

Google Play Store Analysis

Prathamesh Chavan^{1,*}, Kaushal Gadaria², Dheeraj Deore², Manasee Deore³, Supriya Joshi⁴

Abstract

Google play store consists of millions of applications and several thousand apps are added on the play store every day. The competition is so fierce that it is really difficult for the developers to find out whether the app that is the product of his hard work is going to be successful or not. The main goal of the study is to create a tool that helps developers and organizations to understand the current trends in the Play Store and the user bias for applications and utilize the data for making optimal decisions regarding their upcoming app. The success of an app can be determined by factors like ratings, number of installs and reviews. In this study we demonstrate using applied exploratory data analysis to discover correlations among features of an app and direct contribution of features towards success of the app to predict which apps will succeed. Data from Google Play Store was used to train various supervised learning models for prediction of the rating of an app, the models being-Random Forest, Support Vector Machine and Linear Regression. The user reviews are used to learn input bias of users towards applications using natural language processing techniques. Data of reviews is used to train classification models to predict polarity and sentiment of reviews.

Keywords: Google play, app ratings, sentiment analysis, linear regression, support vector machine

INTRODUCTION

It has been observed that the significant growth of the mobile application market has a great impact on digital technology. Mobile app industry is booming in a significant way and thus this is giving rise to more competition among the developing community, those who create applications. This is kind of a threat to each individual developer as a greater number of competitions will increase the chance of degrading the value of their app in the store. Therefore, even if an app is launched and has reached a standard rating along with a strong number of installs, it is not enough to retain their position in the Play store forever because there will be many other apps which will be providing the end-users with exactly the same features or even better. Having said that, with the ever-growing mobile app market there is also a notable rise of mobile app developers; eventually resulting in sky-high revenue by the global mobile app industry. With immense competition from all over the globe, it is imperative for a developer to know that he is proceeding in the correct direction. To retain this revenue and their place

***Author for Correspondence**
Prathamesh Chavan

^{1,2}Students, Department of Information Technology, A.C. Patil College of Engineering, Mumbai, Maharashtra, India

³Professor, Department of Information Technology, A.C. Patil College of Engineering, Mumbai, Maharashtra, India

⁴HOD, Department of Information Technology, A.C. Patil College of Engineering, Mumbai, Maharashtra, India

Received Date: May 21, 2021

Accepted Date: May 25, 2021

Published Date: May 29, 2021

Citation: Prathamesh Chavan, Kaushal Gadaria, Dheeraj Deore, Manasee Deore, Supriya Joshi. Google Play Store Analysis. Journal of Open Source Developments. 2021; 8(1): 11–25p.

in the market the app developers might have to find a way to stick into their current position. The Google Play Store is found to be the largest app market in the world. It has been observed that although it generates more than double the downloads than the Apple App Store, it makes only half the money compared to the App Store. So, we scraped data from the Play Store to conduct our research on it. Recently, a good number of researches have been made on this data that we will discuss in Literature of Review [1, 2].

LITERATURE REVIEW

In a research Abdul Mueez, the author scraped data from the Google Play Store to extract as many

features as possible using a web crawler ‘scrapy’ and used the data to train three models to predict the success of an application [3, 4]. To predict success metrics of an app, revenue should be the key feature but as it is not found publicly, the author used the number of installations and average user rating, trained linear model to categorize whether an app will be successful or not and additionally used linear regression to predict the average rating of a system [5]. A system developed by Google also works on the updates. Thus they tracked/mined the data everyday for the application rating and based their analysis around it [2].

PROPOSED SYSTEM

The proposed system is based on supervision which states that the numeric rating as in the learning technique where the model is trained using stars given by users have a huge difference than examples with labels. The data set is divided into two compared to the reviews given by them thus a rating parts of 70–30, 75–25, 80–20 or even 85–15. The system has been proposed by the author which is greater in the training set on which the model is removed to stabilize created by the mismatch of trained and the smaller set is the testing set on the rating and respective review by the same user. It has been seen that how much we as users are dependent on others’ opinion while taking any decision; so according to the author, people install the app based on the rating that is given for that particular app. Here the author describes the problem by dividing it into two sub-problems. Firstly, the ambiguity and secondly it is the biases to the summarized rating of the users. Further explaining the problem, he added that earlier people used to only extract the rating from the comments rather including the star rating associated with it. To solve the problems, he proposed a system that will initially conduct sentiment analysis on the user reviews and will generate a numeric rating from the polarity.

In another paper by Ruiz [1], the author mined the data of the play store everyday in the year 2011. They mined the ratings of the applications along with the number of people that have rated the applications. They were hard set on the analysis based on the ratings of the applications and were picky enough to choose only those applications that were rated by a large number of people. They removed the data about the applications with few ratings. They wanted to capture the user’s satisfaction with the application throughout a certain period where the applications are received which were once the model is trained tested upon to see the accuracy of the prediction. The data set, scraped from the play store contains 9659 unique applications data fields and over 64000 reviews.

Implementation

The dataset which my group has utilized has over 10,000 entries and for sentiment analysis the data set has over 150,000 entries as represented by Figure 1.

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
6831	Bu Hangi Ünlü?	GAME	NaN	1	19M	10+	Free	0	Everyone	Trivia	June 9, 2018	3.3.7z	4.0.3 and up
9815	ES Solar	BUSINESS	5.0	3	4.7M	100+	Free	0	Everyone	Business	May 17, 2018	1.2	4.1 and up
618	FREE LIVE TALK	DATING	4.9	776	4.9M	5,000+	Free	0	Mature 17+	Dating	July 14, 2018	8.2	4.0.3 and up
9390	US Army Training Courses Game	FAMILY	4.1	27130	35M	5,000,000+	Free	0	Everyone	Simulation	December 8, 2017	1.4	2.3 and up
7736	Pocket Prefix Plus	COMMUNICATION	4.5	192	15M	10,000+	Free	0	Everyone	Communication	February 12, 2018	v2.6.0	3.0 and up
563	Mutual - LDS Dating	DATING	3.7	1439	38M	50,000+	Free	0	Mature 17+	Dating	July 30, 2018	1.1.46	4.2 and up
353	MegaFon Dashboard	COMMUNICATION	3.7	99559	Varies with device	10,000,000+	Free	0	Everyone	Communication	July 30, 2018	Varies with device	Varies with device
4023	Learn C# - .Net - C Sharp Programming Tutorial...	FAMILY	4.0	1320	8.6M	100,000+	Free	0	Everyone	Education	June 3, 2018	4.1.0	4.1 and up
7172	CD Music	LIBRARIES_AND_DEMO	NaN	1	2.3M	100+	Free	0	Everyone	Libraries & Demo	March 30, 2018	1.0	4.4 and up
8844	DS Thermometer	WEATHER	3.7	631	3.0M	100,000+	Free	0	Everyone	Weather	May 30, 2015	2.20	2.3 and up

Figure 1. Sample of dataset.

This is what the initial dataset looks like. The dataset has various fields which need to be cleaned and all the category types have been converted to numeric form which makes it easy to compare them. The type category which contains data about the application being free or paid is also converted to numeric-free (1) and paid (0).

Data Processing

While transforming the data from text to numeric form we noticed some discrepancy in the paid applications. The most expensive of the applications were placed at 250\$!! Therefore we had a look at these applications and found out that they were motiveless status quo apps with no real use or meaning. Hence we decided to delete them from the data since they are unimportant and created a heavy bias in the average pricing of applications. Figure 2 shows these applications.

