



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 7, July 2024

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

 9940 572 462

 6381 907 438

 ijirset@gmail.com

 www.ijirset.com

Review of Play Store App Analysis & Rating Prediction using Artificial Intelligence

Manisha Panthi¹, Prof. Firdous Qureshi², Dr. Sneha Soni³

Research Scholar, Department of Computer Science and Engineering, Sagar Institute of Research & Technology Excellence, Bhopal, India¹

Assistant Professor, Department of Computer Science and Engineering, Sagar Institute of Research & Technology Excellence, Bhopal, India²

Professor & HOD, Department of Computer Science and Engineering, Sagar Institute of Research & Technology Excellence, Bhopal, India³

ABSTRACT: The ever-growing popularity of mobile applications has intensified competition within the app market. User reviews and ratings play a pivotal role in influencing download decisions. This paper explores the application of Artificial Intelligence (AI) techniques for analyzing Play Store app reviews and predicting user ratings. We delve into the various AI methodologies employed, including sentiment analysis, topic modeling, and natural language processing (NLP). By harnessing these techniques, app developers can gain valuable insights from user feedback, identify areas for improvement, and ultimately enhance their app's reception within the marketplace. This paper presents survey of machine learning technique for rating prediction of Google play store apps.

KEYWORDS: Google, Play Store, Machine Learning, Online, Rating.

I. INTRODUCTION

The mobile application landscape has become a vibrant ecosystem teeming with innovation and competition. With millions of apps vying for user attention, the ability to understand and cater to user preferences is paramount for success. In this dynamic environment, user reviews and ratings on platforms like the Google Play Store serve as a crucial barometer of user sentiment. These digital word-of-mouth testimonials offer invaluable insights into user experience, highlighting both positive aspects and areas for improvement. However, manually sifting through vast amounts of text data from reviews can be a daunting task. This is where Artificial Intelligence (AI) steps in, offering a powerful toolbox for app developers to unlock the hidden treasures within user feedback.

By leveraging AI techniques like sentiment analysis, topic modeling, and natural language processing (NLP), app developers can gain a deeper understanding of the underlying themes and emotions expressed within user reviews. Sentiment analysis, a cornerstone of AI-powered review analysis, delves into the emotional undercurrents of reviews, pinpointing positive, negative, and neutral opinions. This allows developers to gauge user satisfaction and identify areas where the app might be falling short. Beyond sentiment analysis, AI offers more sophisticated tools like topic modeling. This technique helps developers understand the recurring themes and topics that surface within reviews. For instance, topic modeling might reveal a consistent theme of user frustration with app crashes or difficulty with a specific feature. Armed with this knowledge, developers can prioritize bug fixes and streamline user interactions.

Natural Language Processing (NLP) takes the analysis a step further. NLP techniques allow AI to delve deeper into the semantic meaning of reviews, identifying specific aspects of the app that users are praising or critiquing. This granular analysis allows developers to pinpoint strengths and weaknesses across various functionalities, such as user interface, performance, or content quality. The benefits of AI extend beyond simply understanding user sentiment. By leveraging AI-powered insights from reviews, app developers can make data-driven decisions to optimize their app's performance and target audience. For example, analyzing positive reviews can highlight features that resonate with users, informing future development efforts. Conversely, negative reviews can be used to identify areas for improvement, ensuring a more positive user experience for future users.



Figure 1: Mobile App

Moreover, AI can be harnessed to predict user ratings for new or updated app versions. This predictive power allows developers to anticipate user reception and make adjustments before a major release. By proactively addressing potential issues identified through AI analysis, developers can minimize negative feedback and ensure a smoother launch for their app.

However, the integration of AI into Play Store app analysis is not without its challenges. The accuracy and effectiveness of AI models are heavily dependent on the quality and quantity of data available for training. Furthermore, the inherent complexities of human language can pose challenges for AI algorithms. These challenges and limitations will be explored in a later section of this paper.

AI presents a transformative opportunity for app developers seeking to leverage the wealth of information within Play Store user reviews. By harnessing the power of AI techniques, app developers can gain a deeper understanding of user preferences, identify areas for improvement, and ultimately, enhance their app's success within a competitive marketplace. As AI technology continues to evolve, its role in app analysis and development is poised to become even more significant, shaping the future of the mobile application landscape.

