uses. Knowing who is in your network and to whom they are connected can help you fine-tune your privacy settings. For example, if you have a list of family members, it can be useful to compare this list to the cluster of family members that NodeXL can discover. Second, analyzing a network like Facebook is one of the best ways to get to know some of the basic concepts of network analysis. Why? The network of your friends, family, and acquaintances is a network you know best.

Professionals who use Facebook for "networking" can discover more about their current networking style and whether they want to change this style for different purposes. For example, some people prefer to be "team players" and others prefer to be "brokers." When a team player sees that two of his colleagues are not connected, he wants to introduce them or "close a triad." By contrast, a broker prefers to keep these relationships separate knowing that if people have to go her, then this makes her valuable (this is called "brokerage" [4]).

People who use Facebook to plan large events or promotions can benefit from learning about their audience. Do attendees tend to come from a single dense cluster of Facebook friends or from many parts of the network? If everyone who attends an art show or small concert comes from one dense cluster of friends, this is a signal that one should look to other clusters and reach to people in those clusters directly. By contrast, if the people who attend an event or show are all completely disconnected, this means there is much potential to create a community. These choices depend on your strategy, but NodeXL can help you be more strategic.

This chapter primarily focuses on an example network. This is a modified version of my Facebook network. Why is it modified? Unlike many of the other networks that have been used in this book, Facebook networks are semiprivate, meaning that although much of the data on Facebook is public, the network that you see belongs to your account only. Consequently, I will not be using any personally identifying details, except in the case of several friends whose permission I have secured. Their gender has been kept the same, but I will not be mentioning any other details. This means I have also had to delete the URL for most of the thumbnails in my network. Networks downloaded using the NodeXL data from NameGenWeb will have the real thumbnail URLs.

11.4 WHAT KIND OF NETWORK IS A FACEBOOK FRIENDSHIP NETWORK?

The networks discussed in this chapter are "egocentric" networks. The term "ego" is used to denote a person connected to everyone in the network. An ego network is the social world from ego's point of view. It is convention to use the term "alter" to refer to anyone else in the network. This way one can talk about both friends and followers or fans; they are all alters.

If you recall from the discussion in Chapter 10 about Twitter, there are many ways that you can create an

[4] www.facebook.com/terms.php

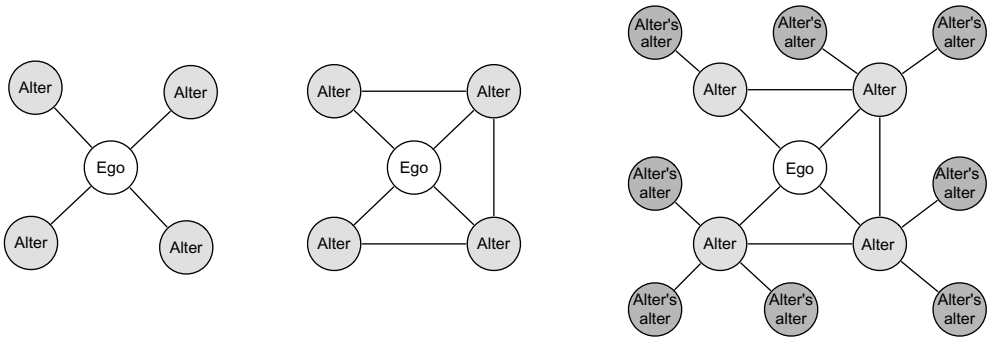


FIGURE 11.2 Examples of a 1.0, 1.5, and 2.0 network. In an unweighted undirected network, the 1.0 network is simply the list of a person's friends. The 1.5 network includes the connections between these people, and the 2.0 network includes their other friends. Facebook will enable a person to see both the 1.0 and the 1.5 list but not the 2.0 list.

egocentric network. One is a star that shows simply ego and all of ego's alters (a 1.0 degree network). Another is a network that includes the ties between these friends (a 1.5 degree network). A 2.0 degree network includes all of alter's other connections, and not just connections to people that ego knows. See Figure 11.2 for a demonstration.

Facebook allows an individual to discover his or her 1.5-degree network, but it does not allow people to discover their 2.0 degree network. That means that I can learn who all my friends are (i.e., ego's alters). I can also learn which of these people are friends with each other (i.e., alter-alter edges). What I cannot learn is whom else they are friends with. It is true that you can see this information when you surf to a friend's Facebook page. However, it is considered inappropriate to use a friend's page for data collection. Instead, Facebook provides a means for capturing data about your network—the Facebook API. I am only using the Facebook API and not screen scraping or other under-the-table ways of collecting data. This is to respect the way Facebook was set up and to be true to the privacy settings that people have put in place. The API respects personally defined privacy settings; therefore, so should we as researchers seeking to leave a light footprint.

Because this ego network is simply the network of friendships, it is also worth pointing out that it is undirected and unweighted. People on Facebook have to agree to be friends. Also, the basic friendship relation is the same for everyone. It is true that people can "hide" certain friends or restrict privacy for people. These individuals may be considered more or less significant to ego (in case you want to build a weighted network). However, this information is only available for your immediate friends (i.e., the 1.0 network). Consequently, this will not make for a very interesting network to analyze.

Finally, Facebook friendship networks tend to naturally cluster around different areas of a person's life: friends, classmates, workmates, and family, along with other groups or hobbies. If someone is particularly active on the site or comes from a culture that is very pro-Facebook, this network may be huge, but do not fear, it is likely that it still clusters in meaningful ways. For this reason, most of this chapter focuses on clustering and important people within and between these clusters.

The basic friendship network is only one of many possible network structures that you can capture from Facebook. For example, one can create a network of all of the people from a single high school class who are friends with each other. One could create a network of people who are participants of a group, and whether these people are friends of each other. Lewis et al. (2008) did something especially clever with their "Taste, Ties, and Time" data set [3]. In addition to the network of friends, they created a network of people who appear in photos together. They argued that this is a network that represents people who are especially close (because they show up in photos and therefore had to be co-present).

11.5 GETTING YOUR DATA INTO NODEXL

To get data into the program, first point your browser to the Facebook app (http://apps.facebook.com/namegenweb). You should choose to download the NodeXL graph. To do this for your own network, you will need to accept the application. The welcome screen will update as the file is being prepared. If you have a large network, this should take a few minutes. In general, it takes about a minute per 200 friends. Once the application is done, it will provide you with a link to your own network. Right-click and "Save as…" this file to your computer. Then click on the "delete my network" link that tells the server you are done with this network. If you do not manually delete your network,

the server will delete it in less than a day in compliance with Facebook's terms of use.

The file from NameGenWeb is not a .xls file but rather a GraphML file. NodeXL is able to import a number of different file formats. However, only one, GraphML, stores information about the nodes. We will be using this information later on in this chapter. To open this file, open a blank NodeXL template. To import this network, click Import → From GraphML file, and choose the file you just downloaded. You will notice that not all of the alters listed in the Vertices worksheet have detailed information that other users have associated with them. Some information is omitted when people adjust their Facebook settings to make their information available to a limited number of friends or to the public. The Facebook API will not deliver such information.

To test that your network has come through just fine, click on "Read Workbook." NodeXL will process all of the vertices and edges and present a generic network graph visualization.

Tip: It is useful to test your worksheet to make sure it is well formed. To do this, make note of the number of edges and the number of vertices before you first press "Read Workbook." Then go to the NodeXL ribbon, click "Merge Duplicate Edges," and then select "Read Workbook." If after you have done that the number of edges and vertices is the same, then the worksheet you have pasted is fully formed.

ADVANCED TOPIC

Roll-Your-Own Facebook Queries: REST API and FQL

If you want to create an application that captures friends, then you will need to know how to query the Facebook API yourself. I query using both php and python (using the modules provided on the Facebook developer site). For the php script, my department, the Oxford Internet Institute at the University of Oxford, has graciously provided me with server space. You will need to find your own. Because the python script runs on the desktop, it does not need a server.

There are two ways to query for friends: the REST API and FQL. The REST queries require you to ask if all friends are friends with each other. To do this, you have to send the server two lists of equal length. It returns a list of that length that says whether the people in these two lists are friends. Figure 11.3 shows this process conceptually with a series of five alters: Homer, Marge, Lisa, Bart, and Maggie.

- *Step 1.* Make a triangular matrix of these people (Fig. 11.3a) with the names 1 to n-1 going down the rows and 2 to n going across the columns. In this case, n equals 5 because there are five names.
- *Step 2.* For the first list, read the row names left to right. The second list is the column names shown left to right.
- *Step 3.* Query the server, and then build an edge list. The list that is returned can then be used to pull out a series of friend pairs: these will form the edge list.
- *Step 4.* Paste this list into NodeXL.

This process is thorough but very slow because it asks about all possible pairs, including those that are not actually friends. The second, and far more efficient but much

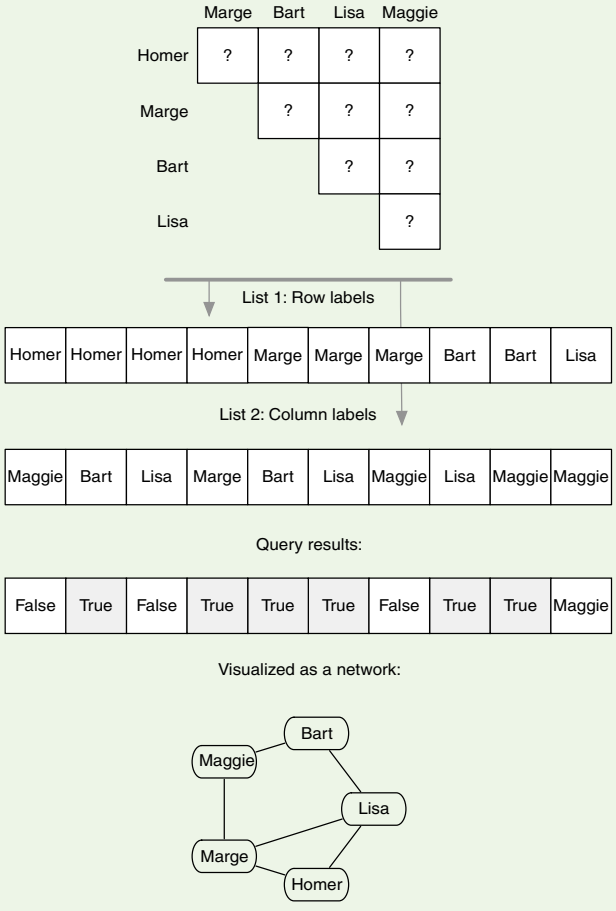


FIGURE 11.3 A diagram showing how to construct a network from a list of friends using the REST API friends.areFriends() query. The algorithm is written in text.

(*Continued*)

ADVANCED TOPIC *(Continued)*

more tricky way is to use an FQL query. FQL stands for Facebook Query Language, which is an adapted form of standard Structured Query Language (SQL). In this query you can ask for people who are ego's friends, and then ask if these people are friends with each other. The basic form of this query is as follows:

```
SELECT uid1, uid2 FROM friend WHERE uid1 IN
  (SELECT uid2 FROM friend WHERE uid1=$user_id)
  AND uid2 IN (SELECT uid1 FROM friend WHERE
  uid2=$user_id )
```

However, if you have more than a few hundred friends, this query might not return all of the edges. That's why I break down the query into a series of smaller queries. You can find the full algorithm on the web site that accompanies this book.

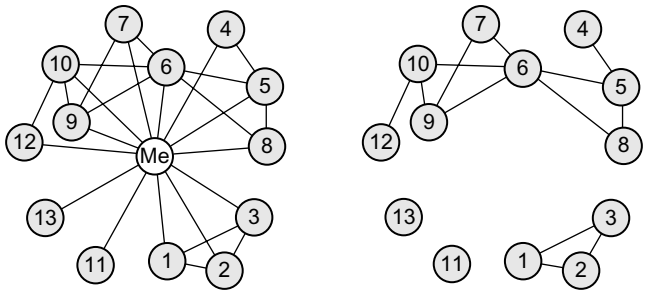


FIGURE 11.4 The same ego network with and without ego. The second visualization (excluding ego) shows the structure of the network more clearly.

11.6 CREATING A BASIC FACEBOOK VISUALIZATION

If you have pressed "Read Workbook," then you will have already seen a default picture of your Facebok network.

11.6.1 Hiding Yourself From Your Own Network

The first thing you might be wondering if this is your network is "Where am I on this graph?" The answer is actually nowhere! In unweighted and undirected ego networks it is customary to exclude the owner of the network from a personal network graph. This is because the owner (who network researchers typically call "ego") is connected to everyone in the network and there is no information gained from showing this. It would look very different if you included ego. For example, all of the individuals who are not connected to the giant cluster in an ego-centered network would all of a sudden be connected and hard to notice. Keeping ego out of the graph (and out of the calculations) makes it easier to identify patterns in this sort of network [5].

To illustrate this point, have a look at Figure 11.4. This is the same network with and without ego. On the left it looks like one single network diagram. On the right it is easy to see that there is one large component,[5] one triad and two isolates. These components represent distinct clusters of individuals who are not friends on the site except through ego. If this were your network, you would be more interested in learning that the five people on the bottom do not connect to the people at the top than learning that you are connected to everybody.

11.6.2 That Networky Look

As you learned in the earlier chapters, NodeXL comes with a number of visualization options. Two of layouts, the Fruchterman-Reingold layout and the Harel-Koren Fast Multiscaling layout, give that classic "networky" look. In this look, vertices that are adjacent to each other (i.e., friends) are shown near each other, whereas vertices that are not adjacent are far apart.

As mentioned before, I will be using my own network for the layouts that follow. That said, I have viewed and analyzed many networks, and for better or worse, I can assure you my network is rather normal. Like most personal networks on Facebook, it was almost completely connected (or "one big component" in network analysis terms). Also like most Facebook personal networks, it includes a number of clusters that represent social contexts alongside individuals of high degree and betweenness who are notable or personally important. Some people have networks that do not show this pattern. For example, I know someone who has a cluster of six people who are not connected

[5]A component in an undirected network includes all the people who are connected to each other. An isolate is a "trivial" component because it is just one vertex that is not connected to anyone.

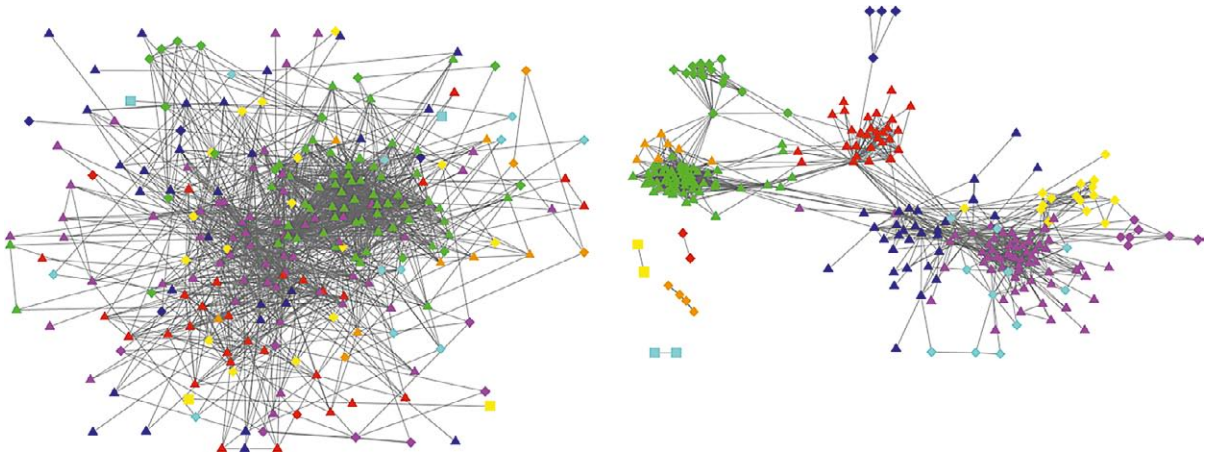


FIGURE 11.5 Comparing two network layouts of same graph. The first uses the default Fruchterman-Reingold settings and the second uses 100 iterations and a repulsion of 3.3. Clusters and bridges between these clusters are more clearly shown in the second layout.

to everyone else. She immediately recognized it as her ex-boyfriend and his friends. She did not want to drop these friends, but clearly she did not want to link them to the rest of her life either. I know someone else who has a huge cluster representing his former church group who are completely disconnected from his university friends. You similarly might become aware of interesting connections (or lack thereof) in your own network.

The Fruchterman-Reingold layout is a classic arrangement, but you may not get the best picture using the default settings. Instead, these networks may need 80 or 100 iterations of the algorithm rather than the default 10 iterations (see Advanced Topic: Changing Advanced Layout Options for Fruchterman-Reingold Layout in Chapter 7). Figure 11.5 shows the same network, first with the default Fruchterman-Reingold algorithm settings and secondly with 100 iterations. The beginning of the clustering is emergent in the first layout, but is especially clear in the second.

11.7 ORDERED AND NONORDERED DATA AND ATTRIBUTES

In a broad sense, there are two kinds of data: ordered and nonordered. Ordered data imply that the values can be sorted in some meaningful way. Degree is an ordered attribute, because we can order the values of degree from lowest (an isolate) to highest (the person with the most friends). Age is also an ordered attribute. Nonordered attributes (often called categorical data) do not have an obvious ordering. Religion is a nonordered attribute, as is gender and favorite sports team. Clusters are typically considered nonordered. Why make this distinction? Network attributes such as size and opacity are ordered, whereas attributes such as color and shape are nonordered.[6] In the visualizations in this chapter, I will try to match ordered data to ordered attributes and nonordered data to nonordered attributes.

The first section that follows discusses nonordered data. The second presents ordered data. In both cases, I show how you can use NodeXL in a simple way to show Facebook data (and network data in general) and a more complicated but arguably more fine tuned way to show the same data.

11.7.1 Visualizing Nonordered Data: Clusters and Categories

In a typical Facebook network, vertices are clustered into clear groups of individuals with more links within the groups than between them. As you learned in Chapter 7, these groups are called "clusters" in NodeXL. To discover the clusters in your network, simply press the Find Clusters button in the NodeXL ribbon and refresh the graph. The networks that were shown in Figure 11.5 were already clustered, which is why the vertices are color coded.

These clusters often signify meaningful sets of individuals, such as friends from high school or co-workers.

[6]Technically, color can be used to represent both ordered or nonordered data in NodeXL. The Autofill Columns feature maps different gradations of two colors to ordered data. In contrast, the Categorized Graph Scheme (see Fig. 11.6) and clusters map color to nonordered data by matching a distinct color to each cluster as is done throughout this chapter.

In fact, this clustering is perhaps the most important and meaningful information you can draw from a Facebook friend network. Once you have clustered the network, consider looking at the network and asking whether each of the clusters represent a meaningful set of people. One place to put this information is in a new column in the Clusters worksheet. Make a new column called "label" and insert the cluster name there. Alternately, it is possible to label the clusters by hand. See the Chapter 7 example on labeling Republicans and Democrats for more details.

Facebook includes a lot of nonordered data or "categorical" data. NameGenWeb will download gender and current location. The simplest way to distinguish vertices by category is to use a predefined graph "Scheme" from the Graph tab on the NodeXL ribbon (Fig. 11.6). This will assign new colors and shapes based on the category of your choosing.

What if you want to show both categories and clusters at the same time? For example, you would like to show both gender and cluster membership. If you want to use color and shape, then you will not be able to use the standard NodeXL cluster features because the shape and color on the cluster sheet override other shape and color values. In this case you have to manually attach the categories.

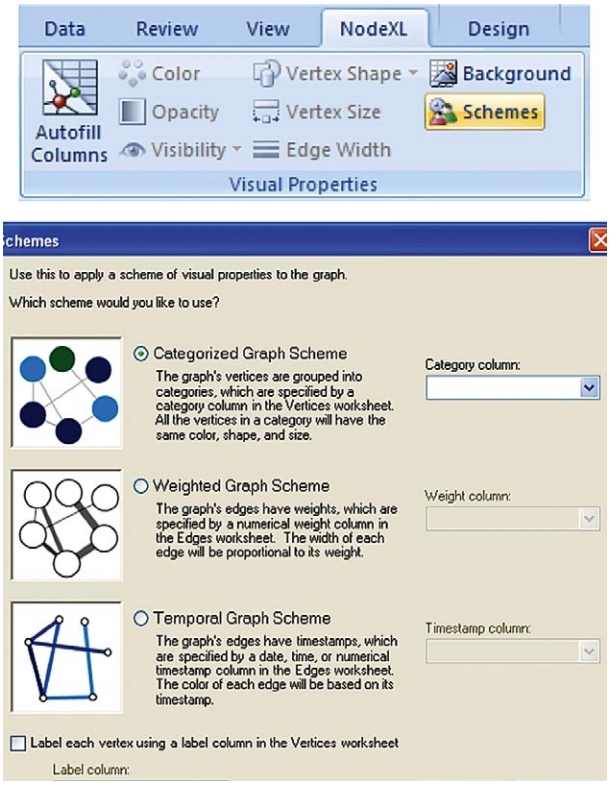


FIGURE 11.6 Detail of the NodeXL Schemes button and the Scheme dialog (which includes a categorized graph scheme).

ADVANCED TOPIC

Manually Attaching Categories to Vertex Attributes

Attaching a category to an attribute of a vertex involves two broad steps:

1. Create a table showing the category and what values it should have
2. Look at each vertex and determine which category it is in, and hence what attribute it should have.

Step 1: Creating a Category and Assigning It Attributes

First create a new worksheet. I call mine "Categories." The first column will be the category names.

Tip: If you know every category, you can just write these down one after another. If you do not know every category (e.g., Hometown), then do the following: Copy the entire column of interest from the Vertices sheet into the Categories sheet. Highlight this column. Then go to Excel's ribbon. Under the Data tab (in the Data tools section) click "Remove Duplicates."

The subsequent columns will be the style attributes. You can give the headers any name you like, but I prefer to use the same names as in the Vertices sheet: Color, Shape, Size, Opacity, and Visibility.

I have decided to use sex. This means I need three rows, one each for "Male," "Female," and "Unknown." Figure 11.7 shows the Category sheet that I have laid out. You may notice that this sheet is colored like other NodeXL worksheets rather than plain white like a generic Excel worksheet. This is a useful feature. To do this, highlight the entire sheet and go to the Home ribbon. Click on "Format as Table," choose a table style that you like and press "OK."

Step 2: Assigning Attributes to Vertices

After filling out the Categories worksheet this information needs to be assigned to the actual vertices and not just to different category values. Here you can use Excel's look-up function, "VLOOKUP." A VLOOKUP

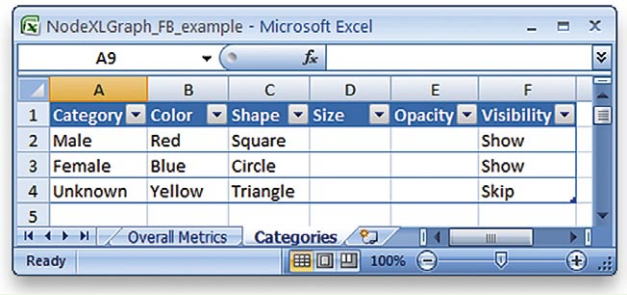


FIGURE 11.7 An example category sheet to be used as a look-up table. The top row has column label details because it has been transformed using the "Format as Table" command.

takes three values, the "value" to be looked up, the "table array" where Excel should look to find the values, and the "relative column" with the value to be returned. The VLOOKUP in this case is as follows:

```
=vlookup([#This Row],[Sex]],Table9[#All],2)
```

The table array: This is not just the first column, but the entire data array. Recall how I formatted the table earlier? This is not just for looks; this command turns the preceding table into a "smart" table. So instead of being

```
Table9[A$1:F$4]
```

(or whatever range refers to your data), it is simply

```
Table9[#All]
```

The relative column: Excel will scan the first column of the table array for the values to look up. In this case it is looking up "Male," "Female," or "Unknown." Then it needs to know which column to return. Column 1 is the first column in this table array, and column 2 is the second. Because the color values are in the second column of this table array, the relevant column is number 2. To ignore all of the vertices that do not have a value for this lookup (i.e., the people whose gender we do not know), the relevant column is number 6 instead (because visibility properties are in the sixth column):

```
=vlookup([#This Row],[Sex]],Table9[#All],6)
```

Figure 11.8 shows the Facebook network with vertices altered to show gender. I chose to show the people with missing data, but give them a hollow diamond instead of a filled in circle or square.

Finally, once you have this information in the Category sheet, it is possible to use this not only for the vertices, but also for the edges. This is a bit more complicated because

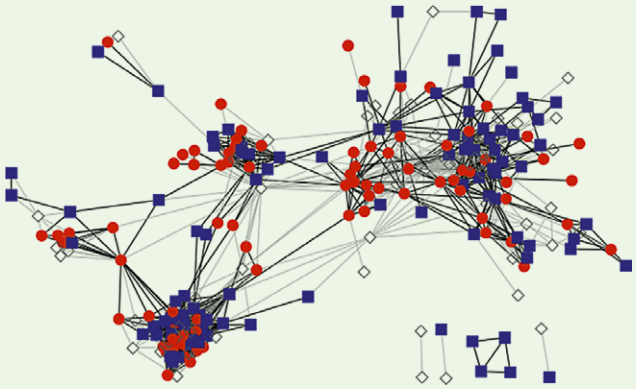


FIGURE 11.8 Author's Facebook network with vertices colored by gender using a Category table. Blue squares are male and red circles are female. Gray hollow diamonds show individuals with no gender data. Edges connected to vertices with missing data are silver and black otherwise. This shows that most of the network is not clustered by gender, although there do appear to be pockets of individuals of like type, notably a red cluster near the center and a blue component of four people near the bottom.

you have to ask if the source or target vertex for an edge has a certain value. The Excel formula takes the form:

```
=IF( OR( condition_for_source_vertex, condition_
  for_target_vertex), False_value, True_
  value)
```

My query looked as follows (bearing in mind that "sex" is the 34th column in my particular Vertex worksheet:

```
=IF( OR( VLOOKUP(Edges[[#This Row], [Vertex
  1]], Vertices[#All], 34, FALSE)="None",
  VLOOKUP(Edges[[#This Row], [Vertex 2]],
  Vertices[#All], 34, FALSE)="None"), "Silver",
  "Black")
```

Tip: It is important to match the type of visual attribute to the type of data attribute. Categories are usually mutually exclusive (one cannot have more than one country of birth) and nonordered. Using an attribute like different sizes for the categories will break this rule; why should "Catholic" be wider or taller than "Jewish"? For categories, it is best to use either different shapes or different colors.

If you are preparing a graph for black and white printing, you might be tempted to use grayscale. Be careful; people will not be able to easily distinguish more than four different tints (black, dark gray, light gray and white). To visualize categories or communities in grayscale consider the Pinwheel layout described later in this chapter.

11.7.2 Visualizing Ordered Data

People can have many friends on Facebook. Users undoubtedly are closer to some of these friends than to others. Having distant friends in one's network has prompted many critics to say, "What does a friend mean anyway?" Having lots of friends does not make the network less meaningful. In fact, having more friends can actually improve the coherence of the network since the friend-friend edges help to cluster the network and improve the calculation of metrics.

Some of these metrics can reveal a lot about the relationship between an individual and their friends. For example, Gilbert and Karahalios [6] showed that data from Facebook can be used to explain half of all the variation in the strength of friendships between any two people. They used a particularly complex model taking into account not just wall posts but the tone of words that people used as well as co-appearances in photographs and mutual interests. It is telling, however, that the strongest predictor was the number of friends in common. In our networks that measure is degree. Other measures, however, might also reveal things about the relationships between people. In analyzing my own network [7], I discovered that betweenness did a reasonably good job of picking out important people. This is because betweenness measures who lies on shortest paths between different parts of the network. Because the clusters represent different groups of friends, people of high betweenness are those who link across the different groups of friends. People who know many of my different sets of friends are typically people who know ego very well.

Metrics are all ordered numerical data. Some of them (such as betweenness) are continuous, whereas others (such as degree) are discrete. But in both cases, they can be applied to a network diagram to expose important individuals and learn about the network's overall structure.

To start, click on "graph metrics" on the NodeXL ribbon. The dialog that pops up shows a host of metrics as well as some explanation of each one. More extensive discussion of these metrics can be found in Chapter 5. However, since this Facebook network is an ego network and not a whole network, these measures take on a slightly different interpretation.

Degree

In a personal network like the ones discussed in this chapter (or more specifically a 1.5 degree network that excludes ego), degree refers to the number of mutual friends between ego and alter. This is because a 1.5-degree network only includes edges between people who know ego. So if an alter named "Bob" has a degree of 5, it means that Bob knows five people who also know ego. Bob may know other people that were not included because they are not ego's friends.

Degree is commonly mapped to size or opacity. Mapping degree to size is not always a good idea, however. In Fruchterman-Reingold layouts, it is easy to see dense pockets of individuals. Because these pockets are dense, it means that it includes many people of high degree. If you make all of these vertices very large it will make the graph hard to see. In this case, it is handy to map it to a color gradient or opacity. Make the opacity or color gradient lighter for vertices of high degree to make the dense pockets easier to discern. You can use the Autofill feature (Chapter 4) to accomplish this in NodeXL, adjusting the options such as colors or minimum and maximum opacity as desired.

Betweenness

As you may recall from Chapter 5, this is a measure of shortest paths within the network. This is one of the measures in a personal network (along with closeness centrality) that simply cannot include ego. If it did the shortest path would either be a direct edge between two vertices (A links to B) or two edges: A links to ego and ego links to B. In Freeman's classic article [8], he described betweenness as a measure of control (i.e., information flow). However, Facebook networks do not really describe the flow of information. Instead, they show paths of mutual awareness. People who went to high school are aware of each other, as are co-workers. Someone who is aware of both these groups and your family and your sports team probably knows you really well. Thus, in personal networks betweenness becomes a measure of how well alters link across different social groups.

As was done with degree, it is possible to use the Autofill Columns feature to map Vertex Size to betweenness. Betweenness is often a great measure for size because vertices of high betweenness often lie in between groups. There is one catch, however. The distribution of values for betweenness is often very skewed. That means a few people will have a very high value and most people will have a very low value.[7] There is an easy way to deal with this and a precise way. The easy way is to use the options in Autofill Columns. If you click on the upside-down triangle next to Vertex Size, you will see an options pane. At the bottom are two checkboxes: Ignore outliers and logarithmic mapping. Clicking on these boxes is a simple way to cope with the extreme skew of betweenness. You may need

[7]To rapidly get a sense of this distribution, click on the Dynamic Filter dialog. On top of the betweenness scale is a histogram that shows this skew.

ADVANCED TOPIC
Altering Betweenness Manually

Betweenness is one measure that could benefit from a log transform, but unfortunately, often betweenness is zero. Log of zero is "undefined" just like trying to divide by zero. This means it will not get included. Also, often outliers are the most interesting individuals, so checking the ignore outliers box is not ideal. Therefore, I recommend calculating size using Excel formulas.

The betweenness measure in NodeXL is standardized so that it scales from zero to one. Therefore, to use it as a size measure the easiest transform is simply to multiply it by 10:

```
=Vertices[[#This Row],[Betweenness
  Centrality]]*10
```

If you try this and discover some vertices are huge, it is because of the skew in the data. To plot this data on a log scale use Excel's LOG formula:

```
=LOG(Vertices[[#This Row],[Betweenness
  Centrality]]*10)
```

If you recall, betweenness ranges from zero to one and log(0) is undefined. Here is a slight tweak to help correct this:

```
=LOG((Vertices[[#This Row],[Betweenness
  Centrality]]*100)+1)+1
```

First multiply the betweenness value by a large number and then add 1.1. Multiplying it by a large number helps to make all the vertices larger once they are in a log scale. Adding 1.1 avoids the log(0) problem. Once that is done there might still be a problem: Size does not go from 0 to 100 but from 1 to 100. Because log(1) equals zero, I add a one to the end to make sure that every node has a value greater than one. This formula still is not ideal, however. I prefer this formula below instead:

```
=(LOG((Vertices[[#This Row],[Betweenness
  Centrality]]*100)+1.1)+1)*1.5
```

The "100" value makes all the vertices bigger, and the "1.5" value makes the difference between the high betweenness and low betweenness vertices more noticeable. The specific parameters you use for "100" and "1.5" will vary depending on your network.

to set the maximum size to smaller than 10 or use the scale slider in the graph display pane to reduce the size of vertices in case some are still too big.

The network pictured in Figure 11.9 is a culmination of the different techniques mentioned earlier. The vertex color is based on different clusters. The opacity represents eigenvector centrality (higher values are more opaque). Because eigenvector centrality is degree weighted by the degree of one's neighbors, this means that dense clusters end up being more transparent and sparse nodes become darker. Size is represented by betweenness using the formula in the advanced section (with a value of 400 in the parenthesis and 1.5 as the multiplier). Thus, someone who links many groups and knows many people will show up bigger and darker than everyone else in the network. I have also tied the shape to gender. You may notice the large yellow diamond linking different groups: this is my spouse. This is unsurprising because my partner knows many people from different parts of my life such as the orange group of family, the yellow group of undergraduate contacts, and the gray group of graduate school contacts. If you have a long-term partner (or even a best friend) who is on Facebook, it is likely that they will also show a similar trace whereby this individual links together different groups and have many connections to your friends. If you have recently started a relationship, try downloading your network at several times to watch your boyfriend or girlfriend become more central in your network (whether you like it or not!). As another example, if you have recently moved cities or started a new job, watch to see a new component appear in the margins of the network as you meet new people from your new social context. If this new component links up

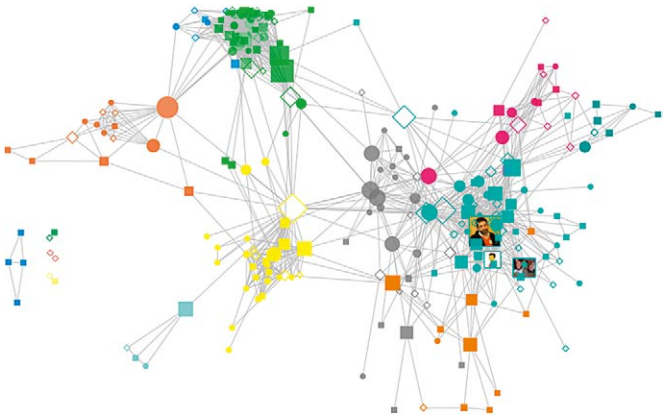


FIGURE 11.9 Complete visualization of Facebook network. Node shape corresponds to gender (circles are female, squares are male, diamonds are unknown through Facebook). Size is scaled betweenness (showing connectors). Opacity is eigenvector centrality (making dense clusters visible). Color is mapped to automatically detected clusters. Thumb-nailed images show fellow authors in this book.

with the giant component, is it because you introduced one of these people to someone from the giant component, or is it a coincidence where someone from the new city or job knows someone you already knew?

Finally, the NodeXL sheet that you download should come with the URL to the Facebook thumbnails. The last thing I did with this network is pick out some people in my Facebook network who are authors or editors in this book. For each of them I show a thumbnail rather than a shape. It is clear that they are all a part of one distinct cluster—the blue "professional contacts" cluster.

11.8 FRIEND WHEEL TO PINWHEEL: A FACEBOOK VISUALIZATION THE NODEXL WAY

One popular Facebook application for visualizing your network is a program called "Friend Wheel" (http://apps.facebook.com/friendwheel/). This program, developed by Thomas Fletcher, captures a user's friends and then groups them in some form around a wheel. By default the edges in the wheel are colored in a rainbow. It is pretty and somewhat illustrates the nature of each user's social network in Facebook. It has the advantage over a force-directed layout in that it is possible to focus on the links between groups. It is an easy network to show to others who may be intimidated by the complexity of a force-directed algorithm. Also, it scales well, so that even if you have 1000 friends, no node will overlap with another node.

It is possible to create a Friend Wheel style visualization in NodeXL. This example will not color-code the edges as is done in FriendWheel, where the nodes are colored around a spectrum (although this is technically possible to accomplish in NodeXL), but it will still cluster people together and arrange them in a colorized ring (Fig. 11.10). Much like Friend Wheel, it is based on the classic radial layout.

To create a Friend Wheel style diagram, you will have to go through a number of steps, including a little data cleaning. One of these data cleaning steps involves a formula. This is not included in the advanced section because this formula is straightforward and can be pasted directly without much fuss.

First, arrange the vertices using the Circle layout. Second, cluster the network using the "Find Clusters" button. Refresh the graph.

You will notice that vertices of different colors are next to each other and there are many lines criss-crossing the circle. This is a useful way to get an overall sense of which clusters connect well to other clusters. If you had a series of contacts from completely separate areas of life, there would be no edges crossing the middle of the network, as all the edges would be within groups.

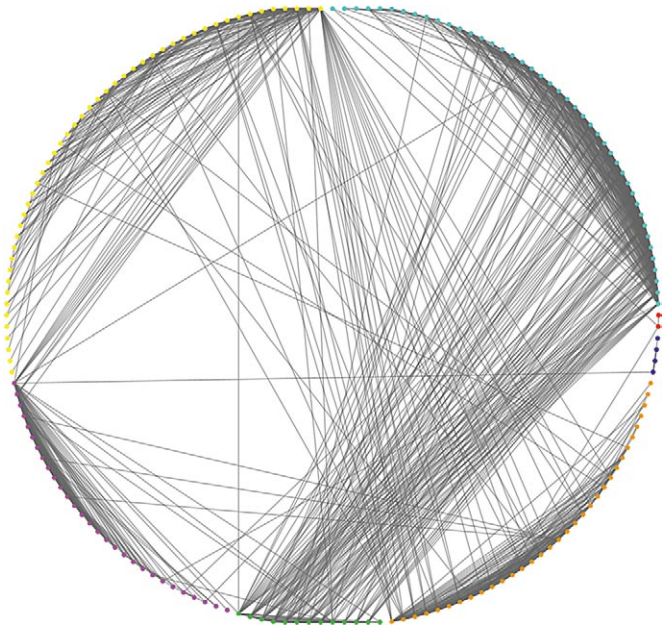


FIGURE 11.10 NodeXL radial layout of a Facebook network. The colors represent different clusters. All of the vertices within each cluster are ordered by degree. This diagram shows how well the different clusters link together. It can also be seen as sort of a pie chart for the clusters.

To group the clusters together, you need to change the order in which the vertices are laid out in the graph. To do this, edit the "layout order" column in the Vertices worksheet. This sheet is hidden by default, so first you will have to first make it unhidden. Go to the NodeXL ribbon and click on "Workbook Columns." This will drop down a box that shows several sets of columns. Select "Layout." Now when you look at the columns there will be six new columns ("Layout Order," "X," "Y," "Locked," "Polar R," and "Polar Angle"). To find the right information to put in Layout Order, you will have to look in the "Cluster Vertices" worksheet.

Unfortunately, Layout Order takes a number, and the clusters are all text—they have a C followed by digits. Therefore, you will need to clean up the "Cluster" column first. Excel only has rudimentary string cleaning facilities, but this formula will get the job done:

```
=TRIM(REPLACE(ClusterVertices[[#This
  Row],[Cluster]],1,1," "))
```

Paste this formula in the column immediately to the right of the other columns in the Cluster Vertices worksheet. What this does is replace the first character with a blank space (or white space) character. Then it trims the white space thereby leaving a number. This column can now be inserted into the Vertices worksheet. There are two ways to move the column: a VLOOKUP statement and copy and paste. Here I will use copy and paste, but

first I have to make sure I've copied and pasted in the right order:

Step 1. Sort the Vertices worksheet by the first column (Vertex). Check the last row number.

Step 2. Sort the Cluster Vertices worksheet by the second column (Vertex). Check the last row number. It should be one less than the number on the Vertices worksheet, because the Vertices worksheet has two header rows whereas the Cluster Vertices worksheet only has one header row.

Step 3. Copy the entire column of numerical values from Cluster Vertices worksheet.

Step 4. Right-click on the first cell under "Layout Order" and select "Paste special." Here you should paste the values rather than the contents of the cell.

Step 5. Refresh the graph and marvel at a rainbow of networkiness.

Because the network is bunched together you can tell by the edges which clusters link closely to each other (notice how in my example network the purple and blue clusters have a thick line crossing between them; this makes sense as the blue cluster is mainly professional contacts and the purple cluster is mainly Oxford contacts). To make these relationships even more obvious, I tweak the edges as follows: All edges colored "Gray," with edge widths of 1.3 instead of the default 1.0 and an opacity of 60 rather than the default 100.

ADVANCED TOPIC
Turn That Friend Wheel into a Fireball

The Friend Wheel is a nice layout. It looks fancy and gives a bit of information about the underlying network structure. However, there is much more that one can do with this layout to help accentuate the underlying structure and thereby make the graph more readable (Fig. 11.11). This layout assumes you are looking at the Friend Wheel displayed earlier.

Step 1. Reorder Vertices within the Clusters

Within clusters the vertices are not presently ordered in any way. To reorder them, you have to alter the layout order. In this case, I suggest ordering the vertices by degree, so that the most connected vertices will all move to one side of the cluster.

Create a new column in the Vertices worksheet called "Cluster" and paste the layout order values there (not any formulas, but the numerical values). Then in Layout Order insert a formula that includes this value plus a new per-cluster ordering:

```
=(Vertices[[#This Row],
  [Cluster]]*COUNTA([Vertex]))+ Vertices[[#This
  Row],[Degree]]
```

Refresh the graph. The graph will look slightly different with more edges from each cluster weighted to one side of the graph. You can try different values but remember two things: (1) the numerical range *within* the cluster has to be smaller than the interval *between* clusters or else you will have overlapping clusters, and (2) layout order only goes to one decimal place, so all of the measures that go from zero to one have to be multiplied by some factor.

Step 2. Convert a Circle Layout to a Polar Layout

To tweak the layout any further this graph has to move from a Circle layout to a Polar layout. The Polar layout arranges the vertices using "degree" (in the coordinate sense of angle, not in the graph metric sense of number of connections), going from 0° to 360° and a radius going from 0 to 1, where 1 is at the edge of the screen. The columns where you insert these values (Polar Angle and Polar R, respectively) should have been revealed when you revealed the Layout Order column.

The Polar R value is easy—its 1. Simply paste 1 into every column.

The Polar Angle value is a little more complicated but it can be calculated using the current X and Y coordinates. These coordinates have to be copied into new columns first because the original X and Y coordinates will change the moment you try to redraw the graph. Create two new columns to the right of the current columns. Remember to right-click → "Insert Table Column to the Right" so that they have the blue headers. Then highlight the X and Y columns and paste them into the new columns. NodeXL is smart enough to rename these columns to X2 and Y3 rather than have two columns named X and two named Y.

Once the X and Y values are copied, it is possible to calculate a degree to place in the Polar Angle column. In the Polar Angle column use this formula:

```
=DEGREES(ATAN2(Vertices[[#This Row],[X2]]-
  5000,Vertices[[#This Row],[Y3]]-5000))
```

(*Continued*)

ADVANCED TOPIC *(Continued)*

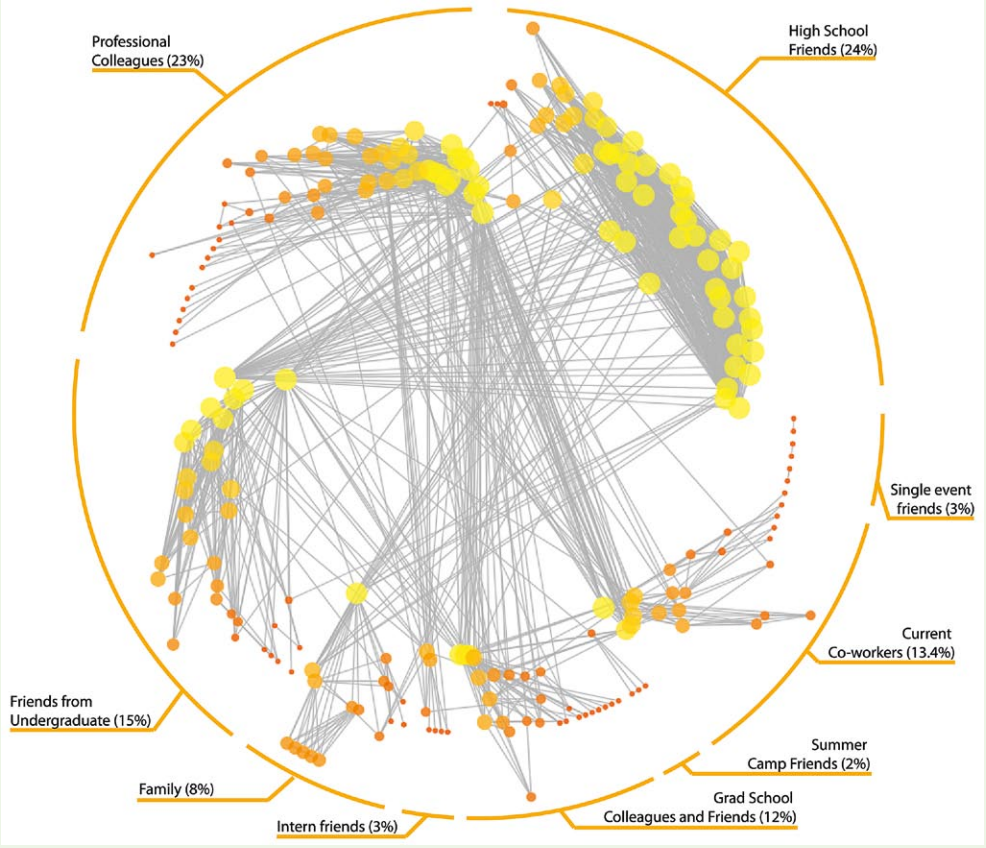


FIGURE 11.11 NodeXL Pinwheel layout. This layout is an extension of the Friend Wheel layout (or clustered Radial layout). The radius is scaled to betweenness and the other attributes (size, color, and ordering are mapped to degree). This layout gives extra information in the cluster relative to the radial layout. One can more clearly see if high-degree nodes link to the rest of the cluster or to other clusters. One can see if low-degree nodes link outside the cluster as well. And clusters are still distinct despite the fact that they all use the same color gradient. The ring around the network and the labels were added in Illustrator.

Because X and Y go from 0 to 10,000, subtract 5000 so that the origin (the midpoint) is in the center of the screen. Now choose the Polar layout and, presto, it should look exactly the same as in the Circle layout.

Step 3. Turn a Ring into a Series of Flames

The last step was the hard part. This is the fun part. Simply use Autofill Columns to set a series of display attributes. I recommend selecting one new variable at a time and then pressing Autofill to watch the network change.

Once it is in the Polar layout it is possible to vary the radius. Remember that betweenness is a measure that indicates how much people link across different clusters. Therefore, this value can be used to make vertices of high betweenness closer to the center, as if their edges are pulling them inward to other clusters. When you select this value for Autofill Columns, you will have to tweak some options. Those of the smallest betweenness should be a 1, meaning they are out toward the edge of the ring. Those of the highest betweenness should be near the center, but not too near. Enter a 0.5 for this value. Refresh the graph.

Now that the pinwheel style graph has spread out the vertices a little more, you can tweak the look of the vertices. I recommend using degree for both the size and the color. Remember that these pinwheels are ordered so that the highest degree is at the front and the lowest degree is at the rear. Therefore, using degree for other values reinforces this ordering rather than confuses the viewer with additional metrics. I choose red and yellow to give it a flaming look. Blue and white on a black background will also look nice. I also use degree for the size of the vertices so that each flame tapers inward from head to tail. Finally, I set the opacity to 60 and tweak the Scale slider until I find an overall shape that I like.

11.9 PRACTITIONER'S SUMMARY

This chapter has covered the process of visualizing and interpreting a personal network on Facebook. Facebook friend networks reveal the social worlds relevant to any given individual. People from the different areas of a person's life tend to cluster together (since these are the people most likely to befriend each other). After downloading an edge list or NodeXL file of these relationships, one can start to interpret this network.

This often involves the application of unordered data such as clusters and categories as well as ordered data (such as wall posts or mutual friends). In both cases, there is an easy way to do this (Category schemes and Autofill Columns) and a hard way to do this involving authoring Excel formulas. As is to be expected, the easy way is useful and clever but limited. The harder way is more involved but gives the user more control. This chapter walked through two general network layouts: force directed and circular. In both cases, it is possible to extensively tweak the default settings to reveal interesting patterns. The resulting force-directed layout can show clusters and categories as well as structurally important people. This is useful for learning to what extent groups intersect. The pinwheel layout starts with categories rather than ends with them. This layout reveals the density of connections within clusters as well as their relationship to other clusters (i.e., whether these links emanate primarily from a single/few individuals or from the cluster in general). The pinwheel also acts as a sort of pie chart with network connections.

In doing so, my goal was to show that although the learning curve for NodeXL may not be steep, nor is it shallow. NodeXL is built on a powerful and clever spreadsheet program. By leveraging features of Excel 2007 such as "Format as Table," "Remove Duplicates," multisheet formulas, and functions like LOG, ATAN2, and VLOOKUP, NodeXL becomes as flexible as it is usable. Many of these features were put into operation in a novel visualization demonstrated here called a pinwheel. This visualization combines an arrangement like a pie chart with features of other network layouts.

The charm in Facebook networks is not just learning about network features that surprise you, but also learning how to read networks by exploring one that you know best, namely your own.

11.10 RESEARCHER'S AGENDA

Clustering is of paramount importance in a Facebook ego network. Different algorithms will lead to different interpretations of what counts as an appropriate cluster. Post hoc interpretations of these clusters may convince the owner of the network that either cluster is meaningful. However, it remains to be seen at what scale clustering is useful. Is it useful to know who co-workers are and how they are connected to high school friends, or is it useful to know the small subset of co-workers that also play on a sports team and emigrated from India? Additionally, the area of soft partitioning (where people can exist in multiple categories) and longitudinal partitioning (where clusters shift over time) are active areas of research [9].

Beyond the work in better cluster algorithms is work on the psychology of social networks. In general, the size of the network that individuals can keep in their head appears to be somewhat smaller than many Facebook networks. Dunbar has suggested that cognitive limits on human ability to maintain active connections (and their interrelationships) at a ceiling of about 150 [10, 2]. However both he and colleagues are mum on the consequence of these new sites. Is Facebook a cognitive enhancement for our age or an inevitable harbinger of social information overload? We do not know yet. Indeed, the answer may change with new visualizations such as the ones shown here.

References

[1] d. boyd, Friends, Friendsters and Top 8: Writing Community into being on social network sites. First Monday 11 (12) (2006).
[2] S.G.B. Roberts, R.I.M. Dunbar, T.V. Pollet, T. Kuppens, Exploring variation in active network size: Constraints and ego characteristics, Social Networks 31 (2) (2009) 138–146.
[3] K. Lewis, J. Kaufman, M. Gonzalez, A. Wimmer, N. Christakis, Taste, Ties and Time: A New Social Network Data Set using Facebook.com, Social Networks 30 (4) (2008) 330–342.
[4] R. Burt, Brokerage and Closure, Oxford University Press, Oxford, UK, 2005.
[5] B. Hogan, J.-A. Carrasco, B. Wellman, Visualizing Personal Networks: Working with Participant-Aided Sociograms, Field Methods 19 (2) (2007) 116–144.
[6] E. Gilbert, K. Karahalios, Predicting Tie Strength with Social Media. CHI '09: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM Press, New York, 2009, pp. 211–220.
[7] B. Hogan, A Comparison of On and Offline Networks through the Facebook API. QMSS2: Communication Networks on the Web, Amsterdam, 2008.
[8] L. Freeman, Centrality in Social Networks: A Conceptual Clarification, Social Networks 1 (3) (1979) 215–239.
[9] M.A. Porter, J.-P. Onnella, P.J. Mucha, Communities in Networks, Notices of the AMS 56 (9) (2009) 1082–1097.
[10] R.I.M. Dunbar, The Social Brain Hypothesis, Evolutionary Anthropology: Issues, News, and Reviews 6 (5) (1998) 178–190.
[11] d. boyd, N. Ellison, Social Network Sites: Definition, History, and Scholarship, Journal of Computer-Mediated Communication 13 (1) (2007).
[12] C. Shapiro, H. Varian, Information Rules, Harvard Business School, Cambridge, MA, 1998.

Additional Resources

Hogan, B. (2008). Analyzing social networks via the Internet. In N. Fielding, R. Lee, & G. Blank, *The sage handbook of online research methods* (pp. 141–160). Sage, Thousand Oaks, CA.

CHAPTER

12

WWW Hyperlink Networks[1]

Robert Ackland

Australian Demographic and Social Research Institute, Australian National University, Canberra, Australia

OUTLINE

12.1 Introduction 181

12.2 Hyperlink Networks 182
12.2.1 Theory of Hyperlinking 182
12.2.2 Methodological Issues 183

12.3 The VOSON Data Provider 185

12.4 Example 1: Who Links to My Organization's Web Site? 186
12.4.1 Preliminaries 186
12.4.2 Constructing Your Own Hyperlink Network Using the VOSON Data Provider 187

12.5 Example 2: What Is the Hyperlink Network of this Field/Industry/Sector? 194

12.6 Blogs, Temporal Changes, and Network Flow 196

12.7 Practitioner's Summary 197

12.8 Researcher's Agenda 198

12.1 INTRODUCTION

The largest machine-readable network graph on earth may be the structure created when web pages link to one another. The tens of billions of pages on the World Wide Web can each be seen as a vertex in a graph whose edges are URL hyperlinks connecting one page to another. These hyperlinks may point to other pages on the same web site or to sites of other organizations, companies, institutions, and nations. The result is literally a web of connections that is often mentioned but rarely seen. The network of related web sites, although based on basic "Web 1.0" technologies, represents important social, economic, and institutional relationships that can reveal significant insights for researchers, policy makers, and corporate strategists. Additionally, many Web 2.0 technologies such as blogs, wikis, and social sharing sites such as Delicious include hyperlink networks that can be captured using web site crawling tools. These networks are now more accessible than ever before for collecting, analyzing, and visualizing.

For most companies or institutions, the primary medium for building a web presence is still the venerable Web 1.0 web site. Organizational web sites now often have Web 2.0 features, incorporating, for example, a blog site or other features for customer engagement, but the mainstay of the typical organizational web site are features that are hallmarks of the Web 1.0 era: text content (including meta tags) describing the organization ("who we are") and hyperlinks to other parts of the web site ("internal" links) and to pages outside of the web site ("external" links).

This chapter is about quantifying and understanding an organization's web presence as constructed via connections within and beyond the web sites: we refer to this variously as organizational/Web 1.0/static

[1]I would like to thank Lin Chen for assisting in the development of the VOSON Data Provider for NodeXL, and Susannah Sabine for research assistance.

web presence. The use of the term "static web presence" refers to the fact that such web sites are typically not designed to be regularly updated (unlike a blog), and in particular, there isn't directly created user- or consumer-generated content. For most organizations, their Web 1.0 web presence is not everything, but simply a component (we argue, still incredibly important) of overall organizational web presence. Other components of an organization's overall web presence (e.g., discussion forums, Twitter feeds, Facebook, Flickr, and YouTube) are increasingly important and are addressed in Chapters 9, 10, 11, 13, and 14, respectively. Unlike the other chapters in this volume that focus on networks that link people to other people, this chapter focuses on relationships between organizations and institutions. Other sources of network data exist on the World Wide Web such as semantic networks of related concepts, but this chapter focuses on the lessons that can be learned from exploring the hyperlink network alone. More advanced approaches integrate these various sources of network data.

Managers of institutional web sites need to know a great deal about their sites. Does the look of the site match corporate brand presence? Does the structure of the site make it easy for customers to find the information they need? What are the traffic metrics? (Which pages do people go to? How long do they stay? Where do they come from?) Web site analytics tools, such as Google Analytics, provide a significant amount of this information and are invaluable to web designers and marketers. However, current versions do not adequately address where your web site sits within the larger constellation of web sites relevant to the organization. For example, analytic tools typically list which sites link your web site but not how the linking sites are connected to one another.

We all know the business mantra "If you can't measure it, you can't improve it," but up until now, it has been difficult for the typical organization wanting to understand its web presence to collect or represent hyperlink networks in a simple manner. This chapter demonstrates new, simple tools and methods that transform hyperlink network analysis into a process that is similar to making a chart in Excel. This chapter is not about search engine optimization (ensuring that the web site is accessible to the Google web crawler[2]) and it is certainly not about setting up link farms or other questionable techniques to improve Google or other search engine rankings. It is about using technology to enable organizations to quantitatively map their online presence and gain actionable information on improving their web presence.

Collecting and analyzing hyperlink network data allow you to answer the following types of questions: How does your organizational web presence compare with your "offline" brand presence? (Does your corporate web site command the position in cyberspace that matches your position in the industry?) If your company has several brands or divisions, how does the web presence of each compare? (Does the web presence of each brand or division reflect its marketing budget allocation?) How has your organizational web presence changed over time (e.g., was a marketing initiative successful in getting new sites to link to yours)? Hyperlink data can also be used to unobtrusively and ethically map and monitor competitive business positions, providing actionable information on the competitive online landscape in which your business operates: How does your organizational web presence compare with that of your competitors (e.g., who is connected to them that you may want to develop relationships with)? Hyperlink data can also help you identify new competitors. If you are entering a new market, these data can give you a sense of who the current players are and how they are related.

12.2 HYPERLINK NETWORKS

Hyperlinks are regarded as the "essence" of the web (e.g., [1]); they are what make the WWW a web, a network of connected pages linked by URLs that point at other pages. Hyperlinks are a significant kind of currency. How many times have you been happy to find that another web site links to your site or disappointed when looking through a list of "important links" on another web site and found that your site is not featured? Although hyperlinks are recognized as important, they are not well understood and until recently they could not be easily explored more than one link at a time. This section outlines the theoretical and methodological challenges of organizational hyperlink analysis. The remainder of the chapter provides a step-by-step guide to analyzing an organization's web site network and neighborhood using NodeXL.

12.2.1 Theory of Hyperlinking

On one level, the web can be conceived of as a vast electronic library of hyperlinked documents. However, Jackson [1] noted that "Once we become critical of the assumption that the Web is a neutral repository of information, the structure of the Web becomes much more

[2]A web crawler is a piece of software that automatically browses a web site, following internal links deeper into the site and external links outward to other sites, downloading each page that it encounters.

interesting." Hyperlink networks serve important organizational functions. A hyperlink network is not simply a network in a mechanical or metaphorical sense, but an organizational network that is amenable to the application of formal social network analysis (SNA) techniques. However, in order to regard a hyperlink as a tie directed from one organization to another, we first need to say something about what motivates an actor to create a hyperlink to another page or site and the benefits of receiving a hyperlink.

There are many potential reasons for an organization to create a hyperlink to another organization; as Thelwall [2] noted, there is no single theory of hyperlinking. Hyperlinks can confer authority or endorsement [3] and reflect trust [4]. Of course, sites can also link to other sites when criticizing them, and thus a hyperlink can reflect a "negative affect" relation. Rogers and Marres [5] emphasized the role of hyperlinks in organizational communicative and strategic behavior and Park, Kim, and Barnett [6] focused on their role in organizational alliance building and message amplification, helping to create a sense of "critical mass" for a particular message or viewpoint.

Receiving hyperlinks from other sites is important because this can convey trust and can help drive traffic to your site. Inbound links are important for driving traffic to your site for two primary reasons. First, the more inbound hyperlinks from other relevant web sites, the greater the number of "pathways" that people who matter to your organization (e.g., customers) can follow to your site. Second, the number of relevant inbound hyperlinks is a primary determinant of a site's ranking on search engines such as Google (e.g., [7]). As Hindman et al. [7] noted, web site *retrievability* is an absolute concept: at a given point in time, the content on my personal web site is either retrievable (if the site is operating) or not retrievable (if the site is not operating) and will be just as retrievable for the marginal viewer as that of Microsoft's corporate web site (assuming both sites are operational). Web site *visibility,* however, is relative, and is largely determined by the number of inbound hyperlinks from other relevant web sites, particularly highly ranked ones.

