A Major Project Report on

Google Play Store App Reviews And Sentiment Analysis

A dissertation submitted for 8th semester major project

Submitted By

1901206086	**Sourav Patnaik**
1901206040	**Abhijeet Kumar Sahoo**
1901206089	**Subhrajit Bastia**
1901206042	**Abinash Nayak**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AJAY BINAY INSTITUTE OF TECHNOLOGY

CUTTACK-753014

(2019– 2023) BATCH

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK

DECLARATION

We do hereby declare that the project report entitled, "Google Play Store App Reviews And Sentiment Analysis" submitted in the Department of Computer science and Engineering, Ajay Binay Institute of Technology, Cuttack of Biju Patnaik University of Technology, Odisha in partial fulfilment of requirement for the award of 8th semester BTech Degree in Computer science and Engineering is an authentic work carried out by us during 2023-2024 under the supervision of. The matter presented in this report has not been submitted by us in any other University/Institute for the award of BTech Degree.

Submitted By

1901206086	**Sourav Patnaik**
1901206040	**Abhijeet Kumar Sahoo**
1901206089	**Subhrajit Bastia**
1901206042	**Abinash Nayak**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK

CERTIFICATE OF APPROVAL

The project report named "Google Play Store App Reviews And Sentiment Analysis" is a bonafied work carried out by

1901206086	**Sourav Patnaik**
1901206040	**Abhijeet Kumar Sahoo**
1901206089	**Subhrajit Bastia**
1901206042	**Abinash Nayak**

In the partial fulfilment for the award of the degree of Bachelor of Technology in Computer science and Engineering, Ajay Binay Institute of Technology, Cuttack of Biju Patnaik University of Technology, Odisha in the year 2023-2024 is an authentic work carried out under our guidance and supervision.

The matter embodied in this report has not been submitted to any other university/institute for the award of any degree to the best of our knowledge.

Prof. Dr. RAJESH KUMAR SAHOO
Head of the Department of Computer science and Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK

CERTIFICATE

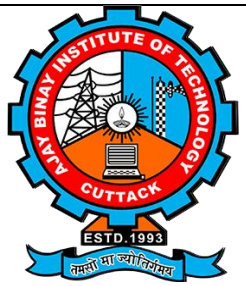
This is to certify that the project report entitled "Google Play Store App Reviews And Sentiment Analysis" is the work done by:

1901206086	**Sourav Patnaik**
1901206040	**Abhijeet Kumar Sahoo**
1901206089	**Subhrajit Bastia**
1901206042	**Abinash Nayak**

of BTech (Computer science and Engineering) of ABIT, Cuttack under BPUT, Odisha submitted in partial fulfilment for the award of degree.

We are satisfied that they have worked sincerely and with proper care.

H.O.D
Prof. Dr. RAJESH KUMAR SAHOO



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mentioning of the people whose constant guidance and encouragement made it possible. We take pleasure in presenting before you, our project, which is result of studied blend of both research and knowledge.

We express our earnest gratitude to the faculties of CSE, for their constant support, encouragement, and guidance. We are grateful for their cooperation and valuable suggestions.

We feel to avail ourself of this opportunity to express our deep sense of gratitude to Prof. RAJESH KUMAR SAHOO , HOD, Dept. of Computer science and Engineering, for the facilities made available and instructions given to us in accomplishing this project successfully.

Finally, we express our gratitude to all other members who are involved either directly or indirectly for the completion of this project.

1901206086	**Sourav Patnaik**
1901206040	**Abhijeet Kumar Sahoo**
1901206089	**Subhrajit Bastia**
1901206042	**Abinash Nayak**

CONTENTS

INTRODUCTION

DATA SCIENCE :-

Data science is an interdisciplinary field that involves extracting knowledge and insights from large volumes of data through various techniques and tools. It combines elements of statistics, mathematics, computer science, and domain expertise to analyze and interpret complex data sets, identify patterns, and make data-driven decisions.

The process of data science typically involves several stages:

Data Collection: Gathering relevant data from various sources, such as databases, APIs, or web scraping. This could include structured data (e.g., databases, spreadsheets) or unstructured data (e.g., text, images, videos).

Data Cleaning and Preprocessing: Transforming raw data into a structured and usable format. This includes handling missing values, removing noise, dealing with outliers, and normalizing or scaling the data.

Exploratory Data Analysis (EDA): Exploring and visualizing the data to gain insights, discover patterns, and identify relationships between variables. EDA helps in understanding the characteristics of the data and formulating hypotheses.

Feature Engineering: Creating new features or transforming existing ones to enhance the predictive power of the data. This step involves selecting relevant features, handling categorical variