# ABSTRACT

Google Play store has five million applications (apps) and large numbers of users regularly download various applications, use them and give a review on the applications according to their experience. Android applications have grown in popularity as a type of software, and user reviews of apps have become a valuable source of feedback. Many users report issues related to the app in their reviews when they are use apps like application crash, performance issue, complaints about the design, their experience with the app and they may also request additional features that may complete desired functionalities, etc. These types of reviews can help developers to improve the apps' features and user experience. There are thousands of reviews in each app which is very difficult to read and gather manually it’s time-consuming, sometimes many reviews were ignored due to the manually reading or gathering. For resolving the above-mentioned issues, This research proposes a framework to automatically classify the review type which is bug reports, performance, feature request, user experience, application crash, and sentiments (e.g.: Positive, Negative, and Neutral) will be classified according to reviews. The proposed framework will be implemented using Natural Language Processing (NLP) techniques and machine learning classification algorithms. The first step involves creating a baseline corpus that will be generated using the Python google scrapper library and the dataset will be prepare from 22 educational or e-learning android applications. And then NLP data pre-processing techniques will be applied which are Tokenization, Lemmatization, Stop Words, Stemming, and a bag of words. After that, feature extraction techniques will be applied to the data set and then these features will be trained and classified using machine learning classification algorithms. Proposed framework performance will be evaluated using evaluation metrics; Accuracy, Precision, Recall, and F1-Score by using review categories and sentiments.

**Table of Contents**

[ABSTRACT 4](#_Toc69988402)

[INTRODUCTION 7](#_Toc69988403)

[RESEARCH QUESTION(S) 8](#_Toc69988404)

[RESEARCH OBJECTIVES 8](#_Toc69988405)

[REVIEW OF LITERATURE 8](#_Toc69988406)

[METHODOLOGY 10](#_Toc69988407)

[CONCLUSION AND FUTURE PROSPECTS 12](#_Toc69988408)

# 

# INTRODUCTION

Vast number of mobile device users has a high influence on a large number of applications (apps) available in the Google Play Store. Apple AppStore, Google Play Store, and Windows Store are the three most used app stores. Where user can easily download multiple applications at the same time and these applications also allow the user to submit their feedback and ratings (Zhang et al., 2017). The reviews of users can help developers in discovering and updating the applications according to user’s requirements. Due to thousands of users’ review and their unorganized nature, manually gathering the reviews is time-consuming and is also an inefficient method (Abrowski et al., 2020). Developers also face issues in gathering the information from the google play store daily due to high numbers of reviews. Several studies (Aldabbas et al., 2020; Deshpande & Rokne, 2018; Stanik et al., 2019) have been conducted on user reviews and analyzing the reviews but these researches are limited mostly to the analysis of users’ ratings. Our research is aimed to improve and classify the user reviews and categorize them along with sentiment classification (e.g.: Positive, Negative, and Neutral) applications feedbacks.

In this research, 22 educational android applications will be used for analyzing user reviews. This research will classify user reviews into five categories which are bug report, user experience, performance, feature request, and application crash. For classification and comparison above mentioned categories, different classifiers will be used. Bug reports identification issues with the app that could be fixed, such as an erroneous application behavior or a compatibility issue. In feature requests, users’ may submit their suggestions for features or missing features and they may also submit suggestions about how to change or add more functionalities in the app. In performance, users can report the application performance through their feedback if the application does not performed efficiently. In User experiences, users’ complaints about the visuals, controls, and design will be considered, and identification of application crashing issues from the user reviews. By examining the above issues, this research aims to improve the review gathering process of android applications by using different supervised learning algorithms and NLP techniques which are tokenization, stop words removal, stemming, lemmatization, and a bag of words. The key contributions of this research are summarized below:

* A Baseline corpus will be created by using the google play store.
* An automated framework will be developed by using NLP techniques and machine learning algorithms.
* Experiments will be conducted using classification algorithms to train the features and predict the review type and sentiment priority.
* In the end, the overall performance of classification algorithms will be evaluated using Recall, F-score, and Precision accuracy measures.

