Analysis of Google Play Store Apps Data and Prediction of an App's Success

Archana J
Department of Computer Science
PES University
Bangalore, India
archanajayakumar4.aj@gmail.com

Vedica Rao
Department of Computer Science
PES University
Bangalore, India
vedica.rao@gmail.com

Swati Naik
Department of Computer Science
PES University
Bangalore, India
naikswati685@gmail.com

***Abstract*— Analyzing android market data particularly pertaining to new-to-market app data variables finding relationships between the data attributes and deriving suggestions according to analysis, thus helping developers popularize their application.**

I. INTRODUCTION

In this digital era usage of smartphones is growing at a high rate of 8 percent; more than a million new smartphones are coming to use everyday. 2.5 billion active android devices in use were announced by Google in 2017. From breakthroughs, improvements on the quality of results to new ways to search, Google helps us understand the world's information. Google launched the Android market in 2008 with only a handful of apps.

After a long way of surpassing 500,000 apps in 2012, it was renamed as Google Play Store. There are nearly 3.04 million apps available on the google play store with 96.5 percent of all android applications available for free and around 125894 paid.
Statistics show google play store sees 84.3 billion mobile application downloads globally and an addition of 6140 apps on average everyday. Popularity of the app is affected by the differences in using app usage criteria, app usage as per age, gender etc.

There are nearly three million apps available on play store, hence development of apps that stand out amongst the others is a challenge for developers. They must pinpoint essential factors which play a role in customers' decision - making process. The sudden uprise of opinion mining and sentiment analysis has created greater possibilities of improving our information gathering interests.

Through our research we analyzed the contributions of different variables towards user popularity for a new-to-market app. We provided consideration towards which factors calculate to a better understanding of customer's satisfaction of an app computing for different variables as dependent variables (example number of downloads, rating, numeric rating from reviews, etc.) to see which give better conclusive results. We also performed NLP and keyword analysis on the user reviews.

II. LITERATURE REVIEW

[1] mainly focus on the different classifier models used for the prediction and find the accuracy rate. Analysis is divided into three phases: data cleaning, data visualization and data modelling. On visualizing the dataset data modelling is performed using K-nearest model, Gaussian Naive Bayes model and Decision Tree model.

In the decision tree model, decision rule in each internal node splits the data. To evaluate splits in the dataset gini index has been used. The drawback is this splitting process rarely generalizes to other data.

The other classifier they used is k nearest neighbour where the euclidean distance between the new point and the existing points are calculated and then choosing the k nearest neighbours and ultimately used k-fold cross validation to overcome the high bias and variance. Logistic regression, one of the machine learning algorithms, has been used to perform binary classification on the data set where sigmoid function is used when the target variables are categorical in nature. While applying the Gaussian Naive Bayes model on the dataset they make an assumption of conditional independence between every pair of the features given the value of the class variables. It has strong feature independence assumptions,which is not usually the case in real life.

Ultimately it was concluded that there was no correlation between the app features like size, rating, number of installs and even between the price and the rating as well but there was a strong negative correlation between the number of installs and the number of reviews. On the basis of the number of installs, apps were divided into two categories i.e successful and unsuccessful. On applying various classifying algorithms concluded that the decision tree fits best to the problem statement whereas the Bayes model gives the lowest accuracy.

The paper [2] has considered two problems. One is the ambiguity which users usually get when they see the conflicting reviews and ratings. This issue can be solved using one of the two review types, star rating and comment. The other problem answered by this paper is the biasness to the summarized rating of the users. To solve this problem, they have used sentiment analysis on the user reviews and considered the starred rating. A numeric rating is generated from the Sentiment polarity of the review contents.The main