Exploratory Data Analysis

After processing the data and visualizing the data in the form of graphs and charts, here is the generated output obtained from the analysis of the data.

The graph in Figure 3 provides the frequency distribution of the applications in various different categories.

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
4197	most expensive app (H)	FAMILY	4.3	6	1.500	100.0	Paid	399.99	Everyone	Entertainment
4362	I'm rich	LIFESTYLE	3.8	718	26.000	10000.0	Paid	399.99	Everyone	Lifestyle
4367	I'm Rich - Trump Edition	LIFESTYLE	3.6	275	7.300	10000.0	Paid	400.00	Everyone	Lifestyle
5351	I am rich	LIFESTYLE	3.8	3547	1.800	100000.0	Paid	399.99	Everyone	Lifestyle
5354	I am Rich Plus	FAMILY	4.0	856	8.700	10000.0	Paid	399.99	Everyone	Entertainment
5355	I am rich VIP	LIFESTYLE	3.8	411	2.600	10000.0	Paid	299.99	Everyone	Lifestyle
5356	I Am Rich Premium	FINANCE	4.1	1887	4.700	50000.0	Paid	399.99	Everyone	Finance
5357	I am extremely Rich	LIFESTYLE	2.9	41	2.900	1000.0	Paid	379.99	Everyone	Lifestyle
5358	I am Rich!	FINANCE	3.8	93	22.000	1000.0	Paid	399.99	Everyone	Finance
5359	I am rich(premium)	FINANCE	3.5	472	0.965	5000.0	Paid	399.99	Everyone	Finance
5362	I Am Rich Pro	FAMILY	4.4	201	2.700	5000.0	Paid	399.99	Everyone	Entertainment
5364	I am rich (Most expensive app)	FINANCE	4.1	129	2.700	1000.0	Paid	399.99	Teen	Finance
5366	I Am Rich	FAMILY	3.6	217	4.900	10000.0	Paid	399.99	Everyone	Entertainment
5369	I am Rich	FINANCE	4.3	180	3.800	5000.0	Paid	399.99	Everyone	Finance
5373	I AM RICH PRO PLUS	FINANCE	4.0	36	41.000	1000.0	Paid	399.99	Everyone	Finance
9719	EP Cook Book	MEDICAL	NaN	0	3.200	0.0	Paid	200.00	Everyone	Medical
9917	Eu Sou Rico	FINANCE	NaN	0	1.400	0.0	Paid	394.99	Everyone	Finance
9934	I'm Rich/Eu sou Rico/ أنا أغنياء/ 我很富有	LIFESTYLE	NaN	0	40.000	0.0	Paid	399.99	Everyone	Lifestyle

Figure 2. Application updates.

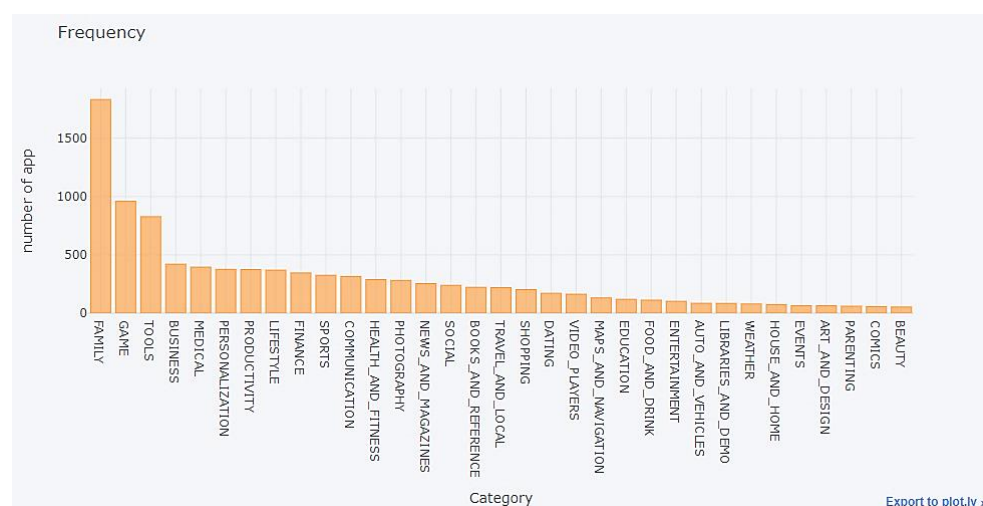


Figure 3. Frequency distribution of apps.

The graph in Figure 4 shows the distribution of the frequency of the applications according to their rating.

The bar chart in Figure 5 shows the frequency distribution of the applications in a grouped fashion according to their number of installs.

The bar chart in Figure 6 shows the numbers of installs of the applications category-wise.

Figure 7 plots the ratings of an application against their price, and it can be observed that most applications are priced between \$1 and 30.

The bar chart in Figure 8 shows the frequency distribution of the free applications and the paid applications categorically.

Figure 9 plots the size of an application against their ratings, and it can be observed that highly rated applications are small in size.

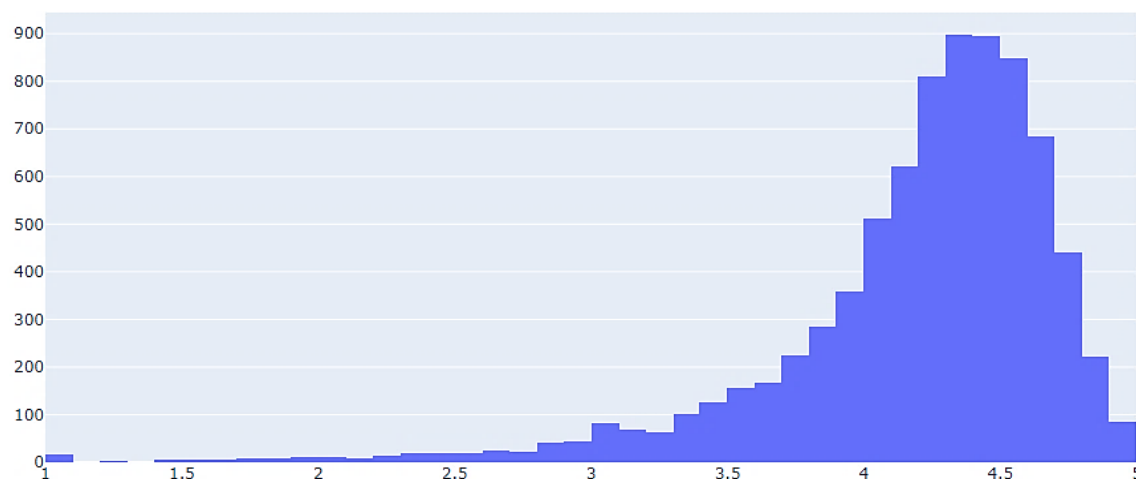


Figure 4. Frequency distribution according to ratings.

