Machine Learning Based Classification of Android Apps through Text Features

Mohamed Guendouz*, Abdelmalek Amine*
Reda Mohamed Hamou*

*GeCoDe Laboratory, Tahar Moulay University of Saida. Saida, Algeria
adresse@email,
http://www.une-page.html

1 Introduction

This paper deals with the problem of Android mobile apps classification using machine learning and text mining methods. Our approach consists in applying some machine learning methods on text characteristics that are extracted from app's description on Google Play Store.

Our proposed approach consists of two main phases. First we collect information about apps from the Google Play store using a web crawler. Then we extract some text information from this data. In our case, for each app we have extracted its description and its category. Second, we train different classifiers, this step involves pre-processing the text which includes removing URLs and digits, tokenization, and calculating TF*IDF to transfer text to a numeric vector which can be used as input for classifiers. Finally, to evaluate performance of our system, we have conducted various experiments on three real datasets using different evaluation metrics. Figure 1 illustrates the architecture of our app classification framework.

Finally, we evaluate our approach on three real datasets, obtained results shows that the use of text features in classifying Android apps can performs well.

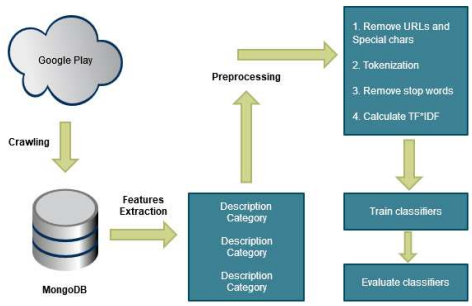

FIG. 1 – *Architecture of the proposed framework*

2 Experimental Results

Since there are not any standard dataset or benchmark for this type of studies, we had to create our own dataset. We collected metadata of free Android Apps from the Google Play store, this task was done automatically using a web crawler developed specially for this. A total of 7893 Android Apps were collected categorized in 4534 Apps and 3305 Games.

We devised our dataset into three other dataset which are : General dataset which contains all Apps classified in two classes : App and Game. Apps dataset which contains only Apps classified in nine classes : COMMUNICATION, EDUCATION, ENTERTAINMENT, MEDIA AND VIDEO, MEDICAL, PHOTOGRAPHY, SOCIAL, TOOLS, WEATHER. Games datast which contains only Games classified in six classes : ARCADE, EDUCATIONAL, PUZZLE, RACING, SIMULATION, SPORTS.

2.1 Evaluation Metrics

In order to evaluate performance and accuracy of our approach, we choose a good number of well-accepted evaluation metrics for classifiers including Precision, Recall, F-Measure, TPR (true positive rate), FPR (false positive rate). In the experiments we utilize ten-fold cross validation to evaluate each classifier.

2.2 Results and Analysis

In this subsection, we analyze results obtained from experiments, we first evaluate classifiers on the first dataset which contains all applications categorized only in two classes : Apps and Games, this means a binary-class classification, we evaluate performance of different classifiers like : Naive Bayse, SVM, RandomForest.

The result shows that the Random Forest algorithm performed better than all other algorithms, the best result of F-Measure is 0.971 obtained from Random Forest algorithm with 150 trees, this means that a large number of samples (97.1% approximately) have been correctly classified by our system. We also note that the SVM algorithm performs good results with a value of F-Measure equal to 0.963 and a small value of FP rate equal to 0.042, this is due to the high classification accuracy of SVM algorithm in binary class classification.

We performed a second experiment on two other real datasets, the first contains only Android Apps of type App categorized in nine categories which means nine classes, the second dataset contains only Android Apps of type Game categorized in six categories which means six classes therefore the problem is not anymore a binary class classification.

These results show that there is a decline in classifiers accuracy especially the Naive Bayes algorithm and this is due to the number of classes in each dataset and also because terms (text) used in describing these Apps are not different, for example the term "play" is used to describes games which belong to different game categories like puzzle, simulation and more, this distribution of terms over multiple classes report incorrect classification results. Although, the Random Forest algorithm performs well and better than all other algorithms and gives good classification accuracy.