Recommendation Systems in Software Engineering

Martin P. Robillard • Walid Maalej •
Robert J. Walker • Thomas Zimmermann
Editors

Recommendation Systems in Software Engineering

Springer

Editors

Martin P. Robillard
McGill University
Montréal, QC
Canada

Robert J. Walker
University of Calgary
Calgary, AB
Canada

Walid Maalej
University of Hamburg
Hamburg
Germany

Thomas Zimmermann
Microsoft Research
Redmond, WA
USA

ISBN 978-3-642-45134-8 ISBN 978-3-642-45135-5 (eBook)
DOI 10.1007/978-3-642-45135-5
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014931203

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Software developers have always used tools to perform their work. In the earliest days of the discipline, the tools provided basic compilation and assembly functionality. Then came tools and environments that increasingly provided sophisticated data about the software under development. Around the turn of the millennium, the systematic and large-scale accumulation of software engineering data opened up new opportunities for the creation of tools that infer information estimated to be helpful to developers in a given context. This new type of software development tools came to be known as recommendation systems, in parallel with similar developments in other domains such as the e-commerce.

Recommendation systems in software engineering (RSSEs) share commonalities with conventional recommendation systems: mainly in their usage model, the usual reliance on data mining, and in the predictive nature of their functionality. Beyond these superficial traits, recommendation systems in software engineering are generally different from those in other domains. Traditional recommendation systems are heavily *user centric*. Users generally create the data items directly, e.g., in the form of ratings. An important challenge for traditional recommendation systems is to infer and model evolving user preferences and needs. In contrast, the major challenge for designing RSSEs is to automatically interpret the highly technical data stored in software repositories.

Realizing that some of the important knowledge that is necessary to build recommendation systems in a technical domain would not be readily found in existing books and other resources on conventional recommendation systems, we set about to capture as much of this knowledge as possible in this book.

About This Book

This book has been a community effort. Prospective authors submitted chapter proposals to an open call for contributions. The proposals and later the selected chapters were reviewed by the editors over four review iterations. In addition, the authors participating in this book were asked to review chapters by other contributors.

A unique aspect of this book was the RSSE Hamburg Meeting in April 2013. The contributing authors were invited to this 2-day event to present their chapter ideas, discuss the RSSE state of the art, and participate in editing and networking sessions. The meeting greatly helped to unify the presentation and content of this book and to further consolidate the RSSE community effort. The meeting has been part of a series of events that started with a workshop on software analysis for recommendation systems at McGill University's Bellairs Research Station in Barbados in 2008 and follow-up workshops at the ACM SIGSOFT International Symposium on the Foundations of Software Engineering in 2008 and at the ACM/IEEE International Conference on Software Engineering in 2010 and 2012. The last workshop in 2012 had over 70 participants, which shows a large interest in the topic.

Structure and Content

This book collects, structures, and formalizes knowledge on recommendation systems in software engineering. It adopts a pragmatic approach with an explicit focus on system design, implementation, and evaluation. The book is intended to complement existing texts on recommender systems, which cover algorithms and traditional application domains.

The book consists of three parts:

Part I: Techniques This part introduces basic techniques for building recommenders in software engineering, including techniques not only to collect and process software engineering data but also to present recommendations to users as part of their workflow.

Part II: Evaluation This part summarizes methods and experimental designs to evaluate recommendations in software engineering.

Part III: Applications This part describes needs, issues, and solution concepts involved in entire recommendation systems for specific software engineering tasks, focusing on the engineering insights required to make effective recommendations.

Target Audience

The book contains knowledge relevant to software professionals and to computer science or software engineering students with an interest in the application of recommendation technologies to highly technical domains, including:

- senior undergraduate and graduate students working on recommendation systems or taking a course in software engineering or applied data mining;
- researchers working on recommendation systems or on software engineering tools;
- software engineering practitioners developing recommendation systems or similar applications with predictive functionality; and
- instructors teaching a course on recommendation systems, applied data mining, or software engineering. The book will be particularly suited to graduate courses involving a project component.

Website and Resources

This book has a webpage at rsse.org/book, which is part of the RSSE community portal rsse.org. This webpage contains free supplemental materials for readers of this book and anyone interested in recommendation systems in software engineering, including:

- lecture slides, datasets, and source code;
- an archive of previous RSSE workshops and meetings;
- a collection of people, papers, groups, and tools related to RSSE. Please contact any of the editors if you would like to be added or to suggest additional resources.

In addition to the RSSE community, there are several other starting points.

- The article "Recommendation Systems for Software Engineering," IEEE Software, 27(4):80–86, July–August 2010, provides a short introduction to the topic.
- The latest research on RSSE systems is regularly published and presented at the International Conference on Software Engineering (ICSE), International Symposium on the Foundations of Software Engineering (FSE), International Conference on Automated Software Engineering (ASE), Working Conference on Mining Software Repositories (MSR), and International Conference on Software Maintenance (ICSM).
- Many researchers working on RSSE systems meet at the International Workshop on Recommendation Systems for Software Engineering, which is typically held every other year.
- The ACM Conference on Recommender Systems (RecSys) covers recommender research in general and in many different application domains, not just software engineering.
- Several books on building conventional recommendation systems have been written. To get started, we recommend "Recommender Systems: An Introduction" (2010) by Jannach, Zanker, Felfernig, and Friedrich.