claim of this paper is that it proposes a unified rating system using sentiment analysis and an optimized probabilistic approach of the app reviews. The mean of the starred rating and the numeric rating generated by the probabilistic polarity approach will be the final rating for the apps to overcome the problem of ambiguity between the user reviews and the starred rating. They have identified whether the two reviews are consistent or inconsistent and formed two networks of them connected by their respective relations. Instead of extracting polarity directly, a polarity probability for each candidate expression was determined first using P-Probability (i.e., the positive sensitivity) and N-Probability (i.e., the negative sensitivity) of the candidate expressions. The sum of squared errors (SSE) has been used in place of absolute error to ensure that the two types of relations do not cancel one another. The candidate expressions in the root were assigned 1 or 0 based on the predefined polarity. The probabilities of the other candidate expressions were found using the sum of squared errors (SSE) function. The L- BFGS- B7 algorithm was used to solve the biasness problem and the polarities were generated. The candidate expressions were assigned with a P- Probability and N-Probability and only the candidates having probability above the threshold value for both positive and negative were removed and the others were considered as the result.

Finally, the final rating for the apps was calculated as the mean of the starred rating and the numeric rating generated by the above procedure. This overcame the problem of ambiguity between the reviews and the starred rating of the apps. A takeaway from this paper is that we could perform analysis on both the numeric rating and the user reviews could be used for predicting whether an app could fare well according to data from other apps in that domain. Another takeaway from this paper is that the probabilistic numeric polarity approach is more efficient in a diverse corpus of writings with different categories of targets.

The research in [3] focuses on identifying key variables that determine the app rating, addressing 3 research questions-(i) which factors effectively determine the app ratings in google play store? (ii)do some sets of factors collectively have a stronger relationship with the ratings? and (iii)are there title keywords that promise better ratings according to statistics? The data techniques used in the study were regression models for variable identification and computation of the variable's importance including Random Forest, SVR, Linear Regression, and Pearson correlation, with the main focus being to analyse the relationship between the attributes (number of reviews, number of installs, app size, app category, price, etc) and the app rating.

The model's computation was 2-fold. The first part of the model focussed on identification of different variables and testing them on correlation models. The performance of the different variables were evaluated with various evaluation techniques namely %IncMSC, RMSE, MAE, and p value and then the most influencing variables were computed. The latter part of the model extracted keywords from app titles and computed new rankings based on the previous rating and the keyword frequencies. The keyword processing included stopword removal and POS tagging.

The main claim of the paper is that peoples' preference towards an app is dominantly decided by the app rating. The graphs of the study demonstrated correlations between variables and their contribution in the app ratings providing conclusive statement results.

III. PROBLEM STATEMENT

The questions explored through our research are:

(i)What variables contribute most towards consumer popularity for a new-to-market app?

(ii)Are there any suggestive results that an app developer can be given through keyword analysis of the constructive reviews from other apps in that domain to help the app fare better in the market?

(iii)What factors contribute towards prediction of app rating?

Our research proves different from previous studies on google play store data since we focus on providing recommendations to new apps in the market. Existing research on app store data mainly focus on the reveal of the strong dependency of number of downloads to the rating of the app. Through our research we provide predictive analysis to help improvement for both market-established apps along with apps just hitting the market.

Google play store provides the ratings, reviews, and number of installs of an app; the variables most accepted as means of identification of whether an app is popular or not.

These factors are heavily influenced by each other and the conclusive results usually benefit those with a large consumer active usage existing. For a new app, one with no existing number of installs or reviews, engaging a large customer crowd would require greater consideration of factors like the reviews and downloads of other apps in that domain, the size and price of the app etc.

The processing of keywords from reviews and calculating numerical ranking is to gain an understanding of customer's needs from and critics of such apps which provides input for improvements for prospecting developers.

IV. OUR APPROACH

Since the raw dataset contains noisy data and some null values, we cleaned the data by checking null values and redundancy, along with necessary modifications such as making the number of installs column purely numeric by removing the '+' (ex. converting 15,000+ to 15000) and taking the lower bound. Similar deletions in the suffixes of the 'Size' and 'Price' columns were also performed.

We converted all the app sizes into MB, dropped columns like 'Current Ver' and 'Android Ver' since the information is not useful for our analysis. We also performed basic Exploratory Data Analysis (EDA) to check the basic relationships between the attribute columns.

