Chapter 24
Building Defect Prediction Models in Practice

Rudolf Ramler
Software Competence Center Hagenberg, Austria

Johannes Himmelbauer
Software Competence Center Hagenberg, Austria

Thomas Natschläger
Software Competence Center Hagenberg, Austria

ABSTRACT

The information about which modules of a future version of a software system will be defect-prone is a valuable planning aid for quality managers and testers. Defect prediction promises to indicate these defect-prone modules. In this chapter, building a defect prediction model from data is characterized as an instance of a data-mining task, and key questions and consequences arising when establishing defect prediction in a large software development project are discussed. Special emphasis is put on discussions on how to choose a learning algorithm, select features from different data sources, deal with noise and data quality issues, as well as model evaluation for evolving systems. These discussions are accompanied by insights and experiences gained by projects on data mining and defect prediction in the context of large software systems conducted by the authors over the last couple of years. One of these projects has been selected to serve as an illustrative use case throughout the chapter.

INTRODUCTION

Knowing which modules are likely to contain defects in advance can assist in directing software quality assurance measures such as inspection and testing to these potentially critical modules. With defect prediction, the defect-prone modules in an upcoming version of a software system can be predicted from data about previous versions. Thus, defect prediction is becoming a more and more promising aid to increase the effectiveness and efficiency of these usually costly quality assurance measures. "The net result should be systems that are of higher quality, containing fewer faults, and projects that stay more closely on schedule than would otherwise be possible." (Ostrand et al. 2005, p. 340).

DOI: 10.4018/978-1-4666-6026-7.ch024

A large number of empirical studies on various aspects of defect prediction are available, and several of these incorporate data from industrial projects (e.g., Ostrand et al. 2005, Nagappan et al. 2005, Zimmermann and Nagappan 2007, Nagappan et al. 2010, Bird et al. 2011). Yet, few studies actually provide insights on how defect prediction can be applied in an industrial setting, where defect prediction itself is subject to a trade-off between cost and quality. Among these are the study from Li et al. (2006) who report experiences from initiating field defect prediction and product test prioritization at ABB, from Weyuker (2007) illustrating the research path towards making defect prediction usable for practitioners, and Wahyudin et al. (2008) presenting a framework for conducting defect prediction as an aid for the project manager in software development organizations.

In this chapter, an overview of the state of the art in defect prediction using data mining techniques is presented and key questions that are of practical importance for establishing defect prediction in large software projects are discussed. The presented work is based on and extends the extensive body of available literature on software defect prediction (see, e.g., Catal and Diri 2009, Hall et al. 2012) as well as some previous works (Ramler and Wolfmaier 2008, Ramler et al. 2009a, Ramler et al. 2009b, Ramler and Natschläger 2011, Ramler and Himmelbauer 2013) of the authors.

The key questions derived in the next section of this chapter concern aspects such as the prediction objectives and granularity level at which predictions should be made, the sources and mining of prediction data, the properties of real-world data and approaches to deal with noise, the choice of learning algorithms and validation measures. The questions identified and subsequently (Sections 3 to 7) discussed in detail provide a valuable guidance for constructing defect prediction models in a real-world setting.

In addition to the discussion of these questions, insights and experiences gained by conducting projects on data mining and defect prediction in the context of large software systems are presented. One of these projects has been selected to serve as an illustrative use case throughout the chapter. Details about this project are provided in Section 3. In the remainder of this chapter, specific issues concerning the construction of prediction models are presented which provide guidelines on how to choose a learning algorithm (Section 4), select features from different data sources (Section 5), deal with noise and data quality issues (Section 6) and evaluate prediction models for evolving systems (Section 7). In the final section, a list of open issues and unanswered questions is highlighting areas to which future research on software defect prediction should be directed.

DATA MINING AND KNOWLEDGE DISCOVERY FOR DEFECT PREDICTION

Defect prediction is based on prediction models built from software engineering data. Thus, defect prediction can be understood as an application within the broad area of *data mining* and *knowledge discovery* which refer to general results of research, techniques and tools used to extract useful information and models from (large volumes of) data (Mariscal et al. 2010).

Defect Prediction Models

The critical essence of making predictions about defects in software systems is captured in prediction models. The approach for constructing prediction models is illustrated in Figure 1.

A prediction model is related to a software system at the level of its modules. Modules may be classes, files, components or even sub-systems of a software system. The modules of the software system are described by various *attributes* (e.g., code metrics or the number of recent changes), which have to be made available via extraction from different data sources such as metric databases or