# RESEARCH QUESTION(S)

RQ1: What are the issue types presented in google play store applications?

RQ2: How NLP techniques and machine learning classification algorithms can be combined to classify app reviews?

# RESEARCH OBJECTIVES

The aims of this research are:

1. to classify android application user’s feedback into different review categories e.g.: bug report, user experience, performance, feature request, and application crash.
2. to classify sentiments from user reviews.
3. to conduct a comparative analysis of different supervised learning techniques based on their accuracy to find the best technique.

# REVIEW OF LITERATURE

Several studies have been published on app-review classification in English (Abualigah et al., 2020; Gao et al., 2018; Maalej et al., 2016; Maalej & Nabil, 2015; Parente, 2018) and Chinese language(Zhang et al., 2017). A lot of research has been conducted that is focused on user sentiments and different issue types, downloads, application ratings with different methodologies to improve problems. This research aims to automate the method of user reviews and identify their sentiments, based on the reviews, and users’ reviews will also categorize. Therefore, the literature of the previous studies have been discussed below:

(Zhang et al., 2017) proposed a model that is a cost-sensitive learning method (CSLabel) for analyzing and labeling the reviews. They analyzed 3902 user reviews from 11 mobile apps in a Chinese app store, they have covered 17 issue types which include Additional cost, Compatibility issue, Content complaint, Crashing, Feature removal, Feature request, Functional complaint, and User interface. The Support vector machine learning algorithm is used in this research. The result shows that CSLabel can correctly label the data and they also identified the issue types that have a negative correlation with users’ evaluation of apps.

The numeric rating was also predicted by (Umer et al., 2020) in which they proposed an ensemble learning model that considers three TF-IDF features which include unigrams, bigrams, and trigrams. For this, 14 different categories reviews were scrapped from the google play store. Random forest, gradient boosting classifier, extreme gradient boosting classifier, the Adaboost classifier, and extra tree classifier were applied to predict numeric ratings. The result shows that the approach of tree-based ensemble classifiers performed much better than then other classifiers and the study also shows that user reviews are irrelevant with user ratings and that numeric ratings are higher than user reviews would imply. (Shashank & Naidu, 2020) also predicts the ratings of google play store applications using machine learning algorithms which include Random Forest, Support Vector Regression, linear regression, K-Nearest Neighbor, and K-mean clustering. Dataset was collected from Kaggle. The result shows that K-Nearest Neighbor is predicting more accurate results better than other algorithms. (Aldabbas et al., 2020) scrapped user reviews and ratings from google play store and they used Logistic Regression, Random Forest, and Multinomial Naïve Bayes to test the reviews. The result shows that the Logistic Regression algorithm is the most active because its precision rate is high and gives more accurate results. Another research is conducted on the aspect extraction process (Eko et al., 2017) they have used 3 IOS applications for analyzing the reviews. They have compared their proposed method and collocation method for analyzing the reviews and their results show that the aspect extraction process gets the higher precision and recall. (Edara et al., 2020) also proposed aspect-based recommendation framework for analyzing the opinions of users 637 android and IOS health application were used for data set and user reviews were scrapped from google play store and IOS store using Heedzy web crawler. They have compared machine learning algorithms which is Multinomial Logistic Regression, Multinomial Naïve Bayes, Linear Support Vector Machine and deep learning algorithm which is Recurrent neural networks, Long-short term memory networks, Ensemble multi model deep learning, and convolutional neural network. Result shows that their proposed framework is more quick and convenient for analyzing the health app reviews.

Several studies have been performed to categorize the reviews but their studies are limited and mostly they are focused on review ratings without any classification of user reviews into different review types and they are also not classifying the sentiments. However, this research aims to classify user reviews and sentiment through a proposed framework which will be developed using NLP techniques and machine classification algorithms.

# METHODOLOGY

The methodology of the proposed framework is divided into five steps: first step is Data collection, Second step is Data pre-processing, third step is Feature extraction, fourth step is Category and Sentiment Analysis Classification and fifth step is Performances’ evaluation.