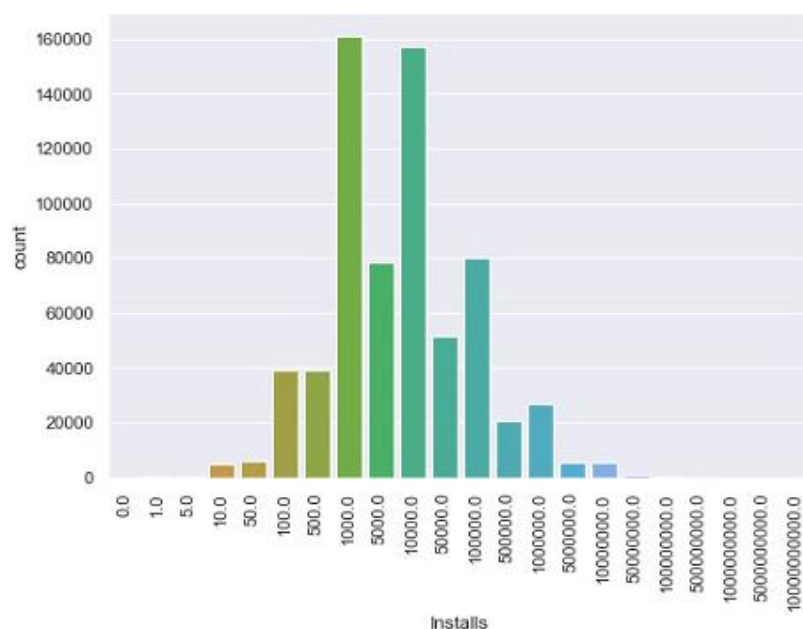


Figure 5. Grouped Fashion Chart.

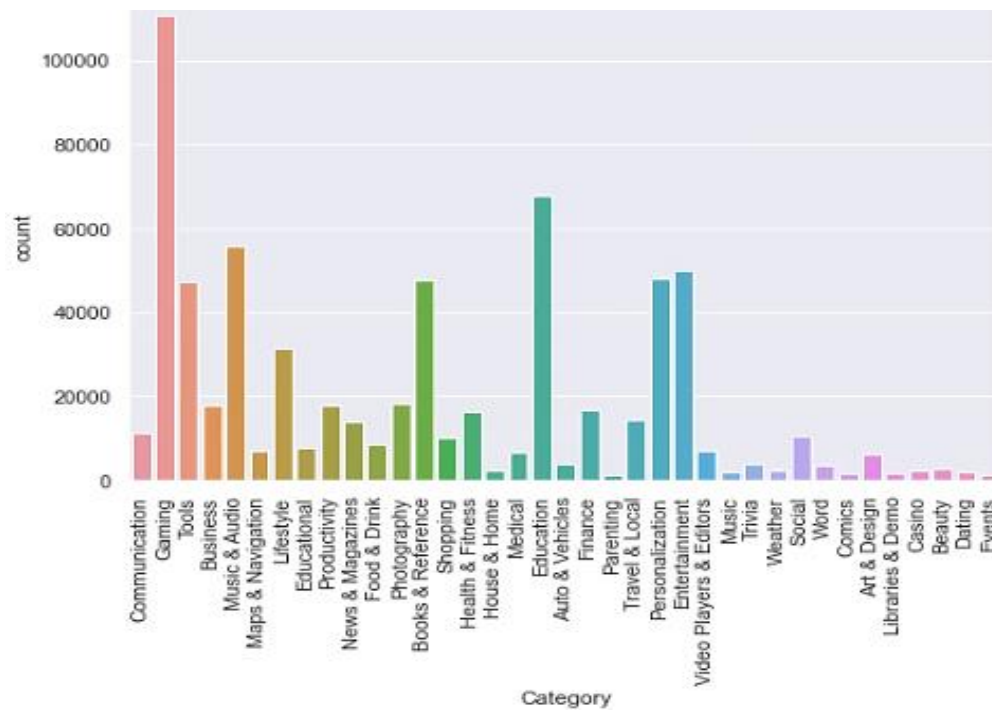


Figure 6. The bar chart.



Figure 7. Plotting of ratings.

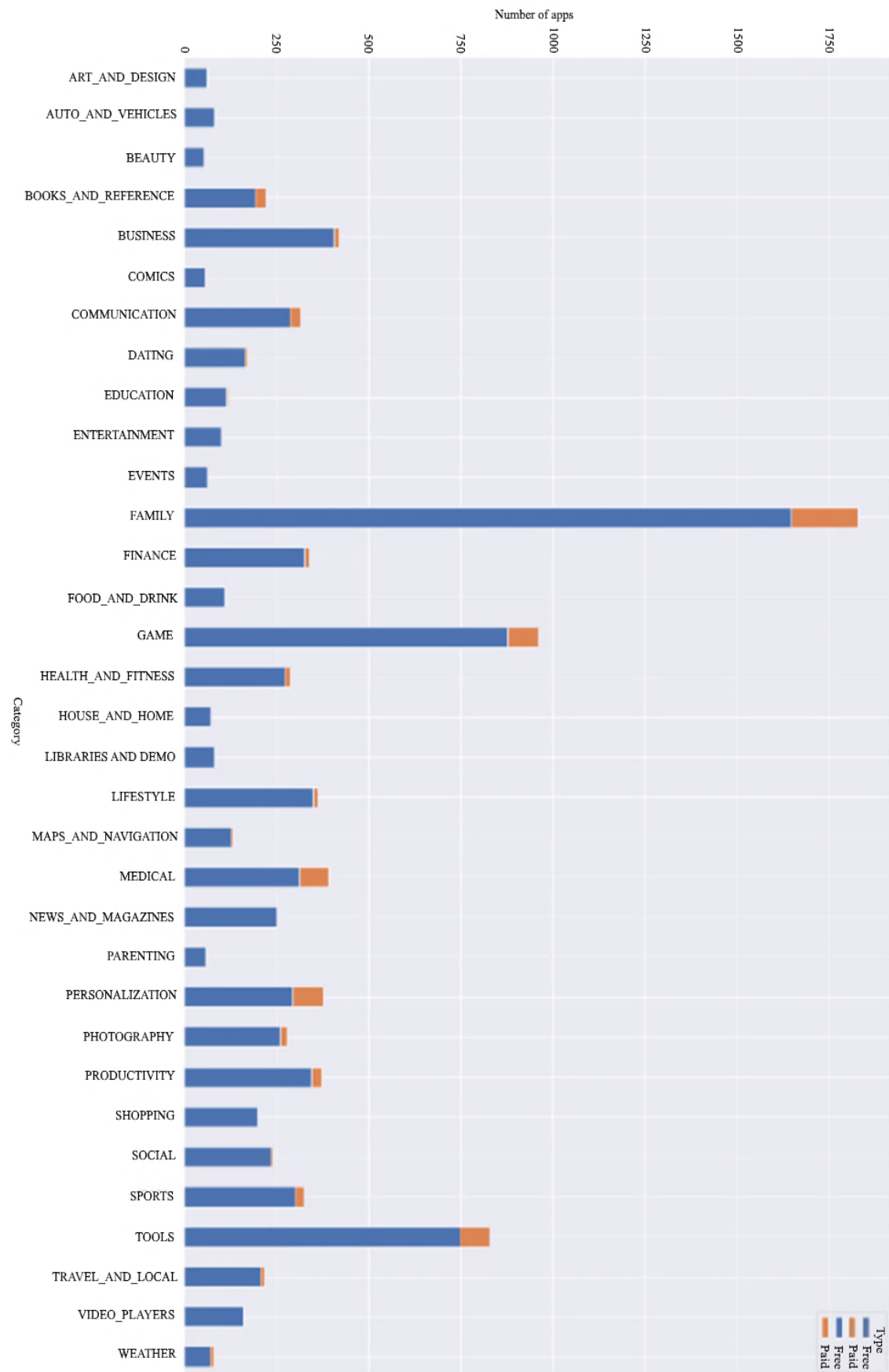


Figure 8. Bar Chart.