A larger number of relevant inbound hyperlinks will generally mean more eyeballs viewing your web site content and for most organizations, this will be regarded as a positive thing. However, the important word in this sentence is "relevant." It's not just the raw quantity of inbound hyperlinks that is important for your organization's web visibility, but who they are from. Most organizations don't simply want to know how many hyperlinks point to them; they want to know *who* links to them and how important these web sites are in the network. In short, they want to construct a complete network that is relevant to their organization and to monitor it over time.

But how does one even start this process? We can distinguish two attributes of the "linker" that will be important. First, there are the graph-theoretical attributes—that is, what is the position in the web network graph of the site that is linking to your site? In terms of web visibility it will be important to garner inbound links from other sites that are themselves visible or prominent within the overall hyperlink network. As explained in Chapters 3 and 5, there are many measures of network centrality that can be computed. For example, it might be important for you to be linked to by a site that itself has many inbound links (high in-degree centrality) or one with links to others who have high in-degree centrality (eigenvector centrality).

Second, there are the nongraph theoretical attributes of sites directing (or not directing) hyperlinks to your web site. For example, you might be aware of particular web sites in your industry or field that are highly important for you to be linked to by, regardless of their network position in the web network graph.

Even for those who are aware of the importance of network structures and their organization's location within them, the challenge has been a practical one: How does one get the data and process it in a simple and end-user manageable way. Now that we have sketched out a conceptual framework for hyperlinking, we turn to the remaining methodological challenges of collecting hyperlink data and computing network measures of web visibility. Once created, these maps of local web network spaces can then be interpreted to provide actionable insights into how to improve web visibility.

12.2.2 Methodological Issues

There are three fundamental and interrelated methodological questions that need to be answered in analyzing any social network: What are the nodes or vertices, what are the network ties or edges, and what are the boundaries to the network?[3] In the context of hyperlink networks, although each of these aspects will be informed by the particular research question that is being posed, the following are some general points.

With regard to the nodes in a hyperlink network, the situation is more complicated than with offline social networks and even other types of online social networks. In a friendship network, the nodes are obviously people. Identifying the nodes in a Twitter network (Chapter 10)

[3]See Lusher and Ackland [8] for a detailed discussion of methodological issues associated with conducting statistical SNA with hyperlink network data.

and a Facebook network (Chapter 11) is similarly fairly straightforward, as each user account in these social media environments is typically associated with an individual. However, Web 1.0 hyperlink networks may be populated by nodes that are not homogenous in type. For example, it is easy to construct a hyperlink network where nodes represent organizations such as universities, government departments, companies, and nongovernment organizations (NGOs)—these are the web sites of entities that have an offline or real-world presence. But additionally, there may be nodes that represent entities that have no offline presence (i.e., they only exist on the web). Examples are portals (sites that provide organized lists of links to other sites), informational sites (sites that aim to provide commentary or insight on a particular topic and links to relevant resources), online businesses (that have no offline counterpart), web sites operated by individual people, vanity web sites (web sites that have been set up to promote a brand or movie), and the popular special form of web pages known as blogs.

Your decision as to what types of sites to include in your network is dependent on the type of analysis that you are conducting. In some situations, it will make sense to prune the network so that it only contains nodes representing organizations that exist offline. In other situations, it may be desirable for the hyperlink network to be an accurate representation of what exists on the web (i.e., to include all web sites found by the web crawler, regardless of their type).

Let's assume that you want to construct a hyperlink network that is relevant to your organization and you have decided that you will only include those nodes that represent other organizations that have an offline or real-world presence (i.e., you are going to prune sites such as portals, informational sites, online businesses, etc.). Even in this simplified situation, you will be confronted with further complications that are particular to hyperlink network analysis.

Ideally, the nodes in your organizational hyperlink network will represent the entire (Web 1.0) web presence of each organization; nodes will represent web sites, or parts of web sites, rather than individual web pages. Hence, the hyperlink network will need to be populated with meta-nodes representing clusters containing the entire web presence of each organization in the network, rather than dozens or perhaps hundreds or thousands of nodes reflecting the individual pages in each site. This can be achieved algorithmically by grouping together all pages from a particular host name. However, sometimes an organization's web presence is reflected in multiple host names. For example, a single organization may have different domain names (www.mycompany.com, www.mycompany.com.au), different subdomains (brand1.mycompany.com, brand2.mycompany.com), or different subsites (e.g., www.mycompany.com/brand1, www.mycompany.com/brand2).[4] To accurately measure the web presence of the entire organization, all the identified pages from these domains, subdomains, and subsites will need to be properly grouped together. Although a lot of this processing can be done automatically, there will typically still need to be some input from the analyst in order to accurately identify the nodes in the hyperlink network that should be grouped together.

The second methodological question to answer before conducting hyperlink network analysis is: what are the network ties?[5] Assume the simplest case of two organizations whose web presence is reflected in single web sites. Even in this simple situation there are several possibilities as to what could constitute a network tie. If the web site of organization A contains a hyperlink to the web site of organization B, then this could be interpreted as a network tie and it would lead to creating a directed edge in the hyperlink network. But you might be more interested in only recording a network tie if the hyperlinks are reciprocated (A links to B, and B links to A), leading to an undirected, symmetric hyperlink network. Finally, you might want to attach values or weights to the network ties reflecting, for example, the number of hyperlinks directed from A to B (one could argue that more hyperlinks reflect a stronger network relationship) or the depth in the web site where the hyperlink was embedded (one could argue that a hyperlink from the homepage of A has more significance than if it is buried deep within the web site).[6]

[4]Subsites can cause a further problem: it may be that two or more organizations in your hyperlink network have web sites that are commercially hosted (e.g., by Geocities), and in such a situation, you need to be careful that these web sites are not merged into a single node (e.g.,www.geocities.com).

[5]The following discussion focuses on collecting and analyzing hyperlinks *between* rather than *within* web sites (i.e., the focus is on external rather than internal hyperlinks). In many situations, internal hyperlinks are only used so that the web crawler can find additional pages within the site where there might be external hyperlinks. However, internal hyperlinks may have analytic value in some situations (e.g., a government department may want to know the connections between divisions within the organizational web site). Also, the example of the Wikipedia network presented in Chapter 15 involves internal hyperlinks between different Wikipedia pages.

[6]Similarly, it may be that a link directed to a page that is deep within the web site is valued more highly than a generic link to a top-level page.

Because a hyperlink can mean different things, it is important that hyperlink networks be constructed such that each hyperlink can be interpreted in the same way. A working assumption here is that there is a certain level of automation in the data collection process: analysts won't be able to go to each page where there is a hyperlink and check the page content to ascertain the context or meaning of the hyperlink. In the absence of manual checks into the context or meaning of a hyperlink, a straightforward way of ensuring this consistency in the meaning of hyperlinks is to ensure that the network actors are homogenous. Lusher and Ackland [8] constructed a network of web sites of organizations who advocate on behalf of refugee and asylum seekers in Australia. The fact that the network was restricted to these types of actors meant that the authors could interpret the existence of a hyperlink using a collective action framework (i.e., the actors were hyperlinking to other organizations that shared their concern for the plight of refugees and asylum seekers in Australia and were working toward this cause). Lusher and Ackland purposefully excluded government sites from the network because the existence of a hyperlink from a refugee advocacy site to a government site could reflect criticism of government policy, and the selection of nodes thus ensured that hyperlinks were likely to represent positive affect.

The final methodological question to address is, what are the boundaries to the hyperlink network? Network boundaries will often be settled by choice of type of node, but a complication with constructing hyperlink networks (compared with offline networks) is that it might not be possible to use familiar boundaries like geography to define the limits of a web hyperlink network. In an offline friendship network, the boundary might be the school (or classroom) (i.e., if a student has a friendship with someone outside the school, then this friendship will not be counted). However, such geographical or physical boundaries do not necessarily exist or make sense in the "borderless" terrain of the web.

In some situations, geography might be helpful in determining hyperlink network boundaries. Lusher and Ackland [8], for example, only included the web sites of Australian-based organizations who are involved in refugee and asylum seeker advocacy. However, even in such a situation where an obvious network boundary can be used, the analyst may still be faced with the problem of discovering all of the units of observation (nodes) that exist within the boundaries and (if necessary) drawing an appropriate sample of nodes. Lusher and Ackland used a snowball sampling approach to discover relevant web sites—starting with an initial list of known relevant sites and then crawling these in an attempt to find additional relevant sites—but unless the analyst has available a sampling frame, this process is problematic as you will never know if you have included all relevant web sites into the hyperlink network.[7]

12.3 THE VOSON DATA PROVIDER

The VOSON Data Provider is a NodeXL data import plug-in for the collection and analysis of hyperlink network data. It provides a front-end to a data collection and processing service that is part of the Virtual Observatory for the Study of Online Networks (VOSON) System [8–10].[8] This data collection and processing service consists of a web crawler that crawls "seed sites" nominated by the user, extracting hyperlinks to other sites. The service also finds (via a query to the Yahoo! Application Programming Interface, API[9]) inbound hyperlinks to a given seed site.

There are other tools for collecting and analyzing hyperlink data that are available, most notably (because they are freely available and used quite extensively in the social sciences) Issuecrawler[10] (e.g., [11]) and LexiURL Searcher/SocSciBot[11] (e.g., [12]). What makes the combination of VOSON and NodeXL different is that it is a specialized hyperlink research service (VOSON) available as a plug-in service within a client-side network analysis and visualization software framework (NodeXL). The result is a remarkable ease of use for nonprogrammers to engage in complex data

[7]This problem is obviously compounded if you are including sites that do not necessarily have an offline presence into the network.
[8]The VOSON System is a web-based e-research tool for the collection and analysis of hyperlink networks, which was initially developed under an Australian research Council–funded grant titled "New Methods for Researching the Existence and Impact of Political Networks on the WWW" (Robert Ackland and Rachel Gibson). The VOSON System has been used for studying the hyperlink networks formed by political parties [9], the nanotechnology industry [10], and advocates of refugee and asylum seekers in Australia [8]. See the VOSON Project web site (http://voson.anu.edu.au.) for more details.
[9]http://developer.yahoo.com
[10]www.issuecrawler.net
[11]http://lexiurl.wlv.ac.uk

collection, scrubbing, analysis, visualization, and interpretation tasks.

Other chapters in this volume provide examples of the NodeXL client creating a data connection to a social media service via an API. These other services (YouTube, Flickr, Facebook, Twitter) are themselves social media platforms, whereas VOSON is a research service that enables research into hyperlink networks. VOSON itself is not a social media platform but a tool for gathering information about other web sites and connections. VOSON is part of an emerging trend toward "boutique" services aimed at mapping selective parts of cyberspace (Spinn3r[12] for the blogosphere is another example of this type of service).

Integrating VOSON with NodeXL reflects a growing demand for independent, specialized, and distributed research resources being made available to end users via web services and a client platform (in this case, NodeXL). This is what was envisaged by the cyber infrastructure program in the United States and related e-Science programs in the United Kingdom, Europe, Australia, and elsewhere (e.g., [13]).

12.4 EXAMPLE 1: WHO LINKS TO MY ORGANIZATION'S WEB SITE?

There are two basic ways to create a hyperlink network using the VOSON Data Provider. Either you already have a set of seed sites (e.g., the URLs of brands that are owned by your company, or the URLs of the corporate web sites of your company or competitor's web sites) or you have a term or phrase (or a series of these) that describes the field, industry, or sector that you want to map. This section illustrates the former approach (we will use a single seed site, the VOSON Project site, http://voson.anu.edu.au), whereas the next section shows how to map the hyperlink network of a field, industry, or sector.

Typically, you know whom you link to via your organization's web site; you can find this information by simply looking through the web site and noting the external hyperlinks. (*Note:* This might not be easy to establish with very large web sites, but let's assume your web site is reasonably small.) But do you know who links to your organization's web site? This exercise is about constructing a hyperlink network map that is relevant to your organization; in social network analytic terms, we would describe it as an ego network, where your organization is the "ego" and you identify your "alters," the sites that are connected to your site, and the connections between them.

[12]www.spinn3r.com

12.4.1 Preliminaries

The most recent version of VOSON Data Provider is available from www.uberlinkcorp.com. The plug-in consists of a single dynamic link library (DLL) file, which will be named NodeXL_VOSON_Spigot_x.dll (where the "x" refers to the version number). To install the VOSON Data Provider, you simply need to download the DLL file and place it in the following folder (assuming you have installed Excel 2007 on your C drive): "c:\Program Files\Microsoft Research\Microsoft NodeXL Excel Template\PlugIns." You will then need to close down NodeXL (if it was running) and restart it.

Selecting the NodeXL Data menu item "Import → From Web 1.0/Blog Network (via VOSON)" will bring up the VOSON login dialog (Fig. 12.1). If you don't already have a VOSON user account, then you need to apply for an account (see http://www.uberlinkcorp.com).

Once you have your VOSON user account details (username and password), enter them in the login dialog and press the button labeled "Log on" (*Note*: the VOSON Data Provider for NodeXL will only work if you are connected to the Internet). A successful login to the VOSON System will result in the "Select existing or build new VOSON network" dialog being shown (Fig. 12.2). The select network dialog provides a listing of all the VOSON databases that are available to you; these are stored on the VOSON server (if you have just created a VOSON user account, you will only have access to tutorial or test databases).

The "Select existing or build new VOSON network" dialog in Figure 12.2 lists a single VOSON database, with the following metadata:

- *Name.* The name of the database. The tutorial database is called "testdbAN."



FIGURE 12.1 NodeXL Data Import "VOSON login" dialog.

- *Project.* This refers to a collection of users who can share data with one another. By default, you will belong to two projects: tutorial (all users belong to this) and a project named after your username (only you will belong to this project). The testdbAN database belongs to the tutorial project.
- *Size.* This is the number of nodes (web pages) in the network.
- *Date created.* This specifies when the database was created.
- *Comment.* Users can optionally enter a comment when a VOSON database is created (for example, giving an explanation of what types of seed web sites the database contains).

The select network dialog has the following six buttons:

- *About.* Press this button to get information on the VOSON Data Provider for NodeXL.
- *Refresh.* Press this button to get an update of the VOSON databases that are available and their metadata.
- *Import network.* Press this button to import the selected network into NodeXL. This button will be disabled (grayed out) unless you have selected a VOSON database from this dialog.
- *Build new.* Press this button to build a new VOSON database (discussed in detail later).
- *Delete.* Press this button to permanently delete the selected VOSON database.
- *Add seed sites.* Press this button to add seed sites to your VOSON database (discussed in detail later). This button will be disabled unless you have selected a VOSON database.

12.4.2 Constructing Your Own Hyperlink Network Using the VOSON Data Provider

There are four steps involved with constructing your own hyperlink network using the VOSON Data Provider:

- create the new VOSON database (including setting the parameters for the web crawl)
- input seed sites into the VOSON database
- import the VOSON database into NodeXL (after the crawler has run)
- (optionally) add more seed sites

Pressing on the "Build new" button in the select network dialog will bring up the "Create VOSON database" dialog (Fig. 12.3).

There are four main parameters or features that you can set via this dialog:

- *Database name.* Choose a name for your database. Note that your database name should only contain

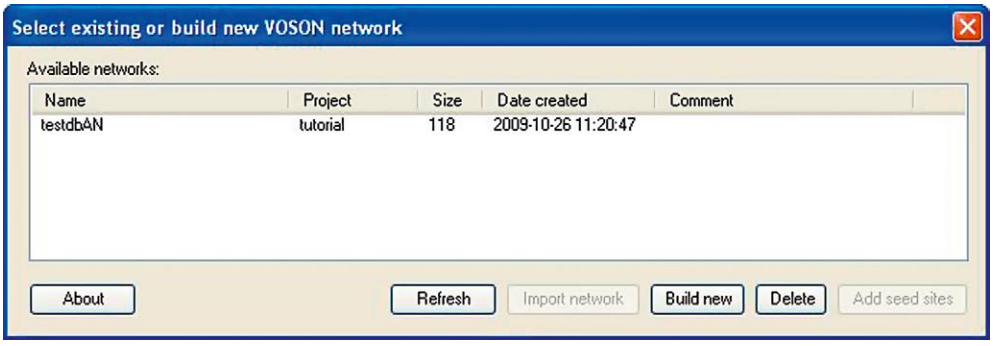


FIGURE 12.2 NodeXL "Select existing or build new VOSON network" dialog.

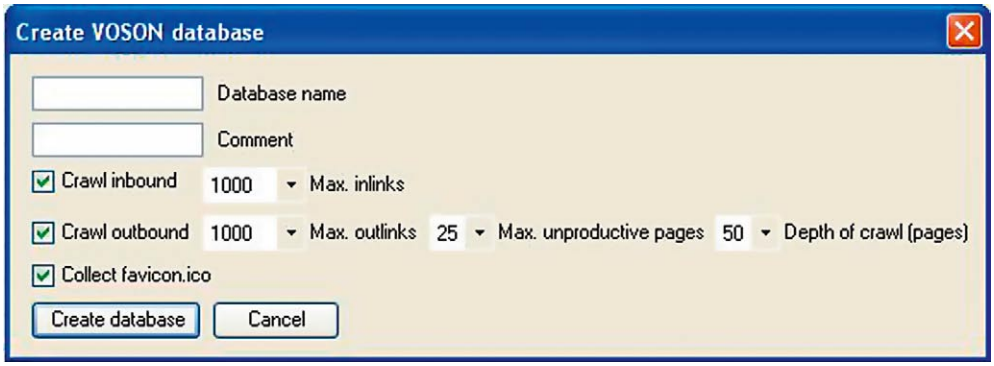


FIGURE 12.3 NodeXL "Create VOSON database" dialog.

letters from the alphabet and/or numbers. Database names are case sensitive and cannot include an underscore "_" or space.
- *Comment*. Optionally enter a comment describing the new database
- *Parameters for inbound crawl*. If you check the "Crawl inbound" checkbox, the VOSON System will find what other sites *link to* the seed sites you have entered into your database (entering seed sites is discussed below) (i.e., it will find inbound hyperlinks). If you choose to "crawl inbound," then you have the option to determine the upper limit on the number of inbound hyperlinks found for each seed site before the crawler stops working on that seed site. You can set the crawler to stop crawling a given seed site after 10, 100, or 1000 inlinks have been discovered.
- *Parameters for outbound crawl*. If you check the "Crawl outbound" checkbox, the VOSON System will find what other sites are *linked to by* the seed sites (i.e., it will find outbound hyperlinks). If you choose to "crawl outbound," then you can control when the crawler will stop crawling each seed site. You can set the crawler to stop when the first of the following conditions is met: (1) X outlinks (or outbound hyperlinks) have been discovered (where X can be 10, 100, or 1000); (2) X "unproductive pages" have been crawled, where an unproductive page is a page that does not return any new outbound hyperlinks (i.e., hyperlinks that have not already been discovered) and X can be set to 5, 10, or 25; (3) the web site has been crawled to a depth of X pages (where X can be 1, 10, or 50). If you choose to crawl outbound, then you also have the option to get the VOSON System to collect favicons for the seed sites (the "Collect favicon.ico" checkbox). A favicon is a small bitmap associated with a particular web site or web page; most web browsers will display the favicon (if it exists) in the address bar and next to the page's name in a list of bookmarks. The following is the favicon for the Wikipedia site: W

Note: It is important to set the web crawl parameters carefully. Web crawling is resource-intensive (in terms of bandwidth and processor cycles), so we want to avoid collecting data that you don't actually need. Further, presently it is not possible to change the web crawl parameters after you have created the VOSON database, so choose carefully! If you are just testing out the service, please set the crawl parameters to a shallow crawl (e.g., so the crawler stops after crawling a single page on each seed site).

After setting the parameters for the crawler, press the Create database button. The Create VOSON database dialog will disappear and you will see the select network dialog, with your new database. *Note:* The letters "AN" (which stand for "analysis") are automatically appended to the VOSON database name. So the system or hidden database is the name that you entered in the Create database dialog, whereas the visible database that you see in the select networks dialog has the name of the system database, plus "AN" appended.

Figure 12.4 shows the select network dialog after the user "rob" created a new VOSON database (the size of the database is 0, because seed sites have not yet been entered and the crawler has not yet run). The new VOSON database was created with the following parameters: database name = "vosonEgo" (the VOSON System has automatically appended "AN" to this name); comment = "egonet for voson.anu.edu.au"; "crawl inbound" checkbox checked with the maximum number of inlinks set to 1000; "crawl outbound" checkbox checked with the maximum number of outlinks set to 100, maximum number of unproductive pages set to 25 and maximum depth of crawl set to 10; "collect favicon.ico" checkbox checked. With these settings, the VOSON System will find up to 1000 inbound hyperlinks to each of the seed sites and will stop crawling each seed after the following occurs: 100 outbound (internal) hyperlinks are discovered, 25 unproductive pages (pages not returning new external links) have been crawled, or 10 pages have been crawled. The crawler will collect the favicon of each seed site, if available.

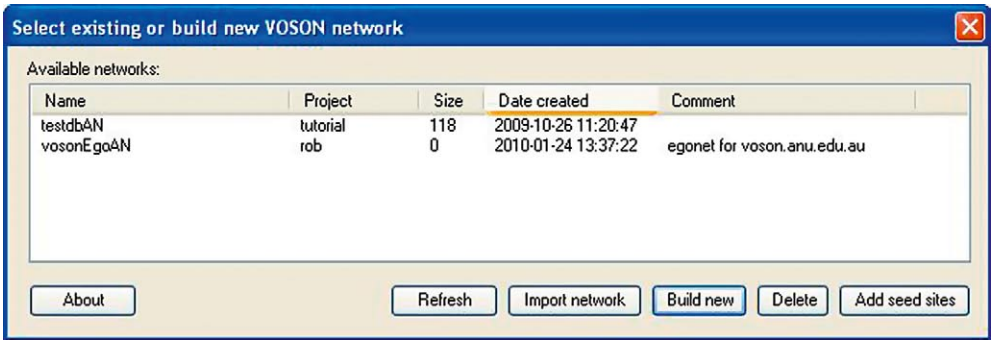


FIGURE 12.4 NodeXL "Select existing or build new VOSON network" dialog after a new database has been created.

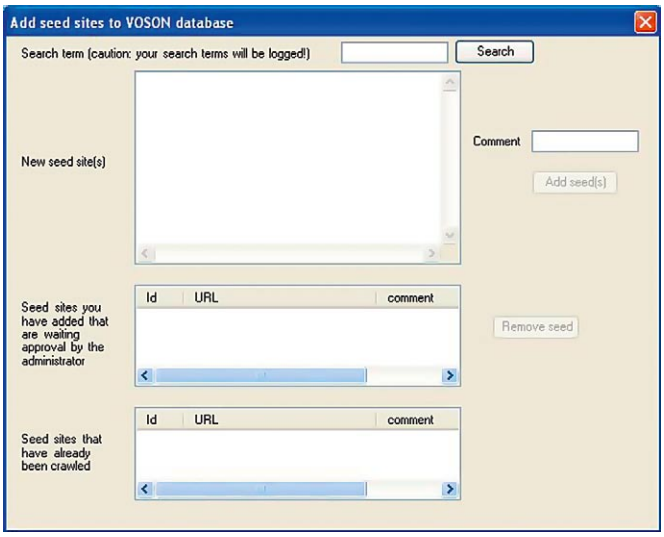


FIGURE 12.5 NodeXL "Add seed sites to VOSON database" dialog.

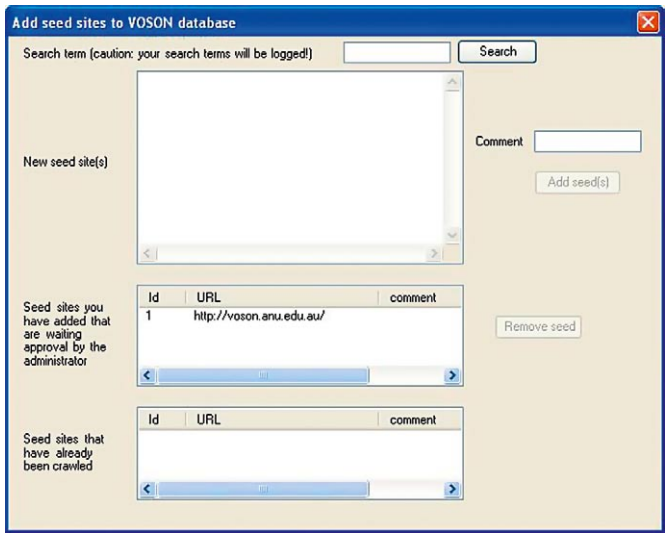


FIGURE 12.6 NodeXL "Add seed sites to VOSON database" dialog after entering voson.anu.edu.au as seed site.

Now you have created your VOSON database, but it does not yet contain any seed sites. To input seed sites into the VOSON database, press the "Add seed sites" button in the select network dialog (when you have selected the row for your new VOSON database). This will bring up the "Add seed sites to VOSON database" dialog (Fig. 12.5).

There are two ways you can choose seed sites: manually inputting seeds and searching for seeds. This example demonstrates the manual approach; the next section illustrates searching for seed sites.

To extract the "ego network" of the VOSON Project web site enter "voson.anu.edu.au" in the "New seed site(s)" textbox of the add seed sites dialog. An ego network will place the VOSON web site at the center of a collection of other web sites to which it links. The "Add seed(s)" button will then become active—press this and the add seed sites dialog will update to show you that "http://voson.anu.edu.au" has been successfully entered as a seed site and is scheduled for crawling (Fig. 12.6). Note that you can enter multiple seed sites simultaneously by putting each seed URL on a separate line in the "New seed site(s)" textbox of the add seed sites dialog.

After completing the entry of seed sites, you can close the add seed sites dialog and wait for the VOSON System to automatically crawl the seed(s). Note that the data collection process will not happen instantly and may take between 15 minutes to an hour for an average size database—the exact time taken to collect and process the hyperlink data will depend on the number of seeds you have entered and how many other users are concurrently accessing the web crawler. Also, you do not need to keep NodeXL running while you are waiting for the VOSON database to be constructed (you don't even need to remain connected to the Internet). You will be sent an email (to the email address you used to register for a VOSON account) when the crawler has finished and your database is ready to use in NodeXL.

Once you have received the email indicating that the VOSON database is ready for use, press the Refresh button on the "Select existing or build new VOSON network" dialog (assuming that you kept NodeXL running while the VOSON database was being prepared; otherwise select the NodeXL Data menu item "Import → From Web 1.0/Blog Network (via VOSON)," which will bring up the VOSON login dialog (Fig. 12.1) again. The dialog should update to show that there are 84 nodes in the "vosonEgoAN" VOSON database, indicating that the seed site http://voson.anu.edu.au is connected to 83 other sites.[13]

To import the vosonEgoAN VOSON database into NodeXL, select this database in the select VOSON

[13]Note that this example was conducted in January 2010; even if you follow this example exactly (in terms of the seed site and the parameters for the crawl), you will most likely get a different number of nodes. This is a reflection of the dynamic nature of the web—by the time you follow this example, the VOSON Project site will have changed and new sites may be listed by Yahoo! as linking to the VOSON Project site.

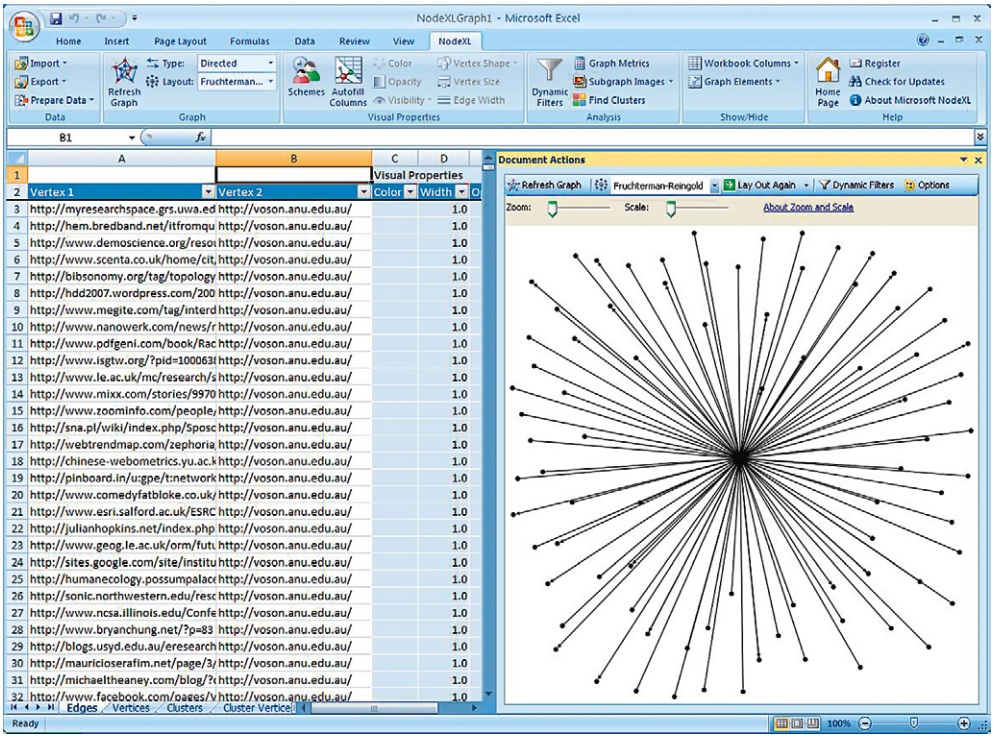


FIGURE 12.7 NodeXL visualization of a VOSON map of the http://voson.anu.edu.au web site (ego) connected to 83 other web sites (alters).

network dialog and then press the "Import network" button. After a short delay (this will depend on the speed of your Internet connection and the size of the network), the "Select existing or build new VOSON network" dialog will disappear and you will see that the network has been imported into the NodeXL Edges and Vertices worksheets. Press the "Refresh Graph" button to display a visualization of the network. Figure 12.7 shows the vosonEgoAN network in NodeXL, using the Fruchterman-Reingold layout algorithm.

The network map presented in Figure 12.7 is not particularly interesting (unless you like porcupines). One easy enhancement involves making use of the vertex attribute data that is imported by the VOSON Data Provider. In the Vertices worksheet you will see that the following columns have been added:[14]

- *Ringset*. This has the value 1 for seed sites and 2 for nonseeds.[15]
- *Country code TLD*. This takes the numeric value matching the country code top-level domain (TLD), which is automatically extracted from the host name. For example, ".au" is 15, ".de" is 55, and ".uk" is 225. When a host name does not include a country code TLD, this value is set to 0.
- *Generic TLD*. This takes the numeric value matching the generic TLD, automatically extracted from the host name. For example, ".edu" (and variants such as ".ac") is 12, ".org" is 9 and ".com" (and variants such as ".co") is 3. When a host name does not include a generic TLD, this value is set to 0.

To set the vertex color to match generic TLD, an Excel VLOOKUP function is used, with the following mapping: 0 (not defined), gray; 3 (".com" and variants), red; 8 (".net"), brown; 9 (".org"), blue; 11 (".gov" and variants), green; and 12 (".edu" and variants), turquoise. The result is a graph with more information (Fig. 12.8); it is immediately apparent from this that while the VOSON Project web site is an academic site, it connects to a wide range of other sites in terms of generic TLD. By selecting the Vertices worksheet and sorting on the Color column, it was found that the VOSON Project site connects to 29 .edu sites, 26 .com sites, 13 .org sites, 7 .net sites, 1 .info site, and 7 sites that do not have a generic TLD.

The most immediately useful and interesting information about this network can be gained by looking at the Edges worksheet; from here it is possible to see exactly what sites are directing hyperlinks to the

[14]In future versions of the VOSON Data Provider, more vertex attributes will be imported into NodeXL.
[15]The term "ringset" comes from the idea that a network can be structured as concentric rings with the seed sites in the first ring, sites discovered from crawling the seed sites in the second ring, and so on.

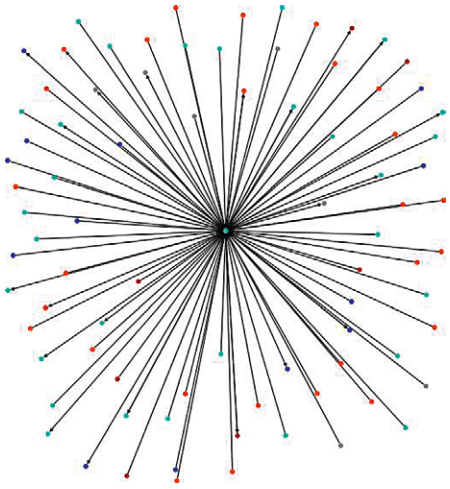


FIGURE 12.8 NodeXL visualization of the 1 degree VOSON hyperlink network with vertex color matching generic TLD.

VOSON Project site. By sorting on the Vertex 2 column in the Edges worksheet, one can quickly see that 67 pages link to http://voson.anu.edu.au.[16] As the VOSON Project leader (and maintainer of the project web site, which has been operational since 2005), reading through this is a fascinating exercise, with reactions ranging from "Wow, X links to us! Do we link to them?" to "Who on earth is Y, and why are they linking to us?" to "Come on! Z knows about the VOSON Project and actively maintain their own related site, so why aren't they linking to us? After all, we link to them!"

Although it is possible to work through the list of pages that link to your site, opening selected web pages in your browser to find out who they are (if you don't already know), this will not be feasible if you have hundreds, thousands, or more inbound hyperlinks. A more efficient way of discovering more about who is linking to your site is to conduct a second round of data collection by "promoting" the discovered sites so they themselves become seed sites and are thus crawled by the VOSON System (so as to find out who else they link to and who links to them). By doing this, you can leverage the power of the network to discover more about who you are connected to in cyberspace.

The following 15 seeds were added to the VOSON database via the add seeds dialog:[17]

- http://blogs.usyd.edu.au/eresearch: eResearch, Education and Educational Research
- http://english-webometrics.yu.ac.kr: The World Class University (WCU) Webometrics Institute at Yeungnam University
- http://humanecology.possumpalace.org/blog: The Human Ecology Forum
- http://julianhopkins.net: A blog about anthropology and blogging
- http://linkanalysis.wlv.ac.uk: The online version of Mike Thelwall's book *Link Analysis: An Information Science Approach*
- http://mahmoudlis.blogspot.com: A webometrics blog
- http://mauricioserafim.net: Administration, education blog, cyberculture, and research
- http://michaeltheaney.com/blog: A blog "focusing on organization in American politics, with particular attention to the importance of social networks"
- www.oii.ox.ac.uk: The Oxford Internet Institute at the University of Oxford
- http://sonic.northwestern.edu: The Science of Networks in Communities (SONIC) research group
- www.bryanchung.net: Digital art, software design, popular culture, and entertainment
- www.connectedaction.net: Sociology and the Internet, social media, and mobile social software
- www.ehess.fr/ldh/Services/liste_liens_web_ADR.php: Resources for the analysis of relational data and social networks
- www.hpcoders.com.au: Consulting in high-performance scientific and engineering computing
- www.ncess.ac.uk: The UK National Centre for e-Social Science

The VOSON System will collect the data for the additional 15 seeds using the same crawl parameters as were originally set, and again the user is informed by email when the data collection process has finished. After the second stage of data collection, there are now 3962 nodes in the network, which is displayed in Figure 12.9 again using the Fruchterman-Reingold algorithm (this

[16]When the VOSON crawler finds two or more pages from the same host name linking to a seed site, only the page that is first collected by the VOSON System is exported to NodeXL (note that all other pages are stored by the VOSON System, but they are presently not exported to NodeXL). For example, there are links to the VOSON Project web site on both the Australian Demographic and Social Research Institute (http://adsri.anu.edu.au) homepage and the academic homepage of my colleague Mathieu O'Neil (http://adsri.anu.edu.au/people/visitors/mathieu.php), but only the URL for Mathieu's page is exported to NodeXL because it was picked up in the data collection process first.

[17]These 15 seeds were more or less randomly selected from the 67 web sites that link to the VOSON Project site. As noted previously, the more correct terminology is that these pages were "promoted" to be seeds, as they were already in the database (they link to the VOSON Project site).

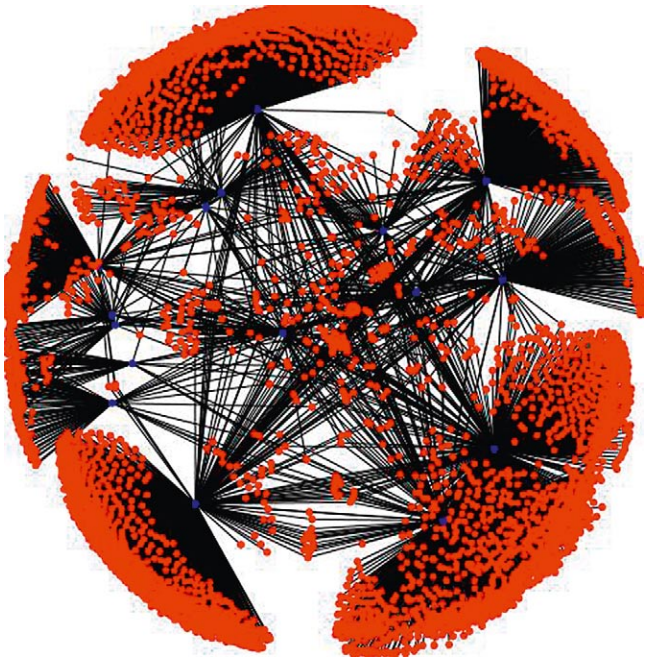


FIGURE 12.9 NodeXL visualization of the VOSON 2 degree hyperlink network created by crawling http://voson.anu.edu.au plus 15 additional seed sites that link to it.

time the Ringset attribute was used to color the vertices, with blue for the seed sites, and red for the rest).

The substantial increase in the size of the hyperlink network for a relatively modest increase in the number of seeds (and recall that the crawl parameters were also set so as to minimize the amount of data returned) is a reminder to the analyst of the vastness of the World Wide Web and the fact that hyperlink networks can easily "blow up" in size. If my goal is to use the hyperlink network to find out something about the slice of cyberspace that the VOSON Project site occupies, discovering related projects that I should be aware of or who should perhaps know about VOSON, the network displayed in Figure 12.9 is going to be very difficult to work with. The majority of the sites in this network are not going to be relevant to the VOSON Project because they are connected to blog seed sites (for example), which most likely focus on a number of issues or topics. We are thus facing the problem that information retrieval specialists refer to as "topic drift."

A way to deal with topic drift is to use the filter tool in NodeXL to create a subnetwork consisting only of the 16 seeds sites plus "important" sites, where a site is deemed important here if it has more than one connection or edge to the rest of the network. So a site will be considered unimportant and therefore not included in this subnetwork if it only directs a hyperlink to a single seed site or, conversely, only has one seed site directing a hyperlink to it. In contrast to undirected networks that have a single degree metric for each vertex, directed networks have two degree metrics for each vertex: in-degree (the number of incoming edges) and out-degree (the number of outgoing links). NodeXL will look to see if the network is directed or undirected and populate the appropriate metrics when you choose Graph Metrics. For this reason, NodeXL will not compute degree for directed networks, so the simplest method for identifying the subnetwork of seeds plus important sites is to take the following steps:

- Set the Graph Type to undirected.
- Merge duplicate edges (Prepare data → Merge Duplicate Edges menu item).
- Compute the graph metrics (Graph Metrics menu item). *Note:* To create this subnetwork, only degree needs to be computed (selecting all metrics will be time consuming in a network of this size). In the Vertices worksheet, sort the Degree column largest to smallest, which will put the sites with the highest degree on top. Go over to the Color column and you will see that all the seed sites (colored blue because we have used the Ringset attribute to color vertices) have a degree of 2 or greater.
- In Autofill Columns, select the vertex visibility to be "Degree" and then in the drop-down box beside it select "Greater than or equal to" and enter the number "2" in the box. Click OK. Click the Autofill button at the bottom of the Autofill dialog.
- Set the graph type back to directed and press the Refresh Graph button. The resulting subgraph is shown in Figure 12.10.

The resulting subgraph has 295 vertices. You can now proceed with analysis of the subgraph, but before doing that, you may like to first save it into its own workbook so if you want to do further analysis later, you don't need to go through the filtering process again. To save the subgraph into its own workbook, right-click in the graph pane and choose Select All → Vertices and Edges (all the vertices and edges will turn red). Then select Export → Selection to New NodeXL Workbook and save the resulting workbook.

Before proceeding with the analysis of the subgraph of seed sites plus important sites, there is another point to be made about the preceding process of filtering vertices. In the previous approach, we only selected on a single attribute (vertex degree). This worked because in this particular network, all the seed sites have a degree greater than 2, But what if some of the seed sites were isolates or had a degree of 1? The previous approach would have not included these sites in the resulting filtered network. In such a situation, the way to proceed is to use a customized formula in the Visibility column of the Vertices worksheet (rather than using Autofill Columns). If you look in this column, you will see that

the earlier use of Autofill Columns has led to a "1" being placed in the Visibility column for sites that are included in the subgraph, and zero otherwise. It is therefore possible to use an IF function in the Visibility column so that a vertex is made visible (i.e., gets a one in this column) if it has degree greater than or equal to 2 *or* a ringset equal to 1 (i.e., a seed site). To achieve this, use the following Excel formula: = IF(OR(Vertices[[#This Row],[Degree]] > 2,Vertices[[#This Row],[Ringset]] = 1),1,0).

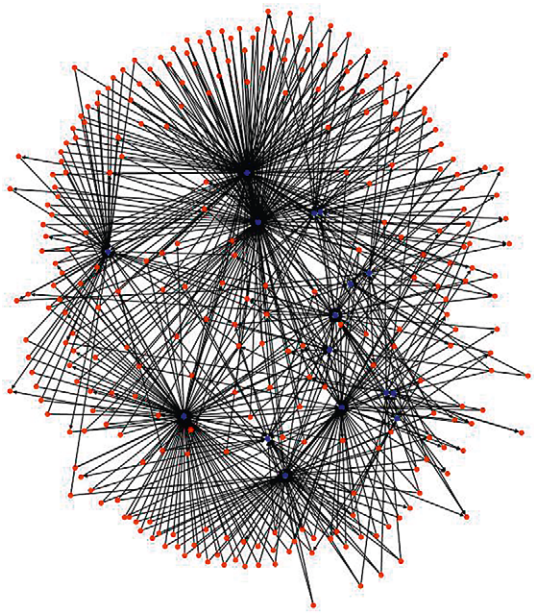


FIGURE 12.10 A filtered NodeXL visualization of the VOSON Project local hyperlink network containing 16 seed sites plus related "important sites."

Although Figure 12.10 is an improvement, it is still somewhat lacking in terms of providing analytic insights into the hyperlink network in which the VOSON Project web site is embedded. Figure 12.11 provides another view of the same network where Autofill Columns and a customized VLOOKUP formula have been used to alter vertex color, shape, and size in the following way: vertex color is mapped to generic TLD, vertex size is proportional to vertex in-degree, and the vertices for the seed sites have been set to solid squares (except that for the VOSON Project site, which is set to a solid triangle).

Figure 12.11 reveals a remarkable amount of information about where the VOSON Project web site sits in cyberspace. First, while the VOSON Project web site occupies a central position in this network (at least based on the visualization), it should be emphasized that this is largely a product of the data collection process (the 15 other seed sites were chosen among those sites that link to the VOSON site). An interesting feature of Figure 12.11 is that the graph is divided into two parts. The bottom two-thirds of the graph is occupied by 10 of the seed sites and all of these (apart from www.hpcoders.com.au and mahmoudlis.blogspot.com) are academic web sites, and the majority of nonseed sites in this part of the graph are either academic sites or .org sites.[18] This part of the graph is dominated by three vertices, which all have a particular focus on e-research: blogs.usyd.edu.au/eresearch, www.ncess.ac.uk, and www.oii.ox.ac.uk. The other portion of the graph is

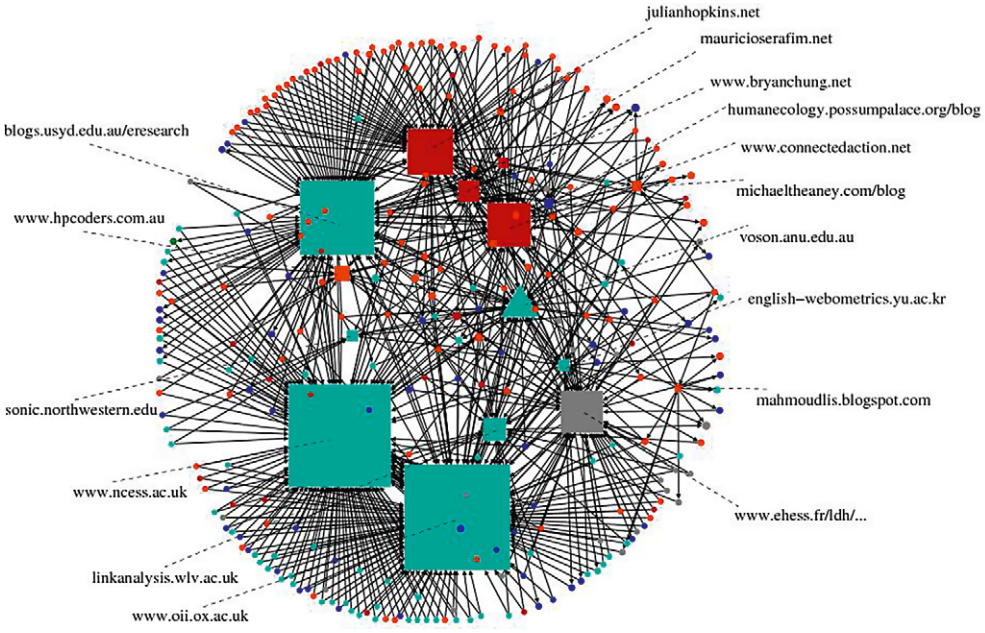


FIGURE 12.11 NodeXL visualization of the local hyperlink network for the VOSON Project (filtered). Seed sites are solid squares (except for the VOSON Project web site, which is a solid triangle), whereas other sites are discs. Vertex size is proportional to in-degree and vertex color corresponds to generic TLD (.edu, turquoise; .org, blue; .com, red; and .net, brown).

[18]Note that www.ehess.fr is the French higher education institution, the School for Advanced Studies in the Social Sciences.

occupied by the remaining six seed sites, all of whom are blog or consulting sites, and the nonseed sites in this part of the graph are primarily commercial sites.

So the local hyperlink network for the VOSON Project is neatly divided into two parts or communities, one reflecting an academic/nonprofit focus and the other a more commercial focus. The VOSON Project site and the academic blog site blogs.usyd.edu.au/eresearch are on the boundary between these two parts of the graph. As the leader of the VOSON Project, I find that this visualization makes inherent sense: the VOSON Project has its origins in academia (in particular, e-research), but the development of VOSON methods and tools have been informed by commercially oriented or funded web research, and there is increasing interest and awareness of VOSON in the commercial sector.

So the visualization of the VOSON Project's online brand presence in Figure 12.11 accords with where the project is currently positioned. To the extent that the VOSON research methods and services are increasingly being used in more commercially oriented web research (and this chapter reflects a move in such a direction), then it will be interesting to see how the composition of the VOSON Project local hyperlink network changes in the future.

12.5 EXAMPLE 2: WHAT IS THE HYPERLINK NETWORK OF THIS FIELD/INDUSTRY/SECTOR?

For this second example of using NodeXL to collect and analyze hyperlink networks, let's assume that you have developed a new software tool for social network analysis and you are ready to start selling it. Let's make the further (very reasonable) assumption that the web will be the primary medium for marketing and sales. You have set up your web site, but how do you get potential customers to find out about your fantastic new social network analysis tool?

The "build it and they will come" strategy is unlikely to work. Another approach could be to buy Google-sponsored links, but this can be expensive and customers are often more interested in the left-hand side of their web search page, the "organic" search results. Furthermore, a lot of customers will not find out about your software by searching Google but will discover it by reading reviews or comments on other authoritative sites (e.g., blogs focused on social network analysis), and it is links from these types of sites that drive organic search engine rankings. Prioritizing these sites for contact and marketing may be a good strategy. But how will you find these sites? One strategy is to use the VOSON Data Provider to construct a hyperlink network map of sites focused on social network analysis.

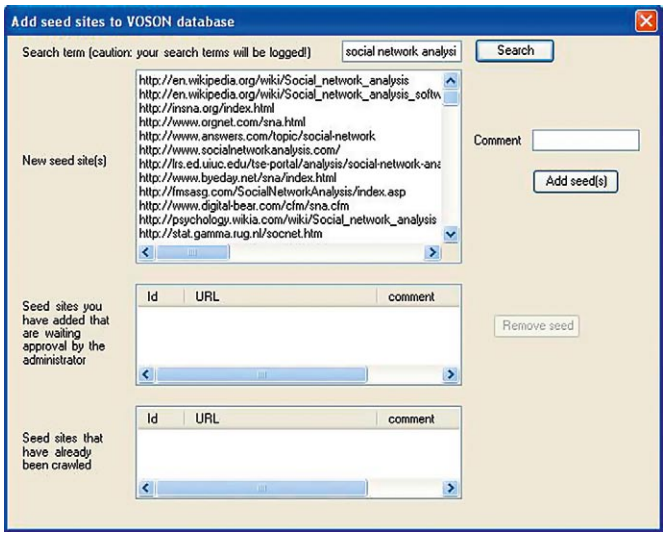


FIGURE 12.12 NodeXL "Add seed sites to VOSON database" dialog with 16 seed sites plus related "important sites."

By studying this map you will be able to discover what other social network analysis tool providers are out there and who is linking to them.

For this example, we create a VOSON database with the data collection parameters set so that only the first 100 inlinks to a seed site will be collected, and the crawler will only crawl the first page of each seed site (these parameters were set to minimize the size of the resulting hyperlink network). In the add seed sites dialog, the search term "social network analysis" was entered and the "Search" button pressed. After a few seconds, the VOSON System returned (via the Yahoo! API) the top 100-ranked pages that are focused on SNA, and they are displayed in the "New seed site(s)" text box (Fig. 12.12).

Once again, simply perusing the list of pages returned by the Yahoo! API can be a fascinating exercise for someone working in the topic area of the search, in this case the area of social network analysis. However, network methods help us understand more deeply who is working in this space and how they are linked to one another. At this point, you might want to quickly look through the list to check that all the returned pages are in fact relevant; irrelevant pages can be simply deleted from the text box by highlighting the row and pressing the delete key. The rest of the pages can then be added as seed sites by pressing the "Add seed(s)" button (for this example, we just added all 100 pages as seeds). You can then close the add seed sites dialog and grab a coffee (or go to bed) while the VOSON System collects and processes the hyperlink network data.

Even with setting the data collection parameters so as to not return too much data, the use of 100 seed

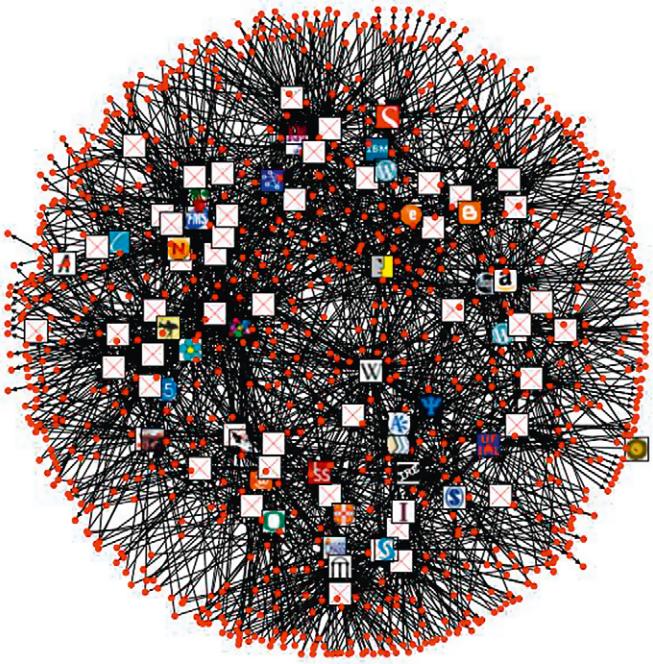


FIGURE 12.13 A filtered NodeXL visualization of seed sites focused on "social network analysis" plus important sites (sites with degree greater than or equal to two). The favicon (where available) is displayed for the seeds, whereas red discs are used to represent non-seed sites.

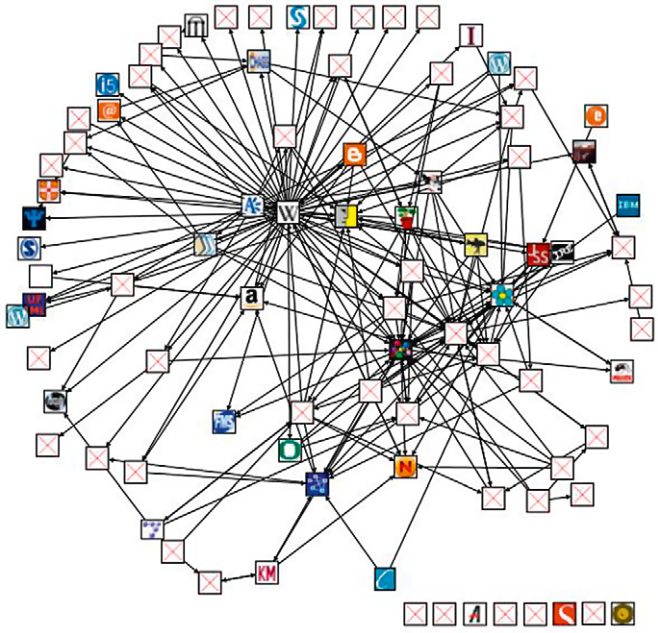


FIGURE 12.14 A filtered NodeXL visualization of seed sites focused on "social network analysis," using the favicon (where available) for the graph vertex.

sites leads to a reasonably large hyperlink network of 6049 nodes. For the reasons noted earlier, this network is likely to suffer from topic drift and a network of this size is going to be difficult to analyze, at least using visualization approaches. So as was done earlier, we create a subnetwork consisting only of the seed sites plus important sites—this is a network with 896 nodes, and is shown in Figure 12.13 (using the Fruchterman-Reingold layout). This figure displays the favicons for seed sites; other sites are represented by a red dot. Note that if the favicon was not collected for a particular seed site (i.e., it does not exist, or you did not check the "Collect favicon.ico" checkbox in the "Create VOSON database" dialog), then a default image (a red cross) is shown. Around 40 of the seed sites have favicons; for the others the default image is used.

A final visualization is made by creating a subgraph containing only the seed sites (Fig. 12.14).[19]

It is immediately apparent from Figure 12.14 that a few sites are taking very central or important roles within this hyperlink network. Although it is possible in NodeXL to identify these actors by interactively working with the graph (placing the cursor on the vertices and finding out who they are from the Vertex worksheet), it is also helpful to make use of the column sort feature in the Vertex worksheet. Sorting the Vertices worksheet by various columns can bring different types of web sites to the top, from the most peripheral to the most central.

In terms of in-degree, the top five sites are the International Network for Social Network Analysis INSNA (http://insna.org); Analytic Technologies, the distributors of the UCINET social network analysis software (www.analytictech.com); the academic homepage of Vladimir Batagelj, who is the creator of the Pajek social network analysis software (http://vlado.fmf.uni-lj.si); orgnet.com social network analysis software and services (www.orgnet.com); and the homepage of the academic Robert Hanneman (http://faculty.ucr.edu/~hanneman). So three of the top five sites (ranked by in-degree) in this hyperlink network are social network analysis software or service providers; a new entrant to this market would most likely want to closely study the web presence of these three entities with the aim of learning exactly who is linking to these sites.

It is possible to identify the sites that are taking the role of linking to other sites by sorting on out-degree. Based on this vertex measure, the top five sites in this

[19]Although 100 seed pages were used to create the data set, the network in Figure 12.14 only contains 88 vertices. The reason for the discrepancy is the fact that seed pages from the same web site (e.g., http://en.wikipedia.org/wiki/Social_network_analysis and http://en.wikipedia.org/wiki/Social_network_analysis_software) were automatically merged to form a single vertex.

hyperlink network are Wikipedia (http://en.wikipedia.org), Answers.com (www.answers.com), the homepage of the academic Steffen Mazanek (www.steffen-mazanek.de), INSNA (http://insna.org), and Analytic Technologies (www.analytictech.com). It is not particularly surprising that Wikipedia, Answers.com, and INSNA are in this top five, but it was interesting to discover that Steffen Mazanek and also the providers of UCINET are taking an important role in directing their visitors to other web-based resources focused on social network analysis.

Finally, betweenness centrality can be used to identify those sites that are connecting disparate parts of this subgraph and on this measure, three sites—Wikipedia, INSNA, and Analytic Technologies—were head and shoulders above the rest.

The reaction to this analysis of someone who is very familiar with the field of social network analysis might possibly be "So Wikipedia and INSNA are important vertices in the hyperlink network relating to social network analysis; so tell me something I *don't* know!" But even with this simple data collection and analysis involving the identification of a handful of top-ranked sites, I discovered one important site that I wasn't aware of (Steffen Mazanek's homepage) and I also learned that Analytic Technologies, though primarily known as a provider of social network software, is also taking an active role in directing people to other web sites focused on social network analysis. If the analysis were extended to the top 10 sites or beyond, then I would definitely be discovering new sites that might be important for my strategy for building the web presence of the VOSON Project. One concrete "take home" for me from this simple analysis is that none of the five top-ranked sites (in terms of out-degree) direct hyperlinks to the VOSON Project web site. If I want to develop the web visibility and prominence of the VOSON software, then I perhaps need to do something about this situation! Submitting a request to these sites that they link to my web site along with a reason for doing so may result is significant increases in traffic and visibility.

If the field that you want to map is one that you are not particularly familiar with, or perhaps it is larger than the (still) relatively small field of social network analysis used in this example, then it will not be possible for you to know all the main actors, regardless of how long you have been in the field. In such a situation, then using NodeXL in the manner described here will be extremely useful.

12.6 BLOGS, TEMPORAL CHANGES, AND NETWORK FLOW

This chapter has provided an introduction to collecting and analyzing hyperlink network data using NodeXL. There are three advanced topics that should be mentioned briefly.

Blogging was one of the first forms of Web 2.0 behavior, and it was popular well before Facebook and Twitter took off. There has been a lot of academic research into blogging networks—for example, Adamic and Glance's [14] analysis of U.S. political blogger activity that illustrated the distinct opposed clusters in American politics. This chapter is not devoted to using NodeXL to analyze blog networks for several reasons. There are certain aspects of collecting and processing blog network data that mean a crawler optimized for Web 1.0 networks (such as the one used by the VOSON Data Provider) cannot be used effectively for collecting blogging network data. In particular, blog researchers are typically interested in identifying the networking behavior occurring in a particular time period (e.g., Adamic and Glance studied the month preceding the 2004 U.S. presidential election), and it can involve sophisticated text mining procedures to parse blog post contents, even when really simple syndication (RSS) feeds are being used to identify the URLs of blog posts at particular time points. Additionally, it is generally useful to distinguish "permalinks" (hyperlinks occurring in the body of blog posts) from hyperlinks in "blogrolls" (lists of hyperlinks that sit to the side of the blog posts), as the former typically reflect the current reading activity of the blogger and the latter have a tendency to go "stale." Parsing out the permalinks from the blogroll links can also be technically challenging.

The challenges associated with extracting (at scale) useful research data from blog sites are so great (and the demand for such data so significant) that there are new companies such as Spinn3r that provide a dedicated service focused on continually crawling the blogosphere and making available the (parsed) data via an API (another early example of a company that provided such an API to collected content was Bloglines[20]). However, as far as we are aware, such services do currently provide their data in a form that can be easily converted into hyperlink networks. What this chapter has shown is that hyperlink analysis using Web 1.0 data is complicated (but is made significantly easier with NodeXL), but blog networks are even more complicated. So we will have to leave blog networks for a future version of the VOSON Data Provider—or perhaps they will be addressed by another NodeXL spigot developer (there is definitely a market for such a plug-in)—and another edition of this book.