In the first step, data will be collected by using the python google play scrapper library while data pre-processing techniques will be applied to the dataset in second phase. In this step user reviews will be converted into topic vector space because machine learning algorithms easily understand this format. In third step, user reviews that have been developed in the previous phase will be categorized based on their similar textual structure by using different NLP approaches. Some additional data pre-processing steps will be performed also using the NLTK toolkit, which is included POS Tagging, stop words removal, a bag of words, and lemmatization. Features will be extracted from the user reviews, and then a feature subset will be created through different feature selection techniques in the next phases. After that, in the fourth step, the top selected features will be trained by using a machine learning model. The performance will be evaluated in fifth step to determine the accuracy of the proposed framework. The framework of this study is shown in Figure 1. Each phase of the framework is described in detail below.

**Data Collection**

Using the python scrapper library, the dataset will be gathered from the Google Play store.

The following standards will be applied:

* The app must lie in the education category.
* The app must have at least 50000 downloads. The dataset will contain the following attributes;

Category, Name, User Review, Review Type, Sentiments while other attributes and related meta-data can be scraped from the Play store in general, This research focuses on investigating the difference between customer perception and review form predictions based on reviews. As a result, irrelevant meta-data will be ignored. The data will be scrapped from 22 different application reviews from the education or e-learning application category.

**Pre-processing**

In this phase, the contents of user reviews i.e. Category, Name, User Review, Review Type, Sentiment will be converted into vectors of topics with the list of top words without unnecessary contents. This phase involves three steps.

* Tokenization
* Lemmatization
* Stop words Removal
* Stemming
* Bag of words

**Feature extraction**

After the completion of a pre-processing phase, features will be extracted then, the feature set of each dimension will be examined. To find the results, feature extraction technique TF-IDF will be used.

**Classification**

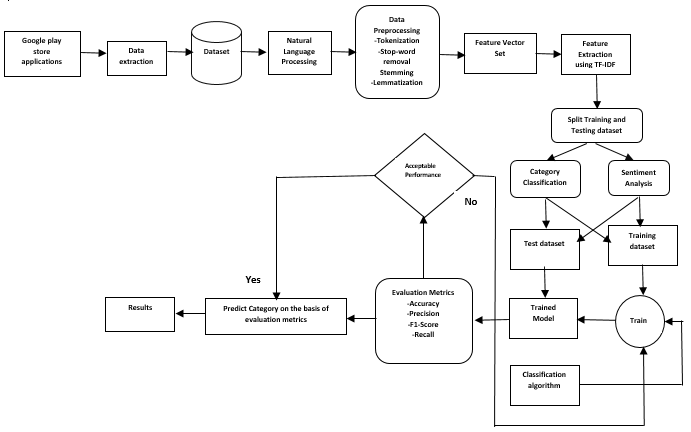
In this phase, different machine learning classification algorithms will be used to model training and then predict the category priority and sentiments of user reviews. This phase will consist of two steps: training and prediction. In the training step, a dataset will be provided to the model with a predefined set of categories. This training model would be helpful to assign new incoming bugs with different level of priorities and categories in prediction step four machine learning classifiers will be performed to compare the results and select the best classification model:

* Naive Bayes Classifier
* Decision Trees Classifier
* Random Forests
* Support Vector Machine (SVM)

**Evaluation Metrics**

In the last phase, classifiers’ performance will be evaluated using the following accuracy measures that will be helpful to examine the output of the proposed framework.

* Accuracy
* Precision
* Recall
* F1-Score



**Figure 1:** Model Framework

# CONCLUSION AND FUTURE PROSPECTS

In today’s era, everyone has android phones and they are using android applications. Millions of users daily download various applications and report issues through their reviews and ratings. Developers cannot read the reviews manually because of the large number of reviews it is an inefficient method. This research proposed an automated framework for the classification of reviews and sentiments using NLP techniques and machine learning classification algorithms. NLP will be used for data preprocessing which includes tokenization, lemmatization, elimination of stop-words, stemming process through Porter Stemmer algorithm, and a bag of words. After applying data processing techniques classification algorithms will be applied on user reviews and sentiments which are Naïve Bayes, Random Forest, Decision tree, and Support Vector Machine then results will be evaluated using evaluation metrics which are Accuracy, Precision, Recall, and F-score measures. Future research may include more android application categories and apply deep learning algorithms to the reviews.