We carried out app category specific analysis calculating the number of apps per domain, the average rating and average

size. Through plots we depicted the relationships between different variables like standardized ratings versus standardized number of downloads. Categorizing the clusters and studying them separately showed some apps with high rating and low number of downloads which could be provided a recommendation to focus more towards marketing. However, the number of reviews is also significant and to be considered to make this conclusion since the rating could be high simply because very few people rated it.

Fig 1. Plot showing Rating vs Reviews

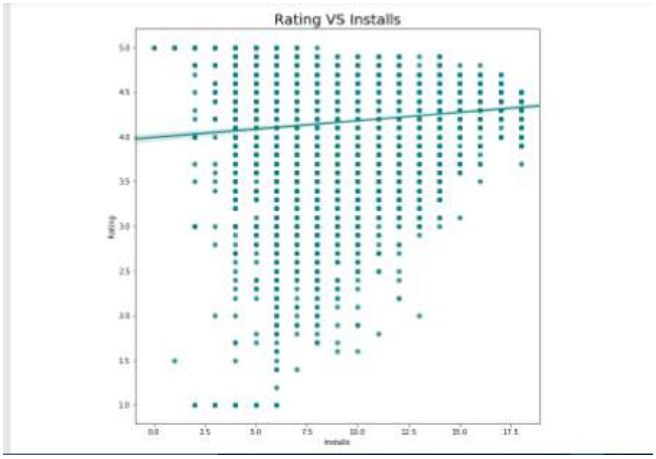

Fig 2. Plot showing Rating vs Number of Installs

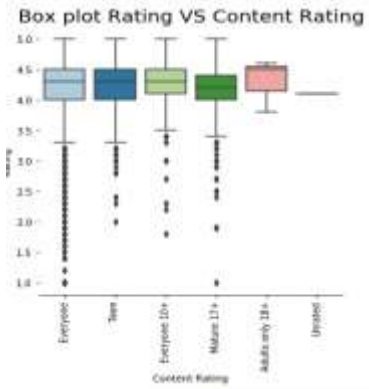

Fig 3. Plot showing Box Plot Rating vs Content Rating

Our approach towards this research focuses on drawing conclusions that would help an emerging app attract consumers in the market. Our research first looks upon finding relationships between variables considering number of downloads as an estimator for app success through data modelling technique Random Forest Regressor for relative feature importance determination of variables including size of the app, type of the app, app age rating(everyone, 10+,15+,adult etc.), along with title extracted details like title word count, title character count etc.

For answering the second question in our research study, we would perform NLP and extract keywords from customer reviews in the specific domain or general to assess what improvements the new app could focus on. For extracting feature analysis on the reviews of other apps in the same domain, we took the domain of the app as input from the user. As the kaggle dataset had one dataset with app specifications and another with the user reviews, we merged the two of them to get category specific analysis. We extracted the rows with the same domain as the one entered by the user. Then we preprocessed the reviews (text data) by removing some hard coded stopwords, punctuations, expanded the contractions in the text, removed various special characters (like :), xD, etc) and performed Stemming on all the reviews.

We also did Sentiment Analysis on every review which resulted in 2 scores for each of them. The first score is the sentiment polarity indicating whether the sentiment is positive or negative. The second score is the subjectivity score which tells us how subjective the text is. We formed a word cloud of the preprocessed text data which could provide category-specific critic review keywords to an app developer. These may be used to understand the customers' likes and dislikes about apps of that category and enable developers and marketers to focus more on improvements in those aspects.

Since the dataset is not very large and the number of apps in each genre vary greatly, with some having significantly less number of apps, the implementation of domain-specific analysis may limit accuracy.