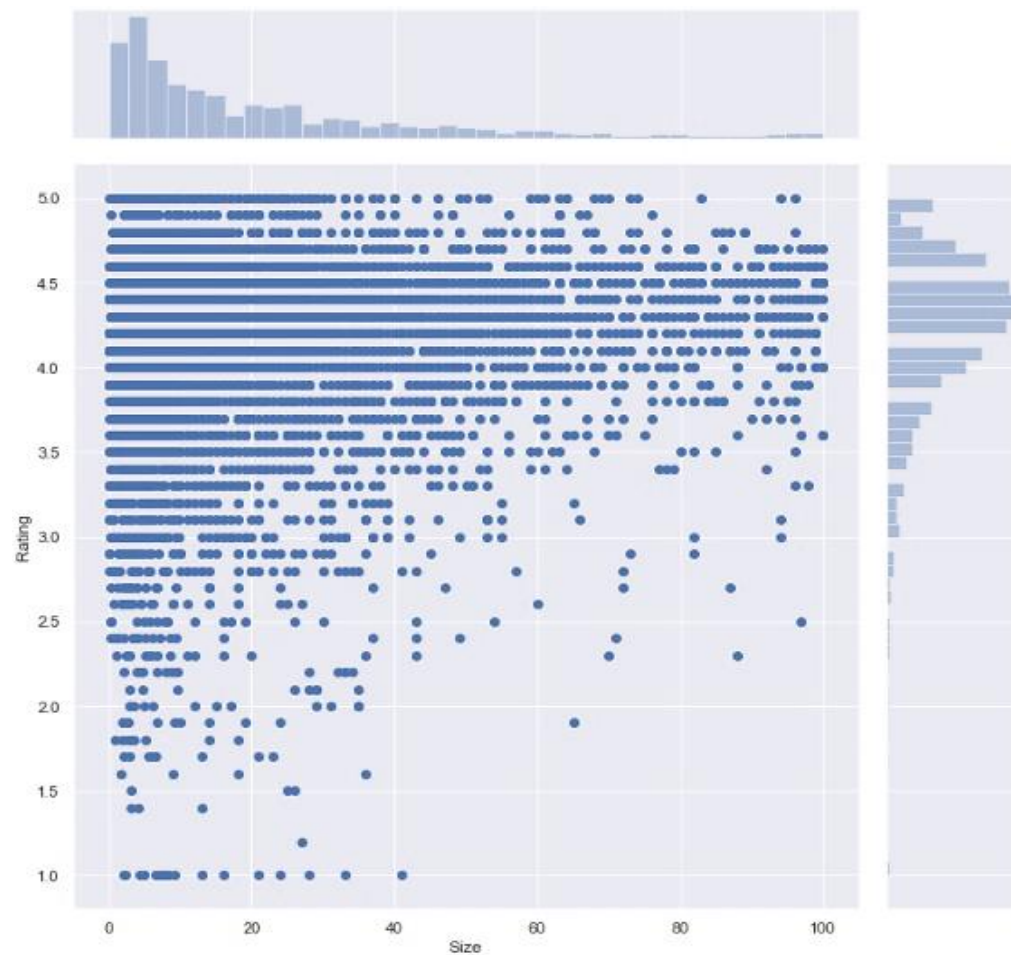


Figure 9. Representation of size of application.

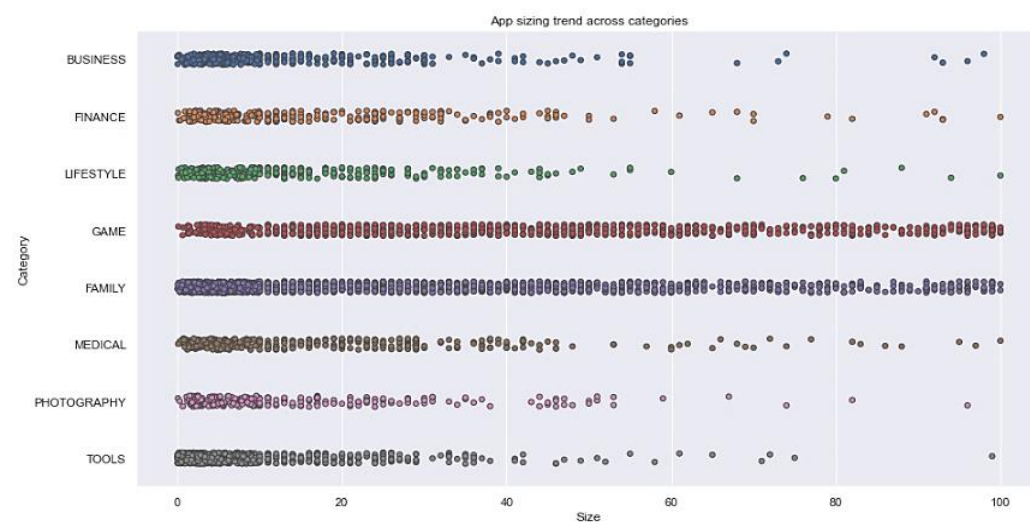


Figure 10. Apps according to their sizes.

The chart in Figure 10 shows the frequency distribution of the application according to their sizes categorically.

The bar charts in Figure 11 shows the frequency distribution of the applications in various categories according to their ratings.

