Preface

The world is driven by connections—from financial and communication systems to social and biological processes. Revealing the meaning behind these connections drives breakthroughs across industries in areas such as identifying fraud rings and optimizing recommendations to evaluating the strength of a group and predicting cascading failures.

As connectedness continues to accelerate, it's not surprising that interest in graph algorithms has exploded because they are based on mathematics explicitly developed to gain insights from the relationships between data. Graph analytics can uncover the workings of intricate systems and networks at massive scales—for any organization.

We are passionate about the utility and importance of graph analytics as well as the joy of uncovering the inner workings of complex scenarios. Until recently, adopting graph analytics required significant expertise and determination, because tools and integrations were difficult and few knew how to apply graph algorithms to their quandaries. It is our goal to help change this. We wrote this book to help organizations better leverage graph analytics so that they can make new discoveries and develop intelligent solutions faster.

What's in This Book

This book is a practical guide to getting started with graph algorithms for developers and data scientists who have experience using Apache Spark™ or Neo4j. Although our algorithm examples utilize the Spark and Neo4j platforms, this book will also be helpful for understanding more general graph concepts, regardless of your choice of graph technologies.

The first two chapters provide an introduction to graph analytics, algorithms, and theory. The third chapter briefly covers the platforms used in this book before we dive into three chapters focusing on classic graph algorithms: pathfinding, centrality, and community detection. We wrap up the book with two chapters showing how

graph algorithms are used within workflows: one for general analysis and one for machine learning.

At the beginning of each category of algorithms, there is a reference table to help you quickly jump to the relevant algorithm. For each algorithm, you'll find:

- An explanation of what the algorithm does
- Use cases for the algorithm and references to where you can learn more
- Example code providing concrete ways to use the algorithm in Spark, Neo4j, or both

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic
> Indicates new terms, URLs, email addresses, filenames, and file extensions.

`Constant width`
> Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements, and keywords.

`Constant width bold`
> Shows commands or other text that should be typed literally by the user.

`Constant width italic`
> Shows text that should be replaced with user-supplied values or by values determined by context.

> This element signifies a tip or suggestion.

> This element signifies a general note.

This element indicates a warning or caution.

Using Code Examples

Supplemental material (code examples, exercises, etc.) is available for download at *https://bit.ly/2FPgGVV*.

This book is here to help you get your job done. In general, if example code is offered with this book, you may use it in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Graph Algorithms* by Amy E. Hodler and Mark Needham (O'Reilly). Copyright 2019 Amy E. Hodler and Mark Needham, 978-1-492-05781-9."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

O'Reilly Online Learning

For almost 40 years, O'Reilly has provided technology and business training, knowledge, and insight to help companies succeed.

Our unique network of experts and innovators share their knowledge and expertise through books, articles, conferences, and our online learning platform. O'Reilly's online learning platform gives you on-demand access to live training courses, in-depth learning paths, interactive coding environments, and a vast collection of text and video from O'Reilly and 200+ other publishers. For more information, please visit *http://oreilly.com*.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at *http://bit.ly/graph-algorithms*.

To comment or ask technical questions about this book, send email to *bookquestions@oreilly.com*.

For more information about our books, courses, conferences, and news, see our website at *http://www.oreilly.com*.

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

Acknowledgments

We've thoroughly enjoyed putting together the material for this book and thank all those who assisted. We'd especially like to thank Michael Hunger for his guidance, Jim Webber for his invaluable edits, and Tomaz Bratanic for his keen research. Finally, we greatly appreciate Yelp permitting us to use its rich dataset for powerful examples.