V. RESULTS AND CONCLUSION

For the next part, the app title has been analyzed, computing word count, char count, symbols count, and whether the word 'free' is present in the title. These computed variables along with app size, app type, app rating(age) is fed as input to a RandomForestRegressor to identify feature importance. The resultant plot indicates that app size, and character count of an app have greater relative importance towards app number of installs which has been taken as the app popularity indicator in this model. This feature importance plot is compared with one with all attributes including established app attributes like reviews and ratings to compute number of installs. Number of reviews and rating of an app majorly contribute to the number of users of the app however the plot suggests that number of characters in the app name/title, and the app size follow in feature importance as suggestive points for a new-to-market app, to fare well in market and build a significant customer base.

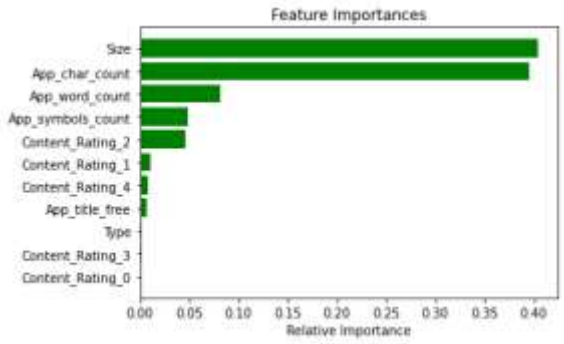

Fig 4.a. Relative importance of new-to-market app attributes

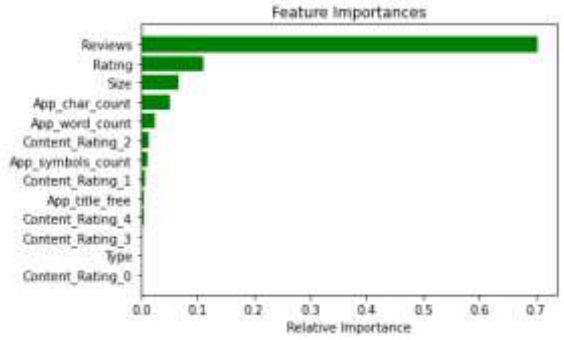

Fig. 4.b. Relative importance of all attributes for a category

The results of the second part of analysis: Sentiment Analysis and the Wordcloud could be used by a user to find which app in a specific domain is the best and how the others could improve.

Fig 5. Word Cloud for Category Art_and_Design

For a given app category, the sentiment average and subjectivity average of all the reviews of the apps in that category are displayed along with a word cloud of the keywords of all the reviews of that category.
The sentiment average polarity suggests if the overall sentiment of that app is positive, negative or neutral(0).
The score average indicates how subjective the reviews are.

```
Easy Origami Ideas :
        Sentiment polarity avg:  0.11695078431372549
        Subjectivity score avg:  0.42102941176470588
Animated Photo Editor :
        Sentiment polarity avg:  0.22781714285714288
        Subjectivity score avg:  0.44809523809523805
ColorFit - Drawing & Coloring :
        Sentiment polarity avg:  0.17603284222677912
        Subjectivity score avg:  0.5349778733652314
Boys Photo Editor - Six Pack & Men's Suit :
        Sentiment polarity avg:  0.27982456140350877
        Subjectivity score avg:  0.38771929824561396
FlipaClip - Cartoon animation :
        Sentiment polarity avg:  0.4191666666666667
        Subjectivity score avg:  0.6666333333333334
Canva: Poster, banner, card maker & graphic design :
        Sentiment polarity avg:  0.20483956893496366
        Subjectivity score avg:  0.5368809824483588
Floor Plan Creator :
        Sentiment polarity avg:  0.2660188643188643
        Subjectivity score avg:  0.3866871416871417
Coloring book moana :
        Sentiment polarity avg:  0.15681818181818183
        Subjectivity score avg:  0.4710858585858585
Garden Coloring Book :
        Sentiment polarity avg:  0.2342251683501684
        Subjectivity score avg:  0.4189311107227836
Anime Manga Coloring Book :
        Sentiment polarity avg:  0.26300505050505046
        Subjectivity score avg:  0.46122685185185188
```

Fig 6. Average of sentiment analysis scores for each app for Category Art_and_Design