The other two advanced topics relate to what many regard as the "Holy Grails" of hyperlink network analysis. Over the years, when presenting my hyperlink network research, the two questions that I have

[20]www.bloglines.com

encountered the most have been: "So now I know what the hyperlink network of field/sector/industry X looks like today, but how did it get to be like this?" and "So you've shown me the 'pipes' (hyperlinks between web sites), but what is the volume of traffic that flows along them?"

Regarding the first question, statistical analysis of dynamic networks is one of the hot and challenging topics in social network analysis. Longitudinal network data can help distinguish social influence (the tendency for people to become like their friends) from social selection (the tendency for people to seek out other people who are like them). Dynamic analysis of hyperlink networks will allow researchers to better understand the exact mechanisms by which online visibility and prominence develop and how these measures correlate with offline characteristics of organizations (e.g., profit or other offline measures of success within the field). However, dynamic hyperlink analysis presents additional challenges in terms of data collection. If you want to know what the hyperlink network of your particular field looked like a year ago, or 5 years ago, then the VOSON Data Provider can't help you (because it only accesses data from the current state of the live web). However, the Internet Archive[21] has been archiving the web since 1997, and while it does not currently have a publicly available API (its archived data are only accessible via a web browser via the Way Back Machine), perhaps a future version of NodeXL will include a spigot that connects up to the Internet Archive.

Finally, the VOSON Data Provider for NodeXL can be used to discover the hyperlinks that exist between web sites, but it provides no insight into what traffic is flowing along these edges. Such information is difficult to access unless you have access to the web logs of the relevant web sites (e.g., you own them). However, there are companies that do collect web traffic data—Hitwise[22] accesses these data from various Internet Service Providers in different countries and Alexa[23] collects them via people downloading and installing a special browser toolbar—and again, perhaps web traffic data will become available to NodeXL users in the future.

12.7 PRACTITIONER'S SUMMARY

This chapter has focused on an approach and associated tool for quantitatively mapping and understanding organizational (Web 1.0) web presence. For those who want to use NodeXL to understand and improve their organizational web presence, the strategy that is used will depend on who you are and what you are aiming to be within your particular field or industry. Kleinberg [3] introduced the idea that there are two types of sites on the web: "hubs" are sites that are primarily focused on directing visitors to interesting/important/useful information that is found on other web sites ("authorities"), with a good hub being a site that hyperlinks to many good authorities, and a good authority being a site that receives hyperlinks from many good hubs.

Although the Kleinberg classification of hubs and authorities will not be useful for understanding all types of behavior or roles on the web, one can get a lot of mileage from this concept. A number of times I have presented hyperlink networks to audiences who are familiar with the field or industry that is being mapped but have never heard of Kleinberg's work, yet they find the concept of hubs and authorities immediately useful as an explanation of the behavior within the online network. In the second example explored earlier, even though the Kleinberg hub and authority scores were not calculated,[24] it appeared that the authority sites (high in-degree) tended to be providers of social network analysis software or services, whereas the hub sites were exemplified by sites such as Wikipedia and Answers.com. An interesting finding was that a software provider (Analytic Technologies) also appeared to be exhibiting hublike behavior.

So if it makes sense for you to be either a hub or an authority in your field or industry, then the actionable insights that can be gained from using NodeXL to map the relevant hyperlink network are clear. For example, assume you have just set up a site focused on social network analysis and you want your new web site to become *the* place for obtaining the latest information on the field (and you plan to monetize the web site with banner advertising). You therefore aspire to be a good hub within your hyperlink network (because you will be hyperlinking to other web sites that provide authoritative tools and specialist information on social network analysis), so your strategy would be to use NodeXL to identify the authority sites that you should be linking to. On the other side of the coin, assume (as was done initially) that your role in the field of social network analysis is to provide specialist and authoritative tools and services. In aspiring to be a major authority within the hyperlink network, your strategy would be to use to NodeXL to identify the hub sites that should be linking to you.

So the simplest example of using NodeXL to improve your organizational web presence effectively boils down

[21] www.archive.org
[22] www.hitwise.com
[23] www.alexa.com
[24] Kleinberg's [3] Hyperlink Induced Topic Search (HITS) algorithm is currently not implemented in NodeXL.

to this: use NodeXL to identify the leaders in your field or industry and try to understand what it is about their hyperlink network that reflects (and probably contributes to) their success. Then try to emulate their success. In such a situation, NodeXL can provide actionable insights for improving your organizational web presence. Used over time, it can provide a series of images and metrics that measure how closely you have come to your goals.

However, we finish this section with two caveats. First, it is important to realize that a graph-theoretical measure of importance within a field, such as good authority or hub scores in the hyperlink network, is only one source of information and needs to be balanced against everything else you know about the field. There will always be cases of organizations whose online presence or importance does not accord with the reality. The most important organization in your field may have a nonexistent web presence (i.e., it may not even appear as a vertex in your hyperlink network), and this could be due to measurement error (you made a typo when entering the URL as a seed, or the web site happened to be offline when the VOSON crawler visited) or it could be a real finding. You need to know enough about the industry or field that you are mapping to be able to pick up such potential issues, as they will skew results and cause embarrassment (or worse).

Second, the importance of proper data preparation and cleaning cannot be underestimated. Hyperlink network analysis necessarily involves automated techniques (the examples presented in this chapter would be infeasible using manual techniques), but a lot of the data processing will still require human input (i.e., domain knowledge), and without this input, the hyperlink network may have no real meaning. The "garbage in, garbage out" phrase is very applicable to Web 1.0 networks.

12.8 RESEARCHER'S AGENDA

Hyperlink networks have existed since the dawn of the web, and they are still an incredibly important aspect of the online behavior of the commercial entities, private individuals, and government. Hyperlink data have been used for research into e-government—for example, Escher et al.'s [15] empirical analysis of the impact of the web on the "nodality" of government (the government's position at the center of social and informational networks). The use of hyperlink data for competitive business intelligence has been investigated by Vaughan and You [16], who constructed networks of telecommunication company web sites using co-link data (if page X and Y are both linked to by page Z, then X and Y are co-linked, and the number of co-links that two pages share is a measure of their similarity) and show that visualization of these networks correctly maps the companies into industry sectors.

We are not aware of examples of e-government research where hyperlink network data have been analyzed using formal social network analysis methods (i.e., via the construction of a complete hyperlink network relevant to, say, a government department or a sector of government). Instead, research such as that by Escher et al. [15] has focused on raw counts of hyperlinks to or from government sites (without ascertaining the attributes of the site sending or receiving the hyperlink) as measures of e-government performance. It is not clear why formal social network analysis has not yet been used in e-government research using hyperlink data, but the vastness of the web certainly poses a challenge; as Ackland [17] noted, using network sampling to extract complete hyperlink networks may be a way forward. We expect the availability of tools such as NodeXL to have a positive impact on e-government hyperlink research.

As mentioned previously, dynamic hyperlink analysis is a fruitful area of research, as is research into approaches for jointly analyzing hyperlink and text content data. Finally, a promising area of methodological research involves mixed-methods approaches where quantitative hyperlink analysis is combined with network ethnographical (e.g., [18]) qualitative methods.

References

[1] M.H. Jackson, Assessing the structure of communication on the World Wide Web, J. of Comput.-Mediated Commun. 3 (1) (1997) 273–299.
[2] M. Thelwall, Interpreting social science link analysis: a theoretical framework, J. of the Am. Soc. for Inform. Sci. and Technol. 57 (1) (2006) 60–68.
[3] J. Kleinberg, Authoritative sources in a hyperlinked environment, J. ACM 46 (5) (1999) 604–632.
[4] E. Davenport, B. Cronin, The citation network as a prototype for representing trust in virtual environments," in the web of knowledge: a festschrift in honor of eugene garfield, in: B. Cronin, H. Atkins (Eds.) , Information Today, Metford, NJ, 2000.
[5] R. Rogers, N. Marres, Landscaping climate change: a mapping technique for understanding science and technology debates on the world wide web, Pub. Understand. of Sci. 9 (2) (2000) 141–163.
[6] H.W. Park, C.S. Kim, G.A. Barnett, Socio-communicational structure among political actors on the web in south korea: the dynamics of digital presence in cyberspace, New Media & Soc. 6 (3) (2004) 403–423.
[7] M. Hindman, K. Tsioutsiouliklis, J.A. Johnson, Googlearchy: how a few heavily linked sites dominate politics online, Paper presented at the annual meeting of the Midwest Political Science Association, 2003.
[8] D. Lusher, R. Ackland, A relational hyperlink analysis of an online social movement, J. Soc. Struct. (2010). Accepted 13th October 2009.
[9] R. Ackland, R. Gibson, Mapping Political Party Networks on the WWW, refereed paper presented at the Australian Electronic Governance Conference, University of Melbourne, April 2004, pp. 14–15.

[10] R. Ackland, R. Gibson, W. Lusoli, S. Ward, Engaging with the public? Assessing the online presence and communication practices of the nanotechnology industry, Soc. Sci. Comput. Rev. (2010). Accepted 19, June 2009.
[11] R. Rogers, Mapping Public Web Space with the Issuecrawler, in: C. Brossard, B. Reber (Eds.), Digital Cognitive Technologies: Epistemology and Knowledge Society, Wiley, London, 2009, pp. 115–126.
[12] M. Thelwall, Introduction to Webometrics: Quantitative Web Research for the Social Science, Morgan & Claypool, San Rafael, CA, 2009.
[13] W.H. Dutton, P.W. Jeffreys, World Wide Research: Reshaping the Sciences and Humanities, The MIT Press, Cambridge, MA, 2010.
[14] L. Adamic, N. Glance, The political blogosphere and the 2004 U.S. election: divided they blog, Mimeograph (2005). Available at www.blogPulse.com/Papers/2005/AdamicGlanceBlogwww.pdf.
[15] T. Escher, H. Margetts, V. Petricek, I. Cox, Governing from the centre? comparing the nodality of digital governments. Paper presented at the 2006 Annual Meeting of the American Political Science Association, Chicago, August 31-September 4, 2006.
[16] L. Vaughan, J. You, Mapping business competitive positions using web colink analysis, Proceedings of 2005 International Conference of the International Society for Scientometrics and Informetrics, 2005, pp. 534–543.
[17] R. Ackland, Social network services as data sources and platforms for e-researching social networks, Soc. Sci. Comput. Rev. Special Issue on e-Social Science 27 (4) (2009) 481–492.
[18] P.N. Howard, Network ethnography and the hypermedia organization: new media, new organizations, new methods, New Media & Soc. 4 (4) (2002) 550–574.

CHAPTER

13

Flickr

Linking People, Photos, and Tags

Eduarda Mendes Rodrigues and Natasa Milic-Frayling†*

***Department of Informatics Engineering, University of Porto, Porto, Portugal**

†Microsoft Research Ltd., Cambridge, UK

OUTLINE

13.1 Introduction 201

13.2 Flickr Social Media 202
- *13.2.1 What Can You Do with Flickr?* 202
- *13.2.2 Flickr Sets, Collections, and Tags* 202
- *13.2.3 Flickr Groups* 204
- *13.2.4 Sharing and Social Interaction* 206

13.3 Flickr Networks 207
- *13.3.1 Tag Networks* 208
- *13.3.2 User Networks* 209

13.4 What Questions Can Be Answered by Analyzing Flickr Networks? 209
- *13.4.1 Personal Sphere* 209
- *13.4.2 Community Sphere* 209
- *13.4.3 Application Sphere* 209

13.5 Importing Flickr Data into NodeXL 210
- *13.5.1 Related Tags Network* 210
- *13.5.2 Flickr User Networks* 211

13.6 Working with the Flickr Data 212
- *13.6.1 Graph Type* 212
- *13.6.2 Merging Duplicate Edges* 212

13.7 Analyzing Flickr Networks with NodeXL 212
- *13.7.1 Revealing Landmarks and Tourist Attractions Using a Location Tag* 213
- *13.7.2 Identifying Tag Clusters to Disambiguate the Meaning of a Tag* 215
- *13.7.3 Flickr User Networks* 216

13.8 Practitioner's Summary 220
- *13.8.1 Data Preparation* 221
- *13.8.2 Graph Layouts* 221

13.9 Researcher's Agenda 222
- *13.9.1 Exploration and Validation* 222
- *13.9.2 Integration* 222

13.1 INTRODUCTION

Flickr[1] is a social media service that enables users to upload, organize, and share digital photos and, recently, videos. Its significance is rather remarkable in the context of the social web, often referred to as Web 2.0, and the evolution of online services. Indeed, Flickr opened up new avenues for photography enthusiasts to express their creativity and talent by posting and discussing their photos. It connected families and friends by enabling them to share special moments through digital photos. For many, it also provided a practical solution for safe storage and access, with no worries

[1]www.flickr.com

about backups, virus attacks, and potential loss of precious personal photo collections. It is not surprising that Flickr now has more than 84 million users globally[2] and hosts an impressive number of more than 4 billion photos.[3]

The scale of the Flickr service requires an effective technical solution for storing and managing content. However, although that is absolutely essential for running the service, on its own it is not sufficient to enable the successful creation of social media. The key is to provide engaging and effective content access. In case of images and videos, this is particularly challenging because search over such media is not as well defined as search over text and audio. Indeed, indexing and searching textual content and audio transcripts is well understood. Search engines extract keywords from text and create inverted indices that provide effective mapping between keywords in the user queries and the documents that contain them. With photos, however, retrieval systems have to rely primarily on textual metadata that describe various aspects of the photos. Flickr's approach has been to encourage users to tag and annotate photos and then enable them to search and browse the photos based on those tags.

Collectively, the Flickr community has created a large, ever-growing vocabulary of tags that is the foundation for accessing and managing content on Flickr. At the same time, the tagging mechanism depends on the social characteristics of Flickr for its effectiveness. Because tags are generated by individual users, their usefulness to other users depends on the shared understanding of the tags' meaning, which in turn is determined by the exposure of content and metadata among the users and the practices they collectively adopt. For that reason it is interesting to study the interplay between social interactions and tagging practices in Flickr. This chapter demonstrates how that can be done using NodeXL. It presents analysis of several Flickr networks to demonstrate how to use NodeXL to gain insights into content and interaction patterns among Flickr users.

13.2 FLICKR SOCIAL MEDIA

Since its launch in 2004, Flickr has grown into an active and engaged community, supported by several key features and services that Flickr integrates. Even if you do not have a Flickr account, you can still use some of these services. You can simply access www.flickr.com through a web browser to start viewing public Flickr photos. Figures 13.1 and 13.2 show Flickr's Explore page that provides search by keywords, filtering by calendar, recommendations of links to follow, and access to sets and Flickr groups.

On Flickr you can easily view content that has been posted over the past seven days or during a specific calendar month (Fig. 13.1). Besides historical data, you can view photos grouped into sets and collections or posted by specific interest groups. As Figure 13.2 shows, you can also look for photos based on a specific location, captured by geo-tags, or based on tags manually added by users. Selecting a tag from the tag cloud takes you to photos that have something in common. This common property reflects the meaning of a tag that is shared across the Flickr community.

13.2.1 What Can You Do with Flickr?

In order to use Flickr to its fullest, you should get a Flickr account. Then you will be able to take advantage of many features that Flickr provides for managing photos and videos (Table 13.1).

Starting with upload, Flickr provides no fewer than five ways for posting images and videos: via the Flickr Upload available for PC and Mac; via iPhoto, Aperture, or Windows XP plug-ins; via Flickr's upload web page; via email; and via various free third-party applications that make use of the Flickr Application Programming Interface (API). If you also wish to edit photos online, you can upload them to Flickr's partner service, Picnik, and change them in several ways (crop, add text, remove red eye, add effects, etc.).

If you wish to find photos posted by someone you know, it is best to use the "All Flickr members" search (Fig. 13.3) and then view the person's photostream (Fig. 13.4). The photostream always shows the most recently uploaded photos. On the right there are also recent sets of photos.

13.2.2 Flickr Sets, Collections, and Tags

Using the Flickr Organizr tool, you can create *sets* and *collections* of photos and videos. *Sets* typically include groups of photos and videos around a certain theme, such as "Holiday Pictures." Common uses of sets include grouping favorite photos and making photo albums.

Sets are normally grouped into *collections.* For example, a collection may refer to the year in which photos were taken or can represent a broader theme ("Travel," "Conferences," "Events," "Portraits," etc.). The Organizr

[2]Yahoo! Quick View Metrics, June 2009: http://mms.businesswire.com/bwapps/mediaserver/ViewMedia?mgid=190921&vid=1
[3]Flickr blog, October 2009: http://blog.flickr.net/en/2009/10/12/4000000000

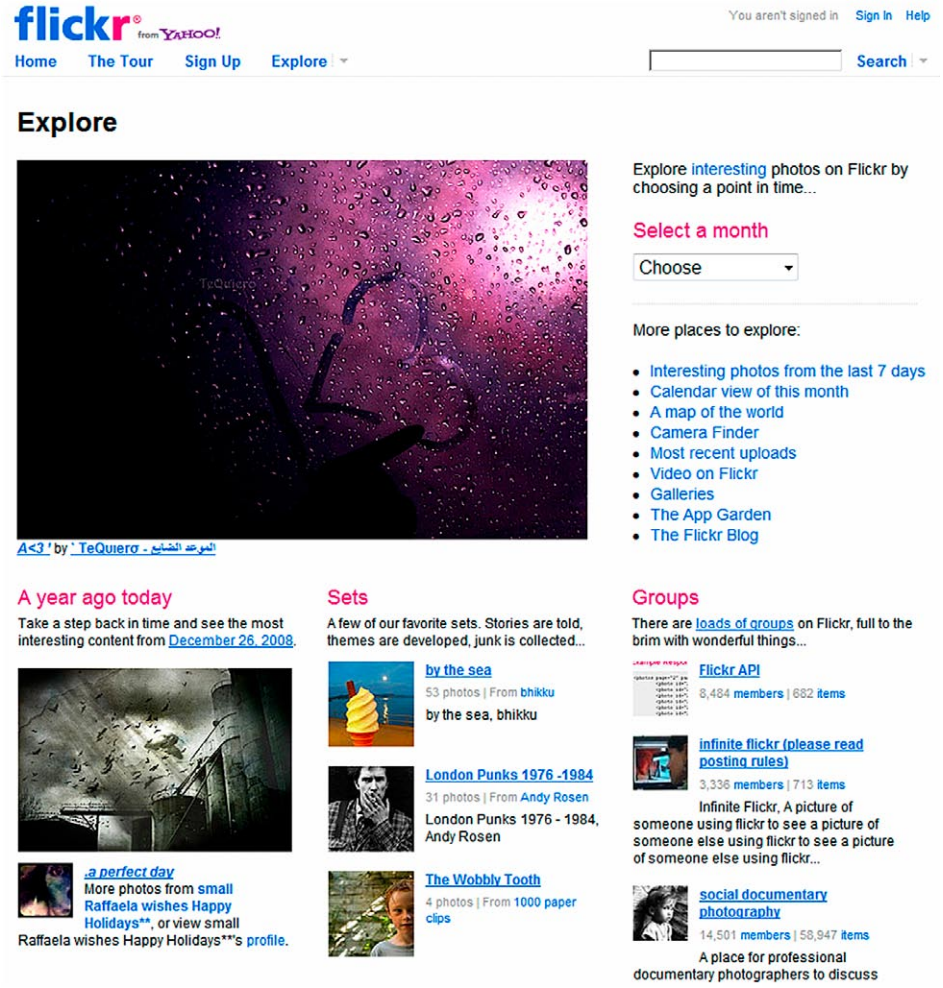


FIGURE 13.1 Flickr enables web users to search and browse through public collections of billions of digital photos (www.flickr.com/explore).

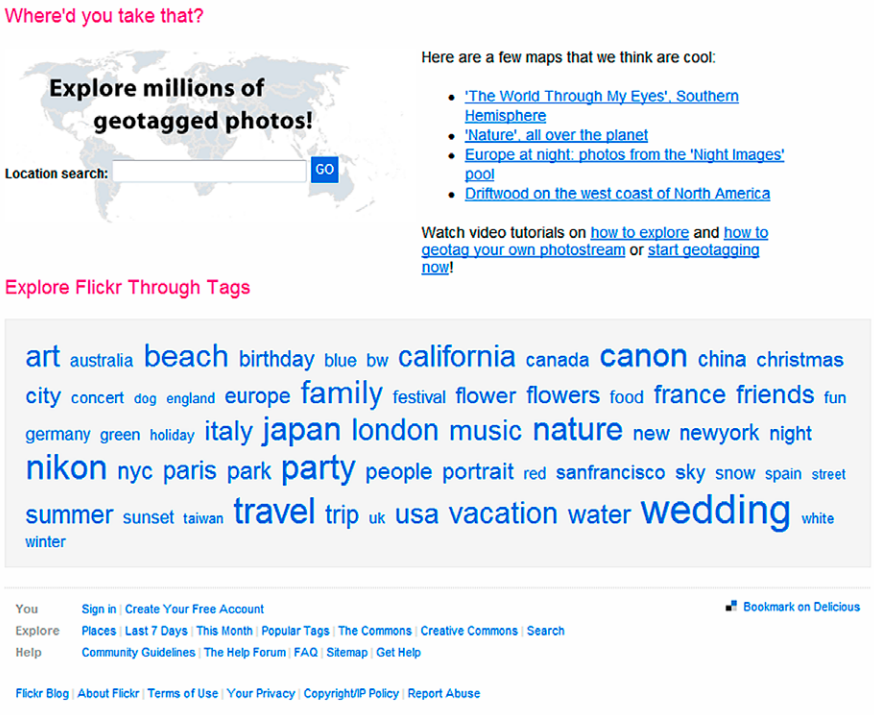


FIGURE 13.2 The Flickr Explore page includes search over geo-tagged images and a tag cloud for browsing photos based on their tags (www.flickr.com/explore).

enables you to manage sets and collections by handling multiple photos and videos at the same time.

On Flickr you can annotate your photos and videos with tags and descriptions. Tags are essentially keywords that facilitate organization and retrieval, as illustrated in Figures 13.1 and 13.2. In many cases, users apply tags with the aim to communicate additional context about the photos, both for personal and public benefit [1, 2]. They may add tags to describe context to friends and family. Some misuse tagging and apply keywords that are not representative of the content or context but have the sole aim to draw increased attention to their photos [3].

Each photo or video can include up to 75 tags. Normally these tags are related. They may complement each other by describing photos at a different level of detail. For example, in Figure 13.5 the user assigned seven tags, two referring to the time of posting the photo (2007, October) and the remaining five referring to the topic, ranging from specific names (e.g., Usenet, Treemaps) to general concepts (e.g., Newsgroup, Information, Visualization).

TABLE 13.1 Basic Tasks That You Can Perform with Flickr

Upload	Upload individual photos or collection of photos from the desktop, via email, or from camera phone. The same applies to short videos.
Edit	Perform basic photo editing using Flickr's partners Picnik (e.g., remove red eye, crop a photo, add text, and create photo effects).
Organize	Organize photos and videos using the notions of collections, sets, and tags.
Share	Create or join groups and use privacy controls to share photos and videos with others.
Maps	Map the photo and video locations onto the maps (geo-referencing) and view other photos that were taken nearby.
Make Stuff	Use photos to create cards, photo books, framed prints, DVDs, and so on.
Keep in Touch	Get updates from family and friends.

13.2.3 Flickr Groups

On Flickr you can create groups and enable a set of friends, and other Flickr members, to share and comment on the groups' photos. Groups bring subcommunities together around a common interest. For example, a group may be interested in photos about a particular theme, specific photographic techniques, and popular locations. Flickr makes it easy to join an existing group or start a new one.

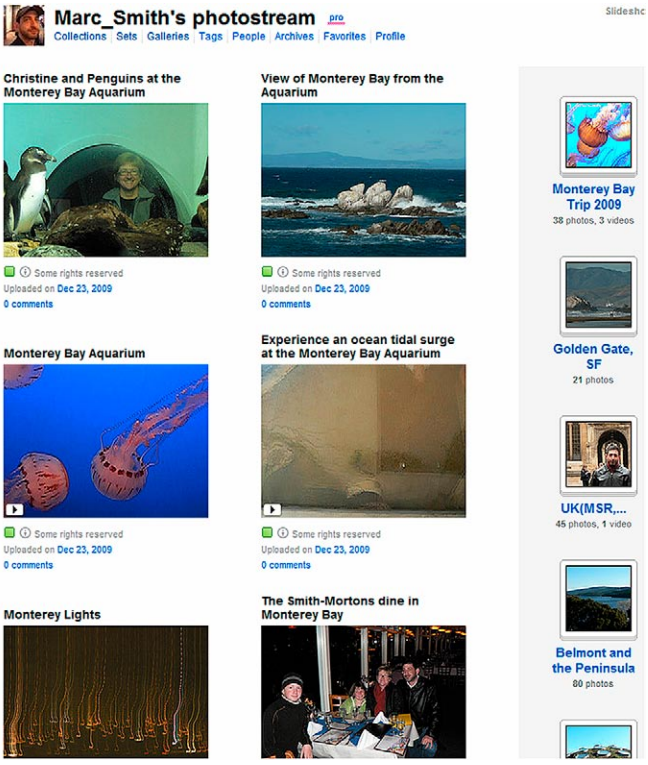


FIGURE 13.4 The Flickr photostream shows photos that the user has recently posted. On the right are sets of photos, sorted by recency.

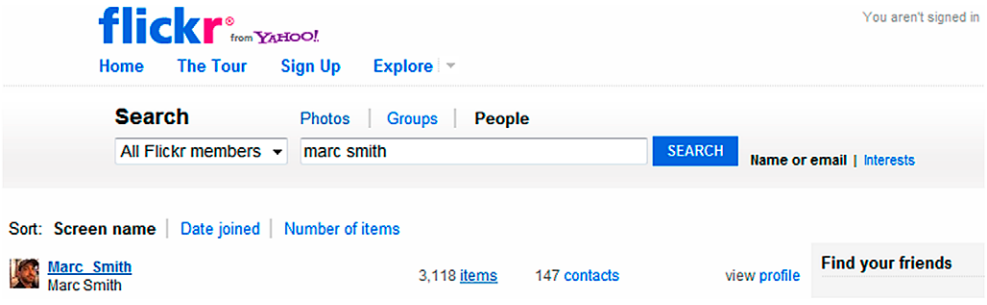


FIGURE 13.3 Flickr members search.

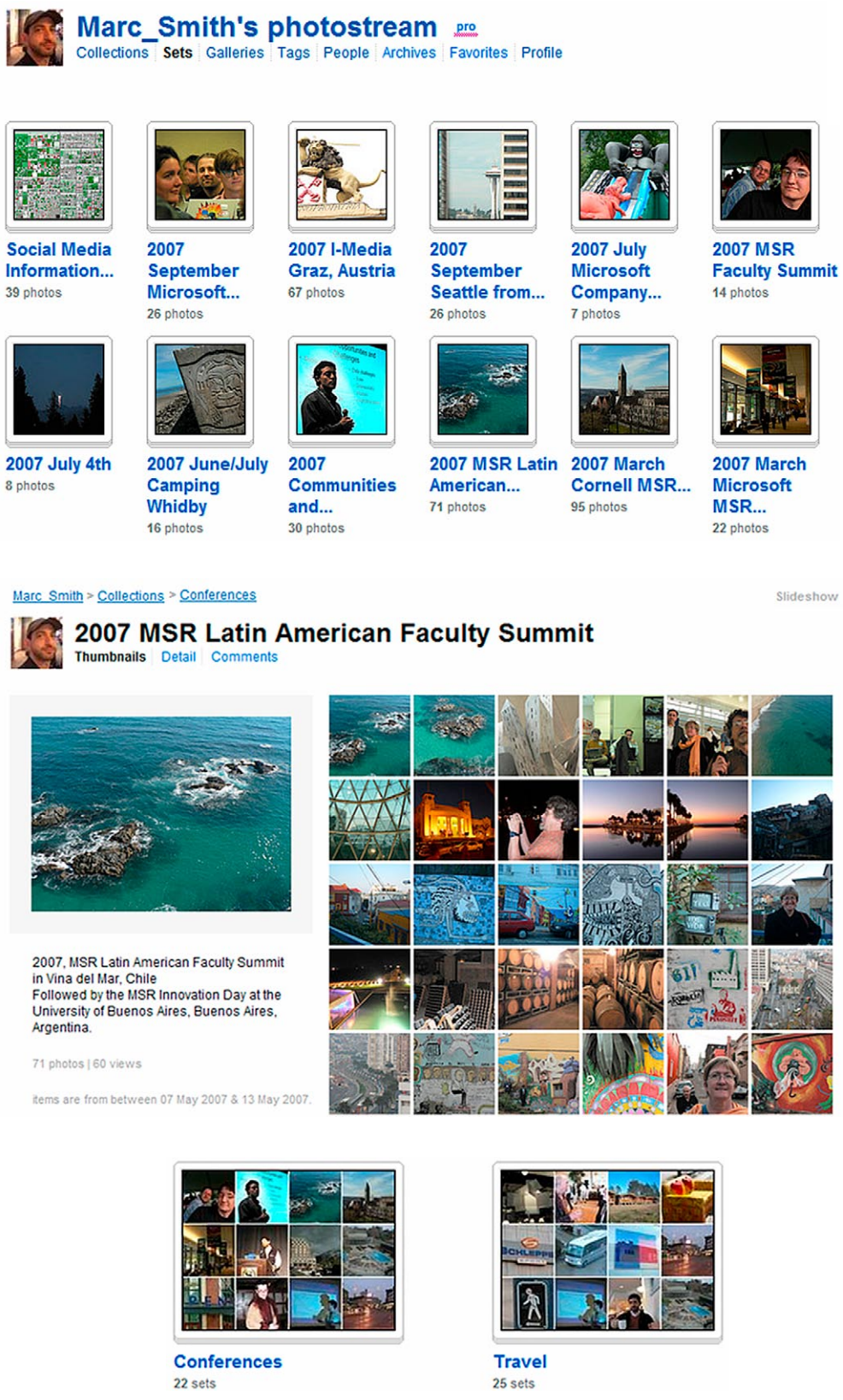


FIGURE 13.5 Top figure shows multiple sets of photos in Flickr. The middle figure shows individual images in the "2007 MSR Latin American Faculty Summit" set. The bottom image illustrates collections containing several sets.

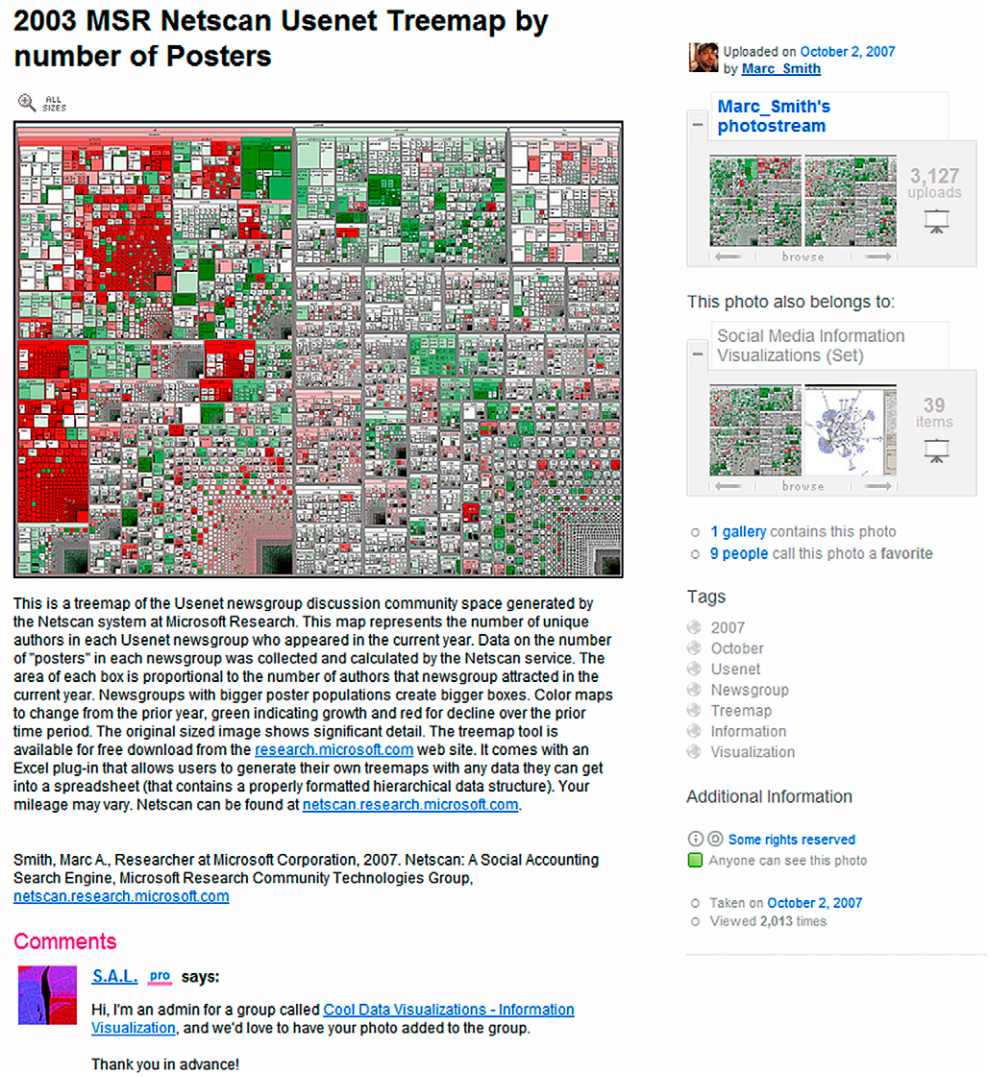


FIGURE 13.6 Flickr image containing seven tags and an extensive description. Nine people liked the photo and called it "Favorite." The image was included in one gallery and received one comment.

Groups can be public (i.e., completely open, public by invitation only) or private. Each group has a "Pool" for sharing photos and videos and a "Discussion Board." You can post a photo to a group using a simple "Send to Group" button located between the photo title and the photo. Once you post to the group, only you or the group administrator can remove the photo.

If a group is public, it will be listed in multiple locations through the Flickr site (e.g., on the Groups page, User Profile pages (Fig. 13.7), and any page that shows photos submitted to the group pool). Private groups are not listed anywhere on the Flickr site and new members can join only by invitation. Generally, on Flickr you can set the privacy level for each individual photo or video. Privacy, in turn, determines the visibility and usage licenses, ensuring that the copyright ownership is clearly stated. If you mark a photo as "private within a group," the members of the group will be able to add comments, notes, and tags but the photo will not be displayed to nonmembers. Normally, with a free Flickr account you can add your photo to 10 different group pools. With the Flickr Pro account this is increased to 60 group pools.

13.2.4 Sharing and Social Interaction

Besides sharing photos within groups, you can also share private photos with your friends. If they are not Flickr members, you can create a "Guest Pass" to grant them access with a special URL web link.

To keep in touch with other users on Flickr, you can add them as "Contacts." You can declare contacts to be friends or family (or both). Using the Contacts page, or by subscribing to a daily email, you can easily keep up to date with what your contacts are doing. You can interact with them using notes and comments. Notes allow contacts to leave messages directly on the surface of photos and videos. They appear when the mouse moves over the image and typically identify people, objects, or locations. Comments are positioned below the image to facilitate further discussion (Fig. 13.6).

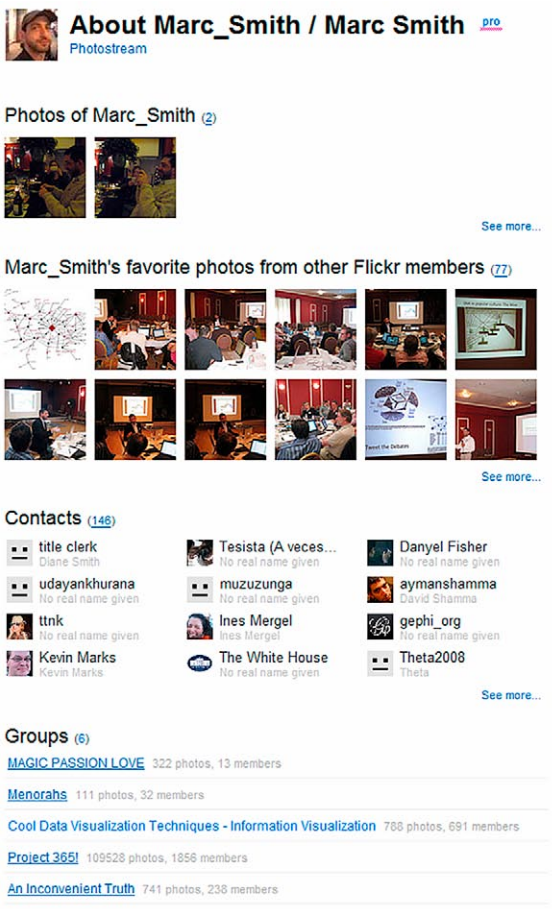


FIGURE 13.7 The user's Flickr profile page shows the number of contacts (146) and group memberships (6).

If you truly like a person's photo and wish to express your appreciation, you can include the photo on your "Favorite" photos list. This is a form of social approval, and Flickr uses it as one of the metrics to identify "interestingness" of a photo. It computes interestingness based on a set of measures including the click-through, the number of comments, and the number of times the photo was chosen as a favorite.

13.3 FLICKR NETWORKS

Through establishing contacts, commenting on photos, and joining Flickr groups, you can develop a substantial social network and increase your social presence. In fact, Flickr allows you to monitor your activities. It creates reports that include the number of views each photo had within a given time period or the list of comments it received. Thus, you can get a sense of your contribution to the community. It is quite rewarding to see how many other members liked and marked your photo as "Favorite." However, at the moment you can learn very little about the Flickr community as a whole. For example, it is hard to get a sense how well you are connected with the community compared to others, or how social ties in Flickr evolve over time, how they affect distribution of photos across groups, and what the structure of the social network is. You might be interested in understanding your own social network that arises from interactions with others. For example, over time you may develop new relationships within Flickr and your engagement may be affected by existing relationships [4]. Your friends may be more likely to provide comments to your photos, and vice versa, than engage with other Flickr users. Social network analysis may reveal interesting patterns, including different roles that users play and the different ways in which they contribute to the evolution of the social network [5, 6].

It is particularly interesting to observe how Flickr users assign tags to photos and how that practice differs across users and photos. With a lack of predefined vocabulary and strict guidelines, it is likely that different users will choose a different set of tags for the same photo. The opposite also holds. The same tag may be used for photos with very different semantic and contextual meaning. By allowing users to assign multiple tags to each photo, Flickr increases the diversity of tags but at the same time increases the chance that photos will share tags if they are in any sense related. Indeed, even though different users assign tags independently, there is typically convergence in tag usage [7]. Tags that co-occur across photos often serve as good descriptors of a photo collection and can be used effectively for browsing and search.

By inspecting common tags and their association with photos and users, you can gain good insights into the characteristics of both the Flickr community and the content. For example, you may want to analyze the following elements:

- *Social relations in Flickr*. Users create *explicit relations* (e.g., by belonging to the same Flickr group or by adding each other as a contact) and *implicit relations* that correspond to social activity (e.g., commenting on each other's photos, assigning the same tags to a photo, and performing similar activities).
- *Content structure in Flickr*. Photos also have explicit relations, such as belonging to the same collection, classified under the same tag, or both flagged for their "interestingness" by Flickr. The tags themselves are implicitly grouped into clusters based on their co-occurrence across photos or common use across users.

All these relationships between pairs of users, photos, tags, or groups arise from users' participation in Flickr and can be represented as networks (Fig. 13.8).

Suppose you have Flickr data in the form of *user-photo-tag* triples—that is, for all Flickr users you obtain a list of all photos that they uploaded to Flickr

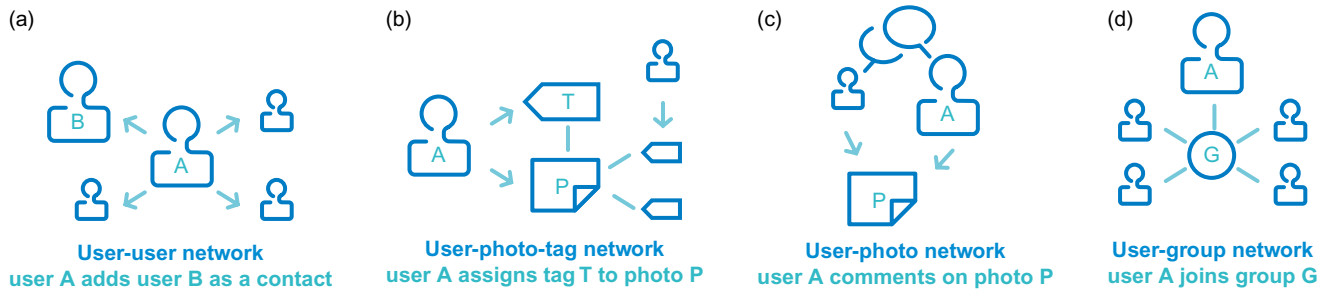


FIGURE 13.8 Networks arising from users' participation and content contributions on Flickr. (a) Adding contacts. (b) Authoring and tagging photos. (c) Adding comments to photos. (d) Joining a group.

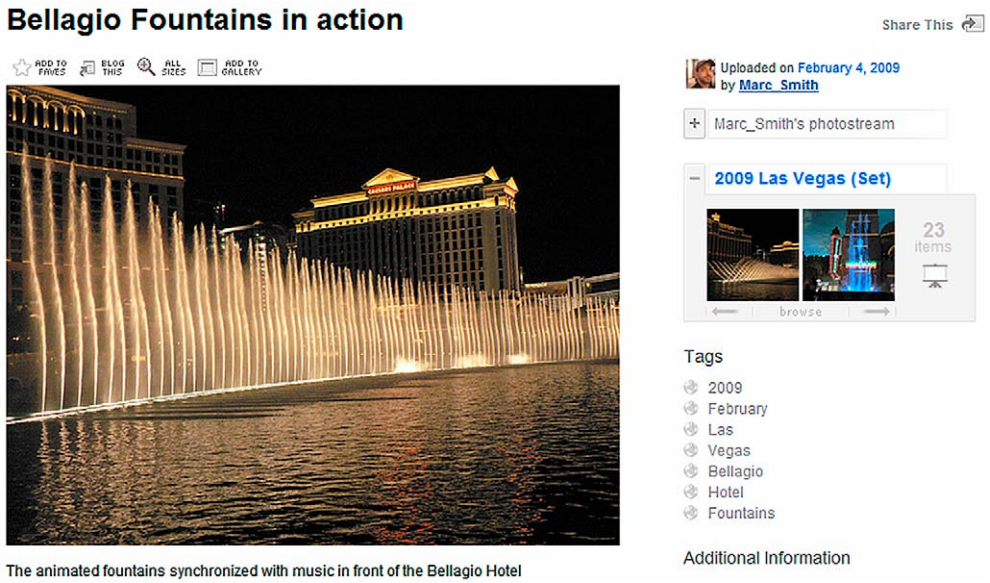


FIGURE 13.9 Tagged photo of the Hotel Bellagio fountains in Las Vegas.

and all the tags they assigned to their photos. You can then derive implicit *user-user* relationships by connecting pairs of users who applied the same tag across several photos. Similarly, you can obtain an implicit *photo-photo* network by connecting pairs of photos that share one or more tags. Finally, you can also derive a *tag-tag* network, connecting pairs of tags that co-occur frequently across images or across users.

Although there are many other implicit networks that could be defined and analyzed, this chapter explores the *tag network* that results from the association of tags with photos, and two user networks, one resulting from *user contact* lists and the other from *user comments* on photos that have been posted by a particular user.

13.3.1 Tag Networks

Flickr members assign tags to their photos at the time they upload them to Flickr. With the help of Flickr tools, they can easily assign tags to multiple photos in a set. They can also apply multiple tags to a photo and thus capture all-important aspects of the photo and its context [2], as shown in Figure 13.9. In this example, the photo contains seven tags indicating where and when it was taken and what specific objects it shows. People interested in photos of Las Vegas or fountains can easily find this photo by using the tags.

Tags are often conceptually related to one another. It would not be surprising if other users who took photos of the Hotel Bellagio applied the same tags "Bellagio" and "Vegas." Indeed, the hotel is located in Las Vegas and the association of the two tags reflects the geographical proximity. Moreover, the use of "Bellagio" and "Vegas" is likely to reflect the relationship between the entities they represent (i.e., the hotel located in the city). Photos with the tag "Bellagio" are likely to contain the tag "Vegas" as a reference to the city. However, many photos tagged with "Vegas" may not refer to "Bellagio." "Vegas" may co-occur with "Caesars Palace," "Venetian," or names of other hotels in Las Vegas. In that sense, "Bellagio" can be considered a more specific tag compared to "Vegas."

Following that observation, it is useful to define the relationship between tags by the proportion of photos that contain each of them. More precisely, two tags that are

applied to the same photos are considered to be related. The relationship is generally asymmetric and can be represented as a bidirectional link with different strength values in each direction. When analyzing tag networks, it is often useful to filter out tags that are generic and focus on more specific ones. The co-occurrence of specific tags in photos is likely to reveal inherent associations among concepts that the tags represent [7] and identify tag clusters that correspond to different meanings of the same tag. Indeed, these clusters of co-occurring tags can help disambiguate the themes or subjects of the tagged photos.

13.3.2 User Networks

User Contacts

In Flickr, users can establish social ties with other users by adding them as contacts (Fig. 13.8(a)). The ties may or may not be reciprocated. The contact relationships can be represented as a social network, showing links between users and their contacts. For two users who included each other as contacts, the link is bidirectional, indicating a reciprocal relationship. Analyzing the network properties, in particular the reciprocity of the contact relationship, helps one to understand the connections of individuals within the community as well as the properties of the community itself.

Users That Comment on Other Users' Photos

Flickr enables users to add comments to photos (Fig. 13.8(c)). Members of individual Flickr groups, for example, are quite active in that respect. They contribute to a pool of photos that all group members can see and comment on. More generally, a person can add comments to any photo that is publicly available on Flickr.

When posting a comment on a photo, the user establishes an implicit contact with the owner of that photo. This interaction can be captured in a user network showing links between users who comment on each other's photos. A "comment link" is, in effect, an expression of interest in the author's photos, possibly with regard to the theme, the quality, or other aspects of the photo. The edges of the network can include time stamps and thus capture the sequence in which users have exchanged comments. Such temporal networks can help explain user interaction patterns over time.

13.4 WHAT QUESTIONS CAN BE ANSWERED BY ANALYZING FLICKR NETWORKS?

The Flickr social media service captures various facets of society. Essentially, Flickr is a reflection of the society through images and the community that has emerged with it and around it. Thus, you can ask yourself questions in different spheres.

13.4.1 Personal Sphere

Looking at your network of Flickr contacts, you may wish to examine the friendship ties and see whether links are reciprocal. You probably expect a large number of reciprocal links with your friends who share interest and engage with you regularly by commenting on photos. Nonreciprocal connections may be of followers, distant friends, or family members who have less affinity to engage in social media.

Considering the network of photos, tags, and comments, you can create clusters around content themes to help with introspection and making sense of content.

13.4.2 Community Sphere

You might be able to discern your friends from others based on their choice of tags for particular photos. It is likely that your group of friends will converge toward common vocabulary and start reusing each other tags.

In Flickr, users are encouraged to create groups. Some members serve as administrators, moderating participation and inviting or banning people from the group. How would you manage a large Flickr group? How would you assess whether the group is lively and healthy. How would you identify important individuals?

First, you can use NodeXL and social network analysis to gain insights about the behavior of any member. For example, you can always check whether users have membership in other groups and observe their track record in such groups. Second, you may want to ensure the group stays focused in its central theme. By analyzing tags, comments, and images, you can see whether there is shift and assess how many people may become alienated if the trend continues. Exposing information to the community may cause self-correcting mechanisms to be activated. Finally, suppose that you need to leave a group and transfer the administrative duty to others. You may want to analyze the group's social network and identify the most authoritative person in the group and those who are well connected, based on the network of comments and replies.

13.4.3 Application Sphere

E-commerce

Influence from friendships in social media is considerable. You may ask, would friends follow friends in their actions? If a user submits an image of the latest car model, will others follow? How will that influence a person's buying a particular car brand, or even

the whole group? The study by Singla and Weber [8] revealed congruence among Flickr friends when it comes to the choice of a camera brand. Product brands may want to analyze the friendship social structure and identify cliques among them. Such insights into the network structure can inform the launch of new products. Businesses could identify the key influencers and connectors and carry out targeted product campaigns.

Service and Infrastructure

From a relatively simple analysis of data usage on Flickr, you can conclude whether a person posted photos for social or archival purposes. In the study by Van House [9], it became clear that many are using Flickr as personal galleries, more or less transient and changing in nature. For a service like Flickr it is important to determine the required infrastructure to support the archival function versus frequent posting and commenting on smaller but socially significant media sets.

Geo-Tagged Applications

Flickr is accessible from different platforms. Thus, geotagged images can be inserted and shown on a geographical map. This provides powerful additional information that can enable specific usage scenarios, from a disaster response to leisure planning. Crandall et al. [10] used content and network analysis on more than 35 million images and began to reveal what matters to us collectively and over time. This can be used for many purposes, from e-commerce to education to policy making.

13.5 IMPORTING FLICKR DATA INTO NODEXL

Flickr makes its data easy to obtain through API services that programmers can use to automatically collect information about users, photos, tags, groups, and the like. NodeXL connects to some of these API methods to import networks and associated metadata directly into the NodeXL network template workbook. This section will walk you through the steps of importing the data that are used throughout the chapter.

The first step is to open or create a new NodeXL template workbook. That will activate the NodeXL ribbon next to the existing Excel ribbon tabs. The data import options are located in the Import menu, in the Data group at the leftmost part of the NodeXL ribbon, as shown in Figure 13.10.

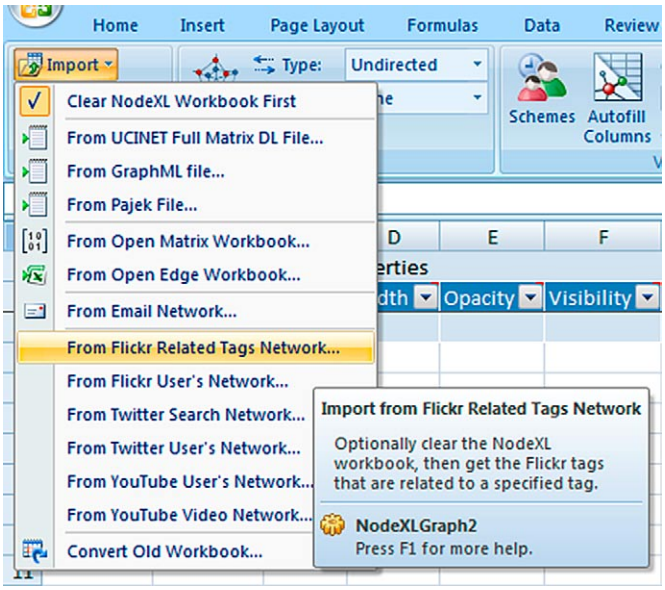


FIGURE 13.10 NodeXL Data Import menu.

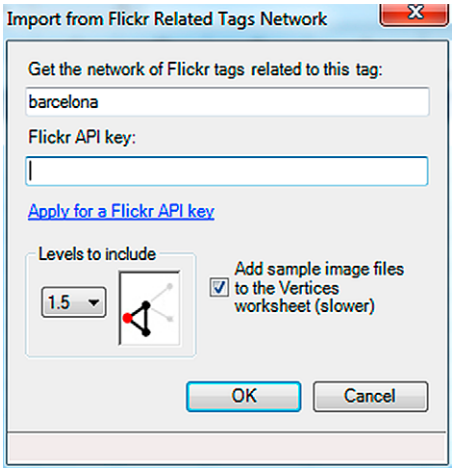


FIGURE 13.11 NodeXL Import from the Flickr Related Tags Network window. Get all tags related with the "barcelona" tag, downloading a sample Flickr picture for each tag.

13.5.1 Related Tags Network

Selecting the option "From Flickr Related Tags Network" opens a new dialog window (Fig. 13.11). This window includes a field to enter the tag for which you want to retrieve the related tags (e.g., "barcelona") and another field to specify the Flickr API key. Flickr requires that all the data requests use an API key, which needs to be obtained from the Flickr web site. In case you do not have one yet, use the link provided in the import window to apply for a Flickr API key. Once you receive it, enter it in the appropriate field and import Flickr networks into NodeXL.

The import dialog also allows you to specify the levels to include in the related tags network. The levels refer to the depth of the *egocentric* network of the specified tag. NodeXL retrieves the tag and all of its related tags and creates edges between them.

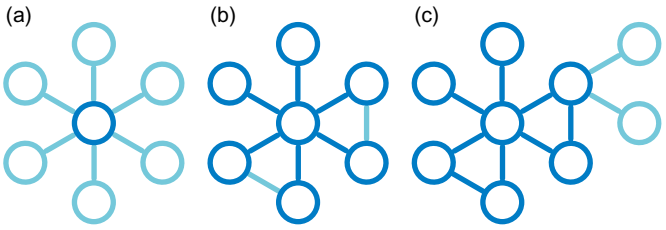


FIGURE 13.12 Egocentric network with a depth level equal to n. (a) n = 1. (b) n = 1.5. (c) n = 2.

Suppose that the specified tag corresponds to the dark blue vertex in the center of Figure 13.12(a) and that the other light blue vertices represent related tags. In this case, the depth level of the egocentric network is 1 because it includes only the edges to and from the central vertex. In Figure 13.12(b), the network also shows edges between related tags, indicating that these tags are related not only to the central tag but also among themselves. This network has a depth of 1.5. Figure 13.12(c) shows a network of depth 2 and includes tags that are not directly related to the central tag but are linked to the level 1 tags.

You can instruct NodeXL to download a sample photo for each tag in the network by checking the box in the dialog window. That enables you to create network visualizations that show the tag vertices as images.

NodeXL imports the tag network by inserting tags into the Vertices worksheet and pairs of related tags as edges into the Edges worksheet. It adds the URLs for the sample image files to the "Image File" column and inserts the links to the Flickr pages for individual tags into the "Custom Menu" column of the Vertices worksheet. These links appear in the NodeXL graph canvas on a context menu, whenever the graph vertices are selected on a mouse right-click.

13.5.2 Flickr User Networks

Selecting the option "From Flickr User's Network" (see Fig. 13.10) opens a slightly different dialog window, as shown in Figure 13.13. In this case, the first input field allows you to specify the name of the Flickr user that is going to be at the center of the social network. The API key and the network levels are set as explained in the previous section.

You can select two types of user networks to import, both mentioned in the Section 13.3.2: (1) the contacts network for the specified user and (2) the network of users who commented on the photos of the specified user. After selecting the network level, you can choose two more aspects of the network. By checking the first box you can add more information for each user in the network, including the real name of the user, the number of photos uploaded to Flickr, whether the Flickr Pro

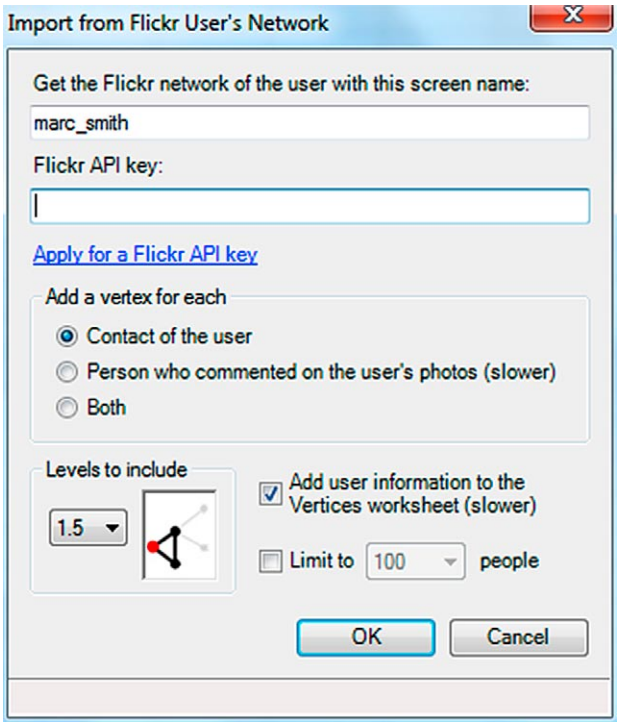


FIGURE 13.13 NodeXL Import from Flickr User's Network window. Get all the contacts of the user "marc_smith," along with information about each user and the respective profile photo.

status was attained, and the URL to the profile photo. The second box enables you to limit the size of the network to a predefined number of vertices in case you expect that the selected network level will lead to a very large dataset.

The time required to download all the data depends on the size of the network. In case of users with many contacts, or users with many photos that receive many comments, the data download can take a while to complete. NodeXL inserts usernames into the Vertices worksheet and pairs of usernames as edges into the Edges worksheet. Each edge represents a comment added by the first user to a photo posted by the second user and includes the time stamp of the comment.

NodeXL stores the URL to the Flickr page that contains the comment in the "Comment URL" column and the time stamp in the "Comment Time" column (Fig. 13.14). It adds a "Relationship" column to the Edges worksheet to indicate the type of relationship, "Contact" or "Commenter," between each pair of users. NodeXL also includes in the "Image File" column the URLs of the Flickr users' profile photos. Furthermore, it inserts the links to the users' profile pages into the "Custom Menu" column of the Vertices worksheet. The "Real Name," "Total Photos," and "Is Professional" columns store information about each user.

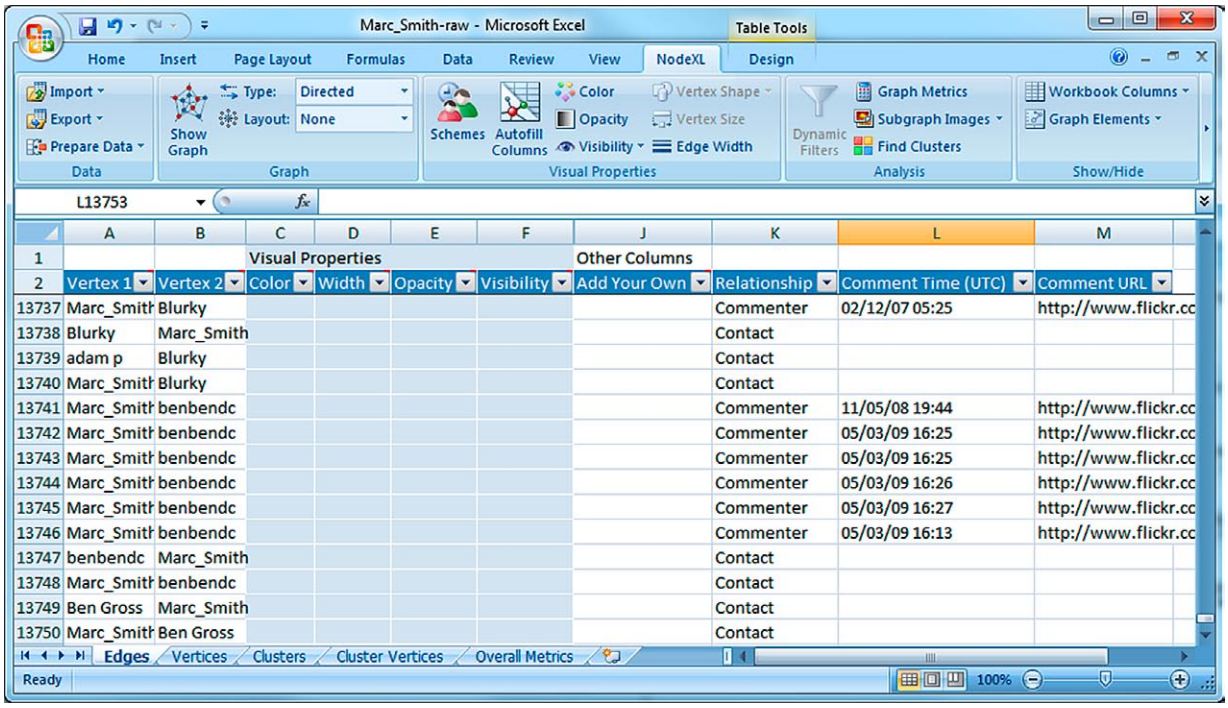


FIGURE 13.14 Raw data in the NodeXL Edges worksheet after importing both user networks (contacts and comments). The Relationship column indicates the type of relationship between each pair of users.