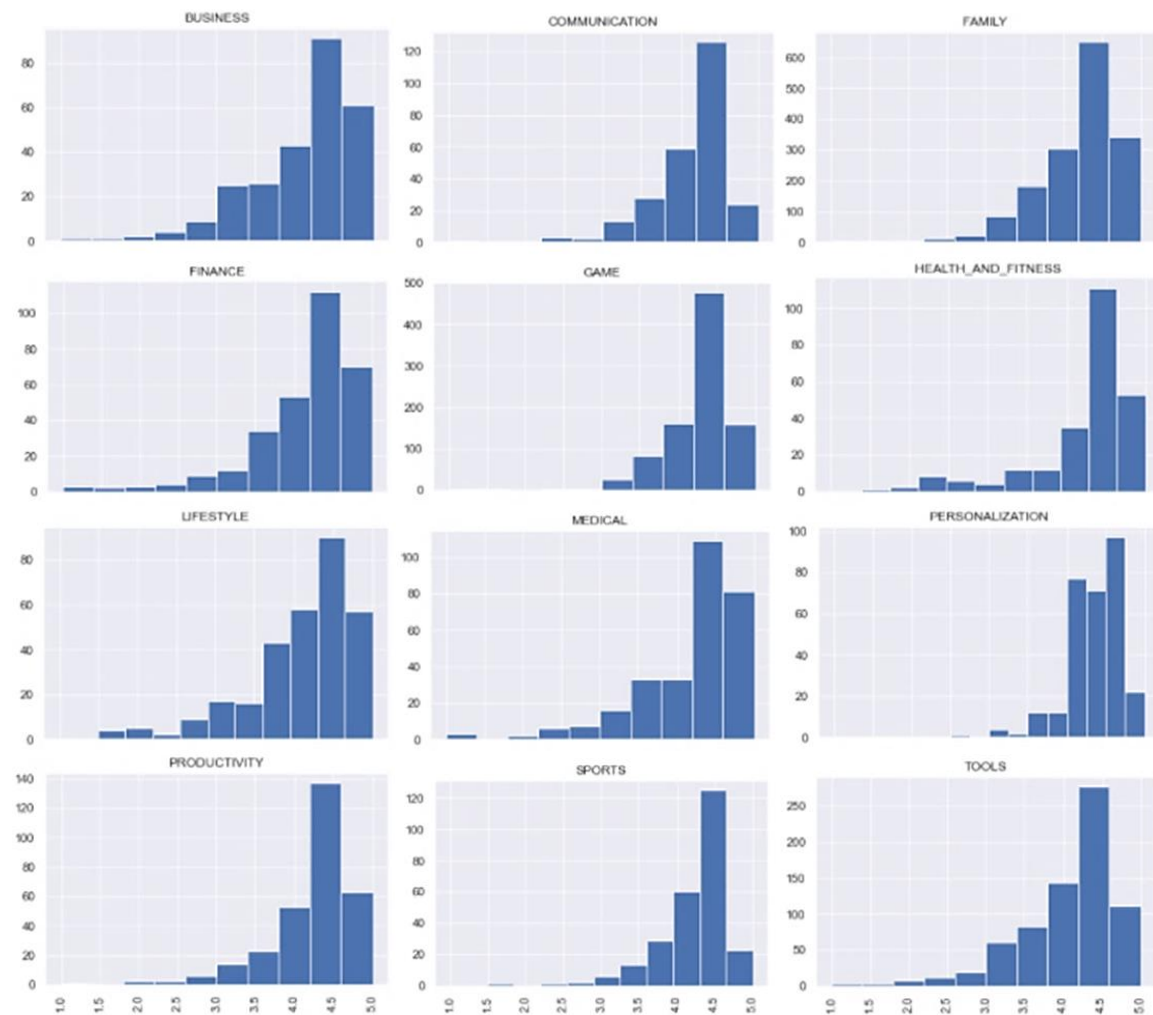


Figure 11. Frequency distribution of apps in various categories.

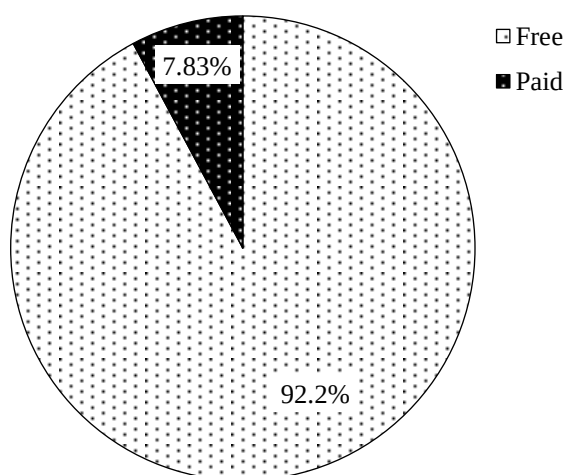


Figure 12. Pie chart.

The pie chart in Figure 12 tells us that free apps completely overshadow paid apps, both in terms of number of apps and installs. Free apps enjoy a dominating market share of about 92%. However some of the applications that are paid are also included in the free category because their demo or trial

versions are free to install. Subscription apps like Netflix and Crunchy roll are also free to download, it is only that usage of their service which is actually chargeable. However this does not create any bias in the data set as the number of such apps is very insignificant.

The logarithmic box graph in Figure 13 shows that even though free apps are more readily available they are not downloaded all that much since the gap between the number of downloads is not huge. This also tells that all the paid apps are very useful or in some cases, very essential services [6, 7.]

Number of downloads of paid apps Vs free apps

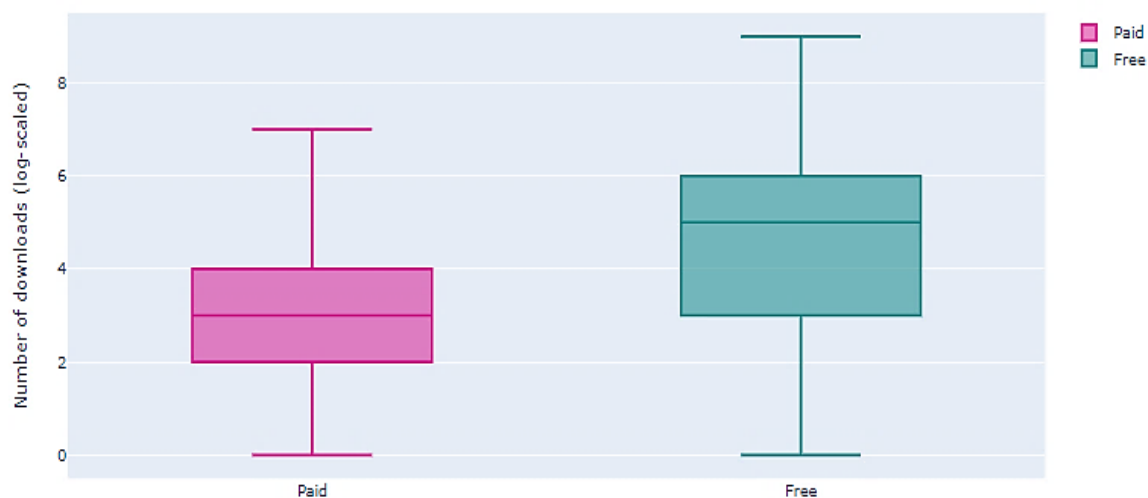


Figure 13. Number of downloads.

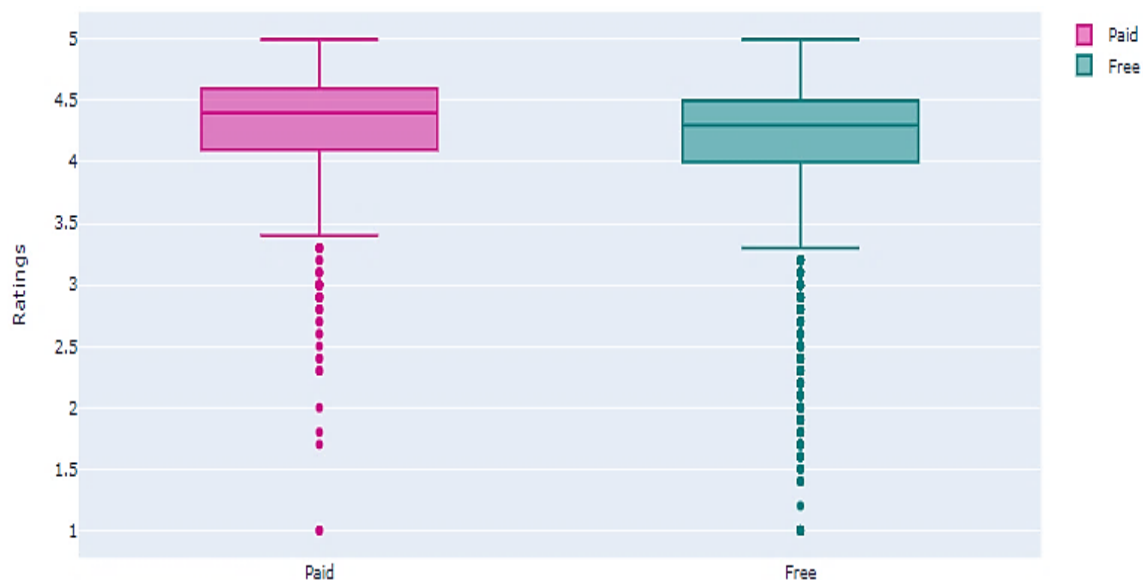


Figure 14. Difference of ratings.

