

# Chapter 21

## Data Mining Techniques for Software Quality Prediction

**Bharavi Mishra**

*Indian Institute of Technology (BHU), India*

**K. K. Shukla**

*Indian Institute of Technology (BHU), India*

### ABSTRACT

*In the present time, software plays a vital role in business, governance, and society in general, so a continuous improvement of software productivity and quality such as reliability, robustness, etc. is an important goal of software engineering. During software development, a large amount of data is produced, such as software attribute repositories and program execution trace, which may help in future development and project management activities. Effective software development needs quantification, measurement, and modelling of previous software artefacts. The development of large and complex software systems is a formidable challenge which requires some additional activities to support software development and project management processes. In this scenario, data mining can provide a helpful hand in the software development process. This chapter discusses the application of data mining in software engineering and includes static and dynamic defect detection, clone detection, maintenance, etc. It provides a way to understand the software artifacts and processes to assist in software engineering tasks.*

### INTRODUCTION

Software engineering is a complex process which has become a prominent human activity at present. Considerable amount of knowledge about the problem domain and the programming domain are needed during the software-development

life cycle. Including this, different techniques are also required to combine this knowledge to provide reliable and robust software solutions. Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; (IEEE, 1987) that is, the application of engineering to software. Software engineering is often thought of as a series of separate, discrete

DOI: 10.4018/978-1-4666-4301-7.ch021

activities (such as design, coding, testing) that lead to a finished product. However, quality software is not composed of discrete processes; instead, it is composed of continuous processes that guide the development activity. Most of these activities are continuous; that is, the activities are performed throughout the entire software-development effort. Some of these activities, such as analysis, design, implementation, and testing are discrete. Selection of the correct life cycle is therefore, extremely important to the success of the overall software project. To support software development process, there is a need of different techniques to ensure a cost-effective and reliable software development with risk minimization.

Data Mining (sometimes called data or knowledge discovery or Knowledge Mining) is the process of analyzing data from different perspectives and summarizing it into useful information that can be used for strategic planning. Data mining techniques allow users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large databases. Data Mining is a series of iterative activities that lead to identification of some interesting patterns in data set. Step by step data mining process is shown in Figure 1. Data mining techniques are now being used by the practitioners to solve several software related problem to ease the task of software development

(Basili, 1996;Jing, 2007; Binkley, 1998; Mishra, 2011, December; Mishra, Sep 2011; Menzies, 2007; Zimmermann, 2007).

## NEED OF DATA MINING IN SOFTWARE ENGINEERING

In the pursuit of good software, engineers have collected huge amount of data in various forms, which can be analyzed to produce better quality software (as it helps in having better comprehension of the software-development process). In software development huge amount of data is produced which can be categorized as:

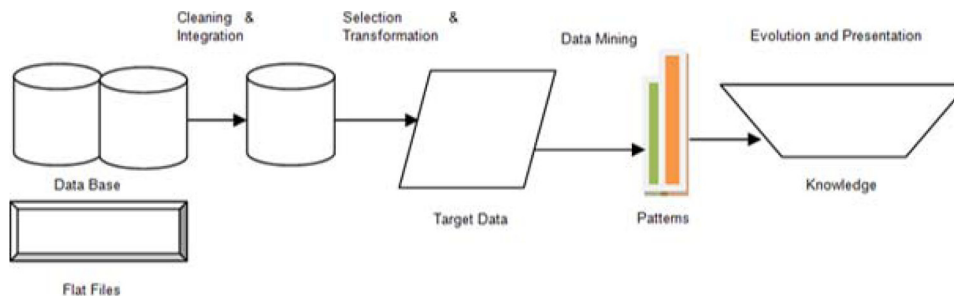
- Data from software repositories
- Data from program executions

Software engineering data have a wealth of information about a software project and processes which includes:

- **Programming:** Versions of programs
- **Testing:** Execution traces
- **Deployment:** Error/bug reports
- **Reuse:** Open source packages

We can explore valuable information regarding software projects and processes to provide higher-quality software within a reasonable time and budget

Figure 1. Step by step data mining process



26 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:  
[www.igi-global.com/chapter/data-mining-techniques-software-quality/77716](http://www.igi-global.com/chapter/data-mining-techniques-software-quality/77716)

## Related Content

---

### Model Based Process to Support Security and Privacy Requirements Engineering

Shareeful Islam, Haralambos Mouratidis, Christos Kalloniatis, Aleksandar Hudicand Lorenz Zechner (2012). *International Journal of Secure Software Engineering* (pp. 1-22).

[www.irma-international.org/article/model-based-process-support-security/69391](http://www.irma-international.org/article/model-based-process-support-security/69391)

### Description, Classification and Discovery Approachesfor Software Components: A Comparative Study

Sofien Khemakhem, Khalil Driraand Mohamed Jmaiel (2011). *Modern Software Engineering Concepts and Practices: Advanced Approaches* (pp. 196-219).

[www.irma-international.org/chapter/description-classification-discovery-approachesfor-software/51973](http://www.irma-international.org/chapter/description-classification-discovery-approachesfor-software/51973)

### A Framework for Understanding the Open Source Revolution

Jeff Elpermand Sergiu Dascalu (2009). *Designing Software-Intensive Systems: Methods and Principles* (pp. 439-454).

[www.irma-international.org/chapter/framework-understanding-open-source-revolution/8244](http://www.irma-international.org/chapter/framework-understanding-open-source-revolution/8244)

### Petri Net Based Deadlock Prevention Approach for Flexible Manufacturing Systems

Chunfu Zhongand Zhiwu Li (2011). *Reconfigurable Embedded Control Systems: Applications for Flexibility and Agility* (pp. 416-433).

[www.irma-international.org/chapter/petri-net-based-deadlock-prevention/50437](http://www.irma-international.org/chapter/petri-net-based-deadlock-prevention/50437)

### A Visual Approach to Business IT Alignment between Business Model and Enterprise Architecture

Boris Fritscherand Yves Pigneur (2015). *International Journal of Information System Modeling and Design* (pp. 1-23).

[www.irma-international.org/article/a-visual-approach-to-business-it-alignment-between-business-model-and-enterprise-architecture/123605](http://www.irma-international.org/article/a-visual-approach-to-business-it-alignment-between-business-model-and-enterprise-architecture/123605)