13.6 WORKING WITH THE FLICKR DATA

13.6.1 Graph Type

NodeXL works with directed and undirected networks. You are expected to specify the type of network that has been imported. You can change the type subsequently in the Graph Ribbon group (Fig. 13.15).

Flickr networks, in particular, should all be treated as directed networks, given the way NodeXL fetches the data and the meaning of the edges. However, in some instances it is useful to convert directed networks into undirected ones. By setting the Graph Type to "Undirected" informs NodeXL not to draw arrows in the graph edges and to treat the network as undirected when calculating graph metrics.

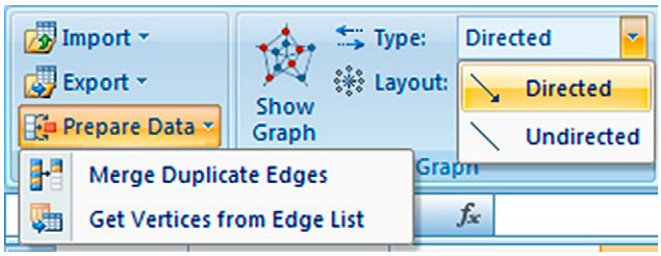


FIGURE 13.15 NodeXL Prepare Data and Graph Type drop-down menus.

13.6.2 Merging Duplicate Edges

The conversion from directed to undirected network is likely to result in duplicate edges due to reciprocal ties that often exist in tag and user networks. Selecting the "Merge Duplicate Edges" option in the Prepare Data menu (Fig. 13.15) aggregates duplicate edges and assigns an edge weight to each edge. NodeXL creates an "Edge Weight" column in the Edges worksheet and assigns a weight of 2 to edges that were bi-directional and weight of 1 to those that were not. In cases where the Edges worksheet already includes an "Edge Weight" column, NodeXL merges duplicate edges and sums the individual weights. Any extra edge data in the worksheet, including time stamps and URLs, are lost in this operation. The weights can be used to aid the network analysis and visualization.

13.7 ANALYZING FLICKR NETWORKS WITH NODEXL

This section presents several examples of NodeXL visualization and analysis of Flickr networks. The first two sections explain how the related tags network can be used to (1) identify landmarks and tourist attractions for a given location and (2) disambiguate tag meaning by identifying clusters of related tags. The remaining section focuses on user networks that result from users' interactions through comments and demonstrates several approaches to analyze them.

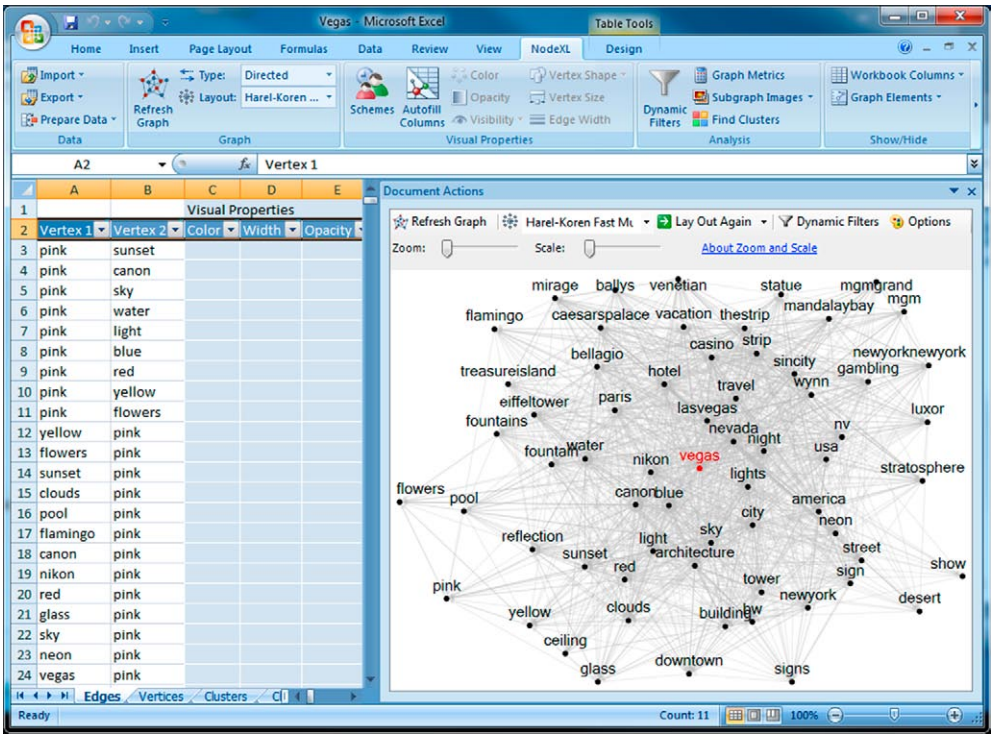


FIGURE 13.16 Raw NodeXL visualization of the Flickr related tags network for the "vegas" tag, which is shown selected selected in the center of the graph.

13.7.1 Revealing Landmarks and Tourist Attractions Using a Location Tag

Consider the following scenario. You have an opportunity to visit Las Vegas for the first time. As a Flickr user you might want to see what others in the Flickr community have visited and captured in their photos. To get a sense for typical places, landmarks, and tourist attractions you could look for tags related to the tag "vegas" and view photos associated with those tags. To create visualizations of the related tag network for "vegas" in NodeXL you can follow the steps described below. The same process can be applied to any other location tag.

Step 1. Import Data into NodeXL

Import data into a new template by specifying the location tag, checking the box to include sample pictures, and selecting 1.5 depth level. Set the Graph Type to "Directed" and choose the "Harel-Koren Fast Multiscale" layout.

Hit the Show Graph button to obtain the first visualization. The Graph Canvas will open displaying a graph similar to Figure 13.16 where the related tags are all visible. However, the graph is very dense and it is hard to explore connections between tags. The density of the graph is calculated and displayed in the Overall Metrics worksheet when you select the Overall Metrics option in Graph Metrics. The density measure ranges between 0 and 1 and reflects the number and distribution of edges in a graph. For example, a graph where every vertex is connected to all other vertices has maximal density of 1. The "vegas" graph has a density of 0.584, which means that, on average, each tag is connected to more than half of the other tags.

The following steps walk you through an approach for filtering out edges and uncovering tag clusters, which group tags that are strongly associated.

Step 2. Prepare Data

As observed in Section 13.3.1, the relationship between two Flickr tags can be modeled by looking at the proportion of photos associated with each tag. The relationship is typically asymmetrical. For example, "pink" is strongly related to "flamingo" but "flamingo" may not be as strongly related to "pink." This results from the aggregate tagging behavior across users in the Flickr community. The community tends to assign the tag "pink" to photos that are also tagged with "flamingo." However, "photos" with the tag "pink" are not equally likely to get the tag "flamingo" because pink can be used in many contexts unrelated to flamingos. Whenever there is a strong semantic connection between two tags, the edge that connects them is bidirectional. If the relationship is nonreciprocal, the direction of the edge is likely to flow from the more specific tag to the less specific one. You can explore the Flickr tag network by filtering out weak semantic connections and displaying only strong connections among tags. To prepare the data for the analysis, set the Graph Type to "Undirected" and then "Merge Duplicate Edges." An edge weight equal to 2 now indicates that the tag pair was originally connected bidirectionally.

Step 3. Compute Graph Metrics and Filter Data

Compute the graph metrics for the network, including degree and betweenness centrality (see Chapter 5). Then, open the Autofill Columns dialog to filter out edges that are not bidirectional, by mapping the Edge Visibility column to the Edge Weight column. Include only edges with weight greater than 1. Skip the vertices that are related to many tags (i.e., that have high degree) because they are too generic for identifying strong connections between tags. You can first use dynamic filters to identify a reasonable cutoff for Vertex degree. The current example uses a cutoff of 45, which vertices are shown below.

Step 4. Visualize the Tag Cloud Graph

The next step is to map graph metrics onto visual attributes of the vertices. Set the vertex label equal to the vertex column, which contains the tag name, and the vertex size proportional to the degree. This creates a "tag cloud" effect on the graph, where tag size varies with the number of related tags (see Fig. 13.2). Set the Vertices Shape to "Label" either through Autofill Columns or directly on the spreadsheet. Figure 13.17 shows the tag graph after using the Harel-Koren Fast Multiscale layout and manually repositioning a few of the vertices to avoid occlusions. After you are happy with the layout, it is recommended that you set the layout to "None." This enables you to preserve the location of the vertices if Autofill is used again. Alternatively, you can lock the position of the vertices. Figure 13.18 shows the same view of the graph after changing the vertex shape to "Image."

Step 5. Interpreting the Visualizations

The graphs in Figures 13.17 and 13.18 reveal interesting facts about Las Vegas. The tags refer to the names of several hotels located in the Las Vegas Strip, as well as various points of interest in these hotels.

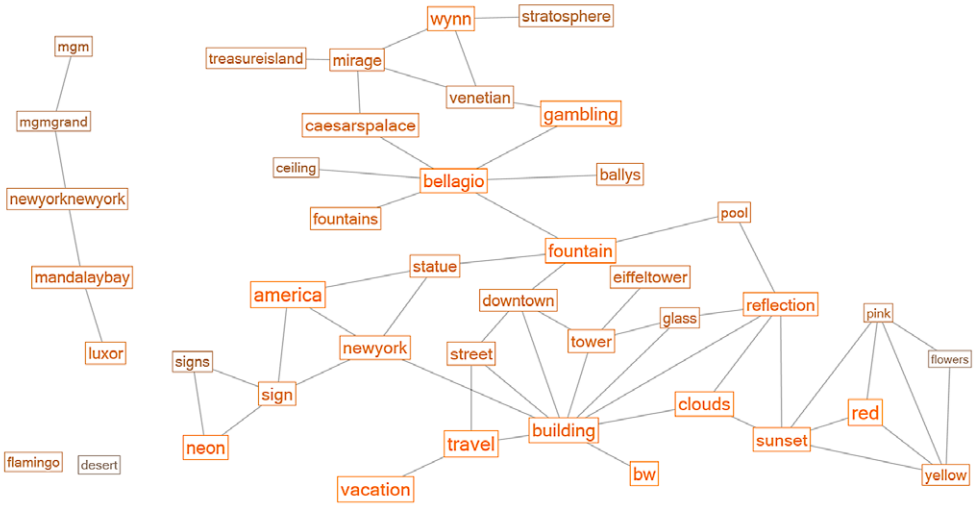


FIGURE 13.17 Filtered NodeXL Flickr tag cloud network. Shows edges with weight >1 and vertices with degree <45. Vertex size is proportional to degree and color is proportional to betweenness centrality (min: gray; max: orange).

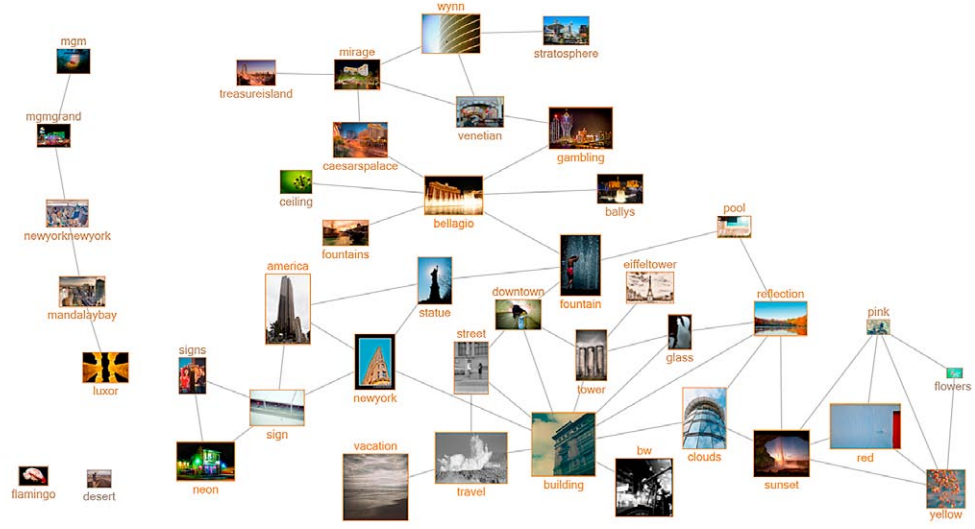


FIGURE 13.18 Filtered NodeXL Flickr tag photo network. Shows edges with the weight >1 and vertices with the degree <45. Vertex size is proportional to degree and color is proportional to betweenness centrality (min: gray; max: orange).

The visualizations suggest relationships between pairs of hotels that seem to be based on their proximity in the Las Vegas Strip. This can be confirmed from the map below. Treasure Island, Venetian, and Mirage, for example, are linked in the graph and located in the vicinity of each other.

"Fountains" are connected to "Bellagio." The Bellagio water fountains are one of the most popular attractions in Las Vegas, renowned for their intricate choreographed shows with lights and music. The connections between "fountain," "pool," and "reflection" refer to the Bellagio fountain pool. The fountain lights and the Bellagio building are reflected in the pool and are best viewed at night. The connection between reflection and sunset is thus intuitive.

"Gambling" appears connected to "Bellagio" and "Venetian." Las Vegas is known for its casinos, and these two hotels are among the most popular.

The "New York" tag refers to the New York New York Hotel and is connected to the "statue," "neon," and "signs" tags. The hotel, indeed, has a replica of the New York Statue of Liberty and signs with neon lights.



13.7.2 Identifying Tag Clusters to Disambiguate the Meaning of a Tag

Consider another scenario. You want to upload your photos to Flickr and want to label them with a good set of tags to make them easily discoverable by other users. You choose the first tag and then decide to explore related tags to get further suggestions. However, you realize that the tag has several meanings and you need to understand the various contexts in which it is used.

This section demonstrates the use of NodeXL automatic clustering and dynamic filters to disambiguate tag meaning. The tag "mouse" is used to demonstrate the methodology.

Step 1. Import Data into NodeXL

Import the Flickr tag network data into a new NodeXL workbook template by specifying the tag "mouse," checking the box to include sample pictures, and selecting 1.5 depth level. Set the Graph Type to "Directed" and choose the "Harel-Koren Fast Multiscale" layout.

After pressing the Show Graph button and setting the default vertex shape to Label, you will obtain a network visualization similar to Figure 13.19. Unlike the previous example, three groups of tags emerge because of a lower graph density, in this case 0.260.

Step 2. Cluster Vertices

One way to view the various contexts of a tag is to identify subgroups of related tags using the automatic clustering algorithm provided by NodeXL. Go to the Analysis Ribbon group and click "Find Clusters" followed by "Refresh Graph" in the Graph Canvas toolbar. The graph will expose clusters by coloring the vertices with the "Cluster Color (Fig. 13.20). You can change the colors manually in the Clusters worksheet.

Step 3. Interpreting the Visualizations

You can follow steps 2 to 4 described in the previous section to preserve only edges that represent reciprocal relationships among vertices and to filter out the vertices with high degree. This process isolates clusters

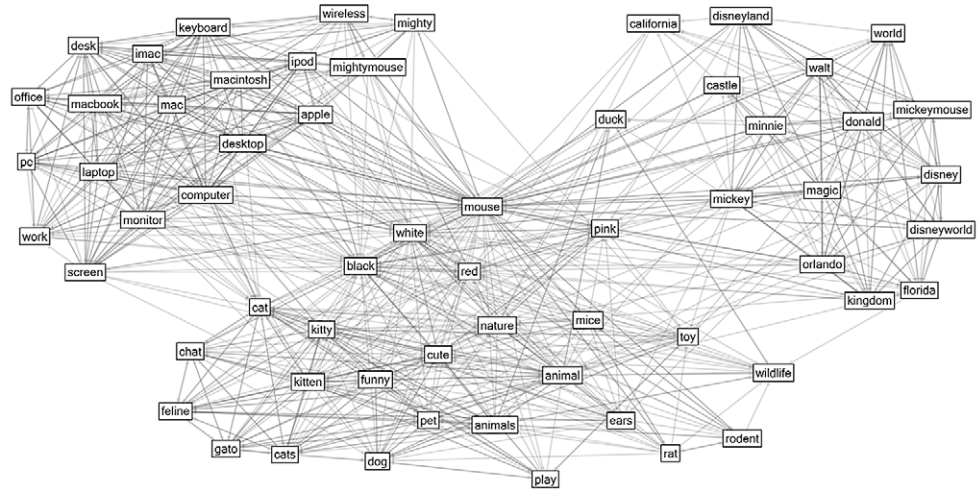


FIGURE 13.19 Raw NodeXL visualization of the Flickr-related tags network for the "mouse" tag.

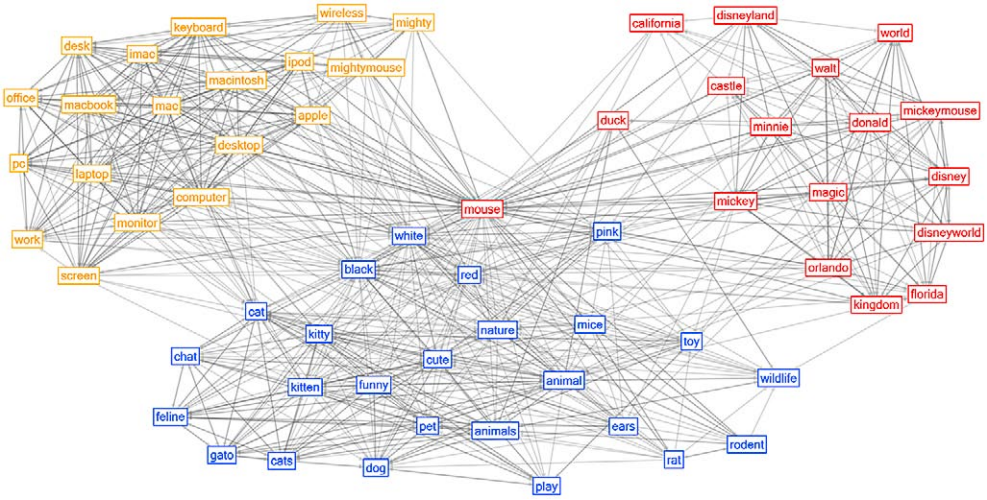


FIGURE 13.20 NodeXL cluster visualization showing three Flickr tag clusters, each representing a different context for "mouse."



FIGURE 13.21 NodeXL display of isolated clusters for three different contexts for the "mouse" tag in Flickr: mouse animal, computer mouse, and the Mickey Mouse Disney character.

as shown in Figure 13.21. The three contexts for the "mouse" tag are quite apparent: the mouse that is a small animal, the computer mouse, and the Mickey Mouse Disney character.

13.7.3 Flickr User Networks

Suppose you would like to get a feel for who posts comments on your Flickr photostream photos and who among them may not be on your list of contacts, for

example. This section illustrates the use of NodeXL to visualize and analyze user networks that result from commenting on each other's photos. The Flickr member "Marc_Smith" is used to demonstrate the methodology.

Step 1. Import Data into NodeXL

Import the user network data from Flickr into a new NodeXL workbook template, specifying the Flickr username (e.g., "Marc_Smith"), selecting the comment network option, checking the box to add user information, and selecting 1.5 depth level. Set the Graph Type to "Directed" and choose the "Harel-Koren Fast Multiscale Layout" in the drop-down list next to the Show Graph or Refresh Graph button. Note this may be set to Fruchterman-Reingold by default.

You can display the users' screen names as vertex labels in the graph to see who is connected to whom. After mapping the Vertex Label column to the Vertex column, using Autofill Columns, and pressing the Show Graph button you obtain a visualization similar to Figure 13.22.

Step 2. Prepare Data and Create New Columns

A Flickr user can post comments to multiple photos from another user at different points in time. In the network representation, these interactions are indicated by multiple edges between the same pair of users, each having a unique time stamp. Given that the current analysis scenario does not focus on temporal aspects of the network, you can ignore the time stamp information and, instead, concentrate on overall interaction. For that, you can merge duplicate edges to obtain an edge weight for each pair of users. Additionally, because users can give comments to other people's photos as well as to their own photos, you can distinguish these two cases. Self-comments are represented in the graph as circular edges (i.e., self-loops) (Fig. 13.22). If you wish to differentiate between these two types of edges, create new columns in the Edges and Vertices worksheets as described in the Advanced Topic: Automatically Identifying Self-Comments and Others' Comments.

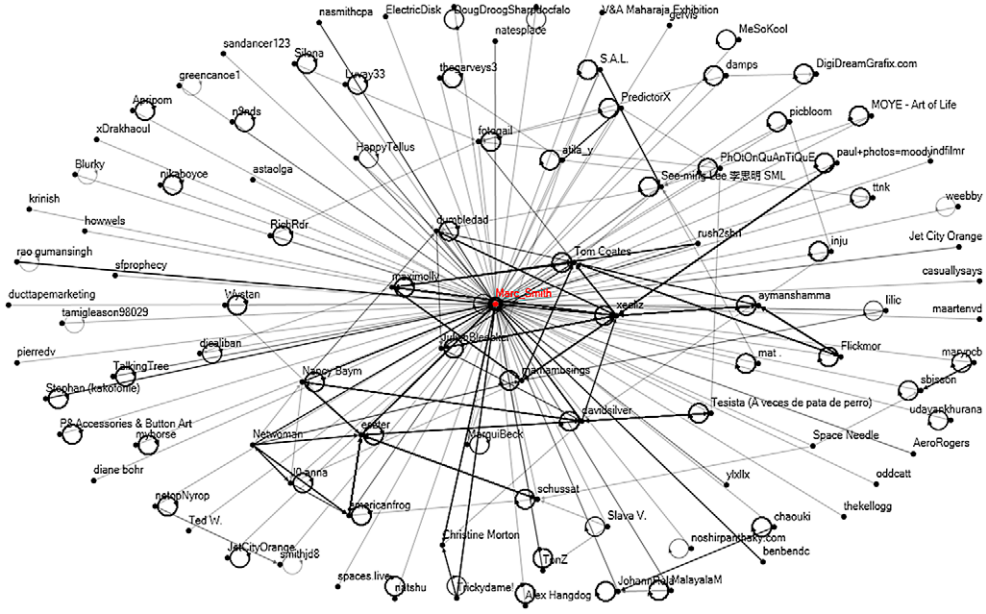


FIGURE 13.22 Raw NodeXL visualization of the Flickr user's comments network for "Marc_Smith."

ADVANCED TOPIC

Automatically Identifying Self-Comments and Others' Comments

New Columns in the Edges Worksheet

The comments that users post on their own photos result in edges where the Vertex1 and Vertex2 cells contain the same username. Create a new column in the Edges worksheet called "Self-Comment," and insert in the first cell the following function:

```
"=IF(Edges[[#This Row],[Vertex 1]]=Edges[[#This
  Row],[Vertex 2]],Edges[[#This Row], [Vertex
  1]],"")"
```

If the comment is a self-comment Vertex1 will be copied into the respective cell in the new column, otherwise and the cell will remain empty.

(*Continued*)

ADVANCED TOPIC (Continued)

Create another column "Edge Weight Others" and insert in the first cell the following function:

```
"=IF(Edges[[#This Row],[Vertex 1]]=Edges[[#This
  Row],[Vertex 2]],0,1)*Edges[[#This Row], [Edge
  Weight]]"
```

In this case, if the edge represents a self-comment the respective cell in the new column will be set to 0, otherwise the value will be copied from the Edge Weights column. This column will be used for filtering out self-comments from the visualization.

New Columns in the Vertices Worksheet

To obtain aggregate statistics about the total number of comments posted by each user listed in the Vertices worksheet, create three new columns "Total Comments," "Comments to Self," and "Comments to Others."

To obtain the total number of comments, insert in the first cell of the Total Comments column the following function:

```
"=SUMIF(Edges[Vertex 1],Vertices[[#This
  Row],[Vertex]],Edges[Edge Weight])"
```

To obtain the total number of comments posted to your own photos, insert in the first cell of the Comments to Self column the following function:

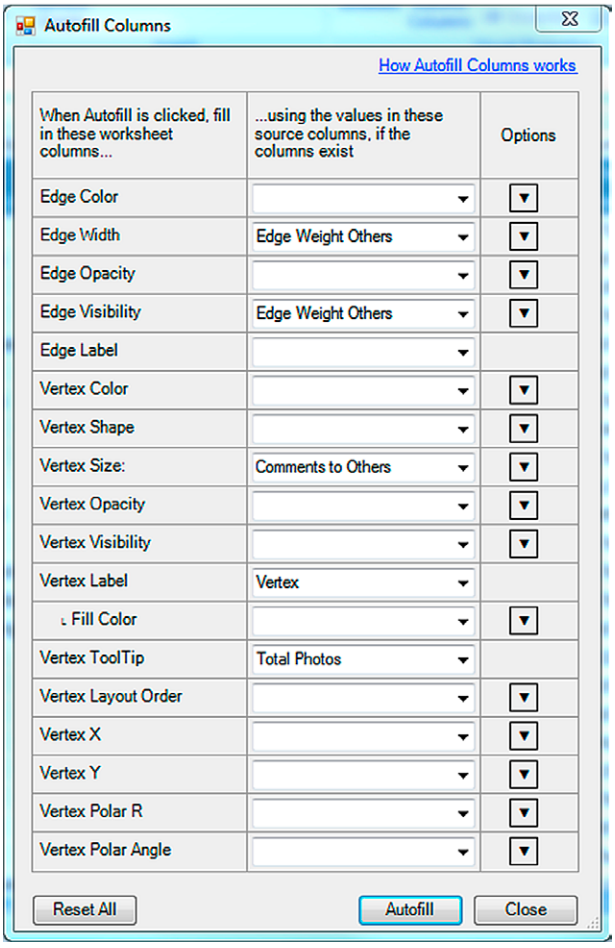
```
"=SUMIF(Edges[Self-Comment],Vertices[[#This
  Row],[Vertex]],Edges[Edge Weight])"
```

The data for the third column is the difference between total comments and self-comments:

```
"=Vertices[[#This Row],[Total Comments]]-
  Vertices[[#This Row],[Comments to Self]]"
```

These columns will be used to set visual attributes of the vertices and for graph layout.

(a)



(b)

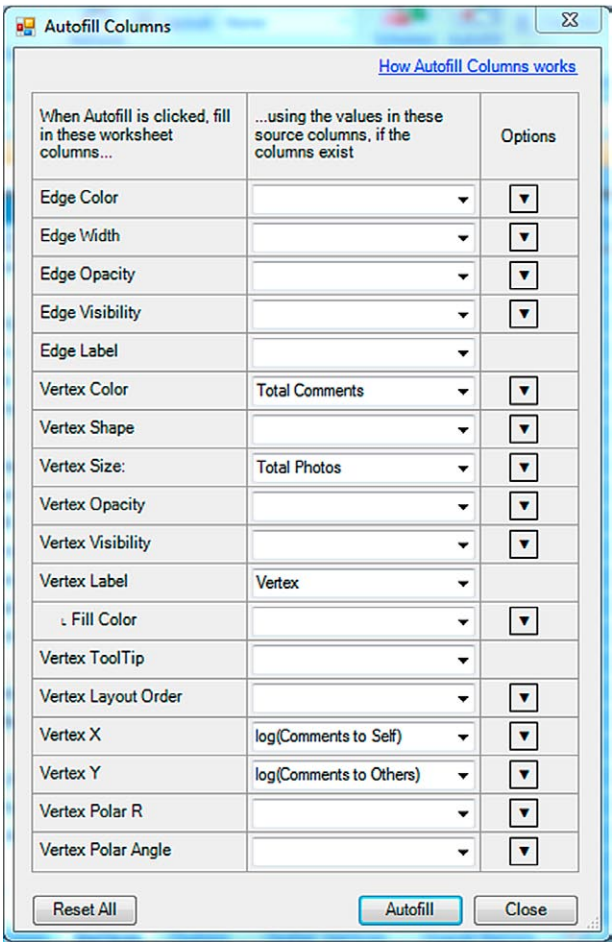


FIGURE 13.23 NodeXL Autofill Columns settings for (a) filtered comments network and (b) comments scatter graph.

Step 3. Filter Data and Set Visual Properties

Some Flickr users exchange many comments, whereas others do so very rarely. You may want to exclude infrequent interactions from the analysis, as they do not contribute much to the characterization of the community as a whole. Open the Autofill Columns dialog and map the network metrics onto the visual properties of the graph, as shown in Figure 13.23(a). The Edge Visibility setting hides self-loops and edges with the weight lower than a specified value, in this case 4. You can first use dynamic filters to identify a reasonable cutoff value for the edge weight. The edge width is proportional to the edge weight and it is set to range between 1 and 3, using a logarithmic mapping. The vertex size is proportional to the total number of comments given to other users. Its range is set between 1.5 and 4.5, using a logarithmic mapping. You can use additional data about the users such as the total number of photos uploaded to Flickr and show it in the vertex tooltip. Set the Vertex Shape column to "Image" to display user profile photos as vertices of the graph.

Step 4. Visualize the User Comments Graph

The next step is to draw the graph. Use the Harel-Koren Fast Multiscale layout and proceed with manual layout adjustments to minimize edge crossings and vertex occlusions. You can also change the positions of individual vertex labels to enhance the readability of the graph. For that, right-click on a vertex and select the "Edit Selected Vertex Properties…" menu option. This will open a window that contains an option to change the label position. Figure 13.24 shows the resulting visualization.

Marked in orange are Flickr users who comment on "Marc_Smith" photos but are not in his contact list. You can obtain and visualize that information by importing the Flickr user contacts network and comparing with the data in the user comments network.

Step 5. Get User Contacts Information

Import the Flickr user's network into a new NodeXL workbook by choosing the "Contacts of the user" network option and setting the depth level to 1.5. Once the Flickr user network is imported, you can use Excel functions to determine who in the comments network is present in the contacts network (see Advanced Topic: Combining Data from Another Network).

Step 6. Visualize the Scatter Graph of Self versus Others' Comments

You can gain further insight into users' commenting behavior by analyzing the number of comments that each user posted to their own photos versus photos from other users. A scatter graph is a suitable visualization to assist in such analysis. For each user vertex, map its X coordinate to the "Comments to Self" column and its Y coordinate to the "Comments to Others" columns (Fig. 13.23(b)). To obtain a global overview, you can show vertices of all the users, including those with few comments to other users' photos. First you need to clear the "Edge Visibility" column so that all the vertices appear again in the graph canvas. Then, to obtain a standard scatter plot, make the edges invisible by setting the default edge opacity to 0 percent. By setting the Vertex Size and Color proportional to the total number of photos and comments, respectively, you can take into account additional information to help with graph interpretation. Figure 13.25 shows the scatter graph after setting the layout to "None" and drawing the graph. You can make the graph axes and legend visible by going to

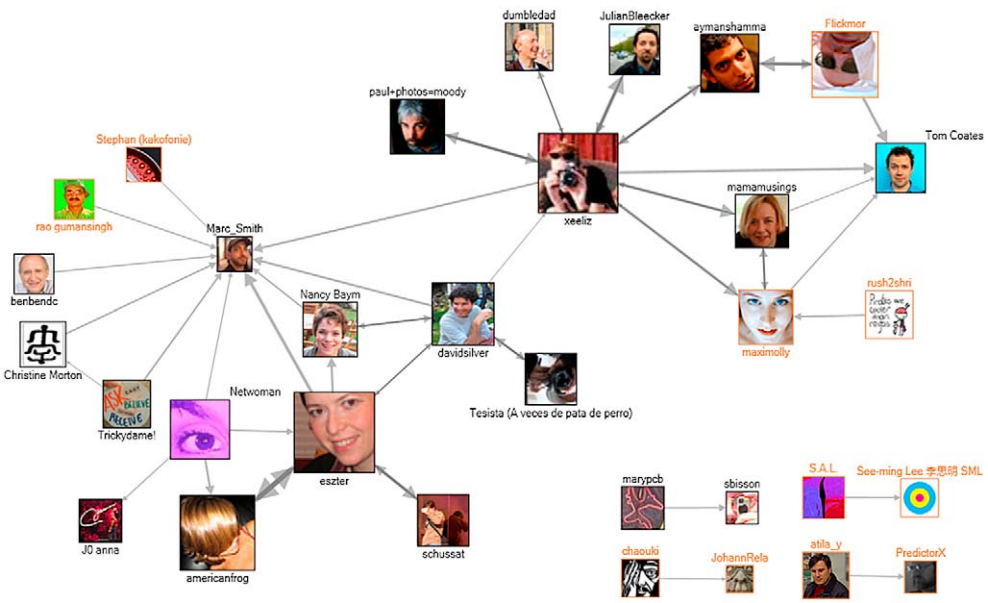


FIGURE 13.24 NodeXL network of Flickr users who comment on Marc_Smith's photos (network depth 1.5; edge weight ≥4).

ADVANCED TOPIC
Combining Data from Another Network

Suppose the contacts network of the Flickr user is saved in the ContactsNetwork.xlsx file and the comments network is saved in the Marc_Smith-Comments.xlsx file. You can create a new column called "Is Contact" in the CommentsNetwork.xlsx Vertices worksheet that indicates whether the user is a contact or not:

```
"=IF(COUNTIF('ContactsNetwork.xlsx'!Edges[[Vertex
 1]:[Vertex 2]], Vertices[[#This Row],
 [Vertex]]),1,0)"
```

The COUNTIF function counts the number of rows in the contacts edge list that include the username specified in the comments network Vertex column. The overall result of the function is 1 if the user is a contact and 0 otherwise. You can use the "Is Contact" column for mapping visual properties of the vertices such as the color. You can also use it to determine the overlap between the comments network and the contacts network by summing up all the values in the column.

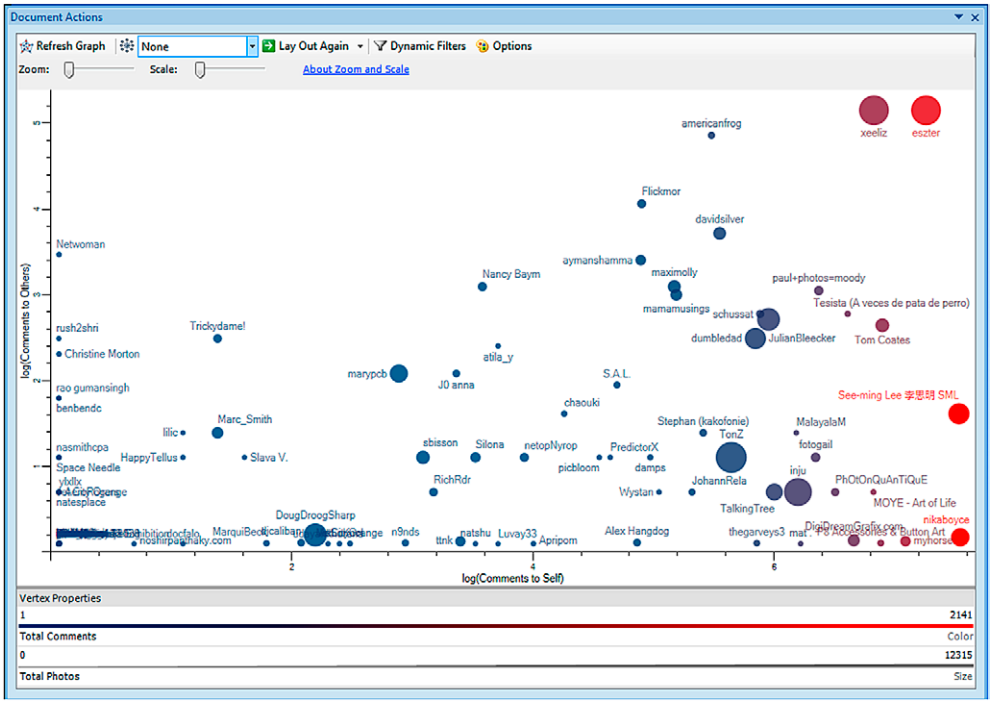


FIGURE 13.25 NodeXL scatter graph of total Flickr comments to self (X axis) versus other users (Y axis), in a log-log scale. Vertex size is mapped to the total number of photos in the user's Flickr photostream, and color is mapped to the total number of comments posted by the user.

the NodeXL ribbon Show/Hide Graph Elements and selecting Axes and Legend.

Step 7. Interpreting the Visualizations

A key finding from the first visualization in Figure 13.22 is that a large percentage of users who commented on Marc_Smith's photos (70.4 percent to be precise) tend to post comments on their own photos, and some do that quite frequently. Self-comments are often used to provide context for the photo, respond to users who commented on the photo, or thank individuals for their appraisal (see the examples in Fig. 13.26). Generally, users with many photos tend to communicate with other users through comments on their own photos (Fig. 13.25). However, some are also very active in giving comments to other users' photos (e.g., americanfrog, xeeliz, eszter).

The second visualization, in Figure 13.24, shows that a relatively small subset of users actively engage with Marc_Smith by commenting on his photos, and most of them are his contacts. Looking more broadly, only 37 out of 99 users who ever commented on Marc_Smith's photos are part of his contact list.

13.8 PRACTITIONER'S SUMMARY

Rich social media, such as Flickr, present opportunities for a wide range of analysis. At the same time, software packages like NodeXL are designed to provide

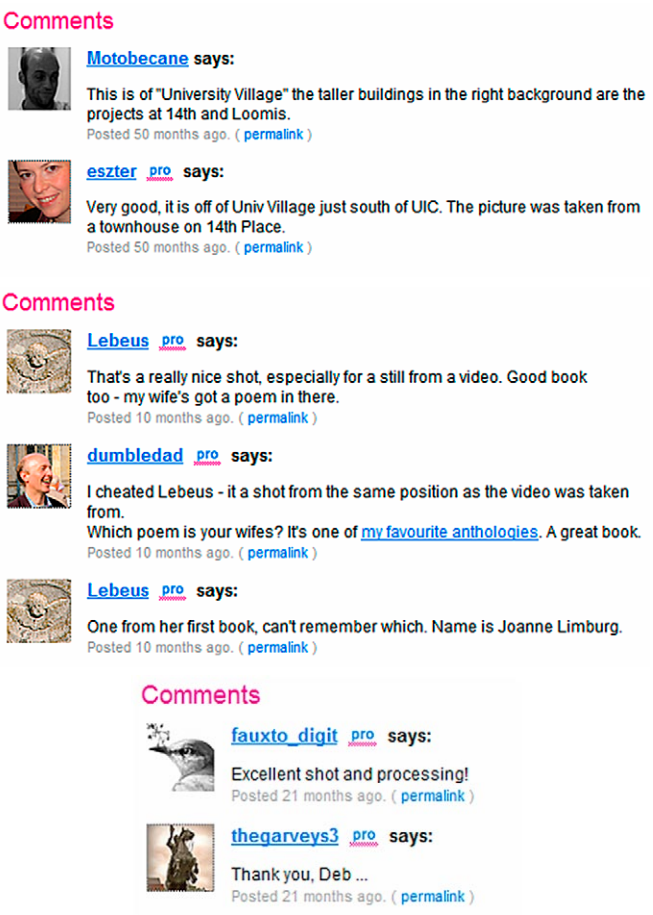


FIGURE 13.26 Three examples of self-comments where the user adds context to the photo (top), responds to other users comments (middle), and thanks other users (bottom).

easy-to-use tools and streamline common types of analysis. NodeXL, in particular, is evolving through a development effort that is informed by the needs of practitioners who provide feedback and new requirements as they reach the limits of the existing NodeXL features. This iterative software design and development process has proven to yield good results, judging from the popularity of NodeXL with its user community.

Effective use of any technology is subject to the practices that users develop and share among themselves. As practitioners ourselves, we here provide some practical advice related to the analysis presented in previous sections. We also discuss possibilities for analyses that are not yet fully supported by NodeXL but could be achieved with a clever repurposing of NodeXL tools and features.

13.8.1 Data Preparation

One important aspect of data preparation is to identify the properties of the network that should be analyzed. This includes deciding whether the network is directed or not; deciding whether there is a single or multiple types of entities in the network, such as photos, people, and tags; understanding the semantics of the edges and other data attributes associated with vertices and edges.

- *Edge merging.* Flickr data are imported as directed networks and include edges A→B where the link may mean, for example, that a user A included a user B in the contact list. In some instances, the edge direction may not matter. Indeed, if you want to analyze who is connected to whom, without regard for who established the contact and whether it has been reciprocated, then you need to adjust the default worksheet that results from the data import. You can do this easily by changing the graph type to undirected. This change is likely to lead to duplicate edges that can be merged to compute graph metrics and to weight the edges. Some care needs to be taken when using the merging operation to avoid loss of information because it does not handle any attributes that may be associated with the edges (e.g., time stamp or edge type), except the edge weights.
- *Temporal data.* Because of the highly dynamic nature of social media, the edge time stamp plays a central role in analyses of network evolution over time. Flickr data may include a time stamp of the event that caused a relation A→B to occur. To view the network within a specific time period or to visualize its changes over time, you need to segment the data into appropriate time intervals. At the moment, you can achieve that in NodeXL by, first, converting the time stamps into suitable time slices (e.g., week, month, or year) using the standard Excel date functions, and then, storing the time intervals into a numerical column that can be used to set the edge visibility and define dynamic filters.

13.8.2 Graph Layouts

Most graph layout algorithms are generic and perform optimizations that are independent from the specific data sets and semantic meaning of the edges. Thus, it is expected that automated visualization needs to be complemented by manual layout modifications. For example, the layouts in Figure 13.17 or Figure 13.24 were achieved semiautomatically by first applying the Harel-Koren Fast Multiscale Layout algorithm and subsequently making manual modifications to improve the readability of the graphs.

Manual and Multiple Layouts

In some instances you may want to produce multiple visualizations of the same network. For example, it might be useful to provide alternative views of the same

data with different visual properties of the graph (e.g., representing vertices as text labels versus images versus geometric shapes). Similarly, you may want to show the network relations at different points in time, while preserving the position of the vertices from one time slice to the next. In NodeXL you can achieve this by first creating a good automatic or semiautomatic layout of the graph that is suitable for the most challenging case. Subsequently, you can lock the position of the vertices so that they remain in a fixed position when different filters are applied. This makes it easier to relate multiple views of the same data or to observe the emergence of new vertices and edges at different time slices.

13.9 RESEARCHER'S AGENDA

The NodeXL project was primarily motivated by the synergy between research objectives in network data analysis and social sciences. Understanding the social web phenomena requires a cross-disciplinary approach and a concerted effort to develop new methods and technologies. This places demands on researchers to evolve theories and algorithms in technical disciplines such as graph analysis, data clustering, and computer graphics, as well as to expand the scope of social sciences to investigate emerging behavior and social norms, and their implications for user privacy and trust, policies and regulations, and the like.

To facilitate cross-disciplinary work, it is of great benefit to consolidate state-of-the-art methods and tools and make them easily accessible to researchers in diverse areas. Such tools enable them to conduct research investigations more efficiently and effectively, leveraging optimized methods for performing standard analyses. That is one of the main objectives of NodeXL. With NodeXL, researchers can gain insights from data and form hypotheses without incurring a considerable cost of implementing tools for data preprocessing or methods and algorithms for graph analysis. NodeXL is designed to be extended and easily integrated with other complementary techniques such as machine learning, text mining, natural language processing, and image and video search. Thus, there are many opportunities for researchers to engage, expand on NodeXL, and enable further advances in their field of study.

13.9.1 Exploration and Validation

This chapter has illustrated a few analyses that you can perform with NodeXL to gain insights into Flickr data. Here we present several other research scenarios in which NodeXL can play a role both in the exploratory phase and in the validation of findings.

The study by Negoescu and Gatica-Perez [11] focused on the analysis of Flickr groups. With NodeXL you can easily conduct similar work, starting with preliminary analyses of individual or multiple Flickr groups and visually depicting patterns of interaction among group members. For example, you can observe patterns in comments on each other's photos and in friendship links. You can also create egocentric networks showing groups that a specific user belongs to and shares photos with. This can enable more specific analyses of groups or specific types of groups that complement the broader statistical approaches taken in [11]. Also, it can help you to investigate the implications of general findings for individual users or groups.

Sigurbjörnsson and van Zwol [12] studied tagging practices and tag recommendation algorithms. Optimal algorithms are likely to combine different aspects of tags in order to arrive at a final scoring and tag suggestions. NodeXL can aid researchers by identifying properties of tags that should be reflected in the metrics for tag scoring and ranking. As illustrated in previous sections, through NodeXL filtering, you can easily visualize tag co-occurrence patterns and see how specific ranges of statistical values affect the graph representation. This is an easy way to gain insights about the potential effectiveness of measures like *tag stability* and *descriptiveness* introduced by Sigurbjörnsson and van Zwol [12].

Furthermore, you can use NodeXL to bridge the quantitative and qualitative studies of social media. Indeed, NodeXL can support and complement qualitative studies of user behavior in a similar way that Ames and Naaman [1] and Nov et al. [2] explored the motivations for tagging photos in Flickr. You can use the tagging profiles of individuals to select those suitable for the study. Also, the quantitative findings for individual users and the sample of study participants could be compared and correlated with their practices before and after the study, purely based on the logs. This can detect the influence of the study method on the user behavior and possibly assist with validation of hypotheses derived from the observations. These are just a few of the situations in which NodeXL can support exploratory phases of research.

13.9.2 Integration

The unique value of NodeXL is the ease in visualizing network data and exploring the properties of vertices and edges. However, in many instances your research questions may require multiple perspectives on the data and application of several analyses. The study of Flickr images by Crandal et al. [10] investigated how to organize a large collection of photos by leveraging geotags as well as tags that users assign to their photos. In their work, they use textual tags and images with precise spatial information to define the structure between photos that are taken at a particular location.

A typical task would be to determine where a particular photo was taken. This is not as simple as you may think, even when geo-location is available. For example, you may take a photo of a Swiss guard by St. Peter's Church in the Vatican. There may be many pictures depicting changing of the guards, but also there may be many other images showing various dignitaries at the same location. Thus, the meaning of the photo changes over time based on the events that happen at that location. To capture the specific meaning of images, one has to resort to sophisticated content and structure analysis. You can use NodeXL in two ways. First, you can analyze the photos using existing information and identify what types of features may be relevant to make other techniques effective. For example, you can easily plot the relationship of photographers, specific geographical landmarks, and tags to see whether some initial patterns surface and whether they might be useful for photo classification. Second, you can use NodeXL to analyze intermediary results of content analysis. For example, classify images using machine learning techniques into groups related to preselected landmarks and then use NodeXL to import the resulting clusters of images, together with information about photographers and tag usage. NodeXL can then help visualize key photos for each landmark or overlay the network of landmarks over a geographical map.

The continued use of NodeXL for social media, like Flickr, will certainly reveal opportunities for integration with tools for complementary analyses and management of large volumes of data, comprising millions of images. These present technical challenges and interesting research opportunities. They are, in effect, the key for making progress in understanding the complex environments that the social media represent.

References

[1] M. Ames, M. Naaman, Why we tag: motivations for annotation in mobile and online media, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (San Jose, California, USA, April 28–May 03, 2007), CHI '07, ACM, New York, NY, 2007, pp. 971–980.

[2] O. Nov, M. Naaman, C. Ye, What drives content tagging: the case of photos on Flickr, in: Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems (Florence, Italy, April 05-10, 2008), CHI '08, ACM, New York, NY, 2008, pp. 1097–1100.

[3] G. Koutrika, F.A. Effendi, Z. Gyöngyi, P. Heymann, H. Garcia-Molina, Combating spam in tagging systems, in: Proceedings of the 3rd International Workshop on Adversarial information Retrieval on the Web (Banff, Alberta, Canada, May 08, 2007), AIRWeb '07, vol. 215. ACM, New York, NY, 2007, pp. 57–64.

[4] A. Mislove, H.S. Koppula, K.P. Gummadi, P. Druschel, B. Bhattacharjee, Growth of the flickr social network, in: Proceedings of the First Workshop on online Social Networks (Seattle, WA, USA, August 18, 2008), WOSP '08, ACM, New York, NY, 2008, pp. 25–30.

[5] R. Kumar, J. Novak, A. Tomkins, Structure and evolution of online social networks, in: Proceedings of the 12th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (Philadelphia, PA, USA, August 20–23, 2006), KDD '06, ACM, New York, NY, 2006, pp. 611–617.

[6] M. Valafar, R. Rejaie, W. Willinger, Beyond friendship graphs: a study of user interactions in Flickr, in: Proceedings of the 2nd ACM Workshop on Online Social Networks (Barcelona, Spain, August 17, 2009), WOSN '09, ACM, New York, NY, 2009, pp. 25–30.

[7] H. Halpin, V. Robu, H. Shepherd, The complex dynamics of collaborative tagging, in: Proceedings of the 16th International Conference on World Wide Web (Banff, Alberta, Canada, May 08–12, 2007), WWW '07, ACM, New York, NY, 2007, pp. 211–220.

[8] A. Singla, I. Weber, Camera brand congruence in the Flickr social graph, in: Proceedings of the Second ACM International Conference on Web Search and Data Mining (Barcelona, Spain, February 09–12, 2009), WSDM '09, ACM, New York, NY, 2009, pp. 252–261.

[9] N.A. Van House, Flickr and public image-sharing: distant closeness and photo exhibition, in: CHI '07 Extended Abstracts on Human Factors in Computing Systems (San Jose, CA, USA, April 28–May 03, 2007), CHI '07, ACM, New York, NY, 2007, pp. 2717–2722.

[10] D.B. Crandall, D.L. Huttenlocher, J. Kleinberg, Mapping the World's Photos, in: Proceedings of the 18th International Conference on World Wide Web (Madrid, Spain, April 20–24, 2009), WWW '09, ACM, New York, NY, 2009, pp. 761–770.

[11] R.A. Negoescu, D. Gatica-Perez, Analyzing Flickr groups, in: Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval (Niagara Falls, Canada, July 07–09, 2008), CIVR '08, ACM, New York, NY, 2008, pp. 417–426.

[12] B. Sigurbjörnsson, R. van Zwol, Flickr tag recommendation based on collective knowledge, in: Proceeding of the 17th international Conference on World Wide Web (Beijing, China, April 21–25, 2008), WWW '08, ACM, New York, NY, 2008, pp. 327–336.

CHAPTER

14

YouTube

Contrasting Patterns of Content, Interaction, and Prominence

Dana Rotman and Jennifer Golbeck

University of Maryland, College Park, MD

OUTLINE

14.1 Introduction 225

14.2 What Is YouTube? 226

14.3 YouTube's Structure 227
14.3.1 Videos 227
14.3.2 Users' Channels 229

14.4 Networks in YouTube 229
14.4.1 Video Networks 229
14.4.2 Users' Networks 230

14.5 Hubs, Groups, and Layers: What Questions can Social Network Analysis of YouTube Answer? 230
14.5.1 Video Network 230
14.5.2 User Networks 231

14.6 Importing YouTube Data into NodeXL 231
14.6.1 Importing Video Data 231
14.6.2 Importing User Data 232
14.6.3 Ethical Considerations 233
14.6.4 Problems With YouTube Network Data 233

14.7 Preparing YouTube Network Data 233

14.8 Analyzing YouTube Networks 234
14.8.1 User Networks 234
14.8.2 Video Networks 238
14.8.3 The YouTube "Makeup" Video Network 238
14.8.4 Healthcare Reform YouTube Video Networks 243

14.9 Practitioner's Summary 245

14.10 Researcher's Agenda 245

14.1 INTRODUCTION

Billions of videos about endless topics have been uploaded to the Internet by hundreds of millions of people. Using the techniques of social network analysis, we can visualize the landscape of connected videos and users and highlight important patterns that are linked to different kinds of roles of producers, commentators, and consumers. Ask a person if he or she has ever watched a video online, and the answer will probably be yes. In just the past few years, online video sharing has become a growing mainstream social practice. Gone are the days when watching a video online was an onerous task, involving the installment of media players and a prolonged wait for the content to download. Today, people easily share private videos with friends and family; amateurs and professionals broadcast artistic endeavors, from music to comedy to directorial experimentation; media corporations distribute TV programs and movie excerpts, and millions of people watch and recommend videos to others, making some of them "viral" and wildly popular. Online video sharing services offer something for almost everyone, whether the users are video content creators or consumers. Video content is used for many purposes, from conveying knowledge and disseminating information to self-promotion to documenting world affairs. The diverse content that can be found on video-sharing sites draws large numbers of users. But not all content, even popular content, is popular in the same way. As we shall see,

networks of producers and consumers of online videos vary measurably.

YouTube has become almost synonymous with watching videos online. Although "video sharing" is a term less familiar to most people, "YouTube" videos are commonplace. Using video-sharing systems to publish or consume videos creates a variety of connections between the people who use these systems, the videos, and the tags that describe them. The social structures that emerge from the interaction of video creators and viewers can be represented as a social network graph. Participation in online video sharing generates a number of network graphs that can teach us not only about users' preferences for video content but also about their habits, motivations, and social interaction. Many people upload, view, and comment on videos, but some become centers of dense communities or are well connected to others. Because of its extreme popularity, YouTube presents an especially inviting domain to explore the social structure and dynamics of networks within a video-sharing community.

14.2 WHAT IS YOUTUBE?

Created in 2005, YouTube is an arena for personal communication, a place to create an online community or egocentric social network, and a platform that can be used for distributing commercial content.

YouTube was one of the first online services to offer users the opportunity to upload videos and share them with the world. Though similar services, such as MetaCafe, Yahoo! Video, and Google Video, emerged around the same time and were later followed by Vimeo, DropShots, and many others, YouTube has become the most popular video-sharing service in the world. Google bought YouTube in 2006, and it is now operating as a subsidiary of Google Inc. Some of the site's current features are now based on Google tools (e.g., the search and suggested list of friends based on Gmail contacts).

The immense popularity of YouTube can be seen in the current statistics offered by the company: on October 9, 2009, the official company blog announced that more than 1 billion movies are watched daily [1]. Internet traffic patterns show that in October 2009 between 21 and 23 percent of all Internet users visited YouTube at least once. According to YouTube's press office, 24 hours of video are uploaded by users every minute.[1] YouTube's popularity can be attributed to several factors: the relative ease of uploading and sharing videos, the site's continuous design changes (which reflect the evolution of online social networking practices), and strategic collaborations with commercial content providers such as broadcast networks, movie studios, and political parties.

The foremost reason for YouTube's success is the relative ease of uploading and sharing videos. Video sharing existed before YouTube prevailed: videos were sent as email attachments or were available through other video hosting services, yet these services were slow, cumbersome, and limited in the amount of storage they offered. Viewers could not watch videos instantly but had to download them to their computers before playing and viewing them using proprietary video players. Metadata descriptions about the video were rarely available. YouTube changed all that, as well as moving the viewing experience from a solitary experience to a social one. A significant part of YouTube's popularity can be attributed to the simplicity of sharing videos and discussions about them. YouTube supports and encourages the embedding of videos in other forms of online communication—from email to microblogging to status updates in social networks—by displaying the relevant link next to each video. All users have to do is cut and paste these video codes into a range of other social media such as blogs, wikis, emails, and status updates. The simplicity of video sharing and embedding contributed not only to YouTube's popularity but also to the phenomenon of "viral videos": usually provocative, quirky, or very creative videos that achieve extreme popularity after being mass-distributed through an electronic word of mouth via various online social tools.

The practice of video sharing creates several types of networks: some networks are based on content and others on social affinity or social ties. Content networks reflect mutual interests or shared hobbies or practices, essentially creating communities of practice [2] that stem from the commonalities between users' interests. These communities can loosely be based on the preexisting categorical definition suggested by YouTube (e.g. "music," "entertainment," "how-to and style," "politics and news," etc.) or sub-networks of people interested in a specific aspect of the overall category (e.g., Japanese anime enthusiasts, Obama supporters, cosmetic makeup aficionados, gamers). Content networks evolve around videos and are based on reciprocal ties (edges) between videos (vertices), which are formed through the use of social tools for creating comments, creating reciprocal videos, collecting "favorites," and tagging content with keywords.

Social affinity networks are formed when users interact with each other. YouTube allows users to "friend" one another or subscribe to other users' "channels" of video collections. Subscription and friendship ties are the edges of the networks, and users are the vertices.

[1] www.youtube.com/t/fact_sheet

Social affinity networks can be based on preexisting relationships (e.g., family members, friends, or fans) or form on the site, as people interact with one another based on mutual interest in content. As you will see, the conceptual distinction between the two networks is sometimes difficult. However, as the next section explains, they are structurally separate.

Analysis of different YouTube networks will allow us to reveal the key positions some people and videos occupy within the mesh of connections created when users collectively create, watch, and comment on content or form personal relationships; in some cases they may even offer insights as to the underlying reasoning for these connections.

14.3 YOUTUBE'S STRUCTURE

According to YouTube's official policy, the purpose of the service is twofold: to provide a distribution platform for video content and to create a community of video enthusiasts. As such, YouTube's structure is based on two sub-layers, clearly differentiating between videos (content) and users (community) while maintaining a close linkage between the two. Similarly, network analysis can be performed on both networks of video or users, either independently of each other or in conjunction.

YouTube features, it must be noted, are constantly changing, with new tools being introduced and others periodically removed. In recent months an effort has been made to make YouTube an even more social space. Social features such as friend suggestions (based on email addresses), alerts about friends' activities, easier video sharing on the site and others are being added constantly. As is the case with almost every online social network, you should always reevaluate the current features of the site and familiarize yourself with the latest changes, before embarking on an analysis of the YouTube network.

14.3.1 Videos

Each video on YouTube is displayed on a separate page, but all video-pages have the same layout, similar to that of Figure 14.1.