II. RELATED WORK

B. Moharana et al.,[1] The reviews and ratings given by end users play an important role for both the developers and other users for the performance and survival of the app in the marketplace. Reviews are just a description for justification of the rating given by a user & the rating plays an important role in the first look. In this paper, we have used the Play Store Analysis Dataset to descriptively analyze the various attributes present in it and create models that predict the rating of an app, given its specifications. The models have been compared based on a common accuracy metric based on which their reliability can be judged and used in real-time rating predictions for a new application in the store for which a certain amount of data is available over some period.

R. Gomes et al.,[2] present the classifiers to meet the success requirements of the Google Play Store app store. Through the techniques of KNN and Random Forest, a statistical analysis was done performing the regressions of the applications according to some characteristics: as hypothesis test, correlation and regression metrics analysis. This work aims to create inference engines, allowing the prediction of application ratings, using the KNN and Random Forest regression techniques. The Random Forest showed better results than the KNN.

G. S. Bhat et al.,[3] present an asthma risk prediction tool based on machine learning (ML). The entire tool is implemented on a smartphone as a mobile-health (m-health) application using the resources of Internet-of-Things (IoT). Peak Expiratory Flow Rates (PEFR) are commonly measured using external instruments such as peak flow meters and are well known asthma risk predictors. In this work, we find a correlation between the particulate matter (PM) found indoors and the outside weather with the PEFR. The PEFR results are classified into three categories such as 'Green'

(Safe), 'Yellow' (Moderate Risk) and 'Red' (High Risk) conditions in comparison to the best peak flow value obtained by each individual. Convolutional neural network (CNN) architecture is used to map the relationship between the indoor PM and weather data to the PEFR values.

Z. Wu et al.,[4] framework augments the labeled dataset of app descriptions to improve the prediction of permissions. FideDroid compares inferred permissions with used ones to reveal the suspicious and unnecessary permissions based on the prediction. It helps developers to refine app descriptions and maintain permission usages. In our experiments on large real-world apps, we analyzed and revealed that the category-based common permissions may cover more unmentioned functionalities without considering all possible permissions during app description analysis. In addition, we discovered three factors causing the inconsistency between descriptions and permission usages to be: 1) human interventions in writing description; 2) bad practices on permission usages; and 3) prolific developers. These findings will facilitate developers to refine app descriptions and optimize permission usages in the apps.

Z. Shen et al.,[5] work aims to predict a set of apps a user will open on her mobile device in the next time slot. Such an information is essential for many smartphone operations, e.g., app pre-loading and content pre-caching, to improve user experience. However, it is hard to build an explicit model that accurately captures the complex environment context and predicts a set of apps at one time. This work presents a deep reinforcement learning framework, named as DeepAPP, which learns a model-free predictive neural network from historical app usage data. Meanwhile, an online updating strategy is designed to adapt the predictive network to the time-varying app usage behavior.

Z. Xu et al.,[6] propose a new cross-triplet deep feature embedding method, called CDFE, for cross-app JIT bug prediction task. The CDFE method incorporates a state-of-the-art cross-triplet loss function into a deep neural network to learn high-level feature representation for the cross-app data. This loss function adapts to the cross-app feature learning task and aims to learn a new feature space to shorten the distance of commit instances with the same label and enlarge the distance of commit instances with different labels. In addition, this loss function assigns higher weights to losses caused by cross-app instance pairs than that by intra-app instance pairs, aiming to narrow the discrepancy of cross-app bug data.

K. Zhao et al.,[7] propose a model, called Simplified Deep Forest (SDF), to conduct JIT defect prediction for Android mobile apps. SDF modifies a state-of-the-art deep forest model by removing the multigrained scanning operation that is designed for data with a high-dimensional feature space. It uses a cascade structure with ensemble forests for representation learning and classification. We conduct experiments on 10 Android mobile apps and experimental results show that SDF performs significantly better than comparative methods in terms of 3 performance indicators.