Another logarithmic box graph in Figure 14 shows the difference of the ratings between the free and paid application. From the above graph, it can be observed that the both categories are pretty much equal when it comes to the rating which suggests there is not much difference of quality in these applications. Although we can note that the low for free apps is lower in comparison to the paid apps which suggests that people are generally harsher while rating a free app compared to when rating a paid app.

The last logarithmic box graph in Figure 15 compares the sentiment of the reviews received by both types of applications. Free apps receive a lot of harsh comments which are indicated as outliers on the negative Y-axis. Users are more lenient and tolerant while reviewing paid apps-moderate choice of words. They are never extremely negative while reviewing a paid app. This proves that users' opinions are more biased towards the paid application than the free application.

From the Figure 16 we can derive that most used words in a review in a positive manner are GOOD, LOVE, BEST, GREAT. While most used words in a negative manner are PROBLEM, AD, FIX, ISSUE.

For a detailed look at the sentiments regarding types of application we can see a clear bias when it comes to paid apps. A box plot of sentiment polarity comparing free apps to paid apps as shown below.

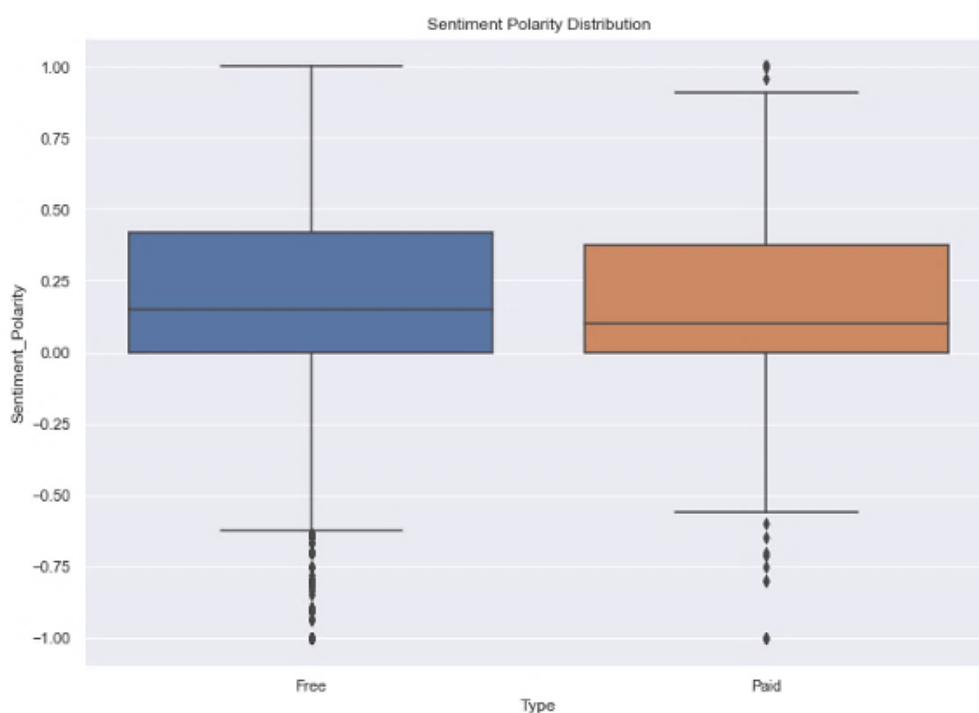


Figure 15. Logarithmic box.



Figure 16. Most used words.

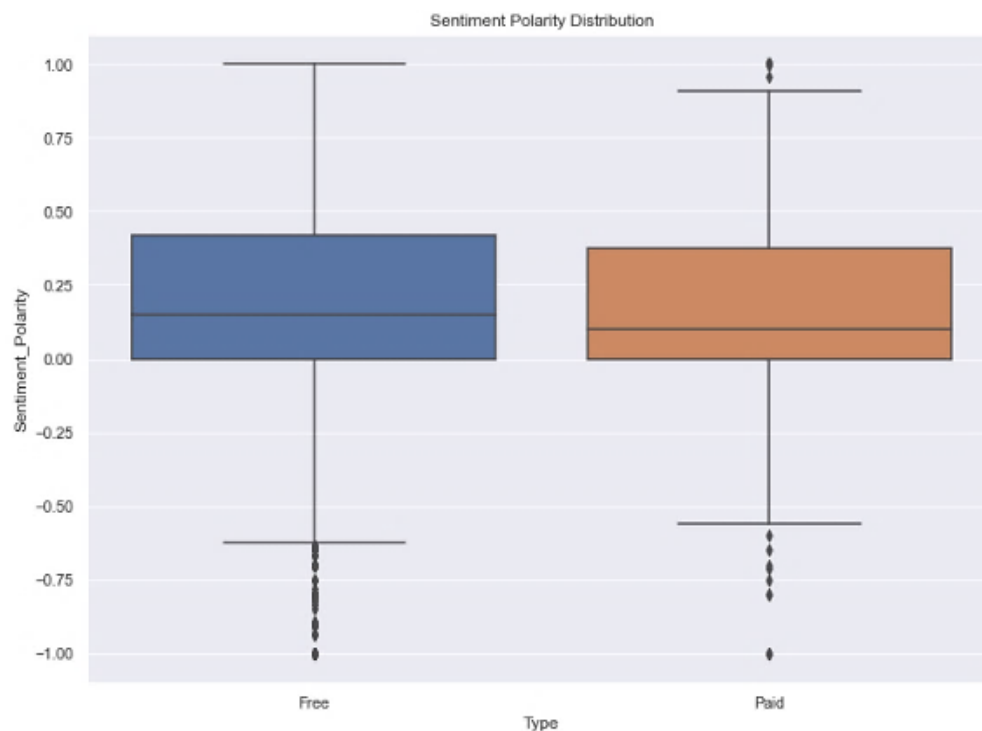


Figure 17. Sentiment polarity distribution.

Free apps receive a lot of harsh comments which are indicated as outliers on the negative Y-axis. Users are more lenient and tolerant while reviewing paid apps-moderate choice of words [8, 9]. They are never extremely negative while reviewing a paid app. This proves the observation we made in the research where we stated that user opinions are more biased towards paid apps than free apps.

Sentiment Analysis

For sentiment analysis we used the user reviews dataset on Kaggle which has over 64000 entries of reviews for over 1000 applications. The data has many null values, this is due to the fact that users can rate an application without writing a descriptive review and that rating is still counted as a review represented in Figure 17.

Therefore we have to drop these rows as they create a heavy bias in prediction of the polarity rating. The new data set has about 38000 entries in total [8].

The reviews are full-fledged grammatical sentences with punctuations, pronouns, etc. We perform operations on the dataset to first convert all reviews to lowercase and then remove any special characters i.e., '\$, #, !, @, ()'.

Then we created a word cloud to observe the frequency of words in the reviews. The figure in this research word cloud used in this project.

Prediction

We have discussed the ratings of an application being the most important parameter which directly indicates success of the application. In this section, we predict the average ratings of all the apps on the play store using all the important numerical parameters such as number of reviews, size, number of downloads, price, etc. [6].