On the upper left-hand side the video itself is displayed, under a caption. Beneath the caption a drop-down menu presents the name of the author and offers the opportunity to watch other videos created by this



FIGURE 14.1 A YouTube video page, presenting the video alongside metadata about it and social tools for interaction with other users.

user and to subscribe to other videos the author created. The video screen allows adjustment to size, volume, and display quality. Dropdown menus at the bottom of the video offer access to metadata related to the video:

Descriptors

- *Description of the video*—given by the user when first uploading the video
- *Tags*—describing the videos, given by the user
- *Category*—chosen by the user from a closed-list of categories provided by YouTube
- *Views, data, and statistics* tab—provides the number of views the video generated, as well as other data regarding the video's traffic since the time it was uploaded, changes in views and ratings with time, and links to the video from other sites that caused a change in its traffic. A map displays the popularity of the video in different geographic regions, based on the number of views of registered users, as well as the video's popularity based on gender and age. Specific honors that the video has received, if any, (e.g. "most viewed") are also displayed
- *Comments*—are displayed under the views and statistics tab. Users can respond to a video in two ways: by clicking on a tab at the bottom of the video, leading to a comment box, or by creating a video-response and linking it to the original video (when uploading a video a user can create this linkage). Textual comments can be replied to as threaded conversations, liked or disliked, and classified as

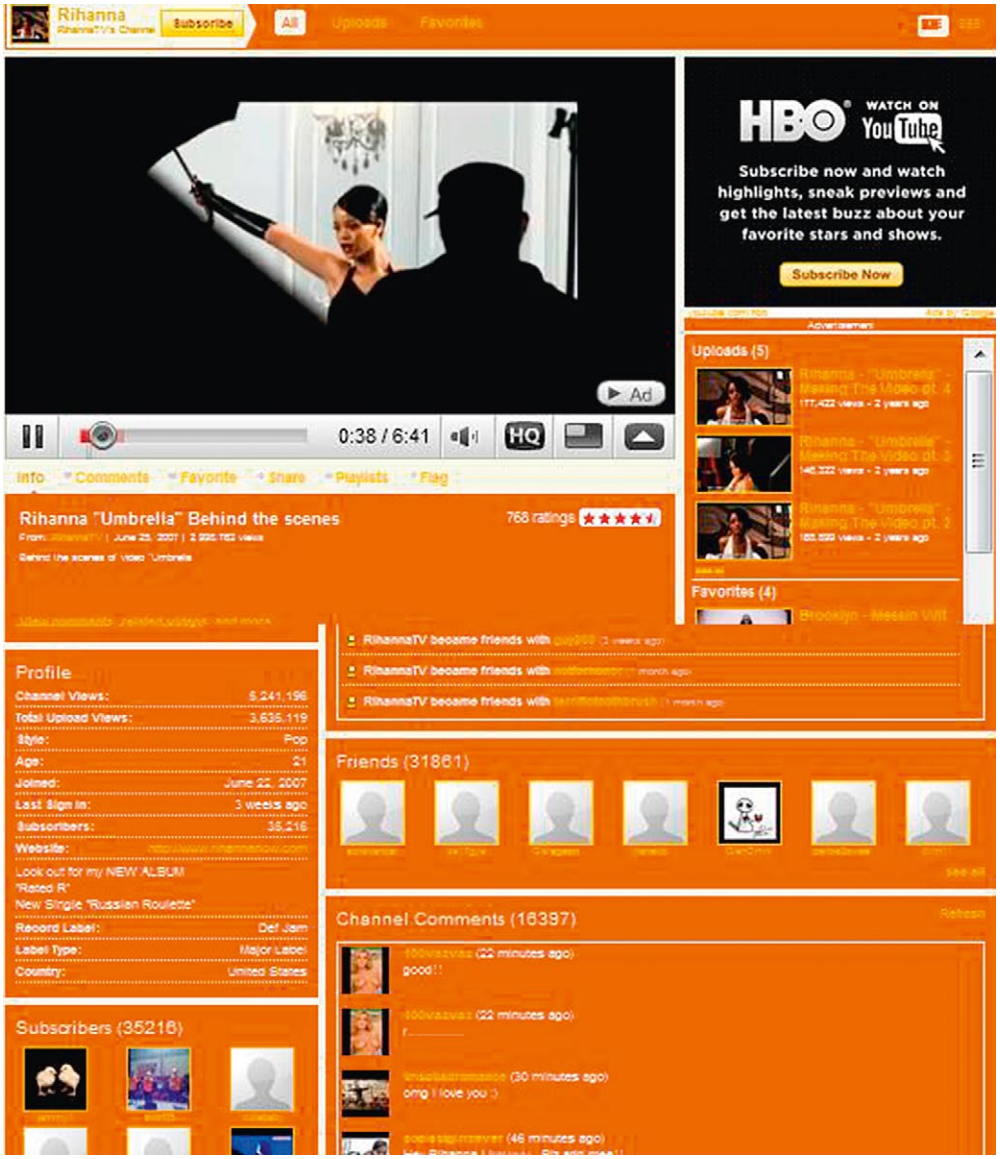


FIGURE 14.2 YouTube user channel for the singer Rihanna, presenting latest videos, information about the user, and tools for social interaction.

spam. This can also be done to video comments, but only on its original, separate, page, and not when it is displayed as a comment. Another way to express a viewer's opinion of the video is by clicking the like/dislike buttons. The aggregate number of likes/dislikes is presented next to the comments column.
- Underneath the information box, a button enables viewers to embed the video manually in other online tools (i.e. personal blog, websites) or directly into their profile in several social media applications (e.g., Facebook and Twitter) or an email.

On the right-hand side of the page a list of related videos is presented. This list is based on commonalities among the videos, as determined by YouTube's search algorithm and on the origin from which the user arrived at the video.

14.3.2 Users' Channels

Similar to other social networks, YouTube users can create personal profiles called "channels" (Fig. 14.2), which are somewhat customizable. Users can choose what information to share with other viewers and which "widgets" to display on their channel. Widgets display small boxes with different types of information, such as personal advertising widgets or connections to other social networking sites such as Twitter and Facebook. Users are able to adjust these settings to reflect their preferences. Thus, users' pages can be strikingly different: some users prefer not to disclose any personal information, whereas others exhibit their social relations and detailed personal information publicly. Most users display information about themselves (e.g., name, age, location, the date they joined YouTube and the last date they logged in, links to other personal web sites, and alternative means of communication—email, instant messaging (IM), or Facebook account), although this can be limited with privacy settings.

Users have the option to display two different social networks on their pages:

1. *Subscribers*. Some viewers ask to receive periodical notification of new content the selected user has uploaded. Subscription is a directed asymmetrical tie that does not require the approval of the channel owner (unless a channel has been made private by its owner). Similar to the "follows" relationship in Twitter (see Chapter 10), subscriptions are a useful tool for keeping up with new content and not necessarily an indicator of a symmetric, reciprocated relationship with a specific user.
2. *Friends*. Friendship requests are similar to social connections in other social network services like Facebook (see Chapter 11). This kind of connection is bilateral and symmetric. Friendship requests have to be approved in order for them to exist. Friends can mutually display their connection to one another on their respective pages.

A users' channel often also includes a personal bulletin board on which other users can leave comments and interact, a public log of the user's recent social activity, videos the user "favorited" or playlists that he or she created, lists of videos uploaded by the user and YouTube honors that were bestowed upon the user (e.g., "most watched," "most commented," "most subscribed").

With effort and time, a dedicated reader or researcher can piece together pieces of information that can assemble an interesting picture of a user's activity and preferences. Automated tools for collecting this information are more challenging; the YouTube API doesn't lend itself well to collecting this information and doesn't provide much of what is contained in the XML metadata that accompanies the videos and user channels.

14.4 NETWORKS IN YOUTUBE

YouTube's rich collection of users, videos, comments, subscriptions, tags, ratings, and favorite videos offers multiple ways networks can be formed. Broadly, video networks are different from user networks in both content and structure. Within these networks are several subnetworks that provide insights into the important people, videos, and events in these video-sharing networks. Interesting connections between people and content can be found. Examples of these different networks will be presented in Section 14.8, but first we will examine the attributes of each network.

14.4.1 Video Networks

Several networks can be constructed from the attributes found on *video* pages:

- *Videos that share the same descriptors*. When users upload videos to YouTube, they must provide a description of the video content, including a title and tags or related keywords. Videos can also be classified according to predefined categories, such as *comedy*, *music*, *education*, *politics and news*, *people and blogs*, *how-to and style*, and so on. Videos that share the same descriptors may also share the same type of content; however, as users are the ones who assign descriptors to the videos, the descriptors can vary widely and we cannot assume content similarity in videos that share descriptors. The emergent collection of connections among videos has varying degrees of density, from tightly knit networks of topical videos that discuss similar content to a

	Videos network	Users network
Comments	Textual and video comments and responses	Textual responses
Friends		Can be found unless checked as private
Subscriptions		Can be found unless checked as private
Similar descriptors	Networks based on tags, headers, description, and categories	A minority of users categorize their "style" (equivalent to video "categories")
Related videos	Based on YouTube algorithm	

FIGURE 14.3 Examples of different networks found in YouTube.

dispersed network of videos that may or may not have something in common.

- *Videos and response videos*. Users can respond to other videos by using textual and video comments. A video comment usually takes the form of a "response video," which is directly linked to the original video. Response videos may share the same topic of the original video or may digress from that to introduce new ideas and people that are interested in the original video are more likely to view the response video as well.
- *Comment networks*. Users leave textual comments in the wake of a video. Textual comments can be recorded while watching the video or after the fact. Comments range from illegible combinations of alphanumeric notations to thoughtful discussions. Currently, YouTube's API and NodeXL provide only basic support for retrieving textual comments.
- *Related videos*. A list of related videos is presented adjacent to each selected video. These lists are based on YouTube algorithms. At the moment, NodeXL does not support related-videos networks.

14.4.2 Users' Networks

In contrast to video networks, which are focused on content, user networks are egocentric—that is, users, not videos, are the center of these networks. User networks can be explicit or implicit evidence of relationships and interactions between users. The direct request or action of at least one user creates explicit networks. Users take the effort to click to create "subscription" and "friendship" networks and display those connections on their channel. Implicit networks are created when two or more users interact through comments, ratings, and favoring on user and video profile pages. Of the three, only comments are visible to external users of YouTube in personally identifiable form; ratings and favoring are anonymized and summed to a single value. When one user comments on another's channel profile, an implied connection is created between them. Not all, or even most, comments are responded to, but all create a connection and allow other users to exhibit their interest in that user and the content he or she provides. Exploring these networks can provide insights into the overall structure of video collections and key videos and users who occupy critical positions in that network, as well as contextual connections among videos and users (Fig. 14.3).

Understanding the nature and structure of the relationships and ties within YouTube networks can help you understand important interaction patterns and information flows within the networks.

14.5 HUBS, GROUPS, AND LAYERS: WHAT QUESTIONS CAN SOCIAL NETWORK ANALYSIS OF YOUTUBE ANSWER?

YouTube does not easily reveal its underlying network structure. The interface displays leaves and branches but not the larger forest of connections it contains. Before deciding which data to collect and analyze, you have to step back and conceptualize the questions that are of interest. Once formulated, questions may be pertinent to both video and user networks.

14.5.1 Video Network

1. *Centrality.* Which videos are central within a category/type of videos? Which videos generate many comments, response videos, and higher ratings? These videos and users may influence the content produced in other videos and attract many relationships (i.e., subscriptions) with people who share an interest in that content. Some videos lead a specific category, whereas others are peripheral. Are there differences between a single video and a series of videos produced by the same user? (Do series increase the overall popularity of individual videos? Can a single video be as pivotal as a series of videos?)

2. *Groups.* Does the network contain hubs of densely interconnected videos that share properties like common tags and descriptors? Which videos are central to those hubs? Is their centrality correlated to other attributes? Are different hubs connected to each other? Which are the boundary videos that connect such hubs? How dense are these hubs? How do they compare with other types of social content?
3. *Temporal comparisons.* How does a video network evolve over time? What affects its development? Are certain descriptors, tags, topics, and types of videos crucial to the evolution of the network? What is the effect of rapidly and widely exchanged viral videos on the development of the video and user network? Do these videos disrupt or reinforce existing networks, or is the effect of viral videos visible mostly outside YouTube's boundaries? What changes occur when a video becomes popular?

14.5.2 User Networks

1. *Centrality.* Which users are central in the network of connected YouTube users? Some users may be central in a specific category but not in others. Is centrality an outcome of the explicit or implicit networks? Which users are boundary spanners between different parts of the networks? Which are peripheral? Can we identify rising YouTube stars?
2. *Groups.* How do users link together to form emergent groups? What brings them together? (For example, is it a certain interest, topic, or another reason?) How do the populations of subscribers and friends overlap? Implicit groups can be found and compared with explicit groups. Are there central and peripheral groups? Do friends-of-friends, or subscribers-of-subscribers belong to the same groups? How dense are these groups?
3. *Temporal and structural comparisons.* How and why does the popularity of a user change? How do users move from being peripheral to central and vice versa? Are boundary spanners changing over time? How do external circumstances affect users, their popularity, and their networks? How do the video, subscription, and friendship networks align? What are the differences between a user's subscription and friendship network? Which is denser? Which is larger? How does a change in a video popularity affect these networks? Are there differences between the explicit networks and the implicit ones? How do they affect each other?

Some of YouTube's features, such as favoriting, number of views, and lists like "Most viewed," "Most subscribed to," or categories such as "Rising videos" can give you an understanding of popularity trends. But you cannot learn about information flows, centrality, and subnetwork structures from these features alone. For that, make use of NodeXL's network analysis metrics, coupled with network graph visualizations.

14.6 IMPORTING YOUTUBE DATA INTO NODEXL

To import data from YouTube into NodeXL, first select which network you are interested in and what type of data to import using one of the data import options in the NodeXL Data menu. When deciding what data to import, remember that YouTube differentiates between videos and users. Although networks of both users and videos can be imported and integrated into the same NodeXL network file and later compared, they cannot be imported simultaneously.

14.6.1 Importing Video Data

From the NodeXL ribbon, choose "Import" → "From YouTube's video network." A box, similar to Figure 14.4 will be displayed. This option allows you to import videos whose descriptors include the keyword that you type into the search box. A YouTube search is conducted using the YouTube/Google search engine; the search algorithm does not differentiate between types of metadata related to the video and will match against all information found in the title, description, tags, and category fields. In Figure 14.4, for example, the search will import videos that include the word "makeup" in any of the descriptors. However, the search cannot be performed, currently, on the audio and textual content of the video itself.

As mentioned, video networks have several attributes, and videos can be imported based on one or more of them. Video attributes include the following:

- *Tags*. When a pair of videos is tagged with the same keyword, an edge is created between them. For example, if video A and video B both have the tag "makeup" (with or without other tags), they will be connected by an edge.
- *Comments by the same user.* An edge between a pair of videos will be created when they are commented on by the same user.
- *Video response.* Videos can also be linked by response videos. If a video response linked video B to an earlier video A, an edge will be created between them.

Your decision about which data to import depends on your research question. Some options will substantially slow the import process and will require further filtering of the data in order to analyze it later. You can

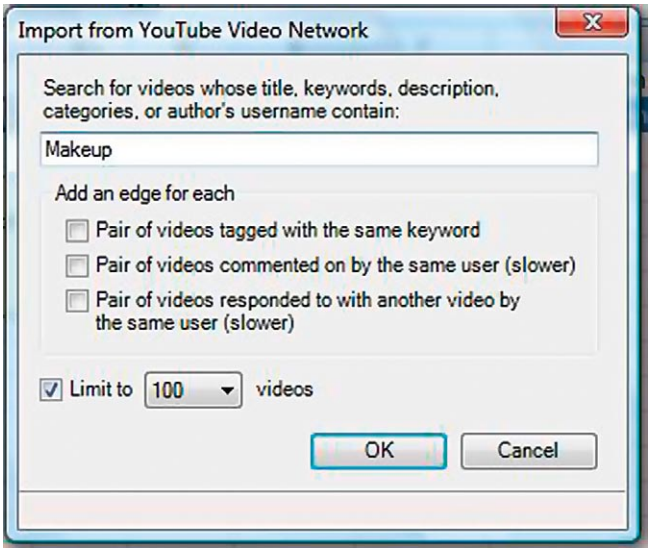


FIGURE 14.4 NodeXL Video network data import dialog box, marked for importing videos that contain the word "makeup" in their title, keywords, description, categories, or username. The number of imported videos is limited here to 100 but can be set higher.

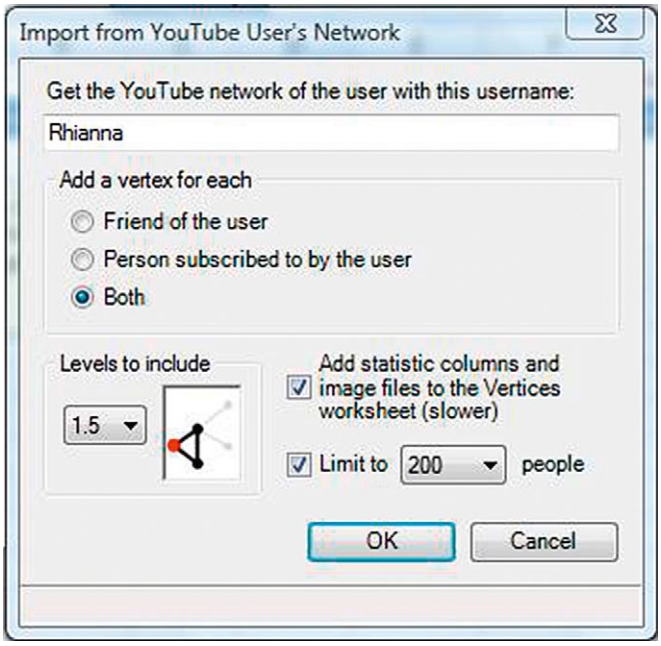


FIGURE 14.5 NodeXL YouTube User network data import dialog box, set to import 1.5 levels of the user's friendship and subscription network. Network size is limited to 200 people and statistics columns about user activity provided by YouTube will be added.

also set up an upper limit to the number of videos that will be imported in any data import. This is very useful when considering the size of YouTube and the staggering number of videos and the limits of desktop spreadsheet programs.

14.6.2 Importing User Data

From the NodeXL ribbon, choose "Import" → "From YouTube User's Network." A dialog box, similar to Figure 14.5, will appear. This option allows you to import some of the networks associated with YouTube users. In the search box type the channel name for the user whose network you want to import. It is important to enter the exact channel name, as several variations (e.g., Dana's, Dana, Danas, DanaChannel) are displayed on different segments of YouTube. The exact channel name is displayed next to the subscription button on the user's channel.

You can then choose which user network ties to include in your data import. You can import the friendship network (an undirected network) and the subscription network (a directed network). The YouTube API does not currently enable importing the list of people who subscribed to the user's channel, only the list of people to whom the selected user subscribes. Both types of explicit user networks can be imported simultaneously or separately.

You can then select the network levels to import. A level is a step in the network from one node to another. A single level starts with a target user and takes a single step out to all of their friends or subscriptions. A second level includes data about all of the friends of their friends. Taking a half step back, you can construct a 1.5 level network, which limits the list to the ties among first level friends. A tie from a friend to someone who is not the selected user's friend will not appear while ties among their mutual friends do appear. The higher the number of levels you choose to go into the network to collect data, the longer it will potentially take to import the data from YouTube into NodeXL. In most cases, asking the spigot to bring back 1.5 levels of the network is sufficient to answer many common questions. You can also limit the number of users (vertices) that will be imported, if needed, to extract a data set of manageable size.

Checking the "Add statistic columns" box will import, along with the vertices and edges, statistical data about the users including number of subscribers and friends and the date the user joined YouTube. A pointer to the image file that represents each user and usually displayed on a user's channel is included. Displaying the profile photos of YouTube users in these network visualizations can make them more compelling and informative.

For analyzing egocentric networks, we advise selecting 1.5 or 2 levels to be included in the imported data. This will encompass the user's friends and subscribers, and their friends and subscribers, as well as any connection between them.

ADVANCED TOPIC
Using Pre-prepared Data

If you would prefer working with data that are not imported into NodeXL through the data import spigots, this data can be gathered independently and opened directly in Excel and imported into NodeXL.

YouTube requires use of the YouTube API to gather data. This is maintained as part of the Google Code system and documentation is available at http://code.google.com/apis/youtube. The API describes how data can be accessed from YouTube. It is based in the Google Data Protocol, a REST-inspired protocol for accessing the data over the web. This means data can be accessed with precisely formatted URLs over the web and through a wide variety of programming languages.

You can write your own code to gather YouTube data through this API. You may want to crawl a network, beginning with a specific user, moving outward through the network. You may also want to send queries and create connections between users or videos based on criteria different from those available in the NodeXL YouTube data spigot.

Once you have gathered the data, it can be loaded into Excel and formatted for use in NodeXL as described in Chapter 4.

14.6.3 Ethical Considerations

YouTube users can choose to make certain segments of the information on their channel private. They can decide that their friends or subscribers lists are private, approve or delete comments, and send private messages to other users. When importing data from YouTube, it is important to consider that data may be missing some components because of privacy preferences (for instance, complete friendship lists will not be imported into NodeXL if they are defined as private). In addition, ethical considerations stem from using other people's data, even when publicly available. When looking at a user's data or performing social network analysis on them, researchers should act carefully and with respect to users' expectations of privacy.

Because not all users are aware of privacy settings and related considerations, researchers should demonstrate care when dealing with information that may be perceived of as private. For example, personal ties and other personally identifying information. In the case of YouTube, there is the danger of inadvertently disclosing extremely personal, embarrassing, or sensitive information. Putting together a name, face, and an opinion may be more revealing than many users expect. Anonymizing data can alleviate some of these concerns; however, it is problematic in a network that offers many facets of extensive data about the users. Therefore the rich metadata offered along with YouTube videos should be handled with care and with respect to the users behind it.

14.6.4 Problems With YouTube Network Data

Data collected for analysis in NodeXL, either through the spigots or through your own code written with the API, is not necessarily complete or accurate. First, the YouTube data API is not 100 percent reliable when used by either your own code or the NodeXL spigot. Because it accesses data over the web, requests may be lost or timed out. Thus, the same query may yield different results at different times. Second, even without errors, imported data may not reveal a complete network. Videos and user profiles may be marked private, preventing them from being accessed and included in the analysis. Third, users can choose to remove a video they uploaded at any time, and YouTube can remove a video when it violates the site's terms of use or is flagged for review by other users. However, there is a delay between the time a video was removed from the site and the time it will stop appearing in search results. This can cause data previously accessed to be incorrect—a video may appear as a vertex in the network but will not actually remain a part of it in the current data on YouTube.

14.7 PREPARING YOUTUBE NETWORK DATA

Using some of the filters that import YouTube data into NodeXL will result in collecting multiple edges for the same vertices. As a result, it is possible for users or videos to be connected to each other more than once. For example, a user may comment repeatedly on the same video; another example may be when two users are connected through mutual subscriptions (user A subscribed to user B and user B subscribed to user A). In such cases, merging duplicated edges will present a clearer picture of the relationships and remove bias when calculating the overall network metrics, but it will also give the appropriate weight to these relationships statistically and visually.

The primary way to clean up the data and create a workable corpus of YouTube network data is to use the

"merge duplicate edges" command in the Data section in the NodeXL ribbon. This command will compare each edge and find the duplicates and merge them to display one connection. The connection will be weighted according to the number of duplicate relationships that were merged. A column containing the edge weight is added to the NodeXL Edge worksheet.

14.8 ANALYZING YOUTUBE NETWORKS

14.8.1 User Networks

YouTube users (or "YouTubers") are content providers, drawing other users, viewers, and participants to the videos they produce. As YouTube itself does not provide video content, lacking the videos produced and uploaded by YouTube users, the site would lose most of its value. However, YouTube users do not limit their activity on the site to video content production. The massive amounts of social interaction that occurs on YouTube strengthen the idea that the site is host to a complex and lively community of users and participants and is more than just a platform through which incidental viewers can watch videos [3–4].

YouTube user networks are important for several reasons. (1) Users are the essence of the YouTube community; the way they congregate and interact may reveal the flow of information in the YouTube network and highlight the importance of certain users to the community or to a specific sub-community of users. (2) Differences between users' networks can demonstrate different participation patterns and aid in improving the interface to accommodate various audiences. (3) These networks can also be used by companies and organizations wishing to use the YouTube platform for advertising, lobbying, or disseminating information.

Users' egocentric networks tell an interesting story about the type of ties created among users. The first example we discuss compares the egocentric networks of two users who—although they come from the same category—use YouTube in very different ways.

The first network belongs to the popular singer Rihanna. To see it, use the "Import from user's YouTube network" option in the NodeXL ribbon. Clicking on that option will open a screen similar to that shown in Figure 14.5. We are interested in both Rihanna's subscriptions and friendship network. Remember that the two networks are different: while subscribing to a user's channel is a unilateral action that does not require approval from the user (and is, therefore, a directed, asymmetric network), befriending a user necessitates his or her agreement (making the network undirected and symmetric). To import both networks, select the option to "add a vertex for both" [networks]. Then use the drop-down menu

	A	B	F	G
1			Other Columns	
2	Vertex 1	Vertex 2	Add Your Own Columns Here	Relationship
3	cubfan31	RihannaTV		Friend of
4	RihannaTV	cubfan31		Friend of
5	supernova2	RihannaTV		Friend of
6	RihannaTV	supernova2142		Friend of
7	enemygam	RihannaTV		Friend of
8	RihannaTV	enemygaming		Friend of
9	RihannaTV	cherrylollipop23		Friend of
10	chickenpatt	RihannaTV		Friend of
11	RihannaTV	chickenpattyboy		Friend of
12	RihannaTV	Emonuelclarinets		Friend of
13	RihannaTV	Sutekh9x9		Friend of
14	RihannaTV	babyprincess92		Friend of
15	RihannaTV	boomtikkerwahwah		Friend of
16	RihannaTV	charmed02835		Friend of
17	RihannaTV	Tombo60		Friend of
18	voodoogun	RihannaTV		Friend of
19	RihannaTV	voodoogunder		Friend of
20	blowj0b	RihannaTV		Friend of
21	RihannaTV	blowj0b		Friend of
22	RihannaTV	antonmullo1990		Friend of
23	krawkadial	RihannaTV		Friend of
24	RihannaTV	krawkadial		Friend of
25	youngtrigge	RihannaTV		Friend of
26	RihannaTV	youngtrigger		Friend of
27	inuyashalo	RihannaTV		Friend of

FIGURE 14.6 Rihanna's egocentric YouTube network NodeXL Edges worksheet, after importing 1.5 levels of friendship and subscription lists.

to specify the scope of the network you want to retrieve. Selecting 1.5 levels will retrieve both users who have ties to Rihanna and the ties among themselves. This will allow you not only to analyze users' relations with the person who is the middle of the egocentric network but also to find if any hubs of relationships exist among the other users. Considering the immense size of YouTube and the sheer volume of users, the number of vertices that are retrieved should be limited; this example requests the first 200 users. This will provide a reasonable sample of the larger egocentric network while not encumbering the retrieval process.

When the data are imported into NodeXL, they are displayed in an Excel workbook. In the Edges worksheet tab you can see each pair of vertices and a new column titled "Relationship," which is added to the standard NodeXL Edges worksheet and filled according to the type of YouTube relationship that connects these vertices. Remember, NodeXL imported both subscription and friendship lists. However, a quick look at the relationship column (Fig. 14.6) will reveal that there are no "subscriptions," just "friends of." We will discuss the meaning of this finding later. For now, we will work with this workbook to create a visual display of the egocentric network that surrounds Rhiana. It is useful to calculate network graph metrics that will describe the

shape of the graph and describe the location of each person or video within it. To calculate many network metrics correctly, the edge list must be prepared to remove the repetition of identical edges in the workbook, as some network metrics will be inaccurate if a connection is represented more than once. You can use the NodeXL Data menu option to "Prepare data" → "Merge duplicate edges." This will remove any duplicates from the Edges tab and will add a column of "edge weight" that corresponds to the number of duplicated edges. For instance, if John is a friend of Rihanna and also a subscriber, it will merge the edges and add an edge weight of "2" to that column. Once duplicate edges have been merged, you can then compute network graph metrics accurately. Notice that Rihanna's egocentric network is extremely sparse with a density level of 0.008.

You can then move on to explore the characteristics of the graphic display of the network.

The first look at the visual display (Fig. 14.7(a)) confirms the story the metrics tell you: this is an egocentric network of multiple ties that stem from the user Rihanna, who sits in the middle of the network. Buried in the network are a few connections among the people connected to Rihanna. To highlight those vertices that are connected to each other, use Excel Data tools to arrange all the vertices according to their degree, sorting from largest to smallest. Looking at the workbook you can see that users who have a relatively large in-degree also have a large out-degree. However, for most users both metrics are 1. Users with higher in- and out-degrees are the centers of small hubs in this network. Highlight them by setting their shape to "image" and in the Autofill Columns feature, and set the tooltip to display the name of each vertex (set Tooltip to Vertex). Last, map the edges by setting their opacity to reflect the edge weight, so that repeated edges will be more opaque than less repeated ones. Then click "refresh graph" to display Figure 14.7(b). This refined network graph tells a more detailed story about the community that surrounds this popular singer.

Rihanna's egocentric network is archetypical for a celebrity or popular figure in which one user is the center around which others congregate. Vertices in the network are rarely connected to each other; fans of Rhiana are not fans or not necessarily friends of each other, so that no community or hub is created. The metrics demonstrate that the network is created only from friendship ties. This reflects the way that Rihanna (or her public relations representatives or recording company) uses YouTube. In the absence of any outward attempt to venture outside the fan base, you can see that Rihanna does not subscribe to any other channel but maintains her fan base through approving their friendship requests. You can deduce that her YouTube channel is used mostly as a promotional showcase for the artist's commercial videos and far less as an interactive channel through which the artist can stay in touch with fans and other artists.

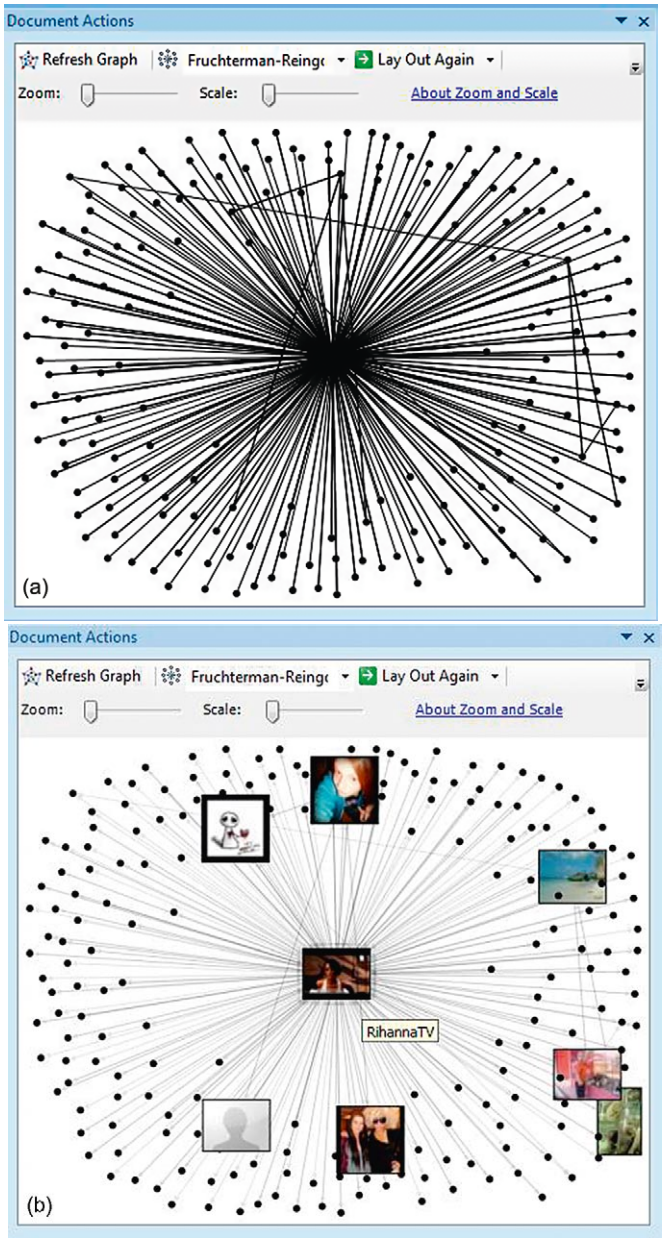
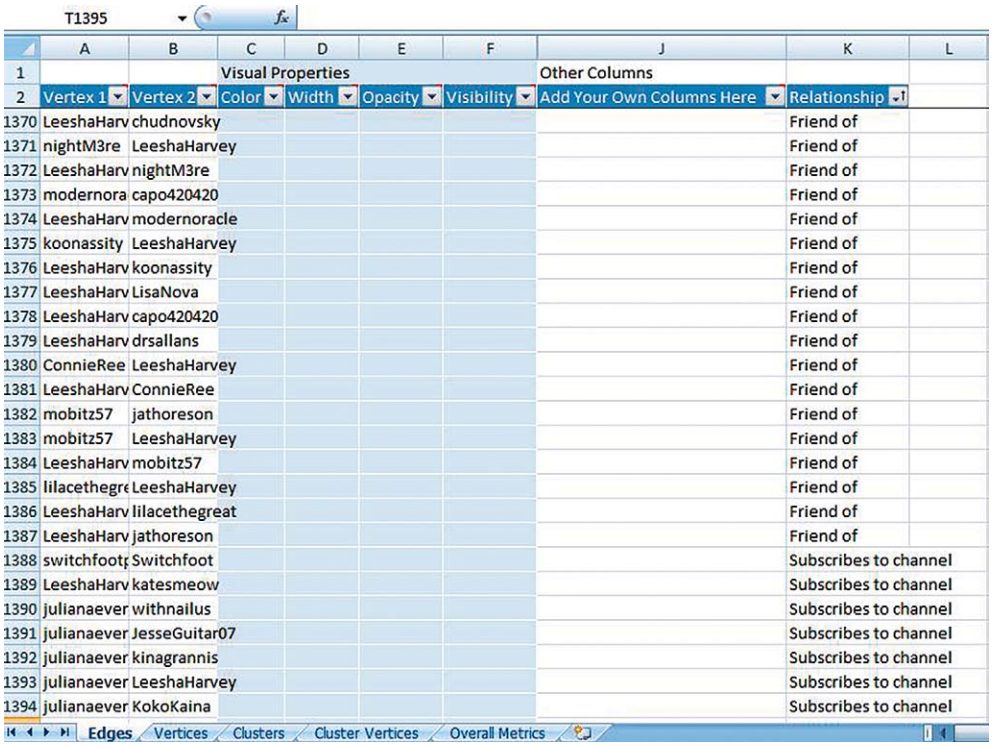


FIGURE 14.7 Rihanna's egocentric YouTube network displayed in NodeXL. The primary layout (a) does not provide any information about the vertices but illustrates a typical single level egocentric layout. After filtering based on degree and edge weight and adding images to the prominent vertices (b), a small number of important connections are revealed.

Compare this network with that of a different artist, Leesha Harvey. This YouTube user defines herself as a folk "singer-songwriter" who, through her music channel on YouTube, tries to reach wider audiences. To retrieve her network, follow the same steps you performed to



T1395

	A	B	C	D	E	F	J	K	L
1			Visual Properties				Other Columns		
2	Vertex 1	Vertex 2	Color	Width	Opacity	Visibility	Add Your Own Columns Here	Relationship	
1370	LeeshaHarv	chudnovsky						Friend of	
1371	nightM3re	LeeshaHarvey						Friend of	
1372	LeeshaHarv	nightM3re						Friend of	
1373	modernora	capo420420						Friend of	
1374	LeeshaHarv	modernoracle						Friend of	
1375	koonassity	LeeshaHarvey						Friend of	
1376	LeeshaHarv	koonassity						Friend of	
1377	LeeshaHarv	LisaNova						Friend of	
1378	LeeshaHarv	capo420420						Friend of	
1379	LeeshaHarv	drsallans						Friend of	
1380	ConnieRee	LeeshaHarvey						Friend of	
1381	LeeshaHarv	ConnieRee						Friend of	
1382	mobitz57	jathoreson						Friend of	
1383	mobitz57	LeeshaHarvey						Friend of	
1384	LeeshaHarv	mobitz57						Friend of	
1385	lilacethegr	LeeshaHarvey						Friend of	
1386	LeeshaHarv	lilacethegreat						Friend of	
1387	LeeshaHarv	jathoreson						Friend of	
1388	switchfoot	Switchfoot						Subscribes to channel	
1389	LeeshaHarv	katesmeow						Subscribes to channel	
1390	julianaever	withnailus						Subscribes to channel	
1391	julianaever	JesseGuitar07						Subscribes to channel	
1392	julianaever	kinagrannis						Subscribes to channel	
1393	julianaever	LeeshaHarvey						Subscribes to channel	
1394	julianaever	KokoKaina						Subscribes to channel	

Edges | Vertices | Clusters | Cluster Vertices | Overall Metrics

FIGURE 14.8 Leesha Harvey's egocentric YouTube network NodeXL Edges worksheet, after importing 1.5 levels of friendship and subscription lists.

retrieve Rihanna's network. A first look at the network metrics for Leesha Harvey reveals a very different picture—her network density is 0.166, significantly higher than the density of Rihanna's network (0.008).

Looking at the edges workbook, you can see that the relationship types vary: some are friends while others are subscribers, and some are both. To ensure that the visual display portrays the different types of relationships, or subnetworks, first use the Excel Data → Sort feature to order them alphabetically, so that friends and subscribers are divided. To distinguish between different ties, map friendship and subscription ties on the Edges tab to different colors: edges between friends are rendered blue, whereas edges between subscribers are rendered orange (Fig. 14.9 (I)). An overall look at the network reveals two subnetworks, however color-coding alone does not offer a deep insight as to their structure.

For a deeper look into these structures, manipulate the vertices data in a way that reveals people who are more connected to others than most in the network. Each user's degree is listed in the Vertices worksheet. Degree measures are divided into in-degree and out-degree for directed networks, counting the number of ties from and to each user. Overall degree is inserted automatically for undirected graphs. In a directed graph, degree can be manually created by adding in-degree and out-degree to create a total count of relationships in the Overall Degree column—this is not similar to the degree that was defined in Chapter 5, but this "Overall Degree" can provide us with an insight as to the total connectedness of the vertices in the network. Sorting that column from largest to smallest you can see, unsurprisingly, that Leesha Harvey has the largest overall degree. Use the dynamic filters in the graph pane to systematically uncover the patterns underlying the network. Changing the filters for overall degree by small increments, for example, can reveal users with higher overall degree. When the graph is filtered so that only users with an overall degree of 25 or more remain (about 10 percent of the network), a pattern emerges; several hubs of dense connections, most based on friendship are hidden in the initial mass of ties, and are highlighted when focusing specifically on the overall degree. These hubs are the central users who create the network. To learn more about these hubs, use NodeXL's ability to directly link from a workbook to its web-based information. Right-click on the vertex to open the context menu; at the bottom of the Vertex context menu, choose the "Open YouTube page for this person." The relevant channel will open in the default web browser. The central users' channels reveal their commonality—all are aspiring artists, singers, and more than that—they share the same musical genre (folk).

The next step in revealing more insights from the YouTube network is to add display attributes that are mapped to individual properties of the central users.

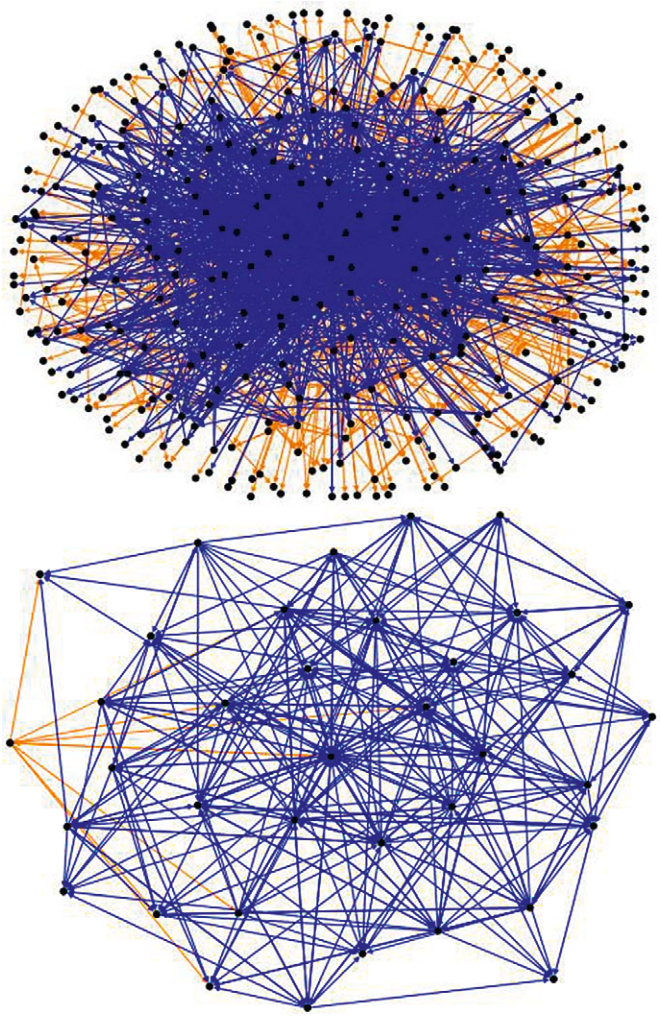


FIGURE 14.9 NodeXL maps of Leesha Harvey's egocentric YouTube networks. Layout I displays the overall friendship (blue) network and the underlying subscription (orange) network; layout II shows them after filtering based on degree ≥ 25 and reveals that the friendship network is denser than the subscription network, with only one subscriber who also befriended many other YouTubers.

Using the Harel-Koren layout and the Autofill Columns feature, map the following data attributes to display attributes of the network visualization:

- Map edge opacity to edge weight.
- Map vertex shape and size. Because you are now looking only at users with an overall degree of 25 or larger, you need to further refine the mapping. This phase may require experimentation to find filter settings that highlight key contributors and positions in the network. After several attempts you may realize that at a setting of degree of 40 or larger you can clearly see several prominent users (who comprise 2.5 percent of the network) in a way that is not too cluttered. Users who have more than the selected overall degree will be displayed using an image whose size will depend on the overall degree and will range between 1 and 6. Other users will be displayed using the default settings.
- Map Tooltip to username. To learn more about the vertices, the tooltip should be set to display the vertex (user) name.



FIGURE 14.10 NodeXL map of Leesha Harvey's egocentric YouTube network with images, edges filtered by edge weight and vertices by degree. This layout shows the most connected users in Leesha Harvey's network, most of whom are her friends, with only one subscriber included in the network. Clicking on the images will link to the users' channels and will show that the majority of them share the same musical genre as Leesha Harvey.

The result is displayed in Figure 14.10.

Leesha Harvey is now clearly located at the center of the network, with several other prominent users arranged around her. Most of these subnetworks are based on friendship ties, but one is centered around a user (RachelPachel) who is both a friend and a subscriber of Leesha Harvey. This user is also a hub of subscription ties. To learn more about her, you can use NodeXL's feature that links a vertex (user) directly to a web page from the workbook. RachelPachel's YouTube channel displays links to other "vlogging" or video blogging users in YouTube with many links to other musician vloggers, portraying her interest in music. You can repeat these steps for other central vertices of the network and see the ways in which they are contextually connected. You can also examine how these ties are reflected when other statistics, such as subscriptions, number of friends, number of videos, or date of joining YouTube are mapped into the visualization.

What is clear, though, just from looking at Figure 14.10, is that this network is extremely different from Rihanna's network. You can easily see that there are many interconnections and hubs among the network vertices, reflecting a more cohesive and interactive network. Though an

egocentric network in nature, it gives you a glimpse into a subcommunity of YouTube users, comprised of lesser known musicians and fans who share a common interest and, perhaps, also support each other in their quest for success.

14.8.2 Video Networks

The second layer of networks that can be found on YouTube are the content-related networks that stem from the various kinds of links between videos uploaded on the site. These networks are less about personal affinity or depiction of preexisting relationships and more about shared topicality and thematic association. Understanding video networks can offer you insights about several important happenings on YouTube, from the phenomenon of viral videos to institutionalized information dissemination; how independent and sponsored videos are connected to each other and how users form these connections and react to them can help others fathom the way YouTube is used for different purposes, and guide their own engagement in this community.

The following examples will look into YouTube's content, through the video networks.

14.8.3 The YouTube "Makeup" Video Network

NodeXL can import YouTube video networks based on several properties. Particularly useful is the feature that enables to import networks of videos linked by shared tags. Videos are frequently tagged with key words that describe its content. By following links between videos created by shared tags, researchers can gather a network of videos that share similar content. This is mostly relevant when trying to identify the ways videos on the same topic are connected to one another. Do connections between some video and users indicate more than just shared concepts? These connection maps inform a strategy that recognizes that a link to some central users and videos may result in higher ratings or a growing number of subscribers.

Let's focus in our example on "makeup" videos. Videos tagged with the word "makeup" may come from several sources: some are cosmetics companies' efforts to extend their marketing to reach viewers online, some videos are from makeup professionals who promote themselves by providing tutorials to the masses, others are created by teenage girls who share their first experimentations with cosmetics. "Makeup" is one of the most popular topics on YouTube, with millions of videos related to the topic and an extensive presence in various categories.

To start analyzing the YouTube makeup video network, look for separate or overlapping groups or cliques of users created around the shared terms. Some uses of a specific term are distinct from each other (i.e., "relationship" and "makeup"), whereas other terms blend together ("eye shadow" and "makeup"). Some videos are clearly personal or amateur, whereas others are professional productions.

Start by importing the video network data into the NodeXL workbook. Using the "Import from YouTube video network" feature in the NodeXL Data menu, first enter the term that is to be used for the search: "makeup." Because we are looking specifically for videos that share the same tags, click on the first box. You can simultaneously import different types of video network connections, but for this example use only videos that are tagged with the same keywords. Although NodeXL can import large numbers of videos, because of the popularity of the topic and the huge number of related videos, limit retrieval to 200 videos.

Searching for the keyword that you have just typed into the search box, the application will look in all the possible search fields combined. Currently, the YouTube API does not allow for distributed searches that differentiate between various fields (e.g., header, description, tags, and category). Therefore, some of the videos will have the keyword "makeup" in their header or their description, and some will not. However, all videos (vertices) will have the "makeup" keyword tag.

After importing the YouTube makeup video network into NodeXL, the workbook will include the following: an Edges worksheet that includes pairs of vertices along with a column that describes their relationship (in our case all edges will be connected by shared tags; in other cases the relationship can be based on comments or video replies), and a Vertices worksheet that will be populated with information about each of the individual vertices (videos). Select the Show Graph button in the NodeXL Graph menu to display a visualization of the set of connections among the population of "makeup" videos.

The first look at a YouTube video tag network can be daunting. What can this blob in Figure 14.11 mean?

Recall that shared-tag networks are extremely and densely interconnected, creating this smeared look. Using a range of tools in NodeXL, you can look at deeper layers within this network, filtering obscuring details to reveal interesting things. The first step in this process is to prepare the data, first by using the Merge Duplicate Edges feature in the NodeXL Data menu. In the case of video networks this is especially important as multiple ties can be repeatedly created based on terms that appear in each of the search fields. In our example, the makeup video network, the Merge Duplicate Edges feature reduced the network from more than 28,000 edges to only 12,000 unique edges.

Once the data set is prepared, it is ready for the creation of network metrics. Compute them using the

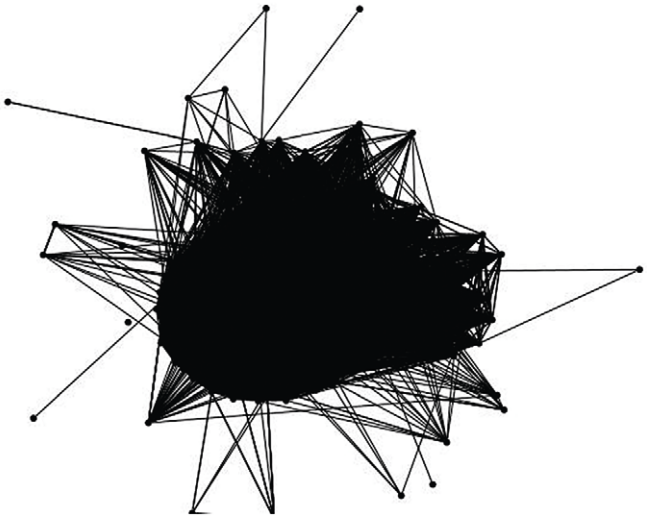


FIGURE 14.11 Initial NodeXL visualization of the YouTube video network for the tag "makeup," limited to 200 videos. This initial, incomprehensible, visualization is a starting point for exploration of the video network.

NodeXL Analysis → Graph Metrics menu option. Summary metrics about the network are reported on the Overall metrics worksheet. Reviewing these values shows that, indeed, this network is denser than the previous user networks that you have seen, with a density of 0.642. This may lead us to also think that many of the videos in the network that share the same tags also share other topic-related properties. Still, the network is difficult to decipher, and the visualization gives us only a point from which to begin our exploration. Filtering the network, based on edge weight, can remove some of the excessive data, especially the peripheral videos that do not belong to the core of the network.

Looking at the range of edge weight you can see that it is quite wide and spans from 1 to 15. There are two approaches to filtering in NodeXL: one operates on the display of nodes and edges in the visualization pane, and the other operates on the rows of data in the spreadsheet that feeds the graph visualization. Unlike dynamic filters, when an edge or a vertex is skipped using the Autofill Columns it will not be read into the graph visualization, and clicking on related edges or vertices will not display it. Filtering at the spreadsheet level can be useful for reducing the size of the data sent to computation intensive tasks like the calculation of metrics, clusters, and layouts. Data filtered at this level will never appear in the graph display no matter how the dynamic filters are set; you have to be careful about filtering this way and not exclude important parts of the network graph. In this case, using the Autofill Columns option is preferred, as at a later stage we will recalculate the network metrics to include only its core components.

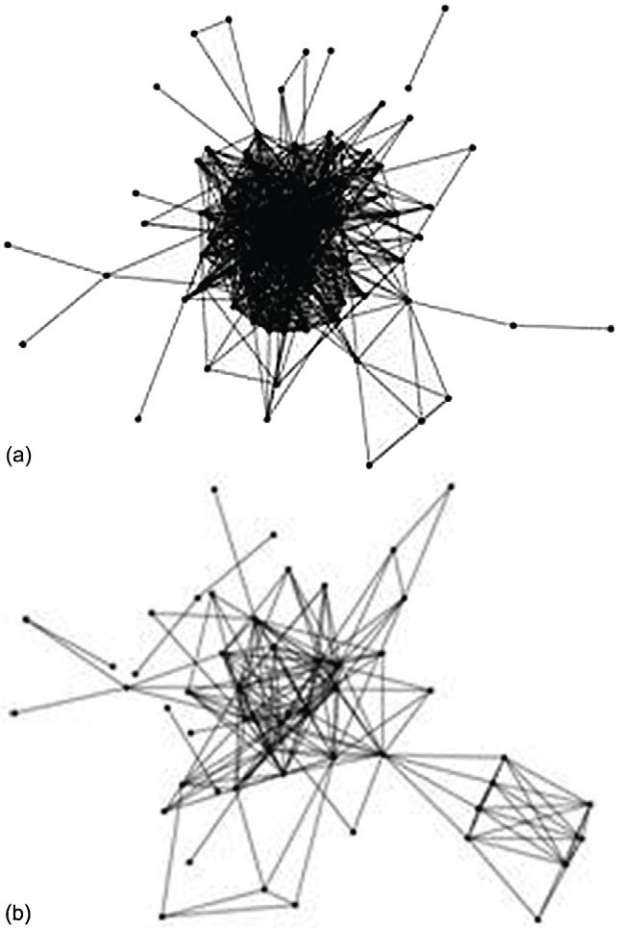


FIGURE 14.12 NodeXL network maps of the YouTube "makeup" tag, showing gradual filtering based on edge weight. (a) The network after filtering where edge weight ≥ 2. (b) The network after filtering where edge weight ≥ 4. Filtering reveals the network patterns from the mass of overall data.

Use Autofill Columns to display the wanted resolution of the network. To do that, select "Edge Visibility" and base it on "Edge Weight." In the right-hand Options tab, select an edge weight starting from 2 and periodically check the resulting visualizations.

In Figure 14.12, the graph has been filtered so that the visibility of edges depends on their weight. In (a), the network has been filtered so that only edges with an edge weight equal or larger than 2 show, and in (b), only edges with edge weight equal or larger than 4 are displayed. As you can see, the optimal visualization of the network, which is not cluttered yet still presents enough information about it, can be found when edge weight is ≥ 5. At this stage you can see several isolates that clutter the visualization. An easy way to de-clutter the graph is by using the "Binned" layout. From the graph pane select "Binned" layout and refresh the graph. The isolates should now be placed at the bottom of the graph, separately from the main graph. This

layout, while useful in dealing with isolates, is not very informative, so it is best to combine the binned layout with another. To do that, select all the isolates by drawing a rectangle around them. Right-click and open the dialog box; choose "Edit selected vertex properties" and select the option "Yes" from the "Locked" drop-down list. Now the isolates are locked in place, and you can choose the layout you prefer for the rest of the graph. In this case the Harel-Koren layout was chosen.

Once the filtering has presented a workable visualization, the next step will be to find clusters within the network. You can often recognize clusters in networks visually, but clusters can also be automatically identified using the NodeXL implementation of the Wakita and Tsurumi clustering algorithm. This feature creates a set of groupings of nodes based on their patterns of interconnection. It finds some obvious clusters but can also identify more subtle distinctions that may not be visually obvious. To use the automated clustering algorithm, use the "Find clusters" option on the NodeXL Analysis menu ribbon. Then, refresh the graph to exhibit the clusters. Look carefully at the clusters that were created: NodeXL assigns each cluster a unique shape or color, and sometimes different clusters may share the same color. If, as in this case, it is hard to differentiate between clusters because of the color similarity, use the workbook to manually set the clusters colors or shapes.

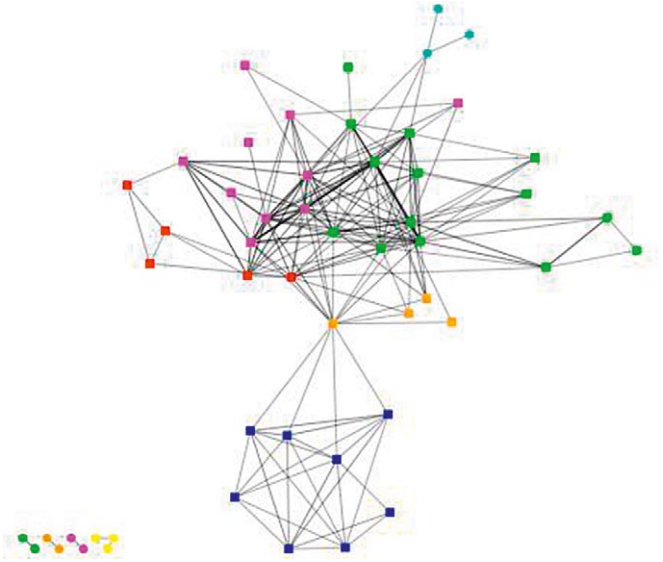


FIGURE 14.13 NodeXL graph of the YouTube Makeup video network after filtering for edge weight ≥ 5, opacity mapped to edge weight, and clusters computed. You can observe five major clusters of videos, two of which (pink and green) are visibly denser hubs of interconnections that span outside the cluster to include videos that belong to other clusters; and the other three (blue, red, and orange) are more sparse and less interconnected. Several isolates are placed at the bottom of the graph.

Figure 14.13 reveals the network clusters. Two clusters are more dominant than the others—the green and pink clusters. The varying opacity shows that they not only have a larger number of vertices but are also densely interconnected, with multiple edges bridging vertices from both clusters.

Automated clusters are created based on attributes the algorithm finds relevant, but understanding what connects these videos requires further investigation. To do that, turn to the Vertices worksheet, which contains a list of all the videos in the "makeup" network. Select the column labeled "Degree," and then select the sort feature from the Excel Data Menu ribbon. Sort the vertices from the largest degree to the smallest. The worksheet now lists videos by how connected they are to other videos. Scrolling down through this list of vertices shows that most videos with a degree larger than 10 belong to the green and pink clusters, which are also the most prominent, connected, and central clusters in the network.

To understand the ties between videos in the different clusters, you can right-click on each video vertex to open the context menu, which links directly to the video's web page. Viewing the videos of the green and pink clusters reveals that the videos in these clusters are tutorial videos in which amateurs or experts detail how to successfully achieve a certain look. The differences between the clusters are relatively small and reflect the fact that most videos in the pink cluster were authored by just two users (panacea81 and MissChievous), whereas the videos in the other cluster were authored by a larger group of users. Similarly, the blue cluster is composed of videos made by one user (Michelle Phan) or videos that were inspired by this user. The pink cluster also has a contextual commonality, many of the videos in this cluster present Halloween and vampire makeup tutorials.

It is interesting to discover that no commercial videos or product placement videos were included in this data set. That may be because of cosmetics companies' reluctance to use YouTube as an advertising venue or due to the popularity of tutorial videos in comparison to commercial content (a search for "makeup commercial" retrieves only about 4800 results, compared with more than 300,000 results for makeup tutorials and 46,000 for makeup tips), it can also be attributed to the sample size.

However, if commercial corporations want to use YouTube as an advertising platform, they should consider how and through which route they can best promote their products. One way is to engage prominent users in product placement (i.e., using specific products in their tutorials or tips). To find the most strategic users to approach advertisers and product producers need to know who the central users that affect the community

more than others are. Some users are more influential than others.

Social network analysis has several measurements that can be argued to relate to the "influence" of a position in the network. One way you can identify influential users in YouTube networks is by sorting users by their "betweenness centrality" measurement. This value captures the extent to which a person is the only path between two otherwise separate parts of the network. Each of us is "between" separate networks, and we "bridge" these networks, linking our families to our workplaces, for example. In YouTube, some videos are linked strategically so that they are the only connection between separate clusters of videos. At this stage it is important to remember that at the beginning of this analysis you have filtered out many of the videos that had edge weight of less than 5. This not only allowed for a clearer visualization but also affected the accuracy of the network metrics that were previously calculated. Videos that were previously central or prominent may change their position in the network after the filtering. To update the metrics to reflect the current network, calculate the graph metrics again. You can now see that the overall graph density doubled to 0.13, meaning that you are looking at the core of the YouTube "makeup" tag network.

Looking at Figure 14.13, it is clear that one video is the only bridge between two major clusters in the network: the blue cluster and the green/red/pink/orange one. Using the tooltip you can see the video title "Natural makeup." To see if the prominence of "Natural makeup" is reflected in the network metrics, turn to the Vertices worksheet and select the Betweenness Centrality column; sort from largest to smallest using the Excel Data Sort menu ribbon. Select the top row and click or scroll through the list and look at the placement of the high betweenness videos within the clusters. It appears almost immediately that the same video also stands out as having the highest betweenness centrality. This video is a boundary object around which the YouTube "makeup" network congregates. Boundary objects are intellectual concepts, artifacts, or objects that connect different communities of interest [2, 5], though each community may interpret or use them differently. Boundary objects are also used as translational mechanisms—they provide a channel for transferring information, ideas, and understandings between different communities, where each community recognizes the boundary object's common structure yet applies to it its own interpretation. A video is not a boundary object per se—it becomes one when different communities give it meaning and use it as such. In this case, the "Natural makeup" video connects one cluster of videos that offers tutorials for naturally looking makeup application (ranging from guy makeup to makeup for women who wear glasses) and videos that offer tips and tutorials for more elaborate—sometimes theatrical—makeup. It is elastic enough to accommodate the different meanings attributed to it by the different clusters, bridging between the everyday and the alternative, while maintaining its own unique content and remaining connected to both groups (as such, it also has one of the largest degrees in the network).

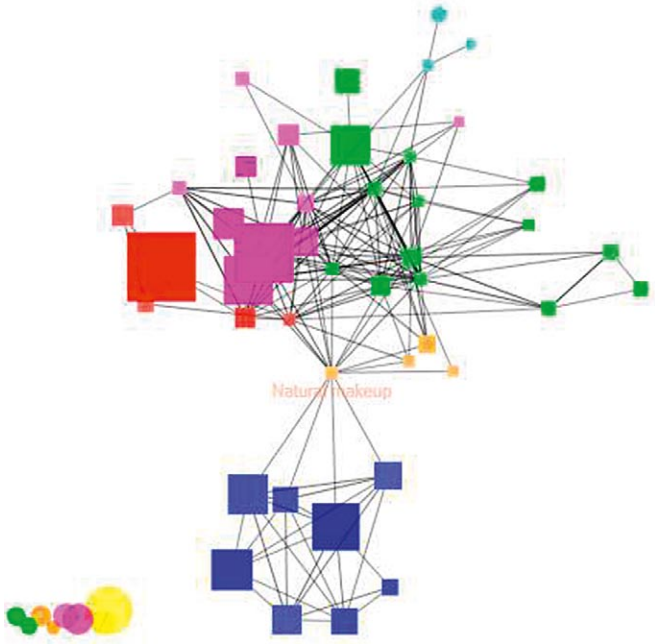


FIGURE 14.14 The position of "Natural makeup" within the NodeXL graph of the YouTube makeup video network. Vertex opacity and size are mapped to views and comments, respectively. The betweenness centrality of this video indicates that it bridges between one otherwise isolated cluster and several other clusters. It is a boundary object that connects several communities of interest in the network; however, this role is not reflected in overall popularity of the video in the YouTube network.

It is interesting to examine the overall placement of the same video in the entire YouTube network. This can be done using the statistical data provided by YouTube (e.g., views, favoring, number of comments), which is also imported into NodeXL. From the statistics it is obvious that "Natural makeup" lags behind many other videos that are less central in the network. It was viewed 123,378 times (compared to more than 3 million views of the most viewed video in the network), commented on 4753 times (compared to the most commented video that received more than 24,000 comments), and favored 9896 times (compared to the favorite video that was favored more than 23,000 times). As Figure 14.14 shows, "Natural makeup" is not the most outstanding video, when considering YouTube popularity statistics.