G. Aceto et al.,[8] propose a novel heuristic to reconstruct application-layer messages in the common case of encrypted traffic. We discuss and experimentally evaluate the suitability of the provided modeling approaches for different tasks: characterization of network traffic (at different granularities, such as application, application category, and application version), and prediction of network traffic at both packet and message level. We also compare the results with several ML approaches, showing performance comparable to a state-of-the-art ML predictor (Random Forest Regressor). Also, with this work we provide a viable and theoretically sound traffic-analysis toolset to help improving ML evaluation (and possibly its design), and a sensible and interpretable baseline

Y. Zhang et al.,[9] propose DeePOP, a popularity prediction model that innovatively leverages time-varying hierarchical interactions. First, we propose Hierarchical Interaction Graph, which is firstly studied in this work, to organically characterize the relationship among apps. Second, DeePOP integrates internal factors and time-varying hierarchical interactions as inputs to build the prediction model. It develops multi-level modules based on Recurrent Neural Network with attention mechanism and generates multi-step time series predictions by fusing the outputs of modules. Experiments on a real-world dataset show that DeePOP outperforms state-of-the-art methods in prediction accuracy, effectively reducing the Root Mean Square Error (RMSE) to 0.088.

Q. Zhu et al.,[10] present a fairness-aware APP recommendation method named FARM. The principal study of this method emphasizes on the fairness issue during the recommendation process. In this method, APP candidates are divided into high visibility and low visibility APPs, and implement recommendation algorithm respectively. For low visibility APPs, we set a fairness factor for everyone, and use the user's latest feedback to make a dynamic adjustment. Based on the fairness factor, the recommendation is implemented by roulette-wheel. For high visibility APPs, we

employ the fuzzy analytic hierarchy process to implement the recommendation. The evaluation results show that FARM outperforms baselines in terms of recommendation fairness.

S. şahin et al.,[11] the dependence of filters on the feedback incurs high amount of latency and computational costs, hence SISO MMSE DFEs with static filters provide an attractive alternative for computational complexity-performance trade-off. However, the latter category of receivers faces a fundamental design issue on the estimation of the decision feedback reliability for filter computation. To address this issue, a novel approach to decision feedback reliability estimation through online prediction is proposed and applied for SISO FIR DFE with either a posteriori probability (APP) or expectation propagation (EP) based soft feedback. This novel method for filter computation is shown to improve detection performance compared to previously known alternative methods, and finite-length and asymptotic analysis show that DFE with static filters still remains well-suited for high-spectral efficiency applications.

S. Rezaei et al.,[12] Many network services and tools (e.g. network monitors, malware-detection systems, routing and billing policy enforcement modules in ISPs) depend on identifying the type of traffic that passes through the network. With the widespread use of mobile devices, the vast diversity of mobile apps, and the massive adoption of encryption protocols (such as TLS), large-scale encrypted traffic classification becomes increasingly difficult. In this work, we propose a deep learning model for mobile app identification that works even with encrypted traffic. The proposed model only needs the payload of the first few packets for classification, and, hence, it is suitable even for applications that rely on early prediction, such as routing and QoS provisioning.

III. CHALLENGES

The mobile app developing grows rapidly. The customer gives the reviews after using the app. The mobile app developers can understand the target audience easily.

- Meeting user requirements.
- Choosing the operating system.
- Choosing the development platform.
- Security.

After developing and using the app, customer rating is very useful to success the app. Therefore during the literature survey some of the observation is carried out, which is as followings-

- Low accuracy rate of true data prediction from given dataset.
- Using traditional System Analysis alone not sufficient for proper feature extraction.
- More classification error.
- No adaptive approach to prediction of true rating of apps.
- Sensitivity, Specificity, Precision, Recall and F measure values is not good optimized.

IV. PROPOSED STRATEGY

- Load the google play store dataset from the Kaggle

In this step, the google play store dataset will be downloaded from kaggle source. It is a large dataset providing company. Then load this dataset into the python environment.

- Visualizing the Dataset

Now open the dataset files and view the various data in term of features like rating, size of app etc.

- Pre-process the Dataset

Now the data preprocess step applied, here data is finalize for processing. Missing data is either removal or replace form constant one or zero in this step.

- Splitting the Dataset into training and testing

In this step, the final preprocessed of dataset is divided into the training and the testing dataset. In the machine learning, firstly the machine is trained through given dataset then it comes in tested period for remaining dataset.

- Classification Using Machine Learning Algorithm

Now apply the machine learning technique to find the performance parameters.

- Performance Metrics

(Accuracy, Precision, Recall, F1 - Score)

Now the performance parameters are calculated in terms of precision, recall, f-1 measure, accuracy etc by using the following formulas-

True Positive (TP): predicted true and event are positive.

True Negative (TN): Predicted true and event are negative.

False Positive (FP): predicted false and event are positive.

