Abstract

Machine learning has been applied on datasets that consist of source codes to perform programming language classification. However, there has been no study that directly investigates the impact of applying various preprocessing methods. This paper aims to analyze the impact of different preprocessing methods and their usage combinations on the machine learning model performance datasets that consists of source codes. To simply the research and to focus on the preprocessing aspect, this research utilizes the preprocessing methods to build a simple code coverage classification (i.e., branch/non-branch). The study uses Support Vector Machines (SVM) and Multi-Layer Perceptron (MLP) to build the models that will be analyzed. The preprocessing methods analyzed are several vectorizations including Vector Count, TF-IDF, and Word2Vec. The results show that the preprocessing methods and their combinations used significantly influences the performance of the machine learning model. Using SVM, TF-IDF vectorization showed the best results, while Count vectorization yielded the best results for MLP. The alphanumeric character extraction preprocessing method contributed the maximum results regardless of the model and vectorization.

Keywords: machine learning, preprocessing, source code datasets, code coverage, classification.