For the third part, we used Rating as a measure of app success, with Random Forest Regression and kNN modelling.

```
[<matplotlib.lines.Line2D at 0x7f7e9a590280>]
```

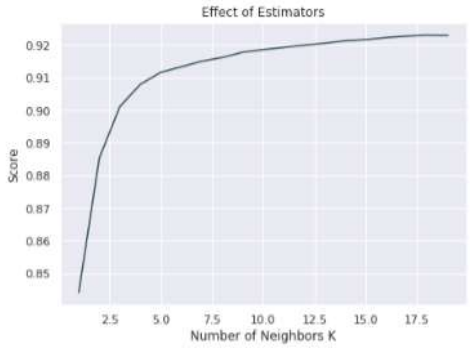

Fig 7. Effect of estimators for kNN model

```
model = KNeighborsRegressor(n_neighbors=18)
```

```
model.fit(X_train, y_train)
```

```
KNeighborsRegressor(n_neighbors=18)
```

```
accuracy = model.score(X_test,y_test)
'Accuracy: ' + str(np.round(accuracy*100, 2)) + '%'
```

```
'Accuracy: 92.3%'
```

Fig 8.Accuracy of KNeighboursRegressor

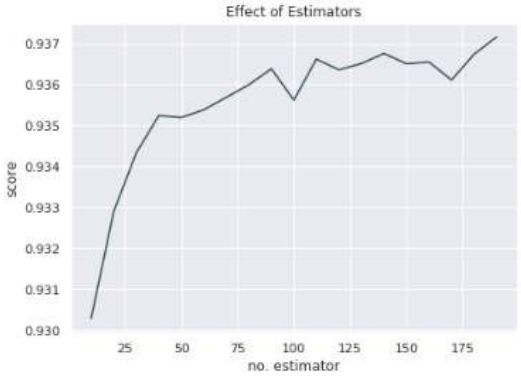

Fig 9. Effect of estimators for RandomForest model

Performance evaluation:
This is an important aspect of the machine learning modelling.
 MSE: Mean Squared Error is the mean of the square of the difference between the original and predicted values of the data.
 RMSE: Root Mean Squared Error is same as MSE but the root of the value is considered while estimating the accuracy of the model.
 MAE: Mean Absolute Error is the absolute difference between the actual value and the predicted values .

Mean Squared Error	0.16166632321724883
Root Mean Squared Error	0.40207750896717515
Mean Absolute Error	0.2450226581213182

Fig 10. Performance Evaluation of Random Forest model

VI. FUTURE WORK

For future work, we would like to optimize our analysis of category specific user reviews and to provide a more customized suggestion. For instance, a particular app would be provided suggestions based on apps most like it in terms of app specifications. Predicting how the app would fare taking into consideration the popularity of other apps in that domain.
Performing analysis to determine which variables as a dependent variable (for example number of downloads, rating, sentiment computed from reviews, etc) give better conclusive results to determine the popularity of the app.

VII. CONTRIBUTION

Our problem statement describes the exploration of three questions through our research. Each of us took upon researching and analyzing one question each-
New-to-market app attribute analysis and contribution towards number of installs using Random Forest Regressor for feature importance extraction and analysis was done by Vedica Rao, Category-specific Reviews' keyword extraction, sentiment average and subjectivity computation done by Archana J, and the Prediction of App Rating using KNeighboursRegression and RandomForest with MSE, RMSE and MAE performance Evaluation performed by Swati Naik.
The data cleaning and exploration was performed by all three of us.

VIII. REFERENCES

[1]Rimsha Maredia. "Analysis of Google Play Store Data set and predict the popularity of an app on Google Play Store. " 2020

[2]Islam, Mir Riyanul. "Numeric rating of Apps on Google Play Store by sentiment analysis on user reviews." *2014 International Conference on Electrical Engineering and Information & Communication Technology*. IEEE, 2014.

[3]Ahsan Mahmood, "Identifying the influence of various factor of apps on google play apps ratings" "Springer Nature Switzerland AG 2019"