Software Analytics: Achievements and Challenges

Dongmei Zhang
Microsoft Research Asia
Beijing, 100080, China
Email: dongmeiz@microsoft.com

Tao Xie
North Carolina State University
Raleigh, NC 27695, USA
Email: xie@csc.ncsu.edu

Abstract—A huge wealth of various data exist in the practice of software development. Further rich data are produced by modern software and services in operation, many of which tend to be data-driven and/or data-producing in nature. Hidden in the data is information about the quality of software and services or the dynamics of software development. Software analytics is to utilize a data-driven approach to enable software practitioners to perform data exploration and analysis in order to obtain insightful and actionable information; such information is used for completing various tasks around software systems, software users, and software development process.

This tutorial presents achievements and challenges of research and practice on principles, techniques, and applications of software analytics, highlighting success stories in industry, research achievements that are transferred to industrial practice, and future research and practice directions in software analytics.

I. INTRODUCTION

Software analytics [7], [8] is to utilize a data-driven approach to enable software practitioners to perform data exploration and analysis in order to obtain insightful and actionable information; such information is used for completing various tasks around software systems, software users, and software development process.

Data under exploration and analysis by software analytics exist in software lifecycle, including source code, software requirements specifications, bug reports, test cases, execution traces/logs, and real-world user feedback, etc. Data play a critical role in modern software development, because hidden in the data is the information and insight about the quality of software and services, the experience that software users receive, as well as the dynamics of software development.

Insightful information obtained by software analytics is information that conveys meaningful and useful understanding or knowledge towards performing the target task. Typically insightful information cannot be easily obtained by direct investigation on the raw data without the aid of analytic technologies.

Actionable information obtained by software analytics is information upon which software practitioners can come up with concrete solutions (better than existing solutions if any) towards completing the target task.

This tutorial presents achievements and challenges in software analytics, to allow the attendees to acquire the skills and knowledge needed to perform research or conduct industrial practice in the field of software analytics and to integrate analytics in their own industrial research, practice, or teaching.

The reasons why the topic is timely and relevant are primarily two folds. First, a huge wealth of various data exist in the practice of software development. Further rich data are produced by modern software and services in operation, many of which tend to be data-driven and/or data-producing in nature. There are huge opportunities for exploiting the data to improve software engineering practice. Second, with rapid progress of the field of Mining Software Repositories (MSR), a variety of effective analytic technologies have been developed for MSR. The community has reached a good timing to explore how to effectively apply analytic technologies in software engineering practice.

II. TARGET AUDIENCE

The tutorial is targeted at researchers and practitioners who have interest in software analytics. Researchers are expected to learn the state of the art in software analytics, including research achievements and research challenges. Practitioners are expected to learn the state of the practice in applying software analytics on software data in practice. The tutorial presents a number of successful research projects [2], [1], [6], [3], [5], [4] on software analytics along with the lessons learned and process of developing such research projects. The tutorial also demonstrates various practical applications of software analytics, so that practitioners could apply the learned skills and knowledge in their industrial environments.

REFERENCES

[1] R. P. L. Buse and T. Zimmermann. Information needs for software development analytics. In *Proc. ICSE, SEIP*, pages 987–996, 2012.

[2] J. Czerwonka, R. Das, N. Nagappan, A. Tarvo, and A. Teterev. CRANE: Failure prediction, change analysis and test prioritization in practice - experiences from Windows. In *Proc. ICST*, pages 357–366, 2011.

[3] Y. Dang, D. Zhang, S. Ge, C. Chu, Y. Qiu, and T. Xie. XIAO: Tuning code clones at hands of engineers in practice. In *Proc. ACSAC*, pages 369–378, 2012.

[4] R. Ding, Q. Fu, J.-G. Lou, Q. Lin, D. Zhang, J. Shen, and T. Xie. Healing online service systems via mining historical issue repositories. In *Proc. ASE*, pages 318–321, 2012.

[5] Q. Fu, J.-G. Lou, Q.-W. Lin, R. Ding, Z. Ye, D. Zhang, and T. Xie. Performance issue diagnosis for online service systems. In *Proc. SRDS*, pages 273–278, 2012.

[6] S. Han, Y. Dang, S. Ge, D. Zhang, and T. Xie. Performance debugging in the large via mining millions of stack traces. In *Proc. ICSE*, pages 145–155, 2012.

[7] D. Zhang, Y. Dang, J.-G. Lou, S. Han, H. Zhang, and T. Xie. Software analytics as a learning case in practice: Approaches and experiences. In *Proc. MALETS*, 2011.

[8] D. Zhang and T. Xie. Software analytics in practice: Mini tutorial. In *Proc. ICSE, SEIP, Mini Tutorial*, page 997, 2012.