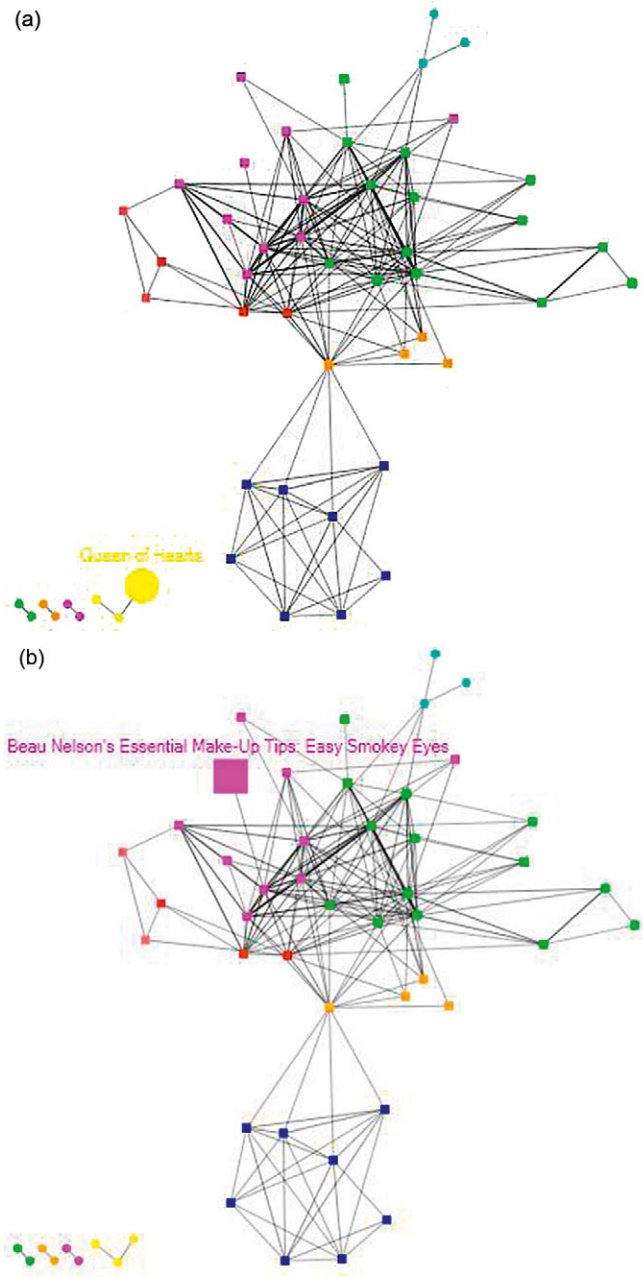


FIGURE 14.15 NodeXL maps of the "Queen of hearts" (a), the highest rated video in the YouTube makeup video network is an isolate, whereas "Beau Nelson's Essential Makeup Tips" (b), the most favorited video in the network, is peripheral to the core of the network, connected to only one other video.

You can now go back to the Vertex worksheet and examine the YouTube ratings for other videos in the network: the highest rated video in the network (and more commented on than "Natural makeup") doesn't belong to a central cluster but is an isolate ("Queen of hearts"), and the most viewed video also belongs to a peripheral edge ("Beau Nelson's essential makeup tips").

Figure 14.15 shows that betweenness centrality and degree do not necessarily indicate the popularity or prominence of a video within the general viewing population of YouTube. But these measures may reflect the place a video has within the smaller scope of a specific community of enthusiasts, as is the case for the makeup network. In such cases, a gap exists between internal importance and centrality of a video and its influence outside its cluster. When advertisers want to select channels through which they will promote their products, they should consider finding the central users who may be less popular but more influential in each relevant subcommunity of interest in YouTube. Videos and users that have widespread popularity among a general population of viewers may lack influence within local communities of interest.

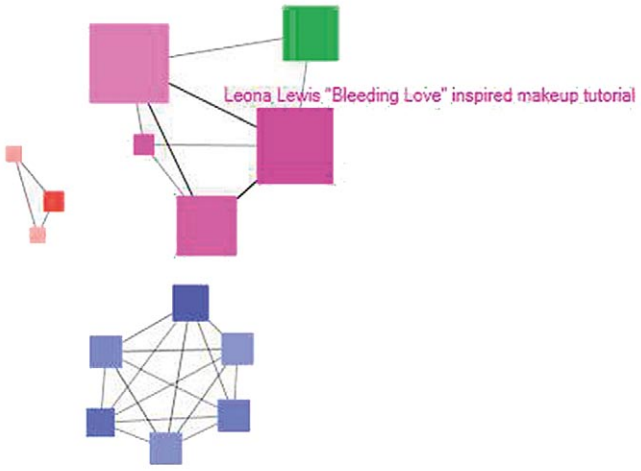


FIGURE 14.16 NodeXL chart with vertex size mapped to degree, opacity to number of views, and vertex visibility to the number of comments a video received, Panacea81's "Leona Lewis 'Bleeding Love' inspired makeup look" video stands out as combining the highest centrality metrics as well as the highest overall popularity measures in the YouTube network. This video, and its author, can be a good starting point for exploring the network and for commercial efforts.

Videos in the YouTube network have many attributes, including the number of times they were viewed, the number of times they were favorited, and the date they were uploaded. Network graphs have various display properties, like the color, size, shape, and location of nodes. To display complex combinations of network attributes, you can map different metrics to different visual properties. A node may hold different attributes like its centrality in its local network and popularity in the overall YouTube population, measured in terms of the number of views or comments, as is done in Figure 14.16. These additional attributes can help distinguish between different kinds of popularity and activity in YouTube, showing that some videos are popular but do not generate discussion and vice versa.

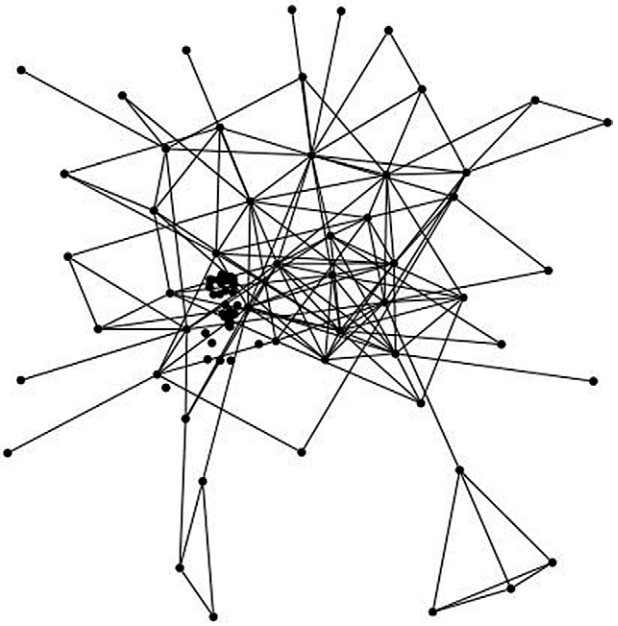


FIGURE 14.17 Preliminary layout of NodeXL map of YouTube "healthcare reform" video commenter's network, using the Harel-Koren layout. This layout is relatively unhelpful in revealing network patterns but is a starting point for further analysis.

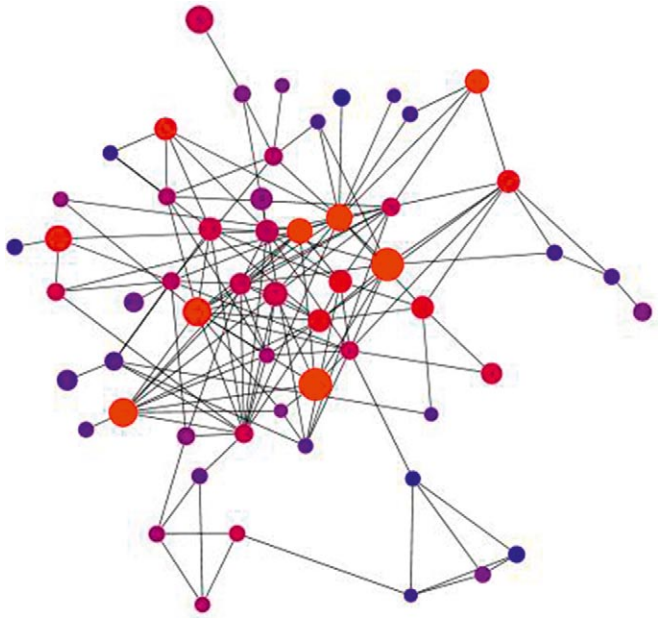


FIGURE 14.18 NodeXL map of YouTube healthcare reform videos with color and size corresponding to views and comments, respectively. The number of comments and views do not necessarily correlate: red vertices, which generated many comments, are not always the most viewed (mapped to larger size), whereas several blue vertices (low number of comments) were viewed as many times as the more commented-on videos.

14.8.4 Healthcare Reform YouTube Video Networks

YouTube videos reflect trends, debates, and popular news. Many videos present news segments from mass media outlets, political campaigns, lobbying efforts, or they provide commentary on these topics by pundits and vloggers. Mapping the networks created by discussions linked to such videos, through textual comments and reciprocal videos, can allow users to understand them better, react to them, or, in case of political and controversial issues, provide counter information.

One of the most heated debates in the United States at the time this chapter is written is the national healthcare reform. A YouTube search of the keywords "healthcare reform" provides more than 53,000 videos dealing with the subject. In looking at those videos, and specifically the comments that users left after watching them, you can gain insight into which videos generate more responses. How does their position within the user and the video networks relate to their rank in popularity and the number of views they generate? And do links from some users drive more traffic than others?

To find answers to these questions, import a sample of videos that have the keywords "healthcare reform" in their descriptors through the NodeXL YouTube Data import feature. Filter the data request to retrieve only pairs of videos commented on by the same user. After importing the data and merging duplicate edges collected, a network data set that contains 147 edges and 102 vertices is created.

You can identify several hubs of videos in this network, with some isolates in its midst. Clean these isolates, to get a better look at the network, by manually deleting them from the workbook (it is advisable to save an original copy of the data set with all the imported data and save the manipulated copy under a different file name).

First, find the most popular videos in the network based on the statistics provided by YouTube. Using the Autofill Columns feature in the Visual Properties section of the NodeXL menu ribbon, map Vertex color to the number of comments each video generated, and map Vertex size to the number of times each video was viewed.

You can see several highly interconnected hubs of videos, which were repeatedly commented on by the same users. Most of the core videos received a substantial number of comments, reflected in their color (red being the most heavily commented videos); they are also quite large, showing the number of times they were viewed. Peripheral videos received fewer comments, for the most part. The less popular videos orbit around the dense core of the network.

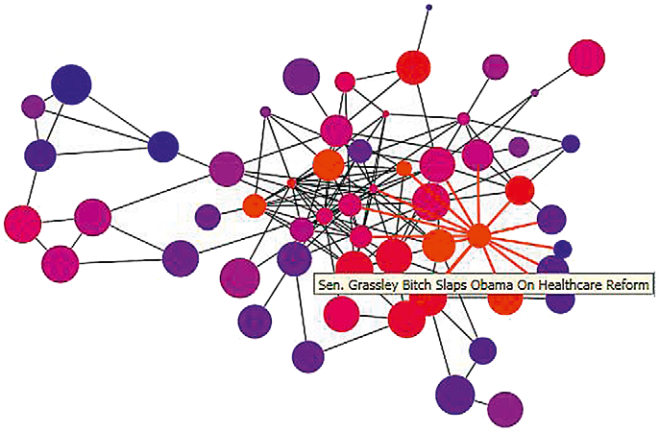


FIGURE 14.19 NodeXL map of YouTube healthcare reform video network with color and size corresponding to the number of comments and ratings for each video, respectively. The blue vertices, which are not frequently commented on, received (in general) higher ratings than the more commented-on videos. This may be the outcome of contentious content that generated heated discussion but dissent that was reflected in lower ratings. The highlighted video has the highest betweenness centrality, making it a pivotal video in the online discussion.

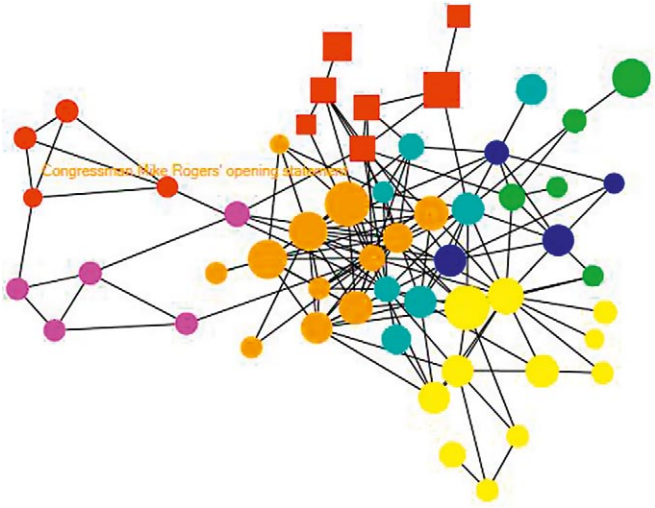


FIGURE 14.20 NodeXL map of clusters of YouTube videos discussing healthcare reform linked by shared comments. With two exceptions (the yellow cluster reflecting opponents to the administration healthcare plan and the red cluster reflecting videos supporting the plan), most clusters do not portray contextual ties between the videos.

Another interesting question is whether the number of comments and the ratings correlate. Videos about such a controversial topic may generate several possible responses: commenters may or may not agree with the content of the video and exhibit their opinion not only through their texts but also by changing the video ratings. To explore this relationship, you can alter the way you visualize the network. Maintain the mapping of vertex color to the number of comments but change vertex size to reflect the rating for each video. This changes the picture somewhat: some of the videos that received fewer comments (colored blue) are the largest, meaning they received the highest ratings, whereas others that were followed by a lively discussion are (colored red) are in fact rated lower than the less commented videos. This corroborates our assumption that highly commented videos are not necessarily the most popular ones, and the high number of comments could, in fact, be the result of controversial content, followed by lower ratings from users who object to that content.

Before looking into the content of specific videos, you can also examine which videos are central to the discussion. Using degree and betweenness centrality you will see that the most connected and central video ("Sen. Grassley Bitch Slaps Obama on Healthcare Reform") is not one of the most commented and, in fact, is among the lowest rated videos (although none of the videos in this set dipped below 3.5).

However, within the network of healthcare-related video commenters it has generated enough attention to become central. You can learn from the title of the video about its contentious content, but a closer look at the video reveals it is a satirical liberal news commentary called "The Young Turks," which offers video content exclusively on YouTube. Their irreverent commentary elicits many responses (through comments and ratings), both in favor of the opinions they express and against them.

You can also use the same method used for the makeup network described in Section 8.2.1 to look for clusters within this network.

In this visualization, size is mapped to the number of comments, and colors and shapes are set according to clusters. The tooltip, again, is mapped to display the title of the video. Although two very distinct clusters can be found: with the yellow cluster including mostly videos that are posted by opponents of the proposed healthcare reform and the red one including videos supporting the plan, most of the other clusters are made of videos that do not share many contextual commonalities. An interesting finding is that several of the clusters hold videos that represent one side of the debate and videos responding to those arguments from the other side of the fence. One example is the video that captured Congressman Mike Roger's opening statement on healthcare in the Congress (orange cluster), which created a heated discussion of more than 13,000 comments and was also replied to by a couple of response videos, proposing both support and objections to the views expressed in the original video.

Looking deeper into the clusters, and using the tooltip, you can observe that the most commented-on videos are, unsurprisingly, those that involve politicians,

followed closely by parody and satirical videos. Personal commentaries and informational videos about the reform details are discussed less.

When considering ways to utilize YouTube to promote discussion or disseminate information about the proposed healthcare reform, researchers and practitioners should take into account which videos generate discussion, commentary, or even controversy. These are the ones that will remain in the spotlight and gain attention from other sources, such as mass media. As you saw, although many users on YouTube express their views, these videos remain mostly outside the mainstream of lively discussion. The analysis of healthcare reform–related videos suggests that proposing a satirical take on the discussion or using excerpts of politicians' media appearances or political rallies will generate a more lively and prolonged discussion around those videos.

14.9 PRACTITIONER'S SUMMARY

Analyzing YouTube social networks can offer many insights into the ways videos become popular, sometimes even becoming "viral," and the way information is disseminated through videos. YouTube's popularity makes it a channel that enables professionals, from marketing experts to political advisers, to gauge popular themes and public trends. Analyzing video networks will make it easier to decide on the types of interventions, the creative routes that will maximize their outcome, and, even more important, what approaches *not* to take in order to avoid negative backlash.

For artists and content producers who are not backed by large-scale media corporations, simple social network analysis—focusing on egocentric and content networks—can give a glimpse of the many facets that affect success and popularity within the YouTube universe.

As YouTube becomes an important tool for information dissemination in other noncommercial domains, such as education and public health (for example, the Center for Disease Control's streaming channel[2]), social network analysis of a different kind—one that explores which audiences can be reached through YouTube videos, for example—can provide a useful tool to coordinate funding and production efforts in an efficient manner.

These observations, coupled with YouTube's immense popularity, provide deep insights into one of the major media outlets in existence today. The size of the YouTube network can be daunting, but by focusing on appropriate samples of data, filtering with relevant metrics, and using visualizations extensively, you can grasp at least part of what shapes our contemporary culture.

[2]www.youtube.com/user/CDCStreamingHealth

14.10 RESEARCHER'S AGENDA

Despite YouTube's immense popularity, the research on YouTube's underlying social networks is in its early stages. Although practitioners, from marketing experts to educators, have attempted to explore these networks to gain an understanding of the best ways to utilize YouTube for information dissemination, researchers have, for the most part, preferred to study more "obvious" social networks such as Facebook and Twitter. Structural studies of YouTube have focused on its overall macrostructure [6–7] or on a category level [3, 8]. The rich composite data found in YouTube offer a compelling reason to map the networks of connection it contains. The combination of user-generated content and social ties can illuminate many phenomena that shape not just our popular culture [9] but also the way institutional information is disseminated or the ways in which public opinion is broadcasted [10].

Using social network analysis, researchers can identify important YouTubers or pivotal videos, as well as the types, structures, and development of the networks that are created around them. Researchers can also explore how the structure of ties and networks on YouTube affect content creation. The interplay between content and structure is one of the more important attributes of YouTube and is worthy of deeper exploration.

As we have shown, different users or videos have different networks built around them. Understanding the nature and evolution of these networks can lead to improved use of YouTube by users and enterprises or by designers of video-sharing interaction. Researchers can extend our knowledge of the social processes that underlie the YouTube interaction and the ways the social networks that exist on the site contribute to the popularity of or disregard certain views, opinions, or video content.

References

[1] C. Hurley, Y,000,000,000uTube, in Broadcasting Ourselves, The Official YouTube Blog vol. 2009, ed. San Bruno, CA, 2009.
[2] E. Wenger, Communities of Practice: Learning, Meaning, and Identity, Cambridge University Press, Cambridge, MA, 1998.
[3] D. Rotman, et al., The community is where the rapport is: On Sense and Structure in the Youtube Community, Presented at the Proceedings of the Fourth International Conference on Communities and Technologies, University Park, PA, June 4–7, 2009.
[4] D. Rotman, J. Preece, The "WeTube" in YouTube: Creating an Online Community Through Video Sharing, International Journal of Web-based Communities, 6 (2010), 2.
[5] S.L. Star, J.R. Griesemer, Institutional ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, Social Stud. of Sci. 19 (August 1, 1989) 387–420.

[6] M. Cha, et al., I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system, presented at the Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, San Diego, CA, October 24–26, 2007.
[7] G. Geisler, S. Burns, Tagging Video: Conventions and Strategies of the YouTube Community, presented at the Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, Vancouver, BC, Canada, June 18–23, 2007.
[8] J.C. Paolillo, Structure and Network in the YouTube Core, presented at the Proceedings of the 41st Annual Hawaii International Conference on System Sciences, January 07–10, 2008.
[9] J. Burgess, J. Green, YouTube: Online Video and Participatory Culture, Polity Press, Malden, MA, 2009.
[10] V. Gueorguieva, Voters, MySpace, and YouTube: the impact of alternative communication channels on the 2006 election cycle and beyond, Social Sci. Comput. Rev. 26 (2008) 288–300.

CHAPTER

15

Wiki Networks

Connections of Creativity and Collaboration

Howard T. Welser[*], *Patrick Underwood*[†], *Dan Cosley*[‡], *Derek Hansen*[§], *and Laura W. Black*[**]

[*]Departmentof Sociology and Anthropology, Ohio University, Athens, OH
[†]Department of Sociology, University of Washington, Seattle, WA
[‡]Information Science, Cornell University, Ithaca, NY
[§]iSchool, University of Maryland, College Park, MD
[**]School of Communications Studies, Ohio University, Athens, OH

OUTLINE

15.1 Introduction 248

15.2 Key Features of Wiki Systems 249

15.3 Wiki Networks from Edit Activity 252
15.3.1 Wiki Networks of General Interest 253

15.4 Identifying Different Types of Editors within a Wiki Project 253
15.4.1 Wiki Social Network Sampling Frame and Data Collection 254
15.4.2 Defining Edges and Attributes in Wiki Social Networks 254
15.4.3 Wiki Network Data Collection 255

15.5 NodeXL Visualization Strategies for Revealing Distinct User Types 256
15.5.1 Making Top Wiki Editors Stand Out by Visually Formatting the Network Graph 256
15.5.2 Interpreting Wiki Network Graphs for Evidence of Distinctive Social Roles 257
15.5.3 Using Subgraph Images to Distinguish between User Types 259
15.5.4 Seeing the Trees and Forest with Wiki Network Analysis 261

15.6 Identifying High-Quality Contributors in Article Talk Pages 261
15.6.1 Tasks and Strategies for Identifying Types of Contributors by Visualizing Article Discussion Page Networks 262
15.6.2 Searching for Structural Signatures of Confrontation and Deliberation in Wiki Article Talk Page Networks 262

15.7 Navigating Lostpedia: Using NodeXL to Reveal the Large-Scale Collaborative Structure of Wiki Systems 264
15.7.1 Creating an Overview Network Map of Lostpedia Content in NodeXL 264
15.7.2 Creating an Overview Map of Lostpedia Users 266
15.7.3 Normalizing Data to Infer Stronger Connections 268

15.8 Data Collection from Wiki Systems 269

15.9 Practitioner's Summary 269

15.10 Researcher's Agenda 270

15.1 INTRODUCTION

A wiki is a web site anyone can edit, where each modification to each page is recorded and archived. The first wiki system, the WikiWikiWeb, was invented by Ward Cunningham in 1995 to allow a group to easily and quickly ("wiki" means "quick" in Hawaiian) edit a set of web pages without having to know HTML or deal with moving files back and forth to a web server. In addition to reducing the technical barriers to creating web pages, wikis make it easy for people to collaborate on writing tasks because the technology of the wiki provides separate spaces for people to create content and to discuss issues related to the content they are creating. Because of their flexible structure, support for discussion, and ease of use, wikis are becoming an increasingly popular platform for supporting online communities. For example, Wikia, a company that hosts wiki communities, has more than 50,000 separate groups, and it is only one of many such platforms. This chapter explores how to use the activity on wikis to create social network representations that can help community designers understand who and what is important to a given community, determine whether the community is healthy, and distinguish between different types of contributors.

Although invented in 1995, wikis remained relatively unknown until about 2003, when Wikipedia, the online encyclopedia anyone can edit, started to come to prominence. Wikipedia has become the dominant source for encyclopedic information online, and it is increasingly understood as a social force that challenges traditional notions of authority, expertise, and knowledge construction [1, 2]. Many organizations and educational institutions such as SocialText, PBWiki, WikiSpaces, Wetpaint, and Wikia, now use wikis as knowledge repositories for users. Open source projects and technical question-and-answer (Q&A) communities use wikis for documentation and support. Although most social media tools support information sharing, wikis are unique in their ability to support collaborative content creation and maintenance.

Wiki systems are also excellent sources of data for social network analysis [3, 4]. Although wikis are rich sources of social network data, they are one of the more technically demanding forms of social media networks to extract and work with. Many wikis give rise to large data sets, which must be either parsed or sampled before any analysis can begin. Wikis offer many types of pages and several ways for editors to interact, providing a more complex social context than systems like email, chat, or Twitter. In this respect, wikis occupy a similar social and practical space as social networking sites like Facebook or MySpace (see Chapter 11).

The majority of wiki communities create information repositories about a specific topic. Whereas some wikis like Gardenology or the Marvel Comics Database take the form of an encyclopedia, other wikis encourage many genres of pages such as lists of annotated links, debates, or how-to documents [5]. The flexibility of wikis makes them highly adaptable to the needs of their users. Company employees use internal wikis to share information and coordinate effort, teachers use the Classroom 2.0 wiki to collaboratively create and share lesson plans, cancer patients share information about clinical trials and doctors, intelligence analysts use Intellipedia to share information across agencies, fans of the TV show *Lost* use Lostpedia to make sense of a complex storyline, and World of Warcraft players use WoWWiki to document the game universe and provide "walkthroughs."

Despite their wide use and potential as data sources, wiki systems constitute a challenging and complex form of online community with numerous modes of interaction and potentially dozens of ways to represent relations in the form of a network graph. Before beginning an investigation of any wiki system, it's important to address several questions. Which modes of interaction are most valuable for understanding your online community? Which definitions of an edge are consistent with socially meaningful interactions in the system? What time frames of interaction should be used to demarcate the boundaries of your data collection? Because of these questions and because of the technical challenges associated with collecting many kinds of network data, wiki systems should be considered an advanced social media system for both practitioners and researchers. However, careful planning and data management techniques can make wikis one of the most rewarding areas for network analysis of social media.

The focus of this chapter is on introducing several exploratory methods for working with data from wikis in a social network analysis framework. There are many types of wikis that might be of interest for social network analysis, we begin with a discussion of these and several questions that could be asked of wiki network data and describe different types of networks that could be extracted. We describe attributes of wiki systems and how those attributes and types of wiki pages are often related to how people interact in wikis.

The core of the chapter introduces three example inquiries employing NodeXL to address three different types of questions in wiki systems. The first example studies a set of wiki pages at the Empire Wiki that are related by the Castle Project, and it seeks to identify different types of contributors to that project based on both their network attributes and key variables related to the types of pages they do, and do not, edit. The data for this example were created using a commonly available web-scraping tool to identify cases and collect data. Understanding and identifying different types of contributors is useful because it can help

web administrators locate highly valuable contributors of different types and figure out whose participation should be encouraged, rewarded, or, in some cases, discouraged or redirected.

The second example examines the quality of online discussion on the "talk" pages associated with a particular project on Wikipedia. This is a small-scale study with an emphasis on qualitative analysis and small group interaction. Much of the interaction on wikis centers on collaborative writing tasks. However, equally important is the discussion that contributors have with one another about what they are writing and why. When the content being created is crucial to the health of an organization, it could be useful to assess the quality of contributors' discussion about their work. This research study illustrates the use of content analysis to measure both edges and vertex attributes. NodeXL is used to visualize patterns in the discussion, identify high-quality contributors, and locate problematic interactions where the discussion is not helping the group achieve its task.

The final example illustrates strategies for revealing large-scale structure of editing patterns in wikis, drawing on data from Lostpedia. The analysis uses network data derived from shared edits of article pages, including comparisons that show how pages are connected by shared editors, and showing how editors are connected by shared interests or areas of attention. This strategy illustrates the use of affiliation networks and large-scale analysis for exploration of an entire wiki, providing a means for making sense of hundreds of thousands of edits and participants.

15.2 KEY FEATURES OF WIKI SYSTEMS

We start with a tour of some of the key features of wiki systems that support studying the ways people behave in them. Wikis have a more complex structure than most of the other social media network data sources discussed in this book, such as Twitter or email. In particular, wikis keep a *history* of all editing activity that can be viewed on a per-page or per-contributor basis; they use *namespaces* to organize different kinds of contributions to the wiki; wikis are built on the premise that *everything is a page*: the content itself, discussion of the content, individual contributors, supporting tools like category pages, and even the policies of the community itself are all contributed by users and evolve over time; and they have *user accounts*, which track activity by each person, allow self-disclosure, and allow direct communication between people. Our discussion will be based primarily on Wikipedia and the MediaWiki software that implements it; however, many wiki sites use the MediaWiki software, and the general concepts of history, namespaces, and everything-is-a-page are common to most wikis.

Most people's view of wikis comes from seeing Wikipedia pages that result from a web search. For example, as of this writing, searching for "Belgian beer" on Google returns the Wikipedia page for Beer in Belgium as the first result (Fig. 15.1). The obvious feature of the page is that it presents information about the page's topic; in Wikipedia, this generally takes the form of an encyclopedia article, although as mentioned in the introduction, these pages can provide other forms of information such as lists of links or instructions.

History. However, the visible topic page is like the tip of an iceberg of activity that happened in the course of creating it. In fact, every time someone edits a Wikipedia page, using the "edit this page" tab, the software remembers who made the edit, why the user said he or she did so, when it happened, and what changed. Clicking the History tab in Figure 15.1 will show a timeline of every change to a page; likewise, there is a tool for seeing "user contributions" (not shown) that shows a record of every edit made by a given user (Fig. 15.2). This historical information has allowed researchers to study everything from the overall growth of wiki systems [6] to effective patterns of collaboration in these communities [7, 8]; it also supports the development of technical tools that make recommendations based on the topics in which people show interest [9]. Wikipedia provides a number of tools and resources for getting the history of a page or set of pages, including database dumps and an official Application Programming Interface (API) that allows external scraping tools access to the database.

Namespaces. Namespaces organize different kinds of contribution. Sometimes, editors disagree on exactly what should be on a given page. Rather than continually editing each other's versions of a page (a phenomenon known as an "edit war") or marking up the page itself with comments and disagreements (which often happens, for example, when people collaboratively edit a Word document), the MediaWiki software separates article creation from article discussion using namespaces. The articles themselves are in the "main" namespace; for each article, there is an associated page in the "talk" namespace where people can talk about the page, ask questions, request clarification, and resolve disagreements without affecting the main page too drastically. For the main content pages of the wiki, clicking on the Discussion tab (Fig. 15.1) will open the related talk page. Although the underlying tool is the same, these talk pages typically take the form of threaded discussions. Figure 15.3 illustrates how contributors use talk pages to discuss page specific edit decisions and to communicate normative standards of appropriate editing. Separating discussion from other editing makes it



FIGURE 15.1 This article page from the English-language Wikipedia displays content and illustrates discussion, edit, and history tabs. These tabs are standard to most wiki systems and they provide access to edit records from which edge relationships and attributes can be measured.



FIGURE 15.2 Wiki pages have a related history page that depicts the timing of every edit, indicates the editor or IP address responsible for the edit, provides space for a brief description of the edit, and displays links to the state of the page before and after the edit. History pages are important sources of network and attribute data in wiki systems.



FIGURE 15.3 This article talk page is used to coordinate decisions about the best contents for the article page. The edits to this page are made by people who have an interest in the content page and are often made by people who actively edit the article page. This page shows evidence both of content-based discussion and the implementation of templates to encourage compliance with community editing norms.

easier to study the specific ways people collaborate and the strategies they use to resolve disagreements [10, 11]. The threaded nature of the discussions provides a natural way to create a social network based on Wikipedia activity, by creating ties between people who reply to each other in a discussion [19]. We will return to the question of how to use this wealth of historical data to create social network structures shortly.

Everything is a page. There are many other namespaces as well, as shown in Figure 15.4 including the "File" namespace, typically used for uploading images, sound files, or videos that are included in a page; the "Category" and "Template" namespaces, where contributors can create technical tools that group pages and create reusable snippets that can support everything from providing a standard set of information to be included for a category to welcoming new users quickly; and the "Wikipedia" namespace, where people create and debate Wikipedia policies. These namespaces further organize and identify the kinds of contributions people make, allowing community designers to understand and support the roles people take on [12, 13, 14]. The fact that every type of wiki page is treated just like any other wiki page with a history of each change and edit supports the study of the evolution of the community itself: how individuals take on specific roles based on the kinds of contributions they make [14] and how the group makes policy and governance decisions [11].

User accounts. Although many wikis allow anonymous access and editing in an effort to reduce the cost of contribution, most regular contributors to wikis create user accounts. The same wiki feature that records the history of edits and changes allow researchers to examine the evolution of a page or study the edit activity of a specific person. Figure 15.4 illustrates a small section of a contributor's edit log. The "User" namespace allows registered users to create a page describing themselves, their interests, their skills, and the parts of the wiki they are most involved in, which is a rich resource for understanding the characteristics of individual community members. The "User Talk" namespace allows for direct user-to-user communication, another natural resource for creating databases representing social network relationships from Wikipedia activity.

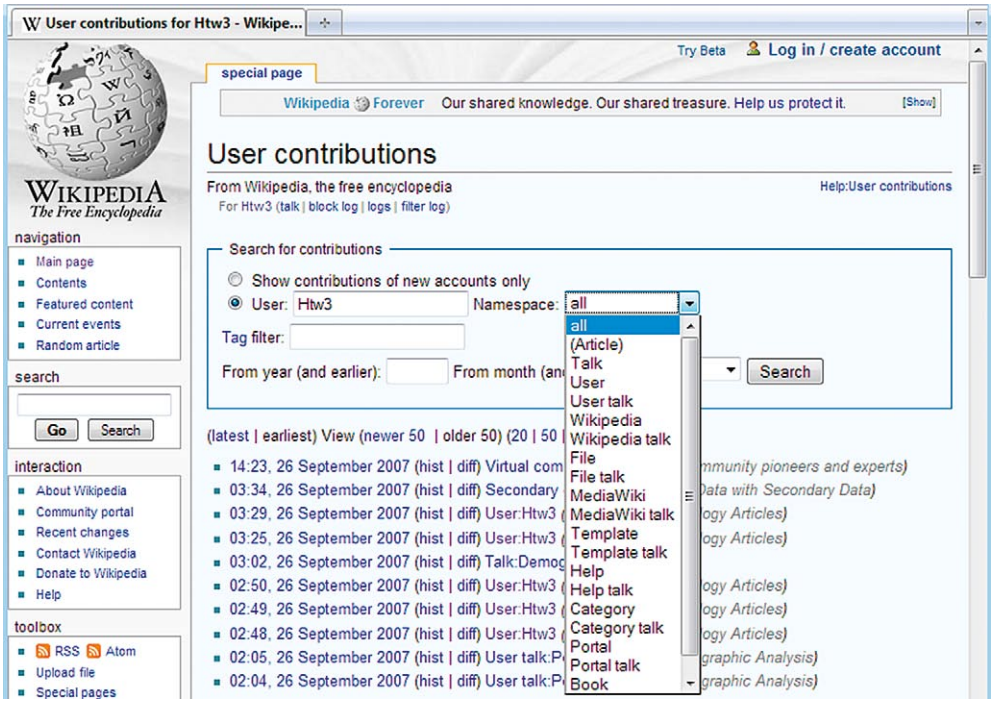


FIGURE 15.4 This page reports a partial history of edits made by a wiki user. These contribution pages are an important source of information about editors. This image also shows a drop-down menu with a range of page types or "namespaces" in Wikipedia and typical to many wikis. The tendency of editors to edit pages in certain namespaces and not others provides important clues about the roles they play in the wiki community.

15.3 WIKI NETWORKS FROM EDIT ACTIVITY

There are many interesting ways to analyze Wikipedia based on the history of activity and interaction of its users. In order to use social networking tools to analyze the community, however, we need to translate the history of editing into a set of social network relationships. This is actually a general question for any use of social network analysis to understand behavior in any kind of system. Even in domains where the social network is obvious and explicitly named "social network services" like Facebook, there are choices to be made. For instance, not all "friends" are really friends, so you may want to only consider "friends" who regularly communicate in Facebook as "real friendships." Or you might be interested in using social network analysis tools to analyze relationships between groups or fan pages or events that people attend, and so you would need to create a social network between these groups using evidence of people's behavior such as joining or posting messages to the same groups or pages.

A recent article by Carter Butts [15] raises several foundational issues related to the challenge of interpreting activity data into a network representation. In particular, networks are composed of vertices, or entities, that are connected through edges that represent the relationships between them. Both vertices and relationships can have attributes, such as the strength of a tie between vertices, or the length of time a vertex has been part of the network. The challenge for the analyst is to choose vertices, relationships, and attributes that give insight into questions that matter. Here, we discuss each of these issues in general in the context of wiki systems and give a number of possible example networks one might choose to construct.

What is a vertex? In many social network studies, vertices represent individual people or groups, companies, or institutions. In a wiki each distinct user account is a possible vertex for networks based on relationships between people using wiki systems. However, for some questions, pages, and even categories of pages might be the appropriate choice for the vertices in a social network. For example, if you want to understand connections between the kinds of topics (or products) a community is interested in, making pages or product categories the star of the show is probably the right choice. We illustrate both of these types of vertices in our examples.

What counts as an edge? If we define vertices as the user identities who contribute edits to a wiki, then we will often want to define an edge as one of many activities

that display some type of interaction between two users. Although there are many types of potential indicators of relationships between users, our first two examples illustrate two common ways to infer edges from edits. First, we use edits by one user on the user talk page of another user as evidence of a communication relationship between the two of them. That is, when editor A makes an edit to editor B's user talk page, we infer an edge from A to B because A is directly communicating to B. In the second case, we use the content of an article's talk page edits to indicate direct communication between one user and another. When editor A makes an edit that refers to a prior edit by editor B, we infer an edge from A to B. The third example uses shared edits of a content page to indicate interaction; in fact, editors who edit a page at about the same time often later go on to interact more directly [16]. Further, edges that link users to one another can be either directed or undirected. For example, if editor A posts on an article's talk page, then editor B posts a reply to A, we know that B is aware of A—but we don't know if A is aware of B. So if we wanted to measure relationships of potential influence, it might make sense to create a directed edge from B to A but not from A to B.

What attributes are important? Many attributes of edits can be used to construct variables assigned to edges or to vertices. A distinguishing feature of an author may be the particular pages that author edits, or the types of pages that author edits or abstains from editing. Pages themselves have attributes that can be coded as variables: for instance, the topics, categories, and the presence or absence of key hyperlinks can be coded as variables. The content of the edit or tendencies to make certain types of edits can be measured. For instance, some wiki users specialize in brief and simple changes like corrections of grammar or spelling. These "minor" edits may be indicative of the role that user plays in the wiki [9, 14], so it might make sense to count the number of minor edits, or the proportion of minor to major edits, that a person makes. Likewise, if you are interested in the influence of the social network on someone's adoption of a given tool, you might count the percentage of neighbors in the social network who have adopted that tool in the past, as Yuan and colleagues [4] did in their exploration of how people adopted an edit suggestion tool in Wikipedia. For these edges between people and pages, it might make sense to count the amount of activity—the number of times a page is edited by each person. Another variable that is of interest stems from the timing of an author's edits relative to the edits of other users. For instance, edits indicative of communication on talk pages are likely to be in close temporal proximity, or to exhibit a structure known as the third turn, where an edit by one editor occurs between edits of another.

15.3.1 Wiki Networks of General Interest

Three of the wiki namespaces lend themselves naturally to defining relationships between people and thus the creation of network data: edits to the user talk pages by other users are clear signs of communication; shared edits to content pages are indicative of clearly overlapping substantive interests, or at least, areas of shared attention; and discussions on article talk pages indicate both shared interest and, depending on the reply structure, direct communication. If the goal is to explore relationships between wiki articles, then co-editing behavior (the same person editing two different pages), category membership, and explicit links in the text of one page to another may be useful indicators, depending on your research question. Other kinds of entities and relationships are possible as well; for instance, one might want to focus on relationships between groups ("WikiProjects"), categories of pages, or even policies, but in general, people and pages are likely to be the main entities of interest in a social network analysis project.

Table 15.1 contains some of the possible networks that could be measured from edits in a wiki system. The networks use different types of edits and pages to infer social relationships, so they have different definitions of vertices and edges. Three of these are demonstrated by the examples in this chapter, but many other network types are possible as evident in the table. Others exist as well. For example, with additional computational effort, semantic networks that reflect the connections among content in wiki pages and technical networks that reflect the ways user's hardware interacts with Wikipedia hardware could be constructed.

15.4 IDENTIFYING DIFFERENT TYPES OF EDITORS WITHIN A WIKI PROJECT

All wikis include ways for users to contribute content and to discuss that content. Frequently, highly dedicated contributors will specialize in making certain types of contributions. For instance, some contributors may specialize in creating new content, whereas others might specialize in resolving minor disputes or sanctioning contributors who break rules or violate norms. Still others might specialize in coordinating the efforts of others. Here we provide an in-depth example that illustrates strategies for distinguishing between some types of editors and provides evidence of each of these types of editors mentioned earlier.

Our approach is to measure some details about types of pages that editors tend to edit and to collect the structure of their communication. We used a commercial web scraping utility to collect these data and focused on

TABLE 15.1 Several Primary Types of Wiki Networks That Can Be Derived from Edit Records

Network	Vertices	Edges	Weighted	Directed
Page Link Network	Pages	Hyperlinks	Yes or No	Yes
User Talk Page (i.e., Profile) Network	Users	Comments on another user's profile page (e.g., user or user talk page)	Yes	Yes
User Discussion Network	Users	Comments posted in reply to each other on an Article Discussion page	Yes	Yes
User to Page Affiliation Network	Pages and users (bimodal)	User edits per page	Yes	No
Page Co-editor Network*	Pages	Co-editors	Yes	No
User Co-edit Network*	Users	Co-edited pages	Yes	No
Category Network	Categories	Shared pages	Yes	No
Project Network	Projects	Shared pages or shared members	Yes	No

Note: Each of these potential networks could be used alone or could constitute part of a composite network.
**Created from the User-Page Affiliation Network.*

a related set of pages in a large public wiki, called the Empire Wiki. We selected this wiki and particular pages because we wanted to try to identify potential leaders involved in a social movement related to the set of pages we sampled [17]. In this study our focus is simply on illustrating strategies for distinguishing between different types of users. However, in several places, the research questions that motivated the original study helped us identify which attributes we should measure and which pages we should include in our sample.

15.4.1 Wiki Social Network Sampling Frame and Data Collection

We drew our sample from users that had made at least one edit to an article related to our focal project (Project Castle). Project Castle emerged from a loosely affiliated set of participants in an online community that organized against Internet censorship. To build our sample, we took the following steps. First we constructed a list of URLs of history pages for every article related to "Project Castle," as tagged by users. Tags are descriptive keywords that editors can add to articles that allow similar articles to be grouped together. Using tags to locate articles related to Project Castle had two advantages. First, it provided a single page with a list of all the project related articles. Second, because tags are user generated, it allowed us to target articles meeting emergent community definitions of what was related to Project Castle rather than the researcher's imposed definition. A commercial web scraping program was next used to generate an Excel spreadsheet containing a list of each user making an edit to each respective article during the sample period (about seven months). Figure 15.5 shows a typical article history page of a wiki article. Having obtained a list of the user name of each edit to each Project Castle related article, duplicate names were removed from the list using the Remove Duplicates tool found under the Data tab in Excel 2007. We defined this list as the sample of users for the study.

15.4.2 Defining Edges and Attributes in Wiki Social Networks

Because we are interested in interpersonal communication between individuals that contributed to a particular project, we felt that the user talk page network gave us the best evidence of interaction without being confounded with edits to the project pages themselves. For instance, one could also measure edges from edits to article talk pages, but this would necessarily be related to edits of the project pages themselves. Instead, edits to user talk pages would be evidence that, for whatever reason, one editor wanted to contact another editor outside the context of the specific project pages. We felt this would be evidence of an interaction likely to be meaningful to both parties. In our network, a directed edge from vertex A to vertex B represents user A making an edit on the talk page of user B.

Network analysis highlights structural aspects of interaction [18], but paying attention to the context, content, and identities of those involved in the interaction can be included in the structural analysis. Thus, in the context of this study, vertex attributes provide a way of linking observed social structure to other aspects of individual behavior.

Two types of vertex attributes were generated and associated with each vertex in the wiki network graph. The first was a set of attributes describing the structural position of each vertex, and the second was a set of attributes generated from measures of participation in the Empire Wiki community and participation in Project

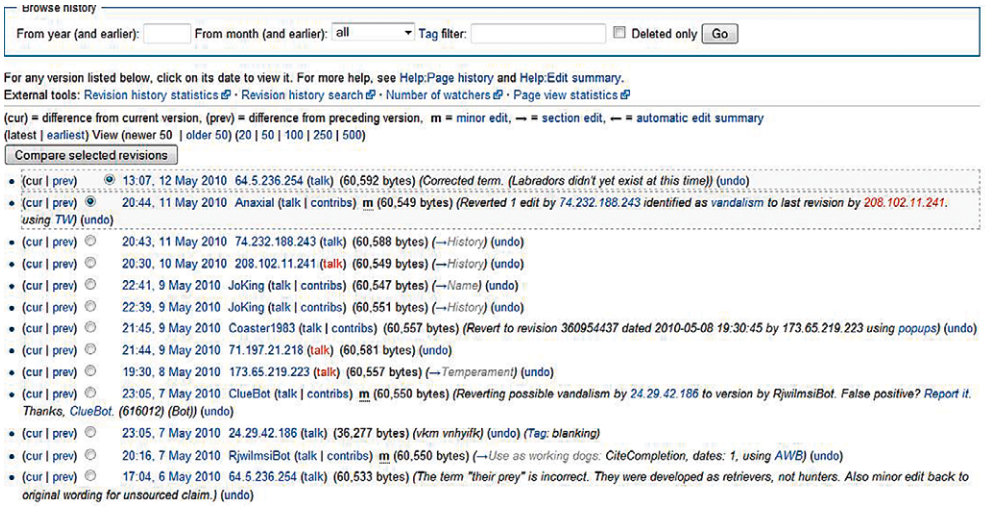


FIGURE 15.5 This study of wiki social networks used the full revision history of the Project Castle page in the Empire Wiki as both a definition of the community of interest and as a source of user IDs. We were interested in the roles played within the community of contributors to these pages. Therefore, when we scraped all of these history pages, we were sure to get all active contributors to this project. Starting from a list of URLs for Project history pages, the web scraping software returns an Excel sheet populated with all text that occurs after the edit date and prior to the (talk & Contribs) link.

Castle related articles in particular. The first set of attributes was selected in order to characterize the importance of an individual in terms of the transmission of information throughout the network and to what degree users looked to a given individual for information. We decided to use betweenness and closeness centralities to measure the importance of an individual for the flow of information. Broadly stated, closeness centrality is a measure of the relative distance of a given vertex from all other vertices in the graph. Betweenness centrality measures the number of shortest paths (geodesics) a given vertex is located on compared to the total number of geodesics in the graph [18]. In general, the higher the values of betweenness centrality measures and the lower the measure of closeness centrality scores, the more important a given vertex is for moving information throughout the network and thus managing communication among individuals within the group.

We used in-degree to measure the relative importance of a given user as a source of information. The in-degree of any vertex N is defined as the total number of edges connecting any other vertices within the graph to vertex N. In this case, in-degree represents the total number of other editors who left messages on the user talk page of an individual, and it is assumed that individuals with higher in-degrees are more important sources of information within the network. In the network itself this represents the number of unique edges directed toward vertex N. Finally, measures of identity investment with the Empire Wiki community and with Project Castle articles were collected. Investment with the Empire Wiki community was defined as how much work the user put into maintaining his or her personal user page on the Empire Wiki and was measured by the total volume of edits, in bytes, users made to their personal user pages. Involvement with Project Castle articles was defined as the number of edits a given user made to articles related to Project Castle. To account for large differences in the total number of edits made by sampled users, the proportion of Project Castle related edits was used.

15.4.3 Wiki Network Data Collection

Starting with a list of users to be used in constructing the network, we first collected the necessary wiki edge list data to allow NodeXL to build the network. As mentioned before, we define an edge as one user leaving a comment (more specifically, editing the user talk page) for another user. To obtain these data, we started with the list of sampled users in Excel. We then used the Web Scraper to go through the history page of each sampled user and build an Excel spreadsheet with the name of the user whose page was being scraped, the name of each user making an edit, and the time stamp for each edit. We imported these data to NodeXL and prepared the edge list for manipulation by using the "Merge Duplicate Edges" function in the Data menu.

Next, we collected data to construct measures of each user's level of involvement with editing the Empire Wiki and Project Castle articles. We used the Webscraper to get the total number of Project Castle related edits

of each user by building a spreadsheet with all of the article edits made by each user and then using filters to separate out the Project Castle related edits from the other edits. Then we used pivot tables to get the number of Project Castle edits and divided each user's total Project Castle edits by his or her total edits to get the proportion of Project Castle edits made by the user. Finally we got the total volume of edits each user made to his or her personal user pages. These values were then copied into the NodeXL template under the "Add your own columns here" column on the worksheet labeled "Vertices." For the purposes of this example, we have included a smaller portion of the original data that best demonstrates the interesting aspects of the overall sample. This subset contains 26 sampled users, 503 total vertices, and 803 edges.

15.5 NODEXL VISUALIZATION STRATEGIES FOR REVEALING DISTINCT USER TYPES

In visualizing these data, we want to illustrate how NodeXL can be used to analyze larger chunks of network data from wiki sites. To learn more about the overall network structure of our group of wiki editors, we will look for unique network structures related to different roles editors perform.

To accomplish these goals, we need to (1) construct a graph of the overall network, (2) visually represent different vertex attributes, and (3) search for structural similarities among individuals exhibiting similar behaviors or occupying similar roles. All three of these tasks can be accomplished with relative ease using NodeXL. We'll go through the basic process of how you can do this.

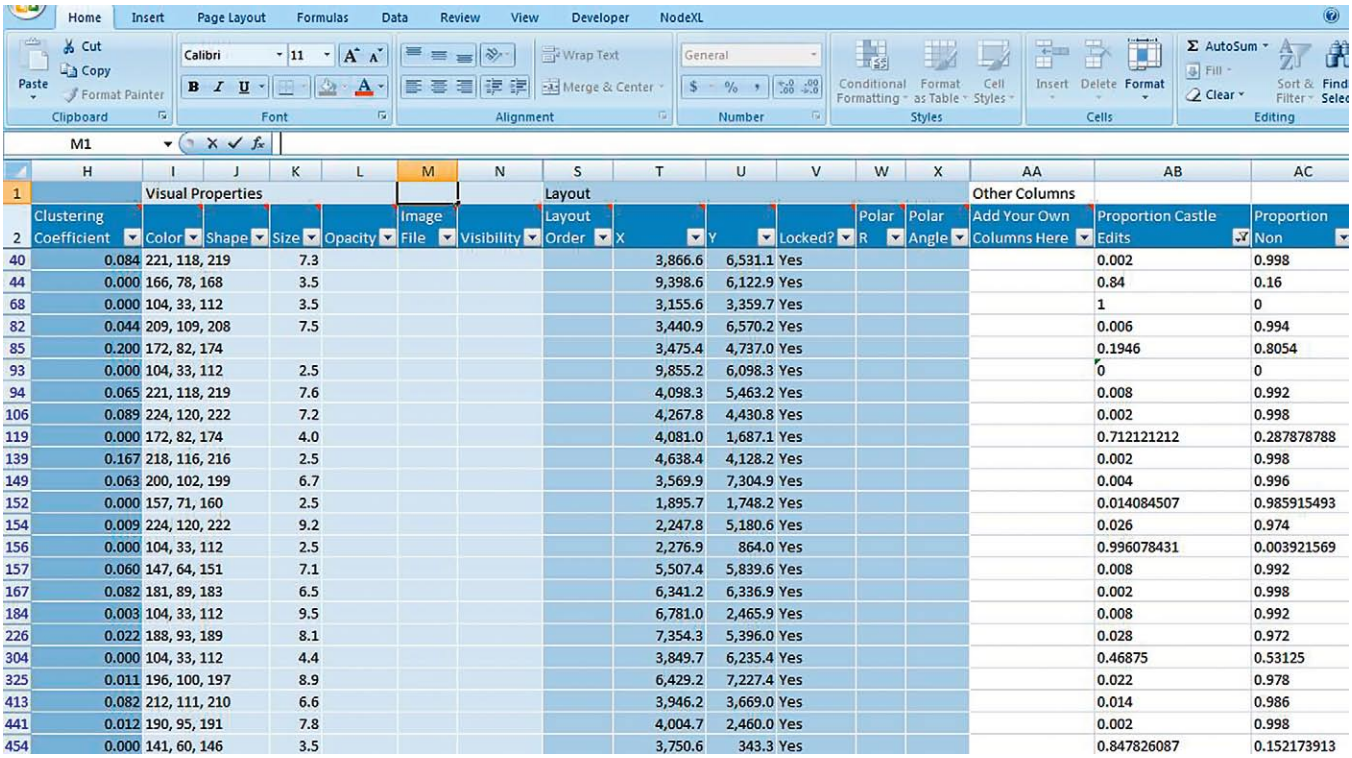
Obtaining a graph of the entire network is straightforward in NodeXL. Because we are concerned with the direction of connections, we select "Directed" from the Type drop-down menu. We then select the Show Graph button in the NodeXL tab. Select "Fruchterman-Reingold" from the Layout drop-down menu, a common layout for graphs that provides a good initial representation of the structure of the network. We can tweak various visual settings of the graph, such as the color of the background or shape of the vertices using the "Visual Properties" section of the NodeXL tab as well as trying out other layouts that may highlight structures in the network most clearly. To make it easier to see the features of the graph, we changed the background to a light shade of gray, increased the size of all the vertices to 2.0, and reduced the opacity of the edges to 40 percent, all from the Options tab on the graph window.

Our next task is to visually represent vertex attributes of interest. We can use the color, shape, and size of each vertex to represent multiple vertex attributes simultaneously. This kind of visualization is commonly used in network analysis as a tool for locating interesting structural features in the network graph or finding interesting patterns. In our example we'll use these visualizations to help us find structural similarities among users that may be performing key roles in the wiki community. To quickly and easily visualize vertex attributes, we can use the Autofill Columns tool under the NodeXL tab. This brings up a menu in which we can select different aspects of the graph, such as vertex color or edge opacity, and have NodeXL automatically assign values to each respective vertex or edge based on other attributes of each vertex or edge, such as in degree or proportion of Project Castle edits.

To use the vertex attributes not calculated by NodeXL, we need to create new columns to the right of the "Add your own Columns Here" header. NodeXL will automatically format these new columns for use in the analysis of your network. After copying the values for each vertex and pasting them into the Vertex worksheet, we can easily use these data to alter the display attributes of each vertex with the Autofill Columns tool. Figure 15.6 shows the newly created columns in the NodeXL workbook.

15.5.1 Making Top Wiki Editors Stand Out by Visually Formatting the Network Graph

With data in place, we are now ready to begin constructing visual representations of the wiki network with the attributes of each editor displayed via color, size, and shape. A useful way to start is to change the color of each vertex based on one vertex attribute at a time and then examine the network graph for any observed patterns. As an example, we describe the process of changing the color of vertices based on the variable "Proportion Castle Edits." The Autofill Columns button in the Visual Properties NodeXL menu opens a dialog box in which you can select "Proportion Castle Edits" in the "Vertex Color" menu. Then click the arrow menu next to this in order to fine-tune the settings. Leaving the settings in this menu in the default settings will have NodeXL assign colors to each vertex starting with solid red for the smallest value in the column running to solid green for the largest value in the column. This works well for our purposes, but using one color range makes the results much easier to see and interpret, so we will vary the color from a dark shade of green for the smallest values to a light shade of green for the highest values. For now, leave the options "Ignore outliers" and "Use logarithmic distribution" unchecked. These can be used to get a good range



	H	I	J	K	L	M	N	S	T	U	V	W	X	AA	AB	AC
1		Visual Properties						Layout						Other Columns		
2	Clustering Coefficient	Color	Shape	Size	Opacity	Image File	Visibility	Layout Order	X	Y	Locked?	Polar R	Polar Angle	Add Your Own Columns Here	Proportion Castle Edits	Proportion Non
40	0.084	221, 118, 219		7.3					3,866.6	6,531.1	Yes				0.002	0.998
44	0.000	166, 78, 168		3.5					9,398.6	6,122.9	Yes				0.84	0.16
68	0.000	104, 33, 112		3.5					3,155.6	3,359.7	Yes				1	0
82	0.044	209, 109, 208		7.5					3,440.9	6,570.2	Yes				0.006	0.994
85	0.200	172, 82, 174							3,475.4	4,737.0	Yes				0.1946	0.8054
93	0.000	104, 33, 112		2.5					9,855.2	6,098.3	Yes				0	0
94	0.065	221, 118, 219		7.6					4,098.3	5,463.2	Yes				0.008	0.992
106	0.089	224, 120, 222		7.2					4,267.8	4,430.8	Yes				0.002	0.998
119	0.000	172, 82, 174		4.0					4,081.0	1,687.1	Yes				0.712121212	0.287878788
139	0.167	218, 116, 216		2.5					4,638.4	4,128.2	Yes				0.002	0.998
149	0.063	200, 102, 199		6.7					3,569.9	7,304.9	Yes				0.004	0.996
152	0.000	157, 71, 160		2.5					1,895.7	1,748.2	Yes				0.014084507	0.985915493
154	0.009	224, 120, 222		9.2					2,247.8	5,180.6	Yes				0.026	0.974
156	0.000	104, 33, 112		2.5					2,276.9	864.0	Yes				0.996078431	0.003921569
157	0.060	147, 64, 151		7.1					5,507.4	5,839.6	Yes				0.008	0.992
167	0.082	181, 89, 183		6.5					6,341.2	6,336.9	Yes				0.002	0.998
184	0.003	104, 33, 112		9.5					6,781.0	2,465.9	Yes				0.008	0.992
226	0.022	188, 93, 189		8.1					7,354.3	5,396.0	Yes				0.028	0.972
304	0.000	104, 33, 112		4.4					3,849.7	6,235.4	Yes				0.46875	0.53125
325	0.011	196, 100, 197		8.9					6,429.2	7,227.4	Yes				0.022	0.978
413	0.082	212, 111, 210		6.6					3,946.2	3,669.0	Yes				0.014	0.986
441	0.012	190, 95, 191		7.8					4,004.7	2,460.0	Yes				0.002	0.998
454	0.000	141, 60, 146		3.5					3,750.6	343.3	Yes				0.847826087	0.152173913

FIGURE 15.6 NodeXL uses spreadsheet columns to store attributes of each vertex and can be transformed using standard Excel formulas. In this case, we see a sample of some Empire Wiki editors' overall activity and the proportion of pages that they edited that were related to Project Castle.

of color even when you have widely different values in the variable you're using. By playing with these different visual options, we can compare and contrast different vertex attributes. For instance, Figure 15.7 shows the tool being used to represent betweenness centrality with vertex color, user page activity with vertex shape, and Proportion Project Castle Edits with vertex size. To learn all the uses of this powerful tool, it is helpful to play around with it, changing values to see how you can visually emphasize different aspects of the network graph structure. All of the visualizations that we will be discussing were created using this basic procedure.

To accomplish our third task it would be useful to be able to look at the structural aspects of the network in the "immediate vicinity" of each of our sampled users. NodeXL has a powerful and simple tool for generating a number of graphs of each individual in the network and others connected to that individual. This is called an ego network graph and it is useful for many network analysis tasks. To create these ego graphs, we simply click on "Subgraph Images" in the "Analysis" NodeXL menu. In the Subgraph Images dialog there are several options for what vertices to include in each subgraph, how to display the subgraphs, and where to save them. For your analysis, select 1.5 adjacent vertices. This will produce a subgraph of a given node, including each direct connection to other vertices, and the connections amongs those other vertices. This can be thought of as your friends and their connections to one another, but not including your friend's friends who are not friends with you. This ego network graph is extremely useful in network analysis of wikis as it provides us with a means of examining the structure of those directly connected to each user. The other options in this dialog allow you to change the size of the generated subgraphs and select whether you want thumbnail images of each subgraph inserted into the NodeXL worksheet. After saving these subgraph images, we now have the visualizations necessary to provide data to answer our research questions.