**REFERENCES**

Abualigah, L., Alfar, H. E., Shehab, M., & Hussein, A. M. A. (2020). Sentiment Analysis in Healthcare: A Brief Review. *Recent Advances in NLP: The Case of Arabic Language, Studies in Computational Intelligence*, *874*(January), 129–141. https://doi.org/10.1007/978-3-030-34614-0\_7

Aldabbas, H., Bajahzar, A., Alruily, M., Qureshi, A. A., Amir Latif, R. M., & Farhan, M. (2020). Google Play Content Scraping and Knowledge Engineering using Natural Language Processing Techniques with the Analysis of User Reviews. *Journal of Intelligent Systems*, *30*(1), 192–208. https://doi.org/10.1515/jisys-2019-0197

Deshpande, G., & Rokne, J. (2018). User Feedback from Tweets vs App Store Reviews: An Exploratory Study of Frequency, Timing and Content. *Proceedings - 2018 5th International Workshop on Artificial Intelligence for Requirements Engineering, AIRE 2018*, *August*, 15–21. https://doi.org/10.1109/AIRE.2018.00008

Edara, D. C., Sistla, V., & Kolli, V. K. K. (2020). Health App Recommendation System using Ensemble Multimodel Deep Learning. *Journal of Engineering Science and Technology Review*, *13*(5), 7–19. https://doi.org/10.25103/jestr.135.03

Eko, B., Galih, D., & Wahyu, E. (2017). Aspect Extraction using Informative Data from Mobile App Data Review. *International Journal of Computer Applications*, *173*(9), 28–32. https://doi.org/10.5120/ijca2017915422

Gao, C., Zeng, J., Lyu, M. R., & King, I. (2018). Online app review analysis for identifying emerging issues. *40th International Conference on Software EngineeringMay 2018*, *December*, 48–58. https://doi.org/10.1145/3180155.3180218

Jacek D ˛abrowski†, Emmanuel Letier, Anna Perini, and A. S. (2020). *App Review Analysis for Software Engineering: A Systematic Literature Review*.

Maalej, W., Kurtanović, Z., Nabil, H., & Stanik, C. (2016). On the automatic classification of app reviews. *Journal of Requirements Engineering*, *21*(3), 311–331. https://doi.org/10.1007/s00766-016-0251-9

Maalej, W., & Nabil, H. (2015). Bug report, feature request, or simply praise? On automatically classifying app reviews. *2015 IEEE 23rd International Requirements Engineering Conference, RE 2015 - Proceedings*, 116–125. https://doi.org/10.1109/RE.2015.7320414

Parente, M. G. (2018). Using NLP and Information Visualization to analyze app reviews. In *Utrecht University*. https://dspace.library.uu.nl/bitstream/handle/1874/368082/MScThesis\_MGarciaParente.pdf?sequence=2

Shashank, S., & Naidu, B. (2020). Google Play Store Apps-Data Analysis and Ratings Prediction. *International Research Journal of Engineering and Technology*, *07*(12), 265–274. www.irjet.net

Stanik, C., Haering, M., & Maalej, W. (2019). Classifying multilingual user feedback using traditional machine learning and deep learning. *Proceedings - 2019 IEEE 27th International Requirements Engineering Conference Workshops, REW 2019*, 220–226. https://doi.org/10.1109/REW.2019.00046

Umer, M., Ashraf, I., Mehmood, A., Ullah, S., & Choi, G. S. (2020). Predicting numeric ratings for Google apps using text features and ensemble learning. *ETRI Journal*, *43*(September 2019), 95–108. https://doi.org/10.4218/etrij.2019-0443

Zhang, L., Huang, X. Y., Jiang, J., & Hu, Y. K. (2017). CSLabel: An Approach for Labelling Mobile App Reviews. *Journal of Computer Science and Technology*, *32*(6), 1076–1089. https://doi.org/10.1007/s11390-017-1784-1