False Negative (FN): Predicted false and event are negative.

$$\begin{aligned} \text{Precision} &= \frac{|TP|}{|TP| + |FP|} \\ \text{Recall} &= \frac{|TP|}{|TP| + |FN|} \\ \text{F1} &= 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \\ \text{Accuracy} &= \frac{|TP| + |TN|}{|TP| + |TN| + |FP| + |FN|} \end{aligned}$$

V. CONCLUSION

The increasing number of Android apps available on Google Play Store with the developers' advantages has attracted many Android apps developers' attention. To benefit from developing Android apps is to know the characteristics of high rated applications on the Google Play Store. The use of applications is part of people daily lives for various activities. In relation to development, the curiosity about the characteristics responsible for success arises. We use classifiers to meet the success requirements of the Google Play Store app store. This paper presents the various research works on rating prediction of the Google play store app. In future we implement the machine or deep learning based classification technique for the prediction model with improved accuracy.

REFERENCES

1. B. Moharana, B. B. Biswal, S. Dey, M. K. Rath and S. Banerjee, "Play Store App Analysis & Rating Prediction Using Classical ML Models & Artificial Neural Network," 2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA), Pune, India, 2023, pp. 1-5, doi: 10.1109/ICCUBEA58933.2023.10391960.
2. R. Gomes da Silva, J. de Oliveira Liberato Magalhães, I. R. Rodrigues Silva, R. Fagundes, E. Lima and A. Maciel, "Rating Prediction of Google Play Store apps with application of data mining techniques," in IEEE Latin America Transactions, vol. 19, no. 01, pp. 26-32, January 2021, doi: 10.1109/TLA.2021.9423823.
3. G. S. Bhat et al., "Machine Learning-Based Asthma Risk Prediction Using IoT and Smartphone Applications," in IEEE Access, vol. 9, pp. 118708-118715, 2021, doi: 10.1109/ACCESS.2021.3103897.
4. Z. Wu, X. Chen, M. U. Khan and S. U. -J. Lee, "Enhancing Fidelity of Description in Android Apps With Category-Based Common Permissions," in IEEE Access, vol. 9, pp. 105493-105505, 2021, doi: 10.1109/ACCESS.2021.3100118.
5. Z. Shen, K. Yang, Z. Xi, J. Zou and W. Du, "DeepAPP: A Deep Reinforcement Learning Framework for Mobile Application Usage Prediction," in IEEE Transactions on Mobile Computing, doi: 10.1109/TMC.2021.3093619.
6. Z. Xu et al., "Effort-Aware Just-in-Time Bug Prediction for Mobile Apps Via Cross-Triplet Deep Feature Embedding," in IEEE Transactions on Reliability, doi: 10.1109/TR.2021.3066170.
7. K. Zhao, Z. Xu, T. Zhang, Y. Tang and M. Yan, "Simplified Deep Forest Model Based Just-in-Time Defect Prediction for Android Mobile Apps," in IEEE Transactions on Reliability, vol. 70, no. 2, pp. 848-859, June 2021, doi: 10.1109/TR.2021.3060937.
8. G. Aceto, G. Bovenzi, D. Ciunzo, A. Montieri, V. Persico and A. Pescapé, "Characterization and Prediction of Mobile-App Traffic Using Markov Modeling," in IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 907-925, March 2021, doi: 10.1109/TNSM.2021.3051381.
9. Y. Zhang, J. Liu, B. Guo, Z. Wang, Y. Liang and Z. Yu, "App Popularity Prediction by Incorporating Time-Varying Hierarchical Interactions," in IEEE Transactions on Mobile Computing, doi: 10.1109/TMC.2020.3029718.

10. Q. Zhu, Q. Sun, Z. Li and S. Wang, "FARM: A Fairness-Aware Recommendation Method for High Visibility and Low Visibility Mobile APPs," in IEEE Access, vol. 8, pp. 122747-122756, 2020, doi: 10.1109/ACCESS.2020.3007617.
11. S. şahin, A. M. Cipriano, C. Poulliat and M. Boucheret, "Iterative Decision Feedback Equalization Using Online Prediction," in IEEE Access, vol. 8, pp. 23638-23649, 2020, doi: 10.1109/ACCESS.2020.2970340.
12. S. Rezaei, B. Kroencke and X. Liu, "Large-Scale Mobile App Identification Using Deep Learning," in IEEE Access, vol. 8, pp. 348-362, 2020, doi: 10.1109/ACCESS.2019.2962018.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details