15.5.2 Interpreting Wiki Network Graphs for Evidence of Distinctive Social Roles

When analyzing visual network data, it is often useful to begin with an inspection of the entire graph, searching for interesting or unique structural forms. Figure 15.8 is the network generated from the sample wiki data. Upon examination of this graph, we see an outer ring of users and a more densely connected inner core. What is

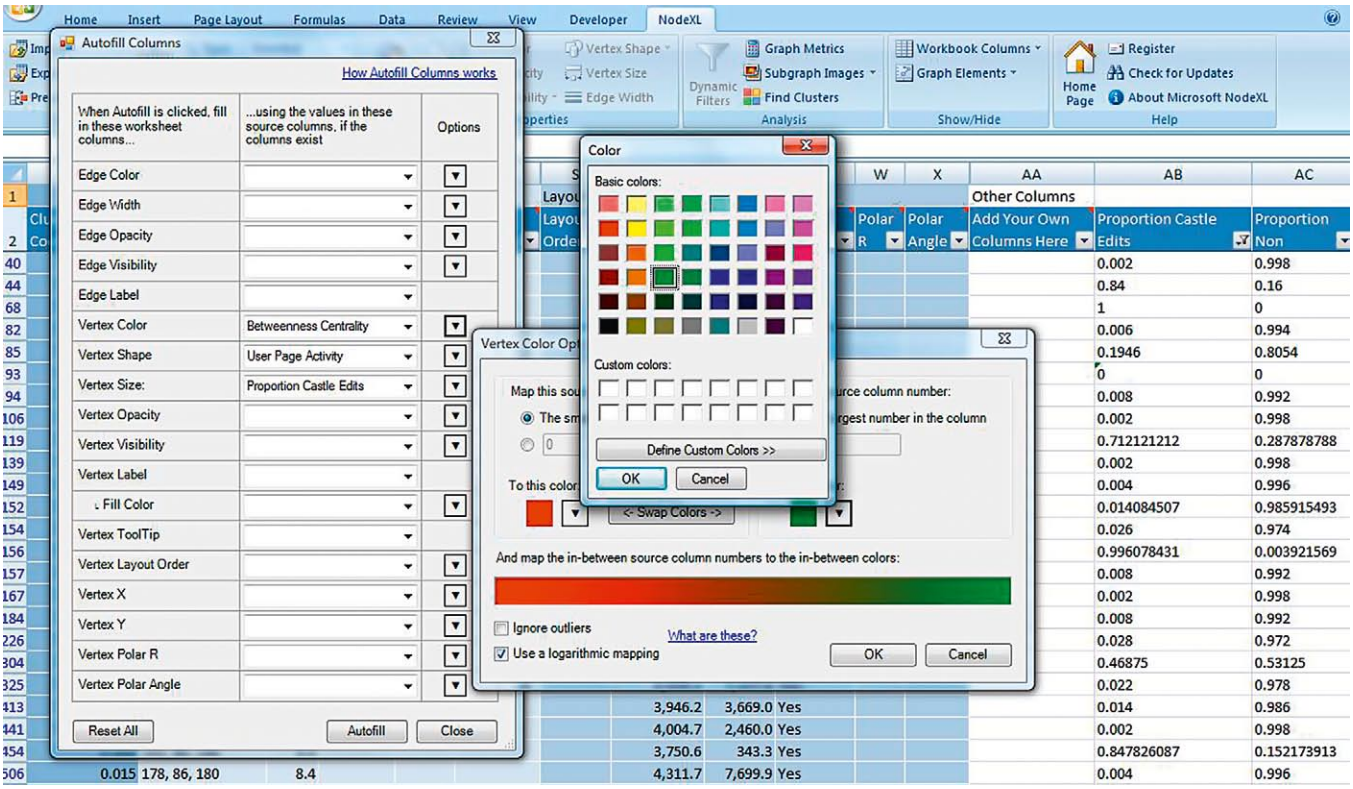


FIGURE 15.7 NodeXL allows you to assign gradients of vertex colors that correspond with data attributes in the spreadsheet. This helps make the resulting graph easier to read and analyze and highlights key features of interest.

particularly interesting is that there appear to be a handful of vertices that connect this outer ring to the inner core. We can infer that without these individuals the inner core would become relatively isolated from the outer ring. Still we need more information to determine whether this structural similarity is meaningful for differing roles or behaviors. This is where our other visualizations become useful.

Figure 15.9 displays two visualizations. On the left, color represents the proportion of Project Castle edits. Dark green vertices have the lowest proportion of Project Castle related edits and light green vertices have the highest proportion of Project Castle related edits. On the right, vertices colored dark blue have the lowest volume of edits to their respective user page, whereas light blue vertices have the highest volume. The vertices that are black are the people that we didn't sample and don't have complete data on. We left them black so that our sample would stand out more. What is interesting is that all of the vertices we identified as being key for connecting the outer ring to the inner core, and most vertices in the inner core, have a very low proportion of Project Castle edits and have some of the larger volumes of edits to user pages. In comparison, the vertices representing individuals with larger proportions of Project Castle related edits appear to be more isolated

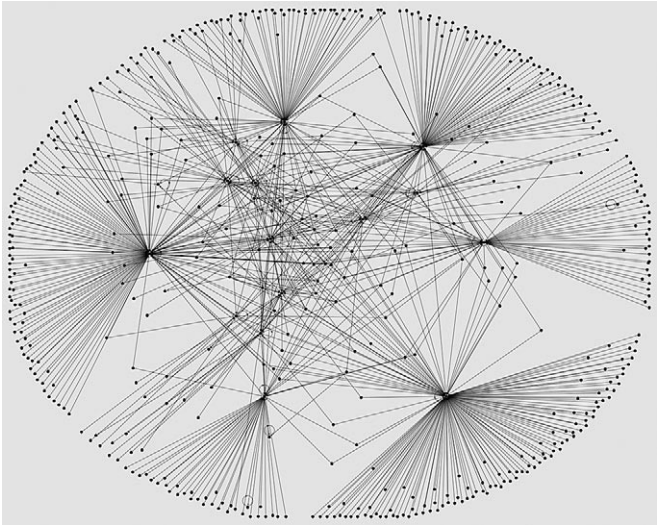


FIGURE 15.8 This NodeXL wiki network graph shows a well-defined outer ring of users and a strong inner core. Only a handful of vertices connect the outer ring to the inner core. Without these nodes, the population would be highly fragmented.

in the graph and less invested in the overall Empire Wiki community. In contrast, the users who appear to be vital in linking different parts of the graph appear to be well invested in the Empire WIki community but do

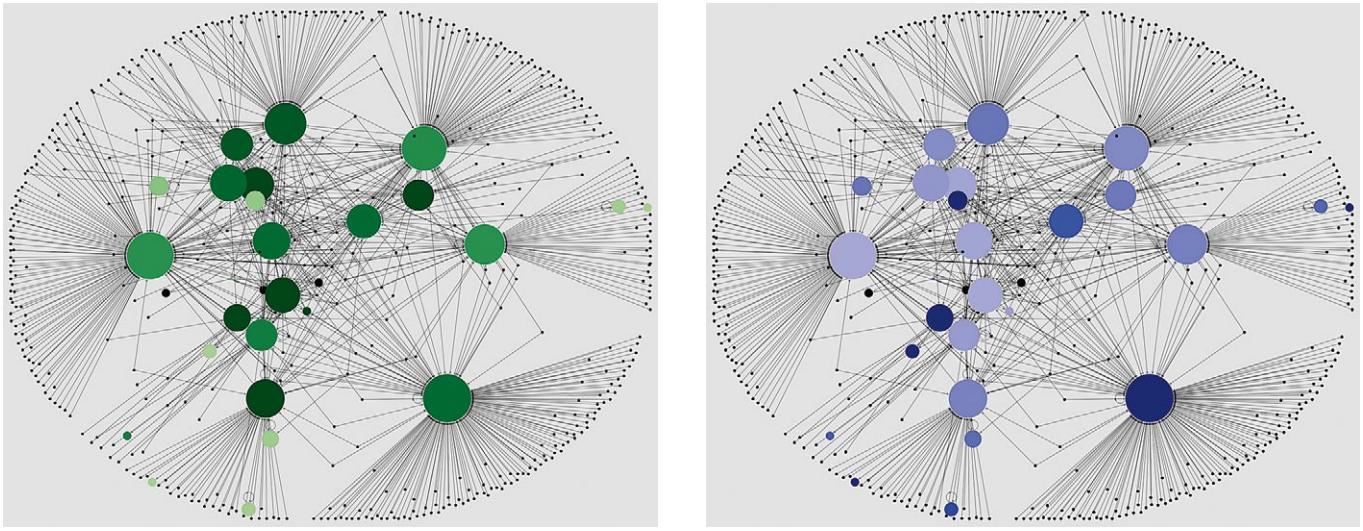


FIGURE 15.9 The NodeXL wiki network on the left displays the relative proportions of Project Castle edits among users sampled. Dark green indicates the lowest proportion of edits, and light green is the highest. The figure on the right displays the volume of edits to the users' respective user pages. Dark blue indicates the lowest edit volume, and light blue represents the highest edit volume. Users who connect the outer ring to the inner core in the previous visualization have few Project Castle edits, and those users who display a high volume of edits are relatively isolated in the previous visualization. This indicates that Project Castle is not strongly connected to the larger Empire Wiki community.

not appear to have been very active with Project Castle. Those involved more heavily with Project Castle are less well integrated into the Empire Wiki community and work on Project Castle related wiki pages largely autonomously from other content on the site. Network analysis is a great way for someone studying an online community or in charge of managing an online community to get at how different groups within the community interact among themselves and with other groups.

15.5.3 Using Subgraph Images to Distinguish between User Types

To further explore the structural aspects of the wiki network and to look for connections between structure and observed behavior, you can inspect the subgraph images. Examining the subgraph images shows four different distinct types, which are displayed in Figure 15.10. The first type is the type 1 sysop. Sysops are those in charge of maintaining the Empire WIki, enforcing the rules of the wiki, answering user questions, and managing disputes between users. It turns out that users with this type of ego graph are all sysops at the Empire Wiki. These are the vertices that connected the outer ring of vertices to the inner core of the graph. This inner core contained the majority of the second type of ego network graph observed—the type 2 sysop. Unlike the type 1 sysops, which have a large number of ties to users with no ties amongst themselves, the type 2 sysops in contrast have a smaller number of overall ties, but the users to which they are connected share a large number of connections amongst themselves. Most of the type 2 sysops have senior positions, whereas type 1 sysops are newer administrators with lower levels of authority. The correspondence between structure and these differing levels of authority is intriguing. It indicates that the lower authority sysops are the ones primarily responsible for day-to-day administrative duties and interacting with other users. They can be thought of as the public face of the sysops. The type 2 sysops in contrast form a tightly connected inner core but deal much less with users, being embedded within dense networks of other sysops. The type 1 sysops provide a key link between type 2 sysops and the broader community.

The third type of ego network graph observed was the low-level Project Castle contributor, which we will call the inactive participant for the sake of simplicity. These users were largely uninvolved with Project Castle but tended to be fairly well embedded within the overall network. This is in contrast to the fourth type, the high-level Project Castle contributor or active participant. The active participant tends to have a very low number of ties and in general is poorly integrated within the overall network. This supports our conjecture from visualizations of vertex attributes. In general our analysis of visual data tells us that the users most actively involved with Project Castle are not well integrated within the Empire Wiki community and tend to work largely autonomously on constructing articles related to Project Castle. In addition, unique structural similarities were found among the administrators of the Empire WIki community. What is particularly intriguing is that these distinct structural

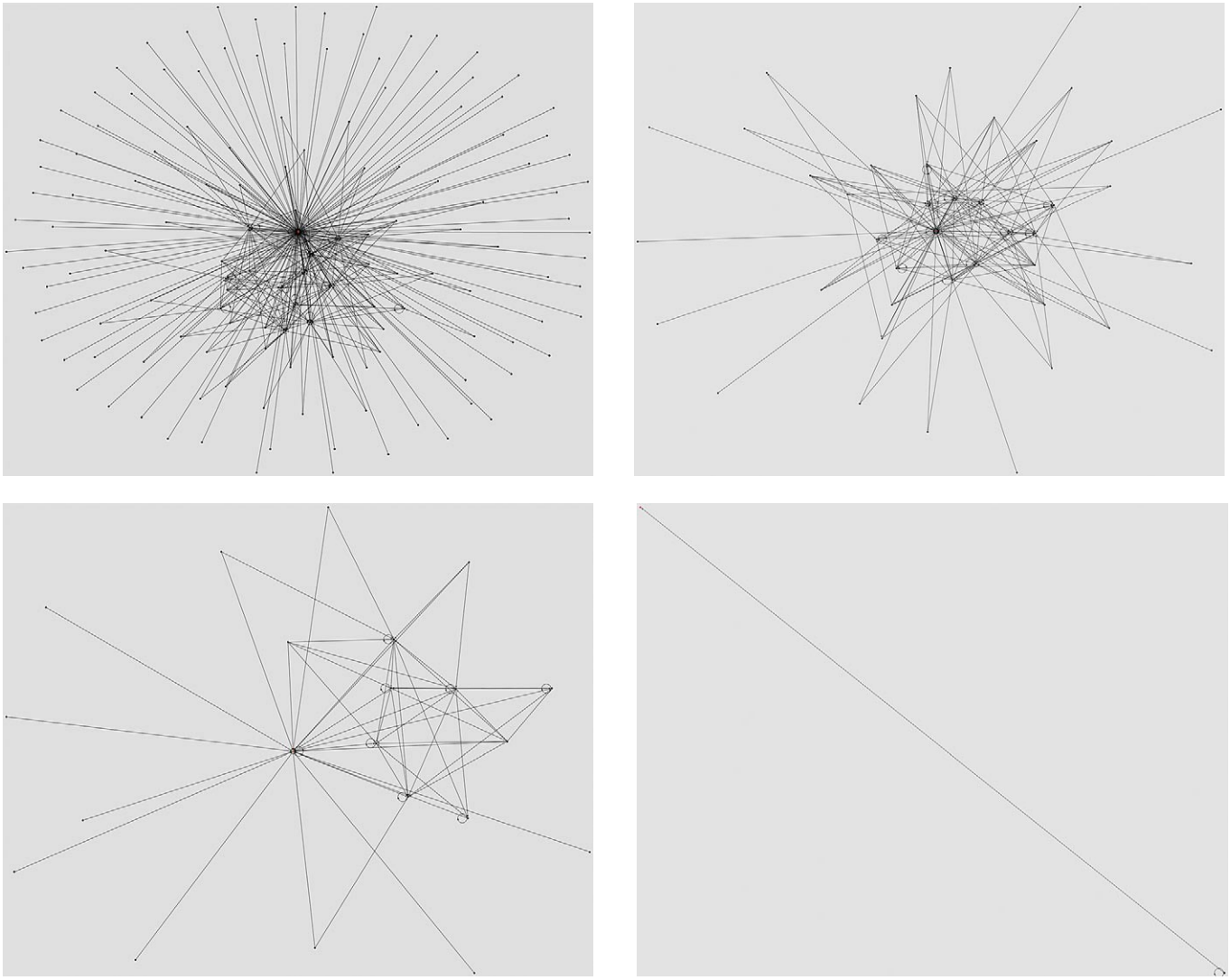


FIGURE 15.10 This figure compares the degree 1.5 ego network graphs of four different exemplary types of Project Castle contributors. Ego network graphs with automated layouts are good ways to identify potential structural signatures of online roles. In this instance, we see evidence that system administrators tend to have more connection to others involved in the project than do the actual substantive experts. Interestingly, for both sysops and substantive contributors, the higher-level contributors tend to have fewer connections.

aspects allow us to locate sysops based on observation of network structure. The structural roles played by sysops are revealed to be highly analogous to their roles within the community. Type 2 sysops tend to be tightly embedded in a small but dense core of other sysops. Type 1 sysops can be seen as the "public face" of the administrators and they are vital for connecting the type 1 sysops to the broader Empire WIki community.

At this point you may be asking, "So what, why is it useful to know that the ego graph of a highly active Project Castle contributor looks different from the ego graph of a low-level Project Castle contributor?" By finding these distinct patterns in the ego network graphs, we can start to find ways to use network structure to identify different types of users within potentially large populations of contributors. We can build a network representing an online community and quickly locate different kinds of users and get a sense of where they fit in within the larger community of users and how they connect to one another. Once we can reliably associate certain types of users with certain types of network structures, we can even build automated methods of seeking out specific types of users or behaviors. As we'll see in the next example, if we can use these techniques to identify behaviors we want to reward or sanction, we have a powerful tool to help with managing online communities. At the same time, it is of great help to those seeking to foster a new online

community by pointing toward the types of structures that are most conducive to desired roles and behaviors.

15.5.4 Seeing the Trees and Forest with Wiki Network Analysis

As we've seen, network analysis is a great tool for examining large communities that form inside wiki sites. The organization of wiki sites such as the Empire Wiki makes it possible to handle large volumes of users and the related connections among those users. This dramatically reduces the time and financial resources necessary for larger-scale observation of communities. What is particularly exciting about this methodology is that it provides a means of establishing the contours of social structure even in groups that have no formal organization and provides a means of locating different behaviors or roles based on that structure. The unique visual aspects of network analysis let us consider many aspects of the entities we're studying in ways that are not possible with other techniques. In general, if you need to get the "big picture" of how an online community is connected, it's hard to beat network analysis. Next we'll look at an example of using network analysis on a smaller scale with rich data, and we'll see how network analysis can be used to draw out certain kinds of behaviors we might be interested in exploring.

15.6 IDENTIFYING HIGH-QUALITY CONTRIBUTORS IN ARTICLE TALK PAGES

You don't need special tools to study how people interact in wiki systems. All you need is the willingness to spend some time coding interaction on the "talk pages" and some basic content analysis methods. This example illustrates research that combines content analysis with visualization of interaction in the context of article talk pages. Studies like this offer great ways to combine qualitative analysis with social network visualization and present viable strategies for professionals and researchers interested in teamwork, small groups, and interpersonal interaction.

The following example illustrates several NodeXL techniques and discusses best practices in the collection of data and interpretation of findings. This example explores how people interact in wiki talk pages, documents both the structure of interaction, and assesses the quality of their discussion. In particular, this example illustrates how vertex attributes, interaction structure, and strength of relationships can reveal differences between high-value contributors and others who may be counterproductive.

In this case, we want to know something about how different people communicate with each other in the process of constructing the content on the wiki. There are challenges to writing content collaboratively. For example, it can be frustrating when a contributor makes an effort to add to the content only to have it changed by someone else without explanation. Such challenges can discourage people from participating or even lead to edit wars or other unproductive uses of time [7]. For wiki systems to be effective, leaders should understand how to encourage and foster high-quality discussions. In this study we ask, "Do people who are more or less deliberative in their communication tend to take different network positions in the conversation?" Understanding how well people have thought through and discussed their content edits can be important. If a wiki system encourages people to discuss edits before or during their collaborative writing, then the quality of the discussion could influence, or even determine, the quality of the content being created. Web administrators will want to encourage productive discussion to help avoid edit wars or other dynamics that can squelch widespread participation in the wiki.

In earlier research, we applied a coding scheme to a sample of discussions on a particular content talk page [19]. The coding scheme measured deliberation quality by assessing two things. First, it measured how well people analyzed the issue by using good evidence, weighing pros and cons, and articulating their criteria for an effective decision. Second, it measured the more social side of group discussion such as respect and consideration of other people's statements. For our purposes, we just need to know that the data that results include vertices that are interacting and those vertices differ in terms of some attribute, deliberation.

How should you determine the sampling frame for collecting data from a wiki? Your sampling frame describes the set of edits that you chose to include in your sample and from which your units of analysis will be assembled. You will want to define a disciplined rational for the inclusion of edits in your study. We identified an important Wikipedia policy and included all edits to the article talk page that occurred during the early stages of the discussion of this policy. The data included with this chapter includes deliberation scores from the three largest conversations in that data set (for simplicity's sake). However, because even internal wikis often grow quite large, it is always best to start your data collection with clear, theoretically based justifications for inclusion of data. Too large a data set may pose problems in terms of the time and resources needed to manipulate and analyze the network graph.

Wiki talk pages do not automatically create an explicit reply structure. However, users tend to make edits that follow patterns similar to those in online

forums. We inferred an edge existed whenever a user made an edit that either (1) directly addressed an earlier editor by name or (2) referred to or reacted directly to content introduced by an earlier editor. When an edit introduced a new idea or was not clearly related to a previous post, we coded that message as directed to "All Editors." Note also that any message that clearly replied to multiple parties was coded as an edge for each recipient.

Many online discussion forums suffer from flame wars or other types of hostile and counterproductive conversation. Although many conversations are more neutral, other conversations and, indeed, particular users will make contributions that are actively demonstrating features associated with principles of deliberation [20]. For instance, some contributors will actively restate the ideas of another or will recognize the validity or importance of another user's contribution. There are many dimensions of deliberative talk, and we used a content analysis scheme to code all messages from our sample frame according to these attributes, and then we calculated the average rate of deliberation for each contributor in our sampling frame.

15.6.1 Tasks and Strategies for Identifying Types of Contributors by Visualizing Article Discussion Page Networks

In this example, our data includes a list of vertices representing all users who participated in the three sampled threads. It includes all of the edges that we observed between that set of users, weighted by the number of directed replies from each ego to each alter. Our visualization goal in this example is to look for evidence of how the structure of interaction and the intensity of ties are both related to the deliberative or confrontational tendencies of the users. This is an exploratory inquiry, and therefore our primary goal is to identify patterns that we could potentially investigate later with more data and across other contexts. With these goals in mind, we have three visualization tasks: (1) weight edges according to number of replies, (2) size vertices according to amount of participation, (3) color the vertices based on deliberation score.

At the bare minimum, we need to transform the edge weights so that they all fall between 1 and 10, as this is valid range for line width inputs. However, we also want to find a range that highlights meaningful differences without distracting the viewer. Our solution is to write an equation that both draws in our high values and imposes a floor value. In practice, we chose to simply take the square root of the number of messages and then add a constant. We employ a similar strategy for vertex size, which we made a function of out-degree with the intent to impose only modest factors of scale differences between our largest and smallest vertices. Our "All Editors" vertex raises an important point: because we defined vertex size based on out-degree and "All Editors" cannot reply to anyone, then we want to make sure we add a floor so that our smallest vertex has sufficient size to be seen.

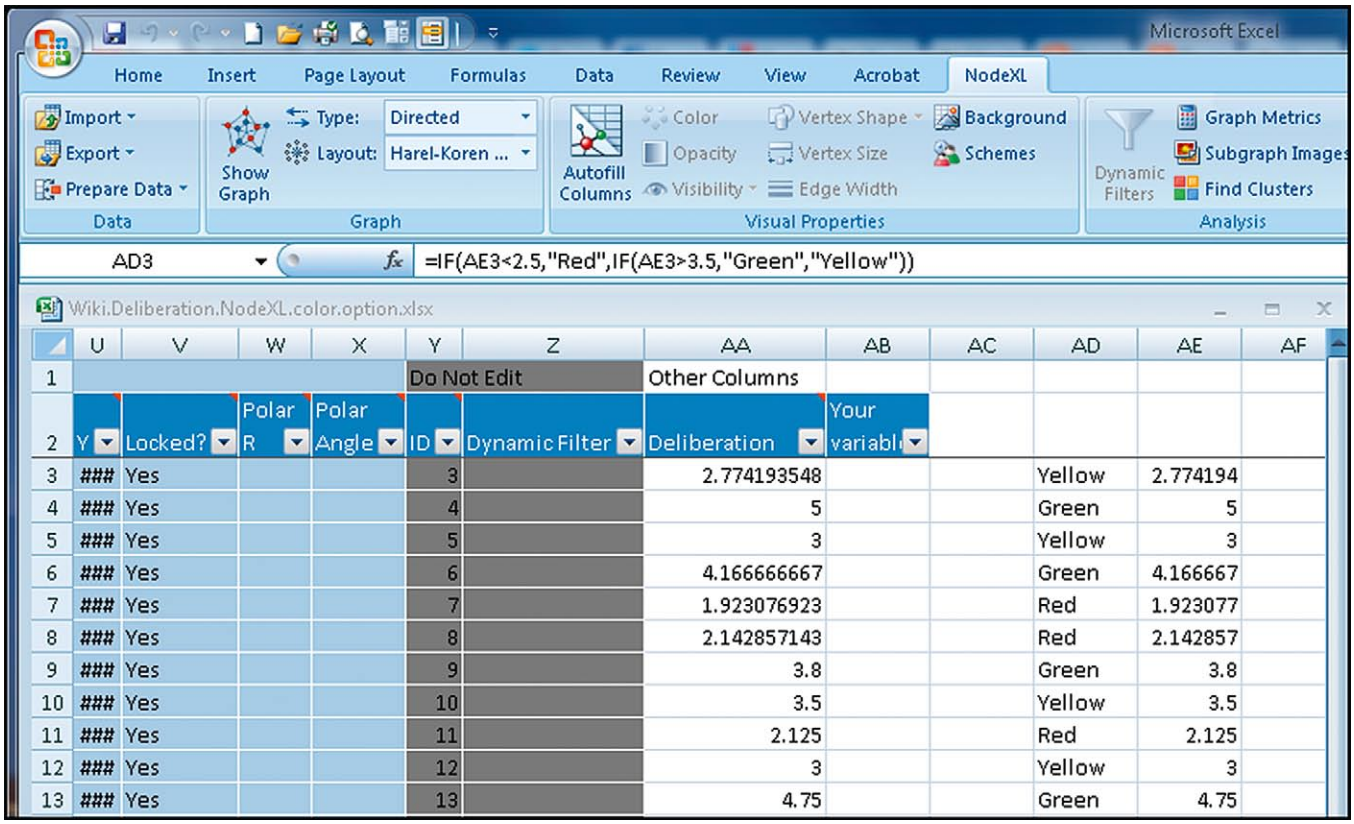
We want to assign color to the vertices based on how deliberative the editors are. Two basic strategies are available. First, like many examples in other chapters, we could use the NodeXL Autofill Columns feature found in the Visual Properties menu to assign a range from high to low. Alternatively, Figure 15.11 illustrates how we can use an "if-statement" to define a few meaningful colors to ranges of variation in deliberation scores. Both strategies have merit, and it is worth considering each depending on your research goals. In this instance, we like the idea of using categorical color definitions that relate to three different orientations toward deliberation: actively deliberative (green), actively confrontational (red), and a third option that is either neutral or closely balanced between the two poles (yellow).

Relying on the metaphor of a stoplight, our three colors represent users whose behavior we would like to encourage (greenlight), users whose behavior we might want to stop (redlight), and users whose behavior is not just in between but that we might want to shift to behave more in the green zone. A second motivation to use categorical color assignment is based on theoretical grounds: we are less interested in differences in degree of deliberation or confrontation than we are in distinguishing between those who are exerting positive or negative effects on the quality of the conversation.

Our third motivation is methodological. Because we know that there is some error inherent in our measurement of deliberation, we want, in general, to draw attention to large differences rather than small differences in deliberation. Therefore, if we simply contrast the reds and the greens, we know we are dealing with differences of at least a substantial amount (determined by the range that we define via our cut points in the if-statement). Finally, please note that we constructed our if-statement using columns to the right and outside of the NodeXL table and then simply pointed the vertex color column at those cells. Because of formatting necessary for NodeXL, the if-statement does not function when using a NodeXL column as a source of values for an if-statement.

15.6.2 Searching for Structural Signatures of Confrontation and Deliberation in Wiki Article Talk Page Networks

Figure 15.12 is the result of several small adjustments for readability and clarity. The image represents



	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF
1					Do Not Edit		Other Columns					
2	Y	Locked?	Polar R	Polar Angle	ID	Dynamic Filter	Deliberation	Your variabl				
3	###	Yes			3		2.774193548			Yellow	2.774194	
4	###	Yes			4		5			Green	5	
5	###	Yes			5		3			Yellow	3	
6	###	Yes			6		4.166666667			Green	4.166667	
7	###	Yes			7		1.923076923			Red	1.923077	
8	###	Yes			8		2.142857143			Red	2.142857	
9	###	Yes			9		3.8			Green	3.8	
10	###	Yes			10		3.5			Yellow	3.5	
11	###	Yes			11		2.125			Red	2.125	
12	###	Yes			12		3			Yellow	3	
13	###	Yes			13		4.75			Green	4.75	

FIGURE 15.11 NodeXL can make use of the full range of Excel 2007 features, for example, using an "if-statement" to assign vertex color according to a categorical definition of low, medium, and high. A categorical assignment like this one is used to highlight large differences in the measured attribute. In this case, we can concentrate on the difference between contributors who are actively improving the quality of the discussion (green) from those who are actively undermining it (red).

the structure of interaction among participants in the three coded threads in the data set. The graph has several notable features relating to the use of NodeXL and in terms of insights into how deliberation relates to network structure. First, the layout results from an initial automatic (Harel-Koren) layout but with several node locations adjusted by hand in order to increase legibility and reduce unnecessary confusion. After the initial layout, we simply set the "fix vertex location" cells to yes and then adjusted the location of the vertices as we saw fit. Second, we adjusted label locations to avoid overlap with edges.

We can make several observations about how the structure of interaction is related to patterns of deliberation in this example. First, the fact that we drew our data from three prominent threads is apparent in the triclustered structure of the network: with the exception of one edge (between *Sedona* and *Michelle*), the only connection between the three threads are the messages that we coded as addressed to "All Editors." We draw three observations related to deliberation. First, many of the larger vertices are colored red, and thus are contributors who were coded as confrontational. For instance, *Pat*, *Elke*, and *Michelle* are all large vertices and are coded as confrontational. Similarly, all of the larger vertices are either red or yellow, indicating that the most deliberative contributors tend not to be talking to as many alters as less deliberative contributors. Second, the most intense edges all involve the confrontational users. This is apparent in each of the three conversations, where the thickest and most intense dyadic relations all involve the large red vertex from each conversation. Third, we note that the deliberative contributors are not necessarily isolated from the confrontational users. In fact, the reason several of them may be behaving in an especially deliberative manner could be to find a middle ground that addresses issues raised by the confrontational authors. Some like *Brynn* and *Anna* may be playing this hostility-diffusing role, whereas others like *Ram* and *Natasha* may simply be involved in a polite and deliberative conversation.

Our inspection of Figure 15.12 suggests a couple of important lessons for using NodeXL to assess the quality of users' contributions in article discussion pages. Editors who participated in high-intensity exchanges and who exhibit relatively high out-degree seem more

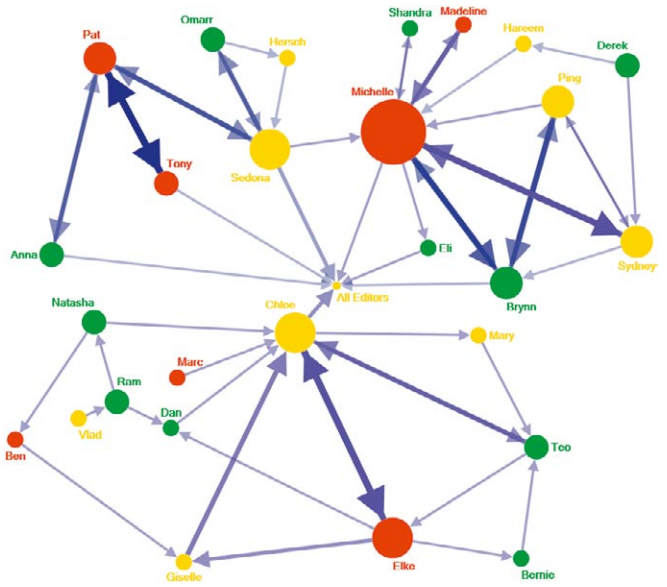


FIGURE 15.12 This NodeXL network graph depicts user-to-user talk page connections from a Wikipedia policy article. The graph illustrates one way that styles of contribution are tied to structural attributes. Note that the red nodes (most confrontational) are involved in the strongest dyadic ties, and they tend to have the highest out-degree. In contrast, the most deliberative contributors tend to have fewer partners and do not necessarily involve themselves in intense dyadic interactions. Observations like these can provide direction for further research that statistically tests the strength of these observer relations. Ultimately, if those measures are robust predictors, they could be used in automated systems for identifying more or less collaborative contributors, assessing community health, and deciding where interventions or support might be most helpful.

likely to be confrontational. Moreover, none of the highly deliberative contributors shared both of these traits. Insights like these can form the basis of a simple set of structural signatures for automatically predicting the quality of editors' contributions, flagging editors for further attention, and assessing the quality of thread conversations without having to read the content. Of course, initial observations would require statistical testing and refinement by comparing results to the actual content of the conversations. However, both practitioners and researchers would benefit from treating their inquiry as an ongoing process of refining definitions of structural signatures and building systems that leverage diagnostics based on those insights [25]. The NodeXL system is well suited to this iterative type of task because of its simplicity, ease of data entry, and clear integration with other Excel 2007 features. This last point reinforces a general lesson that we suggest for all users of NodeXL, which is to take advantage of the fact that the network data can be linked to other standard data analysis and visualizations in Excel.

15.7 NAVIGATING LOSTPEDIA: USING NODEXL TO REVEAL THE LARGE-SCALE COLLABORATIVE STRUCTURE OF WIKI SYSTEMS

Community moderators must make sense of the mass of data that can be generated by active communities. Some of the most active and large communities rely on wiki software as evidenced by Wikipedia. This section illustrates techniques for mapping large-scale collaborative structures in a popular, niche wiki project. Lostpedia is a community of more than 1.7 million fans of the ABC television series *Lost*, a popular show with a complex plot, numerous characters, and a variety of related problem-solving activities. The *Lost* community is centered around the Lostpedia wiki, which contains more than 85,000 pages. However, the community also includes discussion forums, blogs, and an Internet Relay Chat (IRC) chat channel. The community documents nearly every aspect of the show, from character appearances and actor bios to transcriptions of episodes and producer podcasts to songs played in the background of each episode. In addition to the pages standard to most encyclopedic uses of wikis, Lostpedia also includes "Theory" pages where contributors can propose and discuss possible explanations of what is going on. For example, community members theorize about who built a large statue of the Egyptian Goddess Taweret on a remote tropical island (Fig. 15.13). The result is a community of dedicated individuals collaboratively documenting and making sense of a highly complex narrative.

With more than 3.8 million words of text and thousands of images, it is challenging for newcomers to get a sense of the overall structure of the Lostpedia wiki and for core members to keep up on all the activity. The standby wiki tools (e.g., Recent Changes, Watchlists, User Pages) are indispensable in keeping people connected and helping them find projects of interest. However, network analysis and visualization provides a new method for making sense of relationships in wiki data. In this section, you will see three different views, or maps, of activity and relationships in Lostpedia that can help you avoid getting lost in the wiki. They represent only a handful of the many possible visualizations that community members and administrators could use to better coordinate activity, understand social relationships, and get a sense of their place within the larger community.

15.7.1 Creating an Overview Network Map of Lostpedia Content in NodeXL

The goal of the visualization discussed in this section (Fig. 15.14) is to provide an overview of the structure

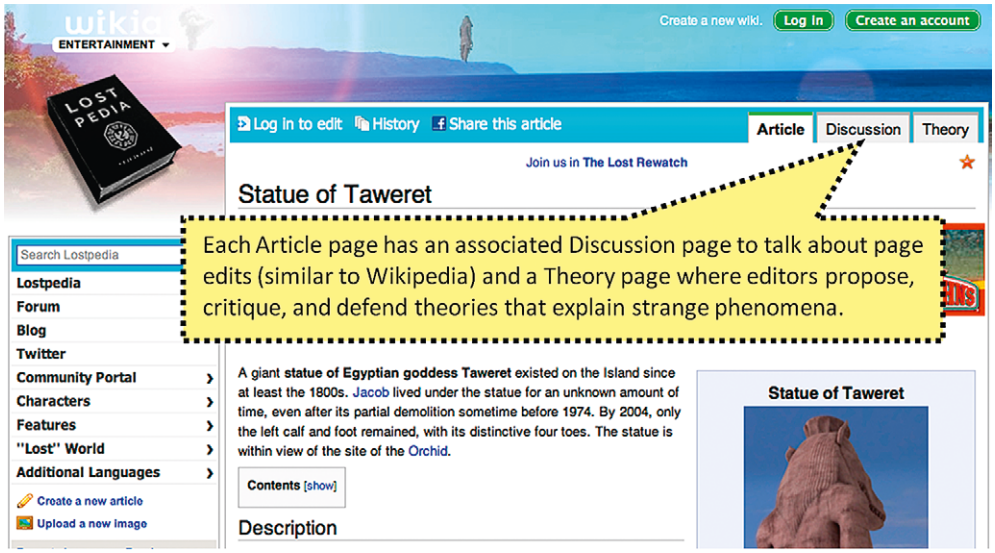


FIGURE 15.13 Lostpedia's article about the Statue of Taweret with links to its associated Discussion and Theory pages. Similar to other wiki systems, Lostpedia include links to History pages and an Edit page. The Theory page is an additional type of page for contributor interpretations of what is happening and why, whereas the articles are more descriptive of what occurred in the show.

of connected wiki pages in the site. Rather than showing which pages link to other pages via hyperlinks or categories (something the wiki navigation does well already), the image focuses on links between pages based on the number of people who have edited both pages. It is a page-to-page network based on co-edits and was derived from the affiliation network linking people to pages they have edited. A link between two pages suggests that "people who edited page X also edited page Y" and is analogous to recommender systems found in popular shopping services on the web that link content products such as books and music to each other based on co-purchasing behavior. The result is a map showing social connections overlaid onto the wiki content.

This overview network map highlights several basic facts about the Lostpedia community:

- Most of the activity is focused on the Article pages, a handful of which are frequently edited (e.g., pages describing the main characters in the show). Thus, editing patterns follow a highly skewed distribution typical of behavior in many online settings.
- Many Theory pages are connected to each other (see the green cluster in Fig. 15.14), suggesting there is a group of "theorists" who edit Theory pages. Theory pages are not connected to Article or Discussion pages often, suggesting that the theorists may be a distinct group of people from those who edit other page types. In contrast, there does not appear to be a comparable "discussion" community on the wiki, likely because they have a separate discussion forum for that purpose. Instead, Discussion pages are linked to their associated Article page, suggesting that they are used in a manner similar to how they are used at Wikipedia—to discuss edits to the Article page.
- Only a few User Talk pages are included and they all belong to administrators (LostpediaAdmin, Jabrwocky7, Nickb123). Most of them are connected to one another, suggesting there is an active group of core members (at least 50) who interact through edits to user pages of prominent members. Lowering the co-edit threshold and limiting the analysis to User Talk pages could provide additional insights about the user community.
- Most pages are located in densely clustered groups (e.g., 83 percent of vertices have a clustering coefficient above 0.5). However, the overall network density is only 0.12, because the average degree is much smaller than is possible for most vertices. The overall pattern is one of many overlapping neighborhoods of articles with a few important article hubs.

Although this high-level view can be helpful in gaining an overview of the community, members are typically interested in the details. The data set used to create Figure 15.14 is available on the book's companion web site at www.mkp.com/nodexl. This allows interested individuals to dynamically explore the data. The file uses NodeXL's custom Vertex Menus feature (accessible via the Options window above the graph pane) to link each vertex with its associated Lostpedia wiki page. If you right-click on a vertex and select "Open Lostpedia Page," an Internet browser will open and take you directly to the associated page.

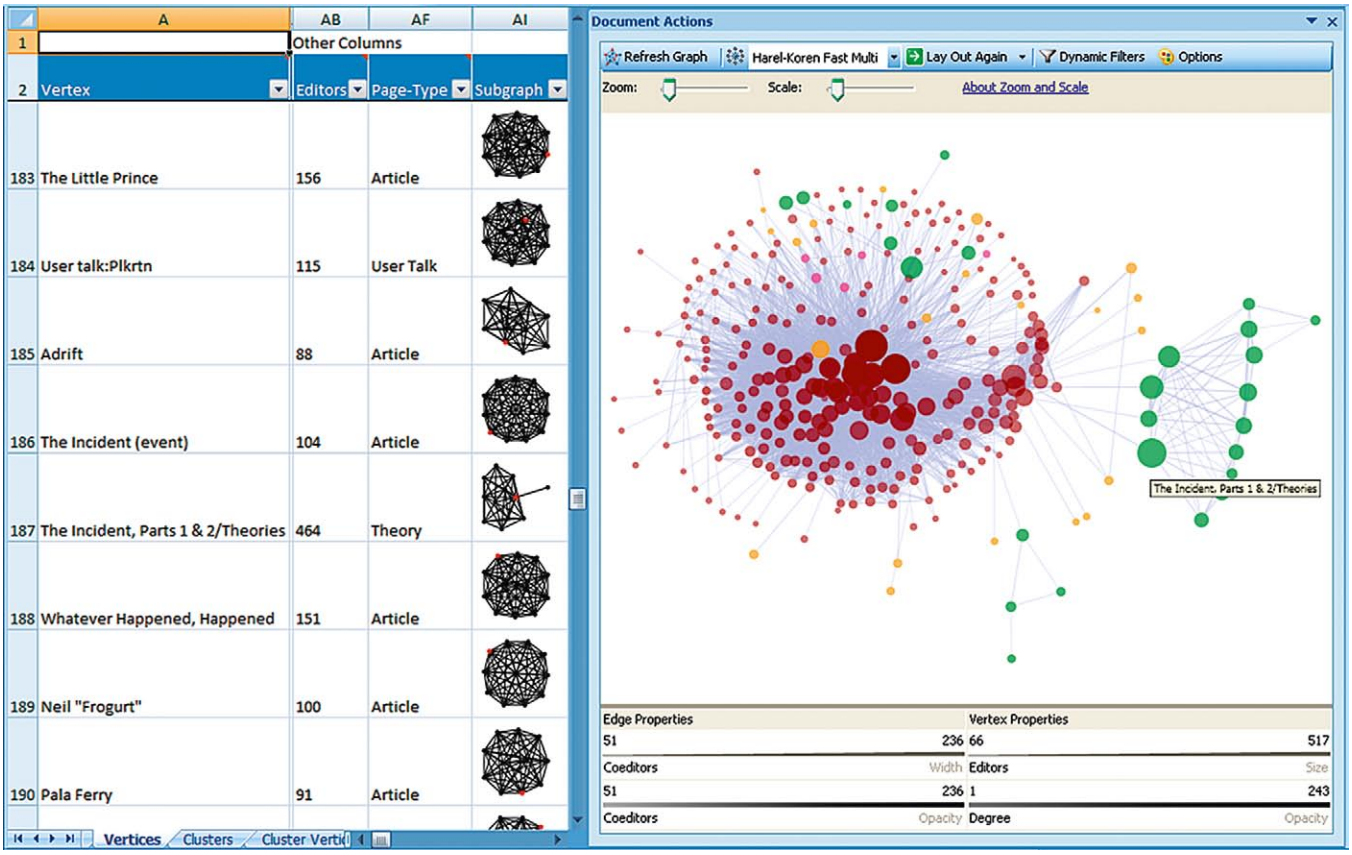


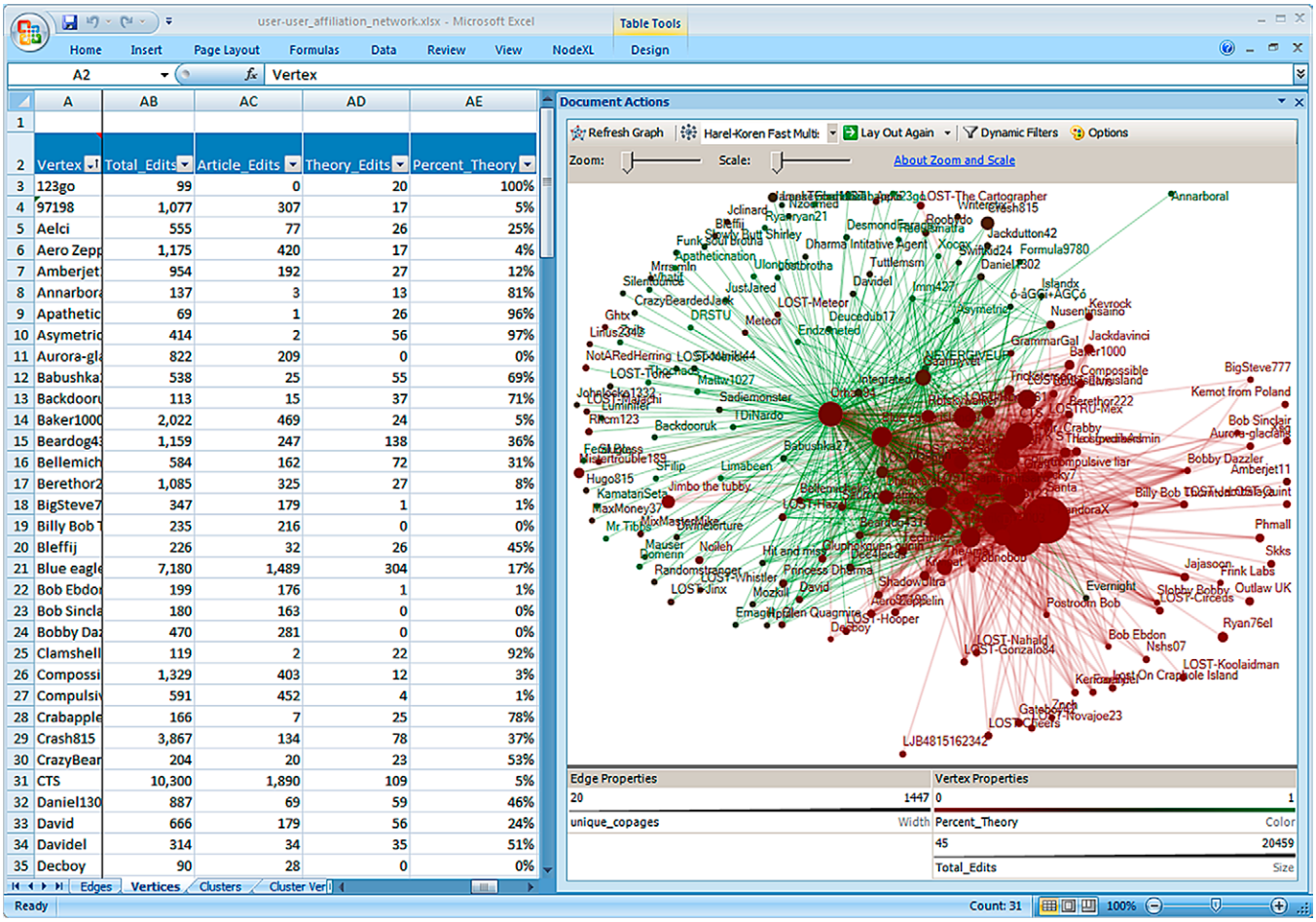
FIGURE 15.14 NodeXL Lostpedia wiki page-to-page co-edit network visualization and Vertex worksheet showing only those pages with more than 50 co-editors. All types of pages were considered, but only Article pages (maroon), Discussion pages (orange), Theory pages (green), and User Talk pages (deep pink) were co-edited enough to show up. The Harel-Koren Fast Multiscale Layout identifies natural groupings such as the main cluster of articles and the cluster of interrelated Theory pages. Size is based on total user edits of a page, and opacity is based on degree. Subgraph images show small dense clusters for the displayed vertices.

To better see patterns within the large Article cluster, you may want to remove some of the major hubs from the graph. Doing so reveals a pocket of closely related articles about episodes, which tend to be grouped by season, which is likely reflective of turnover in membership between years and the close relationship between content of episodes in the same season. It also helps relate pages about important events with people that participated in them because, for example, an episode page is often updated by the same people who update information about events and characters in the episode. Honing in on specific pages such as the "Unanswered Questions" page can quickly identify which articles raise questions (e.g., "A Tale of Two Cities" episode, the character Vincent, the Pearl, the Blast Door Map, and locations like the Barracks), as well as pages that likely help in answering them (e.g., Regularly spoken phrases, Talk: Main Page, Bloopers, and continuity errors).

Although overview graphs like that shown in Figure 15.14 raise as many questions as they answer, they can help provide context to masses of data and identify high-level patterns and important content. They can also serve as a starting point for exploratory analysis of specific relationships.

15.7.2 Creating an Overview Map of Lostpedia Users

The companion graph to the article-to-article affiliation graph is the user-to-user affiliation graph. The user-to-user graph connects users to one another based on the number of unique pages they have both edited. For example, if Derek and Ted have both edited 15 of the same pages, they will have an edge weight of 15. Instead of saying that they know or talk to one another, it says that they "hang out" (i.e., edit) in the same places (i.e., pages) on the wiki. These graphs can get extremely large for wikis with many users. The raw data include nearly 2 million edges with an edge weight of at least 1. To make the network manageable you can filter out edges that are below a certain cutoff point (see Chapter 6).



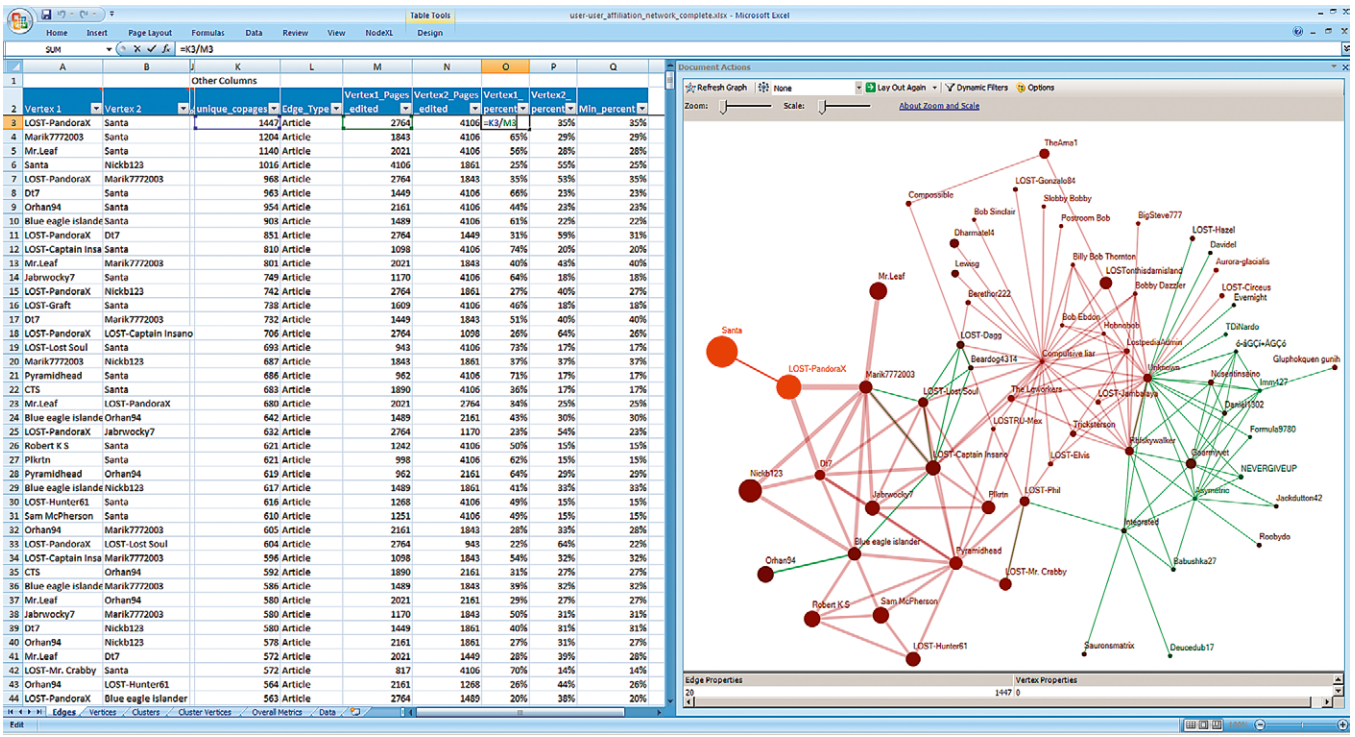
Vertex	Total_Edits	Article_Edits	Theory_Edits	Percent_Theory
123go	99	0	20	100%
97198	1,077	307	17	5%
Aelci	555	77	26	25%
Aero Zepp	1,175	420	17	4%
Amberjet	954	192	27	12%
Annarbora	137	3	13	81%
Apathetic	69	1	26	96%
Asymetric	414	2	56	97%
Aurora-gla	822	209	0	0%
Babushka	538	25	55	69%
Backdooru	113	15	37	71%
Baker1000	2,022	469	24	5%
Beardog43	1,159	247	138	36%
Bellemich	584	162	72	31%
Berethor2	1,085	325	27	8%
BigSteve7	347	179	1	1%
Billy Bob T	235	216	0	0%
Bleffij	226	32	26	45%
Blue eagle	7,180	1,489	304	17%
Bob Ebdor	199	176	1	1%
Bob Sincla	180	163	0	0%
Bobby Daz	470	281	0	0%
Clamshell	119	2	22	92%
Compossi	1,329	403	12	3%
Compulsiv	591	452	4	1%
Crabapple	166	7	25	78%
Crash815	3,867	134	78	37%
CrazyBear	204	20	23	53%
CTS	10,300	1,890	109	5%
Daniel130	887	69	59	46%
David	666	179	56	24%
Davidel	314	34	35	51%
Decboy	90	28	0	0%

FIGURE 15.15 NodeXL visualization of Lostpedia wiki user-to-user affiliation network connecting users (vertices) based on the number of unique pages they have both edited (weighted edges). Two types of edges are included: those connecting users based on co-edits of 20 or more Theory pages (green) and those connecting users based on co-edits of 150 or more articles (maroon). Vertex size is based on total wiki edits, and color is based on the percentage of pages that are Theory pages (green vertices edit mostly Theory pages and maroon vertices edit mostly Article pages). Boundary spanners and important individuals are easily identified.

In the example shown in Figure 15.15, only edges with 20 or more co-edited Theory pages or 150 or more co-edited articles are included. Filtering highlights the most active and arguably most important people, which is desirable for many analyses.

The Lostpedia user-to-user affiliation graph is shown in Figure 15.15. The network is created with the purpose of identifying important individuals and relationships, particularly between the article editors and the theorists. Connections between users are based on the number of Article pages they have both edited (maroon edges) and Theory pages they have both edited (green). This is an example of a multiplex network, as it has different types of edges (see Chapter 3 and [21]). NodeXL can visually represent multiple edges between the same two people by making them different colors and widths and using edge opacity to ensure that you can "see through" edges when there are more than one. Most network metrics in NodeXL assume there is only one type of edge, so interpreting the data must be done carefully. For example, the Degree metric will count each edge independently, so if there are two types of edges connecting two people they would each have a Degree of 2, despite connection to only one other person. A user-to-user affiliation network could be created based on co-edits of any type of page or based solely on co-edits of a specific page type (e.g., User Talk pages) to identify key individuals and relationships within various sections of the wiki.

The network clearly shows the theorists and article writers, and those who span the boundaries separating these groups. The network can be explored within NodeXL by sorting on the data columns in the Vertices worksheet that show the total number of edits, article edits, theory edits, and percent Theory page edits to identify important individuals and find them on the graph. You can access each user's Lostpedia User Page using the customized vertex right-click menu.



	Vertex 1	Vertex 2	unique_copages	Edge_Type	Vertex1_Pages_edited	Vertex2_Pages_edited	Vertex1_percent	Vertex2_percent	Min_percent
3	LOST-PandoraX	Santa	1447	Article	2764	4106	=K3/M3	35%	35%
4	Marik7772003	Santa	1204	Article	1843	4106	65%	29%	29%
5	Mr.Leaf	Santa	1140	Article	2021	4106	56%	28%	28%
6	Santa	Nickb123	1016	Article	4106	1861	25%	55%	25%
7	LOST-PandoraX	Marik7772003	968	Article	2764	1843	35%	53%	35%
8	Dt7	Santa	963	Article	1449	4106	66%	23%	23%
9	Orhan94	Santa	954	Article	2161	4106	44%	23%	23%
10	Blue eagle islande	Santa	908	Article	1489	4106	61%	22%	22%
11	LOST-PandoraX	Dt7	851	Article	2764	1449	31%	59%	31%
12	LOST-Captain Insa	Santa	810	Article	1098	4106	74%	20%	20%
13	Mr.Leaf	Marik7772003	801	Article	2021	1843	40%	43%	40%
14	Jabrwocky7	Santa	749	Article	1170	4106	64%	18%	18%
15	LOST-PandoraX	Nickb123	742	Article	2764	1861	27%	40%	27%
16	LOST-Graft	Santa	738	Article	1609	4106	46%	18%	18%
17	Dt7	Marik7772003	732	Article	1449	1843	51%	40%	40%
18	LOST-PandoraX	LOST-Captain Insano	706	Article	2764	1098	26%	64%	26%
19	LOST-Lost Soul	Santa	693	Article	943	4106	73%	17%	17%
20	Marik7772003	Nickb123	687	Article	1843	1861	37%	37%	37%
21	Pyramidhead	Santa	686	Article	962	4106	71%	17%	17%
22	CTS	Santa	683	Article	1890	4106	36%	17%	17%
23	Mr.Leaf	LOST-PandoraX	680	Article	2021	2764	34%	25%	25%
24	Blue eagle islande	Orhan94	642	Article	1489	2161	43%	30%	30%
25	LOST-PandoraX	Jabrwocky7	632	Article	2764	1170	23%	54%	23%
26	Robert K S	Santa	621	Article	1242	4106	50%	15%	15%
27	Plkrtn	Santa	621	Article	998	4106	62%	15%	15%
28	Pyramidhead	Orhan94	619	Article	962	2161	64%	29%	29%
29	Blue eagle islande	Nickb123	617	Article	1489	1861	41%	33%	33%
30	LOST-Hunter61	Santa	616	Article	1268	4106	49%	15%	15%
31	Sam McPherson	Santa	610	Article	1251	4106	49%	15%	15%
32	Orhan94	Marik7772003	605	Article	2161	1843	28%	33%	28%
33	LOST-PandoraX	LOST-Lost Soul	604	Article	2764	943	22%	64%	22%
34	LOST-Captain Insa	Marik7772003	596	Article	1098	1843	54%	32%	32%
35	CTS	Orhan94	592	Article	1890	2161	31%	27%	27%
36	Blue eagle islande	Marik7772003	586	Article	1489	1843	39%	32%	32%
37	Mr.Leaf	Orhan94	580	Article	2021	2161	29%	27%	27%
38	Jabrwocky7	Marik7772003	580	Article	1170	1843	50%	31%	31%
39	Dt7	Nickb123	580	Article	1449	1861	40%	31%	31%
40	Orhan94	Nickb123	578	Article	2161	1861	27%	31%	27%
41	Mr.Leaf	Dt7	572	Article	2021	1449	28%	39%	28%
42	LOST-Mr. Crabby	Santa	572	Article	817	4106	70%	14%	14%
43	Orhan94	LOST-Hunter61	564	Article	2161	1268	26%	44%	26%
44	LOST-PandoraX	Blue eagle islander	563	Article	2764	1480	20%	38%	20%

FIGURE 15.16 NodeXL Edges worksheet and visualization of a Lostpedia wiki user-to-user affiliation network graph with edges filtered based on the number of pages that users share as a percentage of the total number of edited pages. The number of edges for frequent editors like Santa (highlighted in red) are significantly reduced in the graph, but size indicates that they exist with those filtered out of the graph.

15.7.3 Normalizing Data to Infer Stronger Connections

Notice that in Figure 15.15 the frequent editors (larger circles) are connected to many people (have high Degree). For example, Santa (who apparently keeps pretty busy in the off season) edits so many different pages that he is connected to 88 other people. These frequent editors hang out all over the wiki, making it hard to know who they are most closely associated with and obscuring connections between less frequent editors. To account for this behavior, you can normalize the data, for example, by focusing on percentages of co-edited pages instead of raw count numbers. There are a number of advanced techniques for normalizing data that are beyond the scope of this book (see Chapter 8 of Wasserman and Faust [18] for a discussion of normalization in affiliation networks). Here we use a simple but effective technique based on percentages to create a less cluttered graph that focuses on connections that are significant to both editors in a relationship pair (Fig. 15.16).