Acknowledgments

We are indebted to the many people who have made this book possible through their diverse contributions. In particular, we thank:

- The authors of the chapters in this book for the great work that they have done in writing about topics related to RSSEs and their timely and constructive reviews of other chapters.
- The Springer staff, in particular Ralf Gerstner, for their dedication and helpfulness throughout the project.
- The Dean of the MIN faculty and Head of Informatics Department at the University of Hamburg, for their financial and logistical support for the editing workshop, and Rebecca Tiarks and Tobias Roehm, for their help with the local organization of the workshop.
- The attendees at the past RSSE workshops, for their enthusiasm about the topic and their influx of ideas.
- The McGill Bellairs Research Institute, for providing an ideal venue for an initial workshop on the topic.

Montréal, QC, Canada
Hamburg, Germany
Calgary, AB, Canada
Redmond, WA, USA
October 2013

Martin P. Robillard
Walid Maalej
Robert J. Walker
Thomas Zimmermann

Contents

Part I Techniques

Part II Evaluation

Part III Applications

List of Contributors

Colin Atkinson Software-Engineering Group, University of Mannheim, Mannheim, Germany

Iman Avazpour Faculty of ICT, Centre for Computing and Engineering Software and Systems (SUCCESS), Swinburne University of Technology, Hawthorn, Australia

Gabriele Bavota University of Sannio, Benevento, Italy

Ayşe Bener Ryerson University, Toronto, ON, Canada

Markus Borg Department of Computer Science, Lund University, Lund, Sweden

Bora Çağlayan Boğaziçi University, Istanbul, Turkey

Gül Çalıklı Ryerson University, Toronto, ON, Canada

Carlos Castro-Herrera GOOGLE, Chicago, IL, USA

Jane Cleland-Huang School of Computing, DePaul University, Chicago, IL, USA

Paolo Cremonesi Politecnico di Milano, Milano, Italy

Andrea De Lucia University of Salerno, Fisciano, Italy

Daniel Diaz Université Paris 1 Panthéon-Sorbonne, Paris, France

Cosmin Dumitrescu Université Paris 1 Panthéon-Sorbonne, Paris, France

Alexander Felfernig Institute for Software Technology, Graz University of Technology, Graz, Austria

Thomas Fritz Department of Informatics, University of Zurich, Zurich, Switzerland

John Grundy Faculty of ICT, Centre for Computing and Engineering Software and Systems (SUCCESS), Swinburne University of Technology, Hawthorn, Australia

Lars Grunske Institute of Software Technology, Universität Stuttgart, Stuttgart, Germany

Negar Hariri School of Computing, DePaul University, Chicago, IL, USA

Kim Herzig Saarland University, Saarbrücken, Germany

Reid Holmes David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada

Oliver Hummel Institute for Program Structures and Data Organization, Karlsruhe Institute of Technology, Karlsruhe, Germany

Laura Inozemtseva David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada

Werner Janjic Software-Engineering Group, University of Mannheim, Mannheim, Germany

Michael Jeran Institute for Software Technology, Graz University of Technology, Graz, Austria

Miryung Kim The University of Texas at Austin, Austin, TX, USA

Angela Lozano ICTEAM, Université catholique de Louvain, Louvain-la-Neuve, Belgium

Walid Maalej Department of Informatics, University of Hamburg, Hamburg, Germany

Andrian Marcus Wayne State University, Detroit, MI, USA

Raúl Mazo Université Paris 1 Panthéon-Sorbonne, Paris, France

Na Meng The University of Texas at Austin, Austin, TX, USA

Kim Mens ICTEAM, Université Catholique de Louvain, Louvain-la-Neuve, Belgium

Tim Menzies Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV, USA

Ayşe Tosun Mısırlı University of Oulu, Oulu, Finland

Bamshad Mobasher School of Computing, DePaul University, Chicago, IL, USA

Gail C. Murphy University of British Columbia, Vancouver, BC, Canada

Emerson Murphy-Hill North Carolina State University, Raleigh, NC, USA

Gerald Ninaus Institute for Software Technology, Graz University of Technology, Graz, Austria

Rocco Oliveto University of Molise, Pesche, Italy

Teerat Pitakrat Institute of Software Technology, Universität Stuttgart, Stuttgart, Germany

Florian Reinfrank Institute for Software Technology, Graz University of Technology, Graz, Austria

Stefan Reiterer Institute for Software Technology, Graz University of Technology, Graz, Austria

Romain Robbes Computer Science Department (DCC), University of Chile, Santiago, Chile

Martin P. Robillard McGill University, Montréal, QC, Canada

Per Runeson Department of Computer Science, Lund University, Lund, Sweden

Alan Said Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

Camille Salinesi Université Paris 1 Panthéon-Sorbonne, Paris, France

Martin Stettinger Institute for Software Technology, Graz University of Technology, Graz, Austria

Domonkos Tikk Gravity R&D, Budapest, Hungary

Óbuda University, Budapest, Hungary

Burak Turhan University of Oulu, Oulu, Finland

Robert J. Walker Department of Computer Science, University of Calgary, Calgary, AB, Canada

Annie T.T. Ying McGill University, Montréal, QC, Canada

Andreas Zeller Saarland University, Saarbrücken, Germany