International Journal of Open Source Software and Processes
Volume 12 • Issue 2 • April-June 2021

Quantitative Software Change Prediction in Open Source Web Projects Using Time Series Forecasting

Wasiur Rhmann, Department of Computer Application, Shri Ramswaroop Memorial University, Barabanki, India

ABSTRACT

Software change prediction (SCP) is used for the prediction of changes earlier in the software development life cycle. It identifies the files that are change prone. Software maintenance costs can be reduced with the help of accurate prediction of change-prone files. Most of the literature of SCP deals with the identification of a class as change prone or not change prone. In the present work, the amount of change in a web project in terms of line of code added (loc_added), line of code deleted (loc_deleted), and lines of code (LOC) are predicted using time series forecasting method of machine learning. Data of web projects is obtained from GIT repository using Pydriller Python package extractor. The obtained result showed that support vector machine (SVM) is good for prediction of loc_added and loc_removed while the random forest is good for the prediction of LOC. Results advocate the use machine learning techniques for forecasting changes amount in web projects.

KEYWORDS

GIT Repository, Machine Learning, Software Change, Web Projects

1. INTRODUCTION

In today's era of automation software development process has drastically changed. Now a day DevOps has improved the software development process (Freeman, 2019). DevOps reduces the software development life cycle by combining development with IT operations and delivers quality software. DevOps advocates use of different tools for development of quality software. Different tools like Docker, Ansible, Git, Puppet, JenKins etc. are very popular among DevOps engineers. Git stores the code during project development and it can be used for software change prediction. Software change prediction is needed for cost reduction as it helps in proper allocation of resources during the testing and maintenance phase. Earlier studies of software change predictions have considered the problem as classification in which a module is classified as change prone or not. In the present work quantity of software, changes are forecasted using machine learning-based time series forecasting. In time series forecasting a prediction model is created which is used for prediction of future events based on past events. Software evaluation is a continuous process and change in software comes with the time, prediction of software changes i.e. addition, deletion, or updating line of code. Prediction of the numbers of lines that will be added, deleted, or updated in the future is useful for the allocation of resources for maintenance. Malhotra & Khanna, (2019) have evaluated different models of Software change prediction they have explored search-based techniques and machine learning technique for this. Malhotra & Khanna (2017) have used imbalanced dataset where class distribution is unequal for change prone classes. They predicted change prone classes using sampling method with meta cost

DOI: 10.4018/IJOSSP.2021040103

learners. The authors advocated using resample methods with the replacement for software change prediction in an imbalanced dataset. Malhotra & Megha Khanna (2018) have presented a PSO ensemble technique for software change prediction. They have used weighted voting techniques for software change prediction. The proposed ensemble is compared with machine learning techniques. Goulão et al. (2012) have used ARIMA time series forecasting method for eclipse change request tracking. A change request may be due to the presence of the defect. The ARIMA gives seasonal patterns and validation is done by forecasting change for next months. Changes in software may occur at any stage of the development cycle. Sometimes changes done at one place don't fit to rest of the system. Impact of change must be handle properly to maintain consistency and integrity of the software system. For maintaining integrity of the system stakeholders and developers must focus on how change can affect rest of the system and what should be modified to maintain consistency of the software. Change impact analysis is performed to measure the effects of software changes. Lindvall and San(1998) have discussed Requirements-driven impact analysis (RDIA) that identifies set of entities that need to be change due to changes in requirements. A number of requirements and existing system is given as input and ouput is set of entities that need to be changed. Output of this RDIA is helpful for project planning and cost estimation of project due to change volume of software. Catolino & Ferrucci (2019) have applied ensemble techniques for the Software change prediction. Ensemble models are created based on the different metrics and different types of machine learning techniques are used. Ensemble techniques have used various standard machine learning classifiers as base classifiers. Ten open source systems and their 33 different releases are used for experimental purposes. The results find Random forest as the best technique for software change prediction in terms of F-measure. Yongxin Ge et al. (2018) have created a model of cross-project based software change prediction. Deep Metric Learning model is proposed for the classification of the file as change prediction or not. An over-sampling method is used to train the model for an imbalanced dataset. Several change prediction datasets were used for showing the effectiveness of change prediction. Malhotra and Khanna (2019) have given a novel dynamic fitness function based software change prediction model. An adaptive fitness function selection scheme is proposed for dynamically selection of fitness function of particle selection algorithm for obtaining better prediction. Their model showed better results compared to single fitness-based models. Efe and Demirors (2019) have proposed a change management model for software projects, rework for maintenance, and cost required to monitor the changes. In recent years Time series forecasting based machine learning methods are applied in different domains like cryptocurrency price forecasting(Chowdhury R. et al., 2020), novel Covid-19 forecasting(Zeroual A. et al., 2020) twitter sentiment based cryptocurrency price forecasting(Kraaijeveld **&** Smedt, 2020) etc. Hegde, B. et al. (2019) have used Recurrent neural network(RNN) for lung cancer detection. Jagannatha et al. (2018) have computed different performance metrics of software system. Time series forecasting is also applied in software engineering research like technical debt of software project forecasting (Tsoukalas et al., 2020), Forecasting of commit activity in git repository based software projects(Decan et al, 2020), software vulnerability forecasting (Yasasin et al., 2020). A taxonomy of changes is given by (Jim Buckley et al., 2005) and different changes affecting factors are time of changes, changes types, change history, change frequency etc. Peter and Ramanna(2003) have used two rough set methods for obtaining software change rules to effectively allocation of resources during software development and during maintenance. Rough Set Exploration System (RSES) has provided rough set based classifiers for software change prediction and compared to classification algorithms in Weka software using cross-validation. Fu et al. (2014) have used Semi-supervised LDA technique for automatically change message classification. Domain knowledge is used to label samples and validation of proposed technique is performed on three open source projects. Classification accuracy was found 85% and approach is applicable to cross-project change classification. Change management is challenging in software industry (Johnes, 1996) but in earlier days change management was merely for code and tools were poor in code management however other documents like text, planning documents, cost estimation documents and bug management were poorly managed and