A number of additional columns have been added to the Edges worksheet in the NodeXL workbook that indicate the total number of unique pages each user has edited in the Lostpedia wiki. The number of unique co-edited pages for each pair is divided by the individual's total edited page count to calculate the percent of a person's page edits that are shared with the other person (see row 3 of Fig. 15.16). If this percentage is high for both individuals, there is a strong connection between them. You can use the minimum of the two vertex's percentages as a cutoff point to assure that both individuals have a high percentage. In Figure 15.16 we have used the Edge Visibility attribute setting in the AutoFill Columns feature found in the NodeXL Visual Properties menu ribbon to exclude edges that have a minimum percentage of less than 0.3 (a number we arrived at by trial and error using dynamic filters).

The resulting network graph downplays the prominence of individuals like Santa and Orhan94, while still showing their strongest connections by using Vertex Size to indicate their prominent roles. The less-cluttered graph helps identify other important individuals such as LOST-Captain, Rbfskywalker, and Unknown who span the Theory and Article community, as well as members like Gaarmyvet and Jabrwocky7, who are important in the Theory and Article communities, respectively. Finally, it helps identify relationships between important individuals such as the Lostpedia administrators, most of whom are represented as large red vertices.

15.8 DATA COLLECTION FROM WIKI SYSTEMS

Data collection from wikis is not automatic, and for the time being, practitioners and researchers alike are forced to "roll their own" data collection strategies. This high entry cost is a mixed blessing. It may be harder to get started, but ultimately the richness of the data may make them more rewarding. Wiki data offer unique insights into the temporal flow of computer-mediated collection action. Unlike other social media systems, wiki software records and automatically documents the process and product of collective collaboration. Further, skillful practitioners of network analysis that leverage the complexity of wiki data are likely to gain insights that are more varied, deeper, and more valuable that those likely to emerge from simpler types of social media.

Data collection from wikis requires a combination of technical skill and effort. In many regards the easiest strategy to implement was illustrated in the first example using data from the Empire Wiki. In that example, we used a standard web scraping package to rapidly collect network and attribute data and record it as a series of standard Excel spreadsheets. Such software is useful for Wikipedia studies, as well as those that use data from forums, blogs, or other web pages with standardized page structures. Besides the initial investment in the software, this method requires nothing more than a basic understanding of html to create the collection templates and the basics of Excel, Access, or Structured Query Language (SQL) database tools to organize the data.

Our second example extracted data directly from Wikipedia and required no special tools. All attributes and edges were collected by applying a simple coding scheme to the content in article talk pages. This study illustrated how practitioners or researchers can narrow their focus to the process of interaction surrounding the construction of a single page. This precise focus highlights small group dynamics, issues of leadership, cooperation, and social control. That data, of course, could be augmented with variables collected via other methods, like web scraping, to measure edit patterns, interviews with editors to assess motivations, or even experimental data to test hypotheses about the motivations and types of different wiki editors.

The third example illustrated methods for working with data from an entire wiki and thus employs the most technically demanding data collection. Starting from the original SQL database, the data collection effort can either be performed via queries to the database or, as in the case of many public wikis like Wikipedia, the database is also accessible as a zipped file representing the state of the database at a given time point. Practitioners will need people on their team with skills in SQL database manipulation, large-scale text parsing with tools like Python or Perl, or, ideally, both. Regardless, once the sample frame is established, a data set of a size and structure suitable to Excel can be extracted, and work in NodeXL can proceed.

15.9 PRACTITIONER'S SUMMARY

Wikis are complex social media systems that give rise to many types of relationships, which aggregate into even more complex networks. The complexity inherent in wiki systems is the source of both challenge and opportunity for practitioners. Wikis can provide valuable insights because they are places where collaboration happens and value is created through informal organization. Social network analysis provides a unique window into how people work together to create that value.

We defined and illustrated several structural visualization strategies that can be readily applied to any type of wiki: identifying contributors who play different roles in projects, distinguishing potentially counterproductive contributors from those who actively encourage productive cooperation, and documenting the structure of large-scale collaboration that is distributed through the overlapping edits of contributors. Although practitioners with access to corporate wikis or other internal data sets may find it easy to formulate research questions, it will typically be much more difficult to identify the best path for answering them. This chapter addresses key steps in overcoming these challenges. First, familiarity with the wiki system in general will help practitioners identify which types of pages and which projects warrant special attention. Second, we suggest that practitioners use knowledge of the particular projects, particular populations of editors, or time periods of observation to narrow their attention. Clarifying which questions need to be answered will help practitioners identify which edits to use as evidence of meaningful network relationships and which attributes are relevant and should be measured. Both substantive and theoretical insights are crucial for narrowing the sampling frame in ways that will best answer research objectives.

You can be creative! Wiki network studies remain a young but rapidly developing area of research. With just a computer and an Internet connection, your research may be uncovering patterns never seen before. The studies in this chapter just begin to scratch the surface of what can be done with wiki-edited network data. The building blocks of analysis and measurement throughout this volume can be combined and applied to wikis, and doing so will likely be fruitful. These studies also illustrate the unique "everything is an edit" aspect of wiki systems.

Although communication takes place in wikis, communication actually takes place using the standard page-editing tools rather than an explicit communication system with system-identified recipient structures. Because of its focus on relationships, network analysis practitioners can gain more accurate and valuable insights than those who use an individualistic strategy that ignores patterns of connections among users.

15.10 RESEARCHER'S AGENDA

Part of the impetus behind the development of the NodeXL project is to expand the range of researchers and practitioners who would consider using social network analysis techniques in their research. For many years, network analysis software exacted a high price from people wanting to use these tools in terms of technical skill and willingness to learn proprietary software systems. NodeXL as well as browser-based network visualization tools like TouchGraph are helping expand participation in social network analysis to those without programming skills. We anticipate that established research questions will see increased activity, and, most important, researchers will turn the lens of network analysis on new questions and subjects. Because wikis are so rich in terms of content and interaction, the growth of new research agendas in the social network analysis of wikis will be especially fruitful. We will discuss several promising directions.

Wiki data provide an unprecedented window into the process and product of distributed collaboration. The cooperation and coordination problems inherent in any large task are, to varying degrees, overcome in different wiki projects. How is this collaboration achieved [22, 23]? What are the key conditions that shape differential levels of collaborative success [8]? Which norms, conventions, roles, management strategies, and so on are key [24]?

Online and off, people frequently adopt distinctive patterns of behavior and interaction known as social roles [12, 13, 25]. Wiki systems are no exception; in fact, because of the complexity of the system and diversity of the organizational tasks, wikis are likely to give rise to many distinctive social roles [14, 24].

Wikis are rich settings in which to study the dynamics of diffusion [4]. The temporal history of wiki edits are retained with high fidelity, and editors have both durable identities and numerous internal contexts for interaction; researchers can trace diffusion processes better in wikis than they can almost anywhere else. Further, because creation and communication happen to a large extent within the context of the wiki, researchers can study the role of diffusion processes within the larger context of collaborative creation. Bridging research on diffusion with studies of cooperation, as well as with those of social roles, will be a fruitful direction for further research.

References

[1] Y. Benkler, The Wealth of Networks, Yale University Press, New Haven, CT, 2006.

[2] C. Shirky, Here Comes Everybody: the Power of Organizing Without Organizations, The Penguin Press, New York, 2008.

[3] A. Capocci, V. Servidio, F. Colaiori, L. Buriol, D. Donato, S. Leonardi, et al., Preferential attachment in the growth of social networks: the case of wikipedia, Phys. Rev. E 74 (2006) 036116.

[4] Y. Yuan, D.C. Connie, T. Howard, L.X. Welser, G. Gay, The diffusion of a task recommendation system to facilitate contributions to an online community, J.Comput. Mediat. Commun. 15 (1) (2009) 32–59.

[5] D.L. Hansen, Knowledge Sharing, Maintenance, and Use in Online Support Communities, Unpublished Dissertation, University of Michigan, Ann Arbor, 2007, Available online at: http://hdl.handle.net/2027.42/57608

[6] J. Voss, Measuring Wikipedia, in: Proc. 10th Intl. Conf. Intl. Soc. for Scientometrics and Informetrics, July 24–28, Stockholm, Sweden, 2005.

[7] F.M.W. Viegas, K. Dave, Studying cooperation and conflict between authors with history flow visualizations, in: Proc. of the SIGCHI conference on Human factors in computing systems, April 24–29, Vienna, Austria, 2004, pp. 575–582.

[8] A. Kittur, R. Kraut, Harnessing the wisdom of crowds in Wikipedia: quality through coordination, in: Proc. of Computer Supported Cooperative Work, Nov. 8–12, San Diego, CA, 2008, pp. 37–46.

[9] D. Cosley, D. Frankowski, L. Terveen, J. Riedl, SuggestBot: using intelligent task routing to help people find work in Wikipedia, in Proc. of the 12th international conference on Intelligent user interfaces, January 28–31, 2007, Honolulu, Hawaii, 2007.

[10] A. Kittur, B. Suh, B.A. Pendleton, E.H. Chi., He says, she says: conflict and coordination in Wikipedia, in: Proc. of 25th Annual ACM Conference on Human Factors in Computing Systems, April 28–May 3, San Jose, CA, 2007, pp. 453–462.

[11] I. Beschastnikh, T. Kriplean, D.W. McDonald, Wikipeidan self-governance in Action: motivating the policy lens, in: Proc. of International Conference on Weblogs and Social Media, March 30–April 2, Seattle, WA, 2008.

[12] E. Gleave, H.T. Welser, M. Smith, T. Lento, A conceptual and operational definition of social role in online community, in: Proc. of the 42nd Hawaii International Conference on Systems Sciences(HICSS), January 5–8, Waikoloa, HI. Computer Society Press, 2009.

[13] J. Thom-Santelli, M.J. Muller, D.R. Millen, Social tagging roles: publishers, evangelists, leaders, in: Proc. of the twenty-sixth annual SIGCHI conference on Human factors in computing systems, April 05–10, 2008, Florence, Italy, 2008 [doi > 10.1145/1357054.1357215]

[14] H.T. Welser, D. Cosley, G. Kossinets, A. Lin, F. Dokshin, G. Gay, et al., Finding social roles in Wikipedia, Paper presented at the American Sociological Association annual conference, August 1–4, Boston, MA, 2008.

[15] C.T. Butts, Revisiting the foundations of network analysis, Science 325 (2009) 414–416.

[16] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, S. Suri, Feedback effects between similarity and social influence in online communities, in: Proc. of the 14th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, Aug. 24–27, Las Vegas, NV, 2008.

[17] P. Underwood, New Directions in Networked Activism and Online Social Movement Mobilization, Unpublished master's thesis, Ohio University, 2009.

[18] S. Wasserman, K. Faust, Chapter 8: Affiliations and overlapping subgroups, Social Network Analysis: Methods and Applications, Cambridge University Press, New York, 1994.

[19] L.W. Black, T. Welser, J. DeGroot, D. Cosley, Wikipedia is not a democracy: Deliberation and Policy Making in an Online Community, Presented in the Political Communication Division of the International Communication Association annual convention, May22–26, Montreal, Canada, 2008.

[20] J. Gastil, L.W. Black, Public deliberation as an organizing principle for political communication research, J. Public Deliberat. 4 (2008) Article 3. Available at http://services.bePress.com/jpd/vol4/iss1/art3.

[21] R.A. Hanneman, Introduction to social network methods, 16. Multi-plex relations, www.faculty.ucr.edu/~hanneman/nettext/C16_Multi_Plex.html

[22] S.L. Bryant, A. Forte, A. Bruckman, Becoming Wikipedian: transformation of participation in a collaborative online encyclopedia, in: Proc. of ACM GROUP, '05,Nov. 6–9, Sanibel Island, FL, 2005, pp. 11–20.

[23] A. Forte, A. Bruckman, Scaling consensus: increasing decentralization in Wikipedia governance, in: Proc. of 41st Annual Hawaii International Conference on System Sciences (HICSS), Jan. 7–10, Waikoloa, HI, 2008.

[24] B. Butler, E. Joyce, J. Pike, Don't look now, but we've created a bureaucracy: the nature and roles of policies and rules in Wikipedia, in: Proc. of 26th Annual ACM Conference on Human Factors in Computing Systems, 2008, April 5–10, Florence, Italy, 2008, pp. 1101–1110.

[25] H.T. Welser, E. Gleave, D. Fisher, M. Smith, Visualizing the signatures of social roles in online discussion groups, J. Social Struct. 8 (2) (2007).

APPENDIX

NodeXL for Programmers

This appendix covers NodeXL topics of interest to programmers. If you know C#, VB.NET, or another .NET language, you can customize the NodeXL Excel 2007 template to import network graph data from any data source. You can also create your own graphing application using the NodeXL class libraries.

A.1 IMPORTING CUSTOM GRAPH DATA

The NodeXL Excel 2007 template has built-in support for importing network graph data from a variety of sources, including edge list workbooks, matrix workbooks, UCINET files, Pajek files, GraphML files, and so on. You can import from these sources using the Data, Import menu in the NodeXL tab in Excel's ribbon (see Chapter 4).

But what if you have graph data that is in a format not directly supported by the template, such as a corporate network directory, a public social network API, or an obscure graph file format, and you want to display the graph data in the template? You can do this by creating a "graph data provider," which is a .NET assembly that implements an interface called IGraphDataProvider. When placed in the template's plug-in folder, the assembly adds a custom menu item to NodeXL's Data, Import menu. When the menu item is selected, the graph data provider is told to get the graph data, format it as XML, and provide it to the template. The template then fills in the NodeXL workbook with the custom data.

None of the NodeXL source code is required to create a graph data provider. See Chapter 12 for an example data provider built by Robert Ackland.

A.1.1 Getting Started

To create a graph data provider, you will need the NodeXL assembly that defines the IGraphDataProvider interface (ExcelTemplatePlugIns.dll) and the NodeXLApi help file that documents it. Both are included in the "NodeXL Class Libraries" download on the NodeXL web site. (The download also contains a number of other assemblies that can be used to create a custom graphing application. You may ignore these items.)

A.1.2 The IGraphDataProvider Interface

The IGraphDataProvider interface looks like this:

```
public interface IGraphDataProvider
{
String Name { get; }
String Description { get; }
Boolean TryGetGraphData(out String
  graphDataAsGraphML);
}
```

The Name and Description properties provide NodeXL with the text and tooltip for the custom menu item it adds to its Data, Import menu. The more interesting part of the interface is the TryGetGraphData() method, which NodeXL calls when the user selects the custom menu item.

The TryGetGraphData() implementation typically opens a dialog box in which the user can specify the parameters of the network she wants to import. If the graph data provider is meant to import from some file format, for example, this would be a file selection dialog box. The implementation must then get the graph data, format it in an XML schema called GraphML, and return the GraphML to NodeXL as a string.

A.1.3 GraphML

GraphML is a simple but powerful XML schema for representing network graphs. A good overview can be found here: http://graphml.graphdrawing.org.

Here is sample GraphML that represents an undirected graph with five vertices and two edges:

```
<?xml version="1.0" encoding="UTF-8"?>
<graphml xmlns="http://graphml.graphdrawing.org/
  xmlns">
```

```
<key id="VertexColor" for="node" attr.
  name="Color" attr.type="string" />
<key id="VertexLatestPostDate" for="node" attr.
  name="Latest Post Date"
attr.type="string" />
<key id="EdgeWidth" for="edge" attr.name="Width"
  attr.type="double">
<default>1.5</default>
</key>
<graph edgedefault="undirected">
<node id="V1">
<data key="VertexColor">red</data>
<data key="VertexLatestPostDate">2009/07/05</data>
</node>
<node id="V2">
<data key="VertexColor">orange</data>
<data key="VertexLatestPostDate">2009/07/12</data>
</node>
<node id="V3">
<data key="VertexColor">blue</data>
</node>
<node id="V4">
<data key="VertexColor">128,0,128</data>
</node>
<node id="V5" />
<edge source="V1" target="V2" />
<edge source="V3" target="V2">
<data key="EdgeWidth">2.5</data>
</edge>
</graph>
</graphml>
```

NodeXL imports the specified vertices and edges into the workbook, along with any vertex and edge attributes. The GraphML documents call these "GraphML attributes." If the "attr.name" of the GraphML attribute is the name of an existing column in the Vertices or Edges worksheet ("Color" and "Width" in this example), the specified vertex or edge attribute values are imported to those columns. If the attr.name is not the name of an existing column ("Latest Post Date" in this example), a new column is added to the worksheet and the specified attribute values are imported to the new column. Thus, the graph data provider can fill in any columns in the Vertices or Edges worksheet simply by specifying the appropriate GraphML attributes.

A.1.4 Sample Code

The NodeXL Class Libraries download includes a compilable sample implementation of IGraphDataProvider. Look for the file called "SampleGraphDataProvider.cs." Also, the NodeXL source code, available on the NodeXL web site, includes complete source code for its own Flickr, YouTube, and Twitter graph data providers. These providers support a large number of import options and are considerably more complex than the sample implementation.

A.2 CUSTOM GRAPHING APPLICATIONS

The pane that displays the network graph in the NodeXL Excel template is actually a custom Windows Presentation Foundation (WPF) control that is not specific to the template or Excel. If you are creating a WPF or Windows Forms application that needs to display an interactive network graph, this control will let you do so.

A.2.1 Getting Started

To use the graph control, which is called NodeXLControl, you will need the Microsoft.NodeXL.Control.Wpf.dll assembly, several support assemblies, and the help file that documents everything. All of these are included in the "NodeXL Class Libraries" download on the NodeXL web site.

A.2.2 Populating the Graph

You populate the graph by adding vertices to its vertex collection and then connecting the vertices by adding edges to its edge collection. The following code adds two vertices, connects them with an edge, and draws the graph:

```
IVertexCollection vertices = nodeXLControl.Graph.
  Vertices;
IVertex vertexA = vertices.Add();
IVertex vertexB = vertices.Add();
IEdgeCollection edges = nodeXLControl.Graph.Edges;
IEdge edge1 = edges.Add(vertexA, vertexB);
nodeXLControl.DrawGraph(true);
```

A.2.3 Controlling the Appearance of the Graph

The previous code is all that is required to display a graph that uses small black disks for vertices and thin black lines for edges. Sometimes, though, you'll want to modify the way the vertices and edges are drawn. The NodeXLControl offers extensive control over the graph's appearance.

Take the shape of the vertices, for example. You can change the default vertex shape by setting the control's VertexShape property:

```
nodeXLControl.Shape = VertexShape.Circle;
```

This will cause every vertex to be drawn as a circle instead of a disk. Now what if you want most vertices to be circles but one of them to be a solid square? You can

override the default shape for a vertex by setting metadata on the vertex with the Vertex.SetValue() method:

```
vertexA.SetValue(ReservedMetadataKeys.
  PerVertexShape, VertexShape.Square);
```

There are additional metadata keys for setting a vertex's color, radius, transparency, visibility, label, label fill color, label position, and image. These are all documented in the NodeXLApi help file.

Similarly, edges are drawn with a default width that can be modified with the control's Edge Width property:

```
nodeXLControl.EdgeWidth = 3.0;
```

The default width can be overridden on an individual edge using the ReservedMetadataKeys.PerEdgeWidth metadata key:

```
edge1.SetValue(ReservedMetadataKeys.PerEdgeWidth,
  4.2F);
```

There are also metadata keys for setting an edge's color, transparency, visibility, and label.

A.2.4 Controlling the Layout

By default, the NodeXLControl lays out the graph's vertices using a "force-directed" layout algorithm developed by Thomas Fruchterman and Edward Reingold. Force-directed means that the algorithm models the vertices as bodies that exert physical forces on each other, depending on how they are connected with edges. Although Fruchterman-Reingold works well in many cases, different layout algorithms are more appropriate for other graphs. You can change the control's layout algorithm using the Layout property. This code, for example, causes the vertices to be laid out in a grid:

```
nodeXLControl.Layout = new GridLayout();
```

NodeXL provides additional geometric layouts for circles, spirals, and sine waves; two versions of polar layouts; a random layout; and another force-directed layout optimized for larger graphs, called Harel-Koren Fast Multiscale.

A.2.5 More Features

Other properties and methods on the NodeXLControl let you programmatically select vertices and edges, zoom and scale the graph, add vertex tooltips, change the mouse selection behavior, and more. See the "NodeXL Control Class" topic in the NodeXLApi help file for details.

A.2.6 Sample Code

The NodeXLApi help file includes compilable sample code that adds a NodeXLControl to a WPF application and to a Windows Forms application. See the help file's "NodeXL Control Class" topic.

Index

A

ABC-D email list
  data preparation for analysis of, 137
  finding new administrator for, 136–138
  graph with administrator removed, 138, 138f
  reply network graph in, 137, 137f
Acquiring Twitter data
  options for, 152–154, 152f
  REST API/whitelisting and, 152–153
Administrator, finding new. *See* ABC-D email list
Affiliation network, 36–37
  as threaded network, 130–131
  transforming multimodal into unimodal, 80–81
Aggregate graph metrics, 69
Aggregate networks metrics, 40
Algorithms
  clustering and, 41, 99
  detection, 41
Amber Alert, 6
Analysis, social network
  applications of, 5
  business applications of, 43–44
  early, 38–39
  era of abundant computation and, 44–45
  Gartner Group on, 4
  graph theory and, 35
  history regarding, 4, 5
  individuals/groups who study, 5–6
  introduction, 31–32
  meets the web, 45
  metrics, 39–42
  network perspective in, 32–35
  node-link diagrams and, 47, 48f
  obstacles to development of, 39
  practitioner's summary regarding, 48–49
  questions applied to social media, 47–48
  renaissance of, 42–44
  researcher's agenda and, 8–9, 49
  research/practitioner landscape, 37–39
  tools for, 47
Analyzing Flickr networks, 212–220
  data preparation advise for, 221
  graph layouts and, 221–222
  graph types for, 212, 212f
  identifying tag clusters, 215–216
  manual/multiple visualization layouts, 221–222
  merging duplicate edges for, 212, 212f
  overview of, 212
  researcher's agenda and, 222–223
  revealing landmarks/tourist attractions, 213–215
  user networks, 212f
Analyzing personal email networks, 115
  clean data for, 115
  compute graph metrics/add new columns, 116
  creating expertise email graph, 117f, 118–119, 118f
  creating overview visualization, 115–118
  filter data, 115–116
  import data, 115
  scenario for, 115
  understand visualizations/metrics data, 116–118
  visualize social network, 116, 117f
Analyzing technical support email list
  benefits of, 133
  data preparation for, 132–133, 132f
  important people and, 133–135, 134f
  overview of, 132–135
  social role measures, 134–135, 135t
  social roles identification in, 133–135, 134f
  subgraph images, 133, 134f
  viewing reply network in, 135, 136f
Analyzing Youtube, 234–245. *See also* Healthcare reform video networks; Makeup video network
  Harvey and, 235–236, 236f, 237f
  practitioners summary on, 245
  researcher's agenda, 245
  Rihanna and, 234, 234f, 235f
  user networks, 234–238
  video networks, 238–245
Annotation, 27–28
API. *See* Application programming interface
Application programming interface (API), 152–153
Art, as social sharing, 24
Article history page, Project Chanology, 254, 255f
Article talk pages
  coding scheme and, 261
  collaboration issues and, 261
  conclusions regarding, 263–264
  data collection techniques for, 269
  deliberation and, 262
  edge weights/vertices size and, 262
  identifying high-quality contributors in, 261–264
  Lostpedia and, 265
  sampling frame and, 261
  signatures of confrontation/deliberation in, 262–264, 264f
  vertices colors and, 262, 263f
  visualization tasks for, 262
  visualizing contributor types in, 262
Asymmetric connections, 151
Asynchronous communication, 14
Asynchronous threaded conversation, 17. *See also* Thread networks
  BBS, 17–18
  discussion forums, 17–18, 20
  email, 17, 107
  email lists, 17
  technologies of collaboration and, 17, 18t
  Usenet newsgroups, 17
@ replies/mentions
  ego network graph representing, 157–158, 158f
  symmetric connections and, 151
  Twitter and, 146–147, 147f, 147t
Audio conferencing, 21
Auto-filling data columns, 81–82, 83f

B

Barriers to entry, 15–16
Basic elements
  control of, 15–16, 16t
  genre of, 15
  types of connections and, 16–17
BBS. *See* Bulletin board systems
"Beau Nelson's essential makeup tips" video, 242, 242f
Betweenness centrality
  altered manually, 175
  computing, 71f, 72
  ego network graph using, 157, 157f
  top three sites, 196
  Twitter and, 150–151, 150f
  as vertex-specific networks metrics, 40
  visualizing ordered Facebook data via, 174–176
Black Friday. *See* Trending topics, in Twitter
Blogs, 22–23
  hyperlink networks and, 196–197
  micro, 23
  multimedia, 23
  social media history and, 4
  Twitter compared with, 144, 146t
Bookmarks/books, 24
Bulletin board, Youtube, 229
Bulletin board systems (BBS)
  as asynchronous threaded conversation, 17–18
  history surrounding, 129
Bush, Vannevar, 12

C

Cascading style sheets discussion email list (CSS-D). *See* Analyzing technical support email list

Categories
manually attached to vertex attributes, 172–173, 173f
visualizing, 172, 172f
Category Network, 254t
Celebrity, 151
Channel page layout, Youtube, 228f, 229
Chanologists, 259–260, 260f
Chanology. *See* Project Chanology
Chat, as synchronous conversation, 18
Chinese Academy of Sciences, 7–8
Church of Scientology. *See* Project Chanology
Circle layout, 177–178
Closeness centrality
computing, 71f, 72
as vertex-specific networks metrics, 40–41
Clustering
additional algorithms for, 99
automatically identifying, 95–96, 97f
detection algorithms and, 41
ego network graph using, 156, 157f
FCC lobbying coalition and, 100–101, 100f
filtering edges and, 94–95, 95f
Friend Wheel and, 177
healthcare reform video networks and, 244, 244f
hyperlink networks analysis issues and, 184
introduction, 93–94, 93f
in *Les Misérables* example, 98–100, 99f
makeup video network and, 240, 240f
manually creating, 97–98, 98f
practitioner's summary regarding, 101
raw data/visualization and, 94–98, 94f
researcher's agenda and, 101
visualizing nonordered Facebook data and, 171, 171f
Clustering coefficient
computing, 72–73
as vertex-specific networks metric, 41
Collaboration. *See also* Collaborative authoring; Technologies of collaboration
article talk pages and, 261
Collaborative authoring, 22
shared documents, 22
wikis, 22
Communication patterns, 120–121, 121f. *See also* Asynchronous communication; Synchronous communications
Community detection algorithms, 41
Computation, era of abundant, 44–45
Computing graph metrics, 70–74
betweenness centrality, 71f, 72
calculating/importing additional metrics, 74
closeness centrality, 71f, 72
clustering coefficient, 72–73
degree, 72
dialog box for, 70–74, 71f
Eigenvector centrality, 71f, 72
expertise network email and, 118
Les Misérables co-appearance network, 74–78
Overall Metrics worksheet and, 73–74, 73f
personal email networks and, 116
practitioner's summary, 78
researcher's agenda and, 78
revealing landmarks/tourist attractions by, 214
sorting edge weight column, 74–75, 75f
vertex-specific, 71–72, 71f
visualizing edge weights, 75, 75f
Comte, Auguste, 38
Connections
directed/undirected, 17
explicit, 16
implicit, 16–17
location and, 17
types of, 16–17
weighted, 17
Consumer applications, rise of, 4
Content page layout, Youtube, 227, 227f
Control of basic elements, 15–16
barriers to entry and, 15–16
governance structures and, 16
granularity of control and, 16, 16t
Conway, Drew, 47
Coordinate mapping, X/Y, 77–78, 77f, 78f
Country code TLD, 190
Create VOSDON database, 187–194
adding seed sites, 189, 189f
comment option, 188
crawling seeds, 189
database name and, 187, 187f
dialog, 187, 187f
parameters for inbound crawl, 188
select network dialog, 188, 188f
Creating living organization-chart
creating overview of communication patterns, 120–121, 121f
examining research division, 121–122, 122f
introduction, 119–122
normalizing/filtering data for, 119–120
TechABCs email network, 119
CSS-D. *See* Analyzing technical support email list
Customizing layout options, 82, 83f

D

Data preparation
ABC-D email list, 137
Facebook, 168–170
Flickr, 217–218, 221
technical support email list, 132–133, 132f
thread networks, 131
Data representations, network, 34–35
attribute data *vs.*, 34
edge list, 35, 35t
graph, 35, 35f
matrix, 34–35, 35t
Dating social networking services, 25
Degree centrality
computing, 72
as vertex-specific networks metrics, 40–41
visualizing ordered Facebook data via, 174
Design framework, 12–17
basic elements, control of, and, 15–16
basic elements, genre of, and, 15
blogs and, 11–12
key dimensions influencing, 13
pace of interaction affecting, 14–15
producer/consumer population size and, 13–14, 13t
retention of content and, 17
types of connections and, 16–17
Digital objects, 15
Directed/undirected
connections, 17
edges, 34
Disaster response, 6
Discussion feature, Wikipedia, 249–251, 251f
Discussion forums, 17–18, 20
Dunbar, Robin, 43
Dynamic filtering, 84–87, 85f, 86f, 87f

E

Edges, 34
article talk pages and, 262
computing graph metrics regarding, 74–75, 75f
directed/undirected, 34
filtering, 94–95, 95f
hyperlink networks analysis issues and, 184
list, 35, 35t
merging duplicate, 80f, 81, 114–115, 114f, 212, 212f, 238
NodeXL and, 54–57, 56f, 65, 66f
Project Chanology and, 254–255
weighted/unweighted, 34
Wiki networks and, 252–253
worksheet, 190–191, 217–218
Editors, identifying, 253–256
Ego network, 154–158
aligning vertices and, 158
analyzing, 155
closed triads in, 154
data collection, 155, 155f
graph before filtering, 156, 156f
graph displaying @replies/mentions, 157–158, 158f
graph with cluster detection, 156, 157f
graph with eigenvector/betweenness centrality, 157, 157f
graph without strangers, 156, 156f
reciprocal ties in, 154
terms used in, 154
Egocentric network, 36
Facebook and, 167–168, 168f
Ego/egoless visualization, Facebook and, 170, 170f
Egyptian Goddess Taweret, 264, 265f
Eigenvector centrality
computing, 71f, 72
ego network graph using, 157, 157f
Twitter and, 150
as vertex-specific networks metric, 41
Email. *See also* Analyzing personal email networks; Creating living organization-chart; Email list; Enron email; Expertise network email graph, creating
as asynchronous threaded conversation, 17, 107
broadcast and, 107
calculate total messages sent/received, 116
cleaning email data in NodeXL, 113–115
collections, 108–109, 109t

data, preparing, 110, 111f
data, working with, 110–113
demographics regarding, 105–106
duplicate addresses, managing, 113–114, 113f
duplicate edges, merging, 114–115, 114f
ethical challenges of, 106
filtering and, 111–112, 112f, 118
flexible form and, 107
history/definition of, 106–107
importing and, 111–113, 112f, 118
importing networks into NodeXL, 111–113, 112f
introduction, 105–106
lists as asynchronous threaded conversation, 17
networks, 107–108, 107t, 108f
organizational network questions, 109
permeation of, 105
personal network questions, 109
practitioner's summary regarding, 124
as push technology, 107
questions answered by analyzing, 108–109
researcher's agenda and, 124
self-loops, identifying, 115
technical characteristics of, 107
working with large collections, 111
Email list. *See also* ABC-D email list; Analyzing technical support email list; Ravelry
as asynchronous threaded conversation, 17
Encyclopedia Dramatica. *See also* Project Chanology
data collection techniques for, 269
identifying editors within, 253–256
introduction to, 248–249
what it is, 253–254
Engelbart, Douglas, 12
Enron email
data collection, 122
historical/legal analysis of, 122–124
identifying key individuals, 122–124, 123f
Era of abundant
computation, 44–45
social networks, 46
Ethical considerations
email, 106
Youtube data importation, 233
Euler, Leonhard, 35
European Society of Socially Embedded Technologies, 7
European-based Institute for Prospective Technological Studies, 7
Everything is a page feature, Wikipedia, 251, 252f
Expertise network email graph, creating, 117f, 118–119, 118f
clean data for, 118
compute graph metrics/add new columns for, 118
filter data for, 118
import social data for, 118
understand visualization/data and, 119
visualize network for, 118–119
Explicit connections, 16
Explore page, Flickr, 202–204, 203f

F

Facebook. *See also* Friend Wheel; Visualizing nonordered Facebook data; Visualizing ordered Facebook data
creating basic visualization for, 170–171
data preparation, 168–170
data sharing issues, 167
egocentric network examples for, 167–168, 168f
ego/egoless visualization and, 170, 170f
exclusive origins of, 166
FQL query for, 169–170
friendship network as egocentric, 167–168
history of, 166–167
introduction to, 165–166
network types regarding, 168
networky look and, 170–171, 171f
news feed, 165–166, 166f
ordered/nonordered data/attributes and, 171–176
practitioner's summary, 179
privacy issues surrounding, 166–167
questions answered by analyzing, 166
researcher's agenda regarding, 179
REST query for, 169, 169f
roll-your-own queries for, 169–170
team players/brokers and, 167
why map, 167
Facebook Query Language (FQL), 169–170
FCC. *See* Federal Communications Commission lobbying coalition
Federal Communications Commission (FCC) lobbying coalition, 100–101, 100f
Filtering
analyzing personal email networks by, 115–116
categorical, 84
by controlling visibility, 87–88, 87f, 88f
creating living organization-chart and, 119–120
dynamic, 84–87, 85f, 86f, 87f
edges, 94–95, 95f
email and, 111–112, 112f, 118
for expertise network email graph, 118
Flickr user networks and, 218f, 219
introduction, 79, 82–90
makeup video network and, 239, 239f
ordinal, 84
practitioner's summary regarding, 90–91
putting it all together, 90, 90f
researcher's agenda and, 91
revealing landmarks/tourist attractions, 214
SeriousEats example and, 79–82
subgraph images and, 88–90, 89f
value-based, 82–83
visibility column options and, 88
who links to organization's web site and, 192, 193f
Financial transactions, 25–26
Fireball, 177–178, 178f
Flickr. *See also* Analyzing Flickr networks; Importing Flickr data; User networks, Flickr
comments/notes, 206–207
contacts page, 206
content management in, 202
data preparation advise, 221
exploration/validation regarding, 222
Explore page of, 202–204, 203f
favorites, 206–207
graph layouts and, 221–222
group management in, 209
groups, 204–206, 207f, 209
groups, public *vs.* private, 206
innovations of, 201–202
integration and, 222–223
introduction, 201–202
manual/multiple visualization layouts, 221–222
members search, 202, 204f
photo stream, 202, 204f
practitioner's summary regarding, 220–221
public access to, 202–204
reports, 207
researcher's agenda and, 222–223
sets/collections, 202–204
sharing/social interaction on, 206–207
social media and, 202–207
tag convergence, 207
tagging mechanism in, 202
tags, 202–204, 205f
upload, 202
vertices and, 218
what you can do with, 202, 204t
working with data from, 212
Flickr networks, 207–209. *See also* Analyzing Flickr networks; Importing Flickr data; User networks, Flickr
application sphere and, 209–210
community sphere and, 209
data preparation advise, 221
e-commerce and, 209–210
geo-tagged applications and, 210
graph layouts and, 221–222
implicit, 207–208
manual/multiple visualization layouts, 221–222
personal sphere and, 209
practitioner's summary regarding, 220–221
questions answered by analyzing, 209–210
related tags, 210–211, 210f
researcher's agenda and, 222–223
service/infrastructure and, 210
tag, 208–209, 208f
types of, 207, 208f
user comments and, 208f, 209
user contacts and, 208f, 209
working with data from, 212
Follow Friday, 148
FQL. *See* Facebook Query Language
Friend Wheel, 176–178
clustering/vertices and, 177
creating, 176
program, 176
radial layout used to create, 176, 176f
turned into fireball, 177–178, 178f
Fruchterman-Reingold layout, 96
Facebook and, 170–171, 171f
Full network, 36

G

Games
  hashtag, 147–148
  MMO, 27
  social, 27–28
Gartner Group, 4
Generic TLD, 190
Geo-tagged applications, 210
Governance structures, 16
Granovetter, Mark, 43
Granularity of control, 16, 16t
Graph. *See* Computing graph metrics; Expertise network email graph, creating; Subgraph images; *specific graph*
Graph theory, 35
GraphML, 253

H

Harvey, Leesha, 235–236, 236f, 237f
Hashtags
  games, 147–148
  information diffusion and, 151–152
  origin of, 147
  sign associated with, 147
  Twitter, 147–148
Healthcare reform video networks, 243–245
  central video in, 244
  clustering and, 244, 244f
  comments/ratings correlation, 244, 244f
  conclusions regarding, 245
  number of comments/views visualization, 243, 243f
  preliminary visualization of, 243, 243f
  questions addressed in analyzing, 243
Historical perspective, social networks, 3–4
History
  BBS, 129
  email, 106–107
  Facebook, 166–167
  network science, 3–4
  Project Chanology article history page, 254, 255f
  social network analysis and, 4, 5
  thread networks, 128–129
  Twitter, 143
  Wikipedia history feature, 249, 250f
  wikis, 248
  Youtube, 226
Hyperlink networks. *See also* Virtual Observatory for the Study of Online Networks Data Provider; What is network of this field/industry/sector?; Who links to my organization's web site?
  blogs and, 196–197
  boundaries analysis issues, 185
  challenges of analysis of, 182–185
  clustering and, 184
  constructing your own, 187–194
  context/meaning regarding, 185
  edges analysis issues, 184
  introduction to, 181–182
  linker attributes, 183
  methodological issues, 183–185
  network flow and, 196–197
  practitioner's summary regarding, 197–198
  purposes of, 183
  questions addressed in analyzing, 182
  researcher's agenda, 198
  static web presence and, 181–182
  temporal changes and, 196–197
  theory of, 182–183
  tools for analyzing, 185–186
  trust and, 183
  vertices analysis issues, 183–184

I

Idea generation, 26
Identifying self *vs.* others' comments
  in Flickr user networks, 217–218, 217f
  new columns in edges worksheet for, 217–218
  new columns in vertices worksheet for, 218
Identifying tag clusters
  cluster vertices for, 215, 216f
  to disambiguate meaning of tag, 215–216
  import data for, 215
  interpreting visualizations and, 215–216, 216f
  purpose of, 215
  raw visualization of, 215, 216f
IGraphDataProvider interface, 252–253
Implicit connections, 16–17
Importing
  computing graph metrics and, 74
  custom graph data, 248–251
  edge list, 55–57, 56f
  email and, 111–113, 112f, 118
  identifying tag clusters and, 215
  personal email networks and, 115
  revealing landmarks/tourist attractions and, 213
  Twitter trending topics and, 159, 160f
Importing Flickr data, 210–211
  egocentric network depth levels and, 210, 211f
  import menu for, 210, 210f
  raw data and, 211, 212f
  related tags network and, 210-211, 210f
  user networks and, 211, 211f, 212f, 217
  user networks options and, 211
  working with data after, 212
Importing Youtube data, 231–233
  ethical considerations, 233
  makeup video network and, 238
  preparation after, 233–234
  pre-prepared data importation, 233
  problems with, 233
  user importation, 232, 232f
  video importation, 231–232, 232f
Individual person-level graph metrics, 69
Instant messaging, 18–19

K

Kite network example, 70, 70f
Kleinberg, Jon, 45
Kutcher, Ashton, 144

L

Labels, 63
  alongside shape, 65
  edge, 65, 66f
  formulas to create, 65
  as shape, 64–65, 64f
  vertex, 63
Landmarks. *See* Revealing landmarks/tourist attractions
"Leona Lewis 'Bleeding Love' inspired makeup look" video, 242f
Les Misérables co-appearance network
  clustering of characters in, 98–100, 99f
  finding important individuals in, 75–77, 76f
  introduction, 74–78
  sorting edge weight column, 74–75, 75f
  viewing/hiding elements, 78, 78f
  visualizing edge weights, 75, 75f
  X/Y coordinate mapping, 77–78, 77f, 78f
Lewis, Leona. *See* "Leona Lewis 'Bleeding Love' inspired makeup look" video
Linker attributes, 183
Location, 31–32
  connections and, 17
  sharing, 27–28
  tag, Flickr, 213–215
Lostpedia, 264–268
  article pages and, 265
  basic community facts, 265
  creating overview content map of, 264–266, 266f
  creating users overview map of, 266–267, 267f
  data collection techniques for, 269
  Egyptian Goddess Taweret and, 264, 265f
  navigation of, 264
  normalization of data and, 268, 268f
  scope of, 264
  theory pages and, 265
  user talk pages and, 265

M

Makeup video network, 238–242
  "Beau Nelson's essential makeup tips" videos in, 242, 242f
  boundary objects and, 241
  clustering and, 240, 240f
  clustering conclusions, 240
  complex attributes visualized in, 242, 242f
  filtering and, 239, 239f
  importation of, 238
  influential user identification in, 241, 241f, 242
  initial visualization of, 238, 239f
  "Leona Lewis 'Bleeding Love' inspired makeup look" video in, 242f
  merging duplicate edges in, 238
  metrics and, 238–239
  "Natural makeup" video in, 241, 241f
  "Queen of Hearts" video in, 242, 242f
  tag sources within, 238
Massively multiplayer online games (MMOs), 27
Matrix, network represented as, 34–35, 35t
Merging duplicate edges, 80f, 81, 114–115, 114f, 212, 212f, 238
Metrics. *See also* Computing graph metrics; Network metrics; Vertex-specific networks metrics; *specific metrics*

makeup video network and, 238–239
personal email networks and, 116–118
visualizing ordered Facebook data via, 174–176
Microblogs, 23
Milgram, Stanley, 42
MMOs. *See* Massively multiplayer online games
Mobile Short Messaging Service (SMS), 19
Mobile-based services, 27–28
location sharing/annotation/games, 27–28
Moreno, Jacob, 38, 38f
Multimedia blogs, 23
Multimodal network, 36–37
naming vertices in, 84
transformed into unimodal, 80–81
working with data from, 80–81
Multiplex network, 37
Music, 24

N

Namespaces feature, Wikipedia, 249–251
Nation of Neighbors project, 6
National priorities
applying social media to, 6–7
communities and, 6
disaster response, 6
nongovernmental groups and, 7
public information dissemination and, 7
volunteerism and, 6–7
"Natural makeup" video
as bridge, 241, 241f
popularity of, 241, 241f
Nelson, Beau. *See* "Beau Nelson's essential makeup tips" video
Netviz nirvana, 47
Network flow, hyperlink networks and, 196–197
Network metrics. *See also* Computing graph metrics; Vertex-specific networks metrics
aggregate graph, 69
individual person-level, 69
introduction, 69–70
Kite example, 70, 70f
Network motifs, 41–42
Network Overview Discovery and Exploration add-in for Exel 2007 (NodeXL). *See also* Computing graph metrics; Filtering; Preparing data; Programmers, NodeXL for; *specific Part III Case Study*
cleaning email data in, 113–115
columns, autofilling, in, 61–62, 62f
columns, viewing hidden, in, 64, 64f
data entry in, 54, 55f
descriptive data, adding, in, 60–61, 60f
directed graph type in, 57, 58f
downloading/installing, 54
edge labels, adding, in, 65, 66f
file opening in, 70
formulas to create labels in, 65
getting started with, 54–57
graph appearance changes in, 63–64, 64f
highlighting edge in, 54, 56f
images as vertices, adding, in, 65
import menu, 210, 210f
importing edge list in, 55–57, 56f
importing email networks into, 111–113, 112f
importing Flickr data into, 210–211
introduction, 53–54
labeling in, 63
labels alongside shape, adding, in, 65
labels as shape, adding, in, 64–65, 64f
layout: arranging vertices in graph pane, 57–59
layout, automatic, in, 57, 57f
legend in, 63–66
manual layout in, 58f, 59
opening, 54, 55f
practitioner summary for, 66–67
researcher's agenda, 67
resizing/moving graph pane, 57
saving file in, 66
schemes tool in, 63
showing graph in, 54, 56f
tooltips, adding, in, 65, 66f
updating graph pane in, 59
VBA, use of, 153–154
vertex colors in, 59, 60f
vertex labels, adding, in, 63
vertex placement, fixing, in, 59
vertex size, changing, in, 61, 61f
vertex size advanced options in, 62f, 63
visual design in, 59–63
zoom/scale sliders in, 59
Network perspective, 32–35
Twitter example, 33–34, 33f
Network science, history of, 3–4
News, 24
Niche social networking services, 25
NodeXL. *See* Network Overview Discovery and Exploration add-in for Exel 2007
Nongovernmental groups, 7
Normalization
creating living organization-chart and, 119–120
Lostpedia and, 268, 268f
Noveck, B., 6

O

O'Neal, Shaquille, 144
Online markets and production, 25–26
financial transactions, 25–26
review sites, 26
user-generated products, 26
Open source software, 4
Ordinal filtering, 84
Overall Metrics worksheet, 73–74, 73f

P

Pace of interaction, 14–15
Page Co-editor Network, 254t
Page Link Network, 254t
Partial network, 36
Pattern discovery with Twitter, 154–163
ego network, 154–158
Photography, 24
Podcasts, 23
Polar layout, 177–178
Population
large, 11t, 14
media intention and, 13
medium, 11t, 14
producer/consumer, 13–14, 13t
small, 11t, 13–14
Practitioner landscape, 37–39
Practitioner's summary
of clustering, 101
of computing graph metrics, 78
on email, 124
of Facebook, 179
of filtering, 90–91
Flickr, 220–221
hyperlink networks, 197–198
NodeXL, 66–67
of social media network emergence, 8
social network analysis, 48–49
of technologies of collaboration, 28
of thread networks, 140
Twitter, 163
wiki networks, 269–270
Youtube, 245
Preparing data
auto-filling data columns, 81–82, 83f
customizing layout options, 82, 83f
introduction, 79
merging duplicate edges, 80f, 81
practitioner's summary regarding, 90–91
putting it all together, 90, 90f
researcher's agenda and, 91
SeriousEats example and, 79–82
shapes/colors identifying vertices, 81, 84
sorting vertices, 81, 82f
working with multimodal data, 80–81
Privacy issues, Facebook, 166–167
Professional social networking services, 25
Programmers, NodeXL for
controlling appearance of graph, 256–261
controlling layout, 256–257
custom graphing applications, 254–255
GraphML, 253
IGraphDataProvider interface, 252–253
importing custom graph data, 248–251
more features, 257–259
overview of, 247
populating graph, 255–256
sample code, 253–256, 259–261
TryGetGraphData (), 247
Project Chanology
article history page of, 254, 255f
data collection for, 254–256
defining edges/attributes in, 254–255
identifying editors within, 253–261
inactive/active Chanologists in, 259–260, 260f
interpreting graphs of, 257–259, 258f, 259f
sampling frame for, 254
seeing trees/forest in, 261
subgraph images clarifying, 259–261, 260f
type 1/type 2 sysops in, 259, 260f
vertex color gradients clarifying, 256–257, 258f
visualization strategies for, 256–261, 257f
what it is, 254
Project Network, 254t

Public information dissemination, 7
Push technology, email as, 107

Q

"Queen of Hearts" video, 242, 242f

R

Ravelry
  background on, 138–139
  benefits of analysis of, 140
  graph, 139–140, 139f
  understanding groups at, 138–140
Relational data, 32
Reply networks, 130, 130f
  ABC-D email list graph, 137, 137f
  technical support email list and, 135, 136f
  top-level, 130, 131f
Representational state transfer (REST)
  query for Facebook, 169, 169f
  Twitter and, 152–153
Research landscape, 37–39
Researcher's agenda
  clustering, 101
  computing graph metrics, 78
  email, 124
  Facebook and, 179
  filtering, 91
  Flickr and, 222–223
  hyperlink networks, 198
  NodeXL, 67
  social media network emergence, 8–9
  social network analysis, 49
  technologies of collaboration, 28
  thread networks, 140
  Twitter and, 164
  wiki networks, 270
  on Youtube, 245
REST. *See* Representational state transfer
Retention of content, 17
Retrievability, web site, 183
Retweeting, 148–149
  attributions in, 149
  example, 148, 148f
  information diffusion and, 151–152
  tracking, 149
  trending topics, in Twitter, and, 161, 162f
Revealing landmarks/tourist attractions, 213–215
  compute graph metrics/filter data for, 214
  importing data for, 213
  preparing data for, 213
  raw visualization of, 213, 213f
  semantic connections and, 213
  tag cloud graph interpretation for, 214–215, 214f, 215f
  tag cloud graph visualization for, 214, 214f
  using location tag, 213–215
Review sites, 26
Rihanna, 234, 234f, 235f
Rogers, Everett, 43–44
Roll-your-own queries, 169–170

S

Sampling frame
  article talk pages and, 261
  for project Chanology, 254
Sampson, S., 42
Schemes tool, 63
Science of sociotechnical systems (STSs), 8
Scientology. *See* Project Chanology
Search for Extraterrestrial Intelligence (SETI), 7
Self-loops, identifying, 115
Senate voting analysis
  automatically identifying clusters in, 95–96, 97f
  filtering edges in, 94–95, 95f
  manually creating clusters in, 97–98, 98f
  raw data/visualization of, 94–98, 94f
SeriousEats. *See* Filtering; Preparing data
SETI. *See* Search for Extraterrestrial Intelligence
Shared documents, as collaborative authoring, 22
ShopNPartyGirl. *See* Trending topics, in Twitter
Simmel, Georg, 38
Smart phone, 46
SMS. *See* Mobile Short Messaging Service
Social media
  analysis questions applied to, 47–48
  defined, 12
  examples, 15–26, 18t
  Flickr and, 202–207
  history regarding, 4
  individual contributions in, 5
  individuals/groups who study, 5–6, 8
  introduction, 3
  national priorities and, 6–7
  practitioner's summary, 8
  researcher's agenda, 8–9
  rise of, as consumer applications, 4
  worldwide efforts concerning, 7–8
Social networking services, 24–25
  niche, 25
  professional, 25
  social/dating, 25
Social networks
  analysis, 4
  era of abundant, 46
  historical perspective, 3–4
  individual contributions in, 5
  individuals/groups who study, 5–6, 8
  introduction, 3
  types of, 36–37
Social roles, 41–42
Social sharing, 23–24
  bookmarks/news/books, 24
  music, 24
  photo/art, 24
  video/TV, 24
Social (users' channels) sublayer, Youtube, 229
Social virtual worlds, 26–27
Sociotechnical infrastructure, design influence of, 11–12
Static web presence, 181–182
Strogatz, Steven, 45
Structures, 41–42
STSs. *See* Science of sociotechnical systems
Subgraph images
  analyzing technical support email list, 133, 134f
  clarifying, 259–261, 260f
  creating, 88–90, 89f
Symmetric connections, 151
Synchronous communications, 14
  audio conferencing, 21
  videoconferencing, 21
Synchronous conversations, 18
  chat, 18
  instant messaging, 18–19
  texting, 19
Sysops, 259, 260f

T

Tags. *See also* Hashtags; Identifying tag clusters
  makeup video network, 238
Tags, Flickr, 202–204, 205f, 208–209, 208f. *See also* Identifying tag clusters
  conceptual relationship and, 208
  convergence, 207
  geo-tagged applications and, 210
  identifying clusters of, 215–216
  mechanism, 202
  networks, 208–209, 208f
  related, 210–211, 210f
  tag cloud graph interpretation, 214–215, 214f, 215f
  tag cloud graph visualization, 214, 214f
  using location, 213–215
Talk pages. *See* Article talk pages
Taweret, 264, 265f
TechABC. *See* Creating living organization-chart
Technologies of collaboration
  asynchronous threaded conversation, 17, 18t
  basic elements, control of, and, 15–16
  basic elements, genre of, and, 15
  design framework and, 12–17
  examples, 15, 18t
  introduction, 11–12
  mobile, 46
  pace of interaction affecting, 14–15
  practitioner's summary of, 28
  producer/consumer population size and, 13–14, 13t
  researcher's agenda regarding, 28
  retention of content and, 17
  sociotechnical infrastructure and, 11–12
  synchronous conversations, 20
  types of connections and, 16–17
Television (TV), 24
Temporal changes, hyperlink networks and, 196–197
Texting, 19
Theory pages, Lostpedia, 265
Thread networks, 130–132. *See also* ABC-D email list; Analyzing technical support email list; Ravelry
  affiliation, 130–131
  analysis questions applied to, 129–130
  data preparation for analysis of, 131
  differences in, 128
  history surrounding, 128–129
  introduction, 127–128
  key properties of, 128, 128f

practitioner's summary regarding, 140
reply, 130, 130f
research on, 129
researcher's agenda and, 140
top-level reply, 130, 131f
TLD. *See* Top-level domain
Tomlinson, Ray, 106–107
Topic drift, 192
Top-level domain (TLD), 190
Top-level reply networks, 130, 131f
Tourist attractions. *See* Revealing landmarks/tourist attractions
Trending topics, in Twitter, 158–163
import dialog and, 159, 160f
information diffusion via, 159
mentions network and, 160, 161f, 162f, 163f
overview of, 158–159
retweeting and, 161, 162f
search phrase criteria, 159
TryGetGraphData (), 247
TV. *See* Television (TV)
TweetDeck, 144, 145f
Twhirl, 144, 146f
Twitter. *See also* Acquiring Twitter data; Pattern discovery with Twitter; Retweeting; Trending topics, in Twitter
@ replies/mentions, 146–147, 147f, 147t
attention/importance/eigenvector centrality and, 150
betweenness centrality and, 150–151, 150f
blogs compared with, 144, 146t
follow Friday, 148
friends/followers and, 149–150, 149f
growth of, 144, 144f
hashtags, 147–148
history of, 143
information diffusion in, 151–152
interfaces for, 144, 145f, 146f
introduction, 143–144
Kutcher and, 144
message size, 144
network example, 33–34, 33f
networks in, 149–152
nuts/bolts of, 144–149
O'Neal and, 144
practitioner's summary, 163
researcher's agenda and, 164
symmetric connections and, 151
third-party programs, 144

U

Unimodal network, 36–37, 80–81
Usenet newsgroups
as asynchronous threaded conversation, 17
history surrounding, 129
User accounts feature, Wikipedia, 251, 252f
User networks, Flickr, 209, 216–220
analyzing, 212f
combining data from another network, 220
data preparation for, 217–218
filter data/set visual properties for, 218f, 219
get user comments information in, 219
identifying self *vs.* others' comments, 217–218, 217f
importing, 211, 211f, 212f, 217
interpreting visualizations of, 220
options, 211
raw visualization of comments network in, 217f
visualize scatter graph for, 219–220, 220f
visualize user comments graph for, 218f, 219
User networks, Wiki, 254t
User networks, Youtube
analyzing, 234–238
attributes, 230, 230f
data importation, 232, 232f
Harvey and, 235–236, 236f, 237f
questions for analysis, 231
Rihanna and, 234, 234f, 235f
social sublayer of, 229
value of, 234
User talk pages, Lostpedia, 265
User-generated products, 26

V

Value-based filtering, 82–83
VBA. *See* Visual Basic macro
Vertex-specific networks metrics, 40–41. *See also* Les Misérables co-appearance network; *specific metric*
betweenness centrality, 40
closeness centrality, 40–41
clustering coefficient, 41
computing, 71–72, 71f
degree centrality, 40–41
Eigenvector centrality, 41
Vertices, 34
aligning, 158
article talk pages and, 262, 263f
categories manually attached to attributes of, 172–173, 173f
color gradients menu, 256–257, 258f
color matching visualization, 190, 191f
Flickr comments and, 218
Friend Wheel and, 177
Fruchterman-Reingold layout and, 96
hyperlink networks analysis issues and, 183–184
shapes/colors identifying, 59, 60f, 81, 84
sorting, 81, 82f
Wiki networks and, 252–253
Video. *See also* Healthcare reform video networks; Makeup video network
conferencing, 21
data importation, 231–232, 232f
network analysis, 238–245
network attributes, 229–230, 230f
network questions for analysis, 230–231
as social sharing, 24
sublayer of Youtube, 227–229
Viral videos, 226
Virtual Observatory for the Study of Online Networks (VOSON) Data Provider. *See also* Create VOSDON database; What is network of this field/industry/sector?; Who links to my organization's web site?
constructing network using, 187–194
example 1, 186–194
example 2, 194–196
overview of, 185–186
practical application of, 194
preliminaries for using, 186–187, 186f, 187f
select network dialog and, 186–187, 187f
steps for using, 187
Virtual worlds, 26–27
MMOs, 27
social, 26–27
Visibility
column options, 88
web site, 183
Visual Basic macro (VBA), 153–154
Visualizing nonordered Facebook data, 171–173
categories and, 172, 172f
categories manually attached to vertex attributes, 172–173, 173f
clusters and, 171, 171f
data/attributes and, 171–176
Visualizing ordered Facebook data
betweenness altered manually, 175
betweenness for, 174–176
data/attributes and, 171–176
degree for, 174
metrics for, 174–176
techniques combined, 175–176, 175f
VLOOKUP function, 113–114, 114f
Volunteerism, 6–7
VOSON. *See* Virtual Observatory for the Study of Online Networks Data Provider

W

Watts, Duncan, 45
Web 1.0. *See* Hyperlink networks
Web Science Trust, 8
Weighted/unweighted
connections, 17
edges, 34
Wellman, Barry, 40
What is network of this field/industry/sector?
betweenness centrality top three sites, 196
conclusions regarding, 196
data collection parameters for, 194, 194f
filtered visualization of, 194–195, 195f
final visualization of, 195, 195f
in-degree top five sites, 195
out-degree top five sites, 195–196
practicality of answering, 194
Whitelisting, 152–153, 153
Who links to my organization's web site?, 186–194
basic visualization of, 189–190, 190f
conclusions, 194
constructing network to answer, 187–194
edges worksheet and, 190–191
filtering and, 192, 193f
information rich visualization of, 192–193, 193f
introduction, 186
preliminaries, 186–187, 186f, 187f
seed site promotion visualization and, 191–192, 192f
topic drift and, 192
vertex color matching visualization of, 190, 191f

Wiki networks. *See also* Article talk pages; Encyclopedia Dramatica; Lostpedia; Project Chanology
analysis issues surrounding, 248, 252–253
attributes in, 253
data challenges of, 248
data collection techniques for, 269
derived from edit records, 253, 254t
edges in, 252–253
from edit activity, 252–253
of general interest, 253
introduction to, 248–249
practitioner's summary regarding, 269–270
researcher's agenda for, 270
vertices in, 252
Wikipedia. *See also* Lostpedia
discussion feature of, 249–251, 251f
everything is a page feature of, 251, 252f
history feature of, 249, 250f
key features of, 249–251
namespaces feature of, 249–251
popularity of, 248
typical page on, 249, 250f
user accounts feature of, 251, 252f
Wikis
as collaborative authoring, 22
data challenges of, 248
data collection techniques for, 269
flexibility of, 248
history surrounding, 248
introduction to, 248–249
key features of, 249–251
practitioner's summary regarding, 269–270
researcher's agenda for, 270
social media history and, 4
what are, 248
Windows Search utility, 110, 111f
World Wide Web, 21–22

X

X/Y coordinate mapping, 77–78, 77f, 78f

Y

Youtube. *See also* Analyzing Youtube; Healthcare reform video networks; Importing Youtube data; Makeup video network; User networks, Youtube; Video
bulletin board, 229
changing landscape of, 227
channel page layout, 228f, 229
content networks and, 226
content page layout of, 227, 227f
ease of use of, 226
friends and, 229
history of, 226
introduction, 225–226
networks in, 229–230, 230f
popularity of, 226
practitioners summary on, 245
purpose of, 227
questions for analysis of, 230–231
researcher's agenda, 245
social affinity networks and, 226–227
social (users' channels) sublayer of, 229
structure of, 227–229
subscribers and, 229
users' networks attributes, 230, 230f
video networks attributes, 229–230, 230f
video (content) sublayer of, 227–229
what is, 226–227

Z

Zoom/scale sliders, 59