Since ratings are a continuous parameter we use regression methods for determination. We use various models such as Linear Regression, Support Vector Machine and Random Forest Regression

for our calculations. We also calculate the losses produced by these methods, we use three metrics for error calculation namely MAE (mean absolute error), MSE (mean square error), and MSLE (mean squared log error).

Linear Regression

The Linear regression model performs fairly well on the play store dataset [4]. The model was run with a test size of 33% on a dataset of 9360 entries. The predicted value very closely resembles the ground truth. The Figure 17 plots the predicted ratings of the applications versus the actual ground truth. As we can observe in Figures 18 and 19, there is a lot of deviation in ratings between 4 and 4.2 but higher up in the graph the deviation is significantly low.

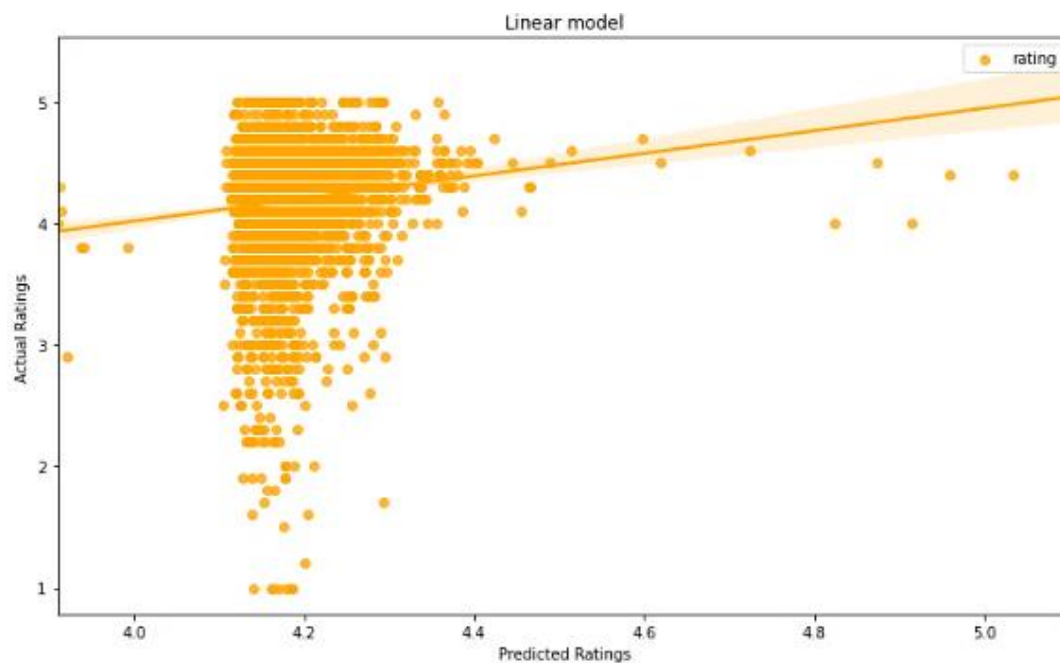


Figure 18. Mean Rating: 4.190023855763878.

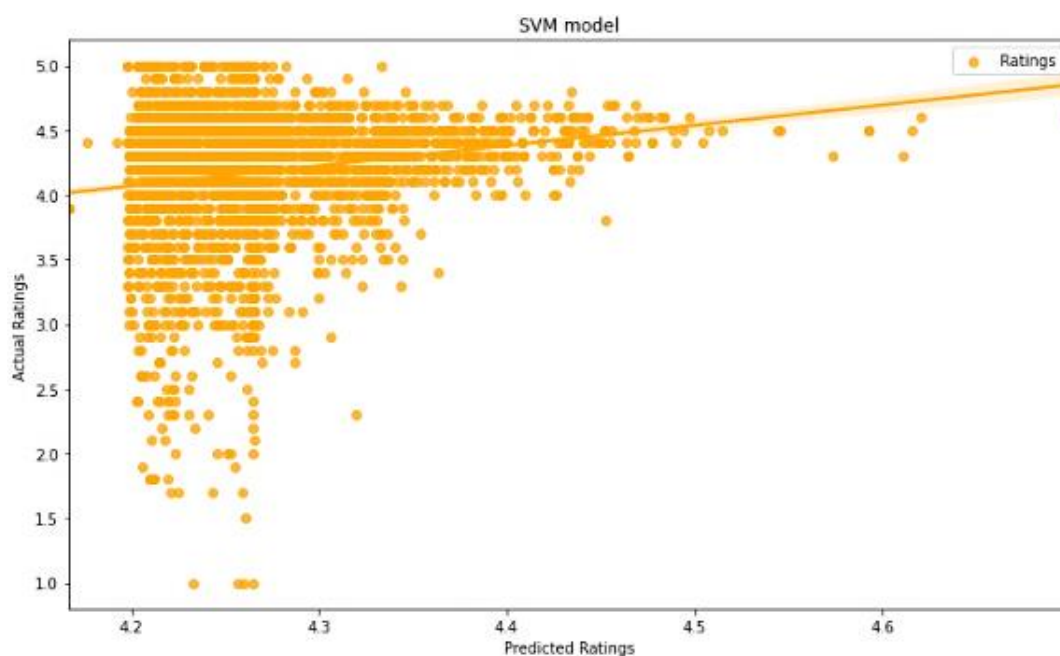


Figure 19. Mean Ratings: 4.272760678920066.

Support Vector Machine

At first, simple SVM was fit with the training data. However, due to the non-linearly separable nature of the data, the accuracy obtained was slightly lower than the linear model. Upon extensive runs with different test-train splits, the results were still less accurate than the other two models. SVM performed the worst on the Google play store data set.

Random Forest Regressor

Random Forest Regressor is an ensemble method that has characteristics of both regression and classification models. Random forest is a bagging technique and not a boosting technique. The trees in random forests are run in parallel. There is no interaction between these trees while building the trees. It operates by constructing a multitude of decision trees at training time and outputting, the Support Vector Machine on the contrary provides class that is the mode of the classes (classification), the highest degree of error when predicting on or mean prediction (regression) of the individual, the Google play store on dataset.trees. Therefore it is expected to perform the best among the three models we used in the study. The ratings predicted by the random forest method are closest to the ground truth; it performs the best on the Google play store dataset as shown in Figure 20.

Error Comparison

Figure 21 is the comparison of error calculations of all three models. We can observe that the Random forest model produces the lowest error on all three metrics of error calculation.

CONCLUSION

Free applications dominate the Play Store market and it will be like this the coming years. Even though free applications dominate the market, the number of downloads of the paid applications is very close to the number of downloads of the free applications. Both free and paid applications are equally rated based on their performance. Not just the price but size of an application is also critical factor in the making and breaking of an application. The highly rated ones will always will be the small and medium sized applications, and if you want to launch a large application, you must have a solidifying feature to justify its large size. From the sentiment analysis, we can observe that users really care about the performance of the paid app and are willing to help in their improvement by providing useful reviews and comments. Large proportions of the users do not really care about the performance of the free applications as they can be deleted without a second thought if they do not perform well.

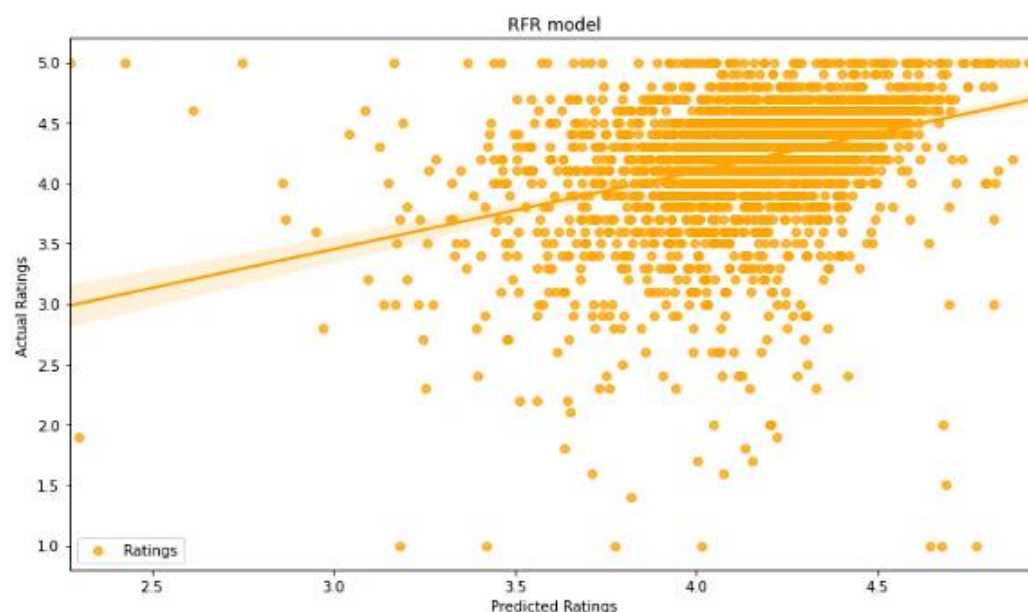


Figure 20. Mean Ratings: 4.1765151353276355.

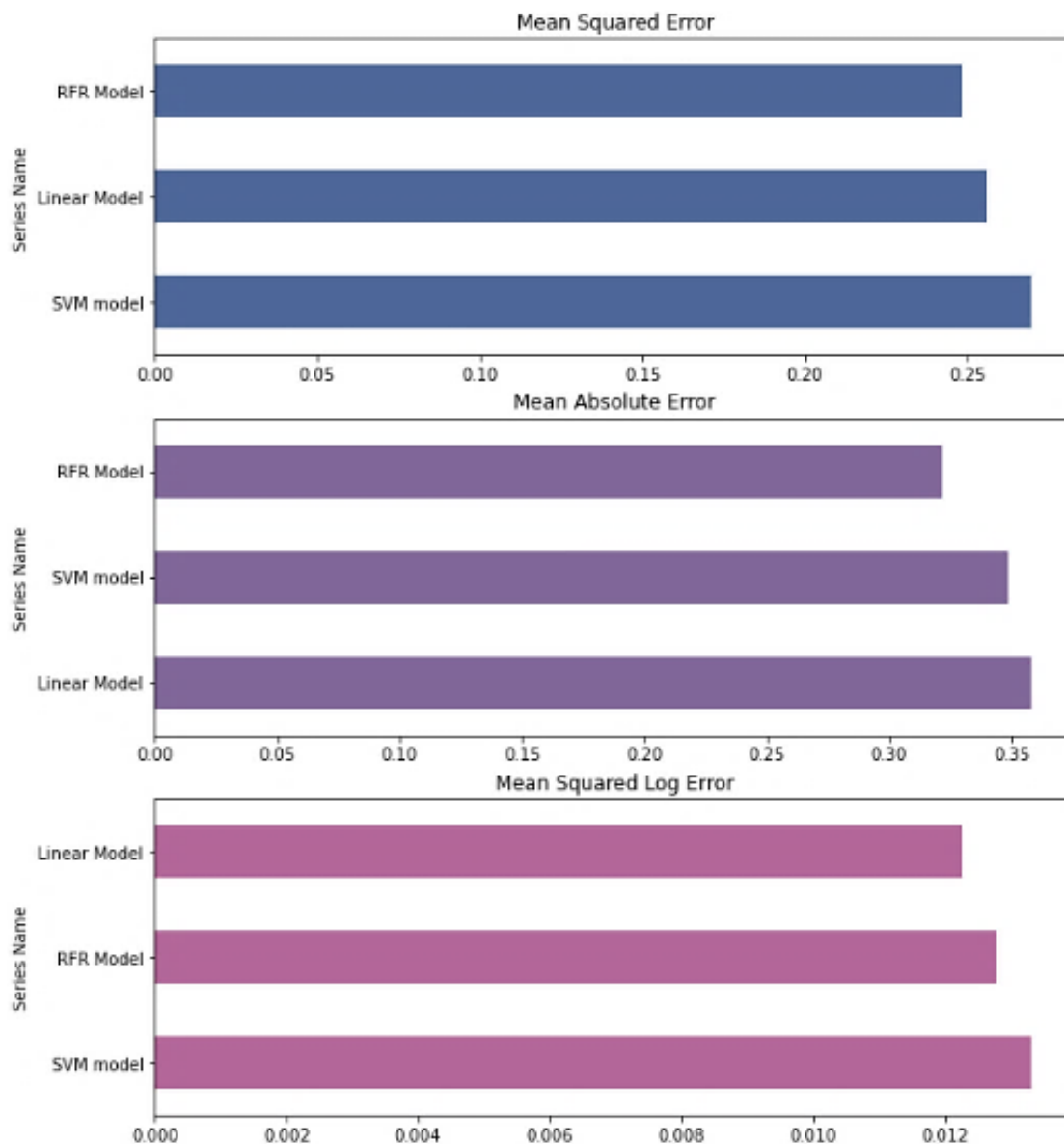


Figure 21. Comparison of error of three models.

ACKNOWLEDGMENT

Me and my group members would like to express our special thanks for gratitude to our guide Mrs. M. N. Deore and Mrs. S. P. Joshi for their guidance and their assistance in completing our project. They always helped us by showing us the right path for the project and clearing our doubts whenever approached by us at the various stages of the project.

REFERENCES

1. Tuckerman CJ. Predicting mobile application success. 2014 Dec.
2. Singh K, Wajgi R. Data analysis and visualization of sales data. 2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave). 2016; 1–6.
3. Muneez Abdul Ahmed, Khushba Islam, Tuba Iqbal, Waqqas. Exploratory Data Analysis and Success Prediction of Google Play Store Apps. BRAC University; 2018.
4. Cramer JS. The Origins of Logistic Regression. Tinbergen Institute. 2002 Dec.
5. Bird Steven, Loper Edward, Klein Evan. Natural Language Processing with Python. O'Reilly Media Inc.; 2009.

6. Aralikkatte R, Sridhara G, Gantayat N, Mani S. Fault in your stars: an analysis of android app reviews. In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data. 2018; 57–66.
7. Maskey Sameer. MapReduce for Statistical NLP/Machine Learning. Columbia University; 2012 Oct 17.
8. Harman M, Jia Y, Zhang Y. App store mining and analysis: MSR for app stores. In 2012 9th IEEE Working Conference on Mining Software Repositories (MSR). 2012; 108–111.
9. Ruiz IJM, Nagappan M, Adams B, Berger T, Dienst S, Hassan E. Examining the rating system used in mobile-app stores. IEEE Software. 2016; 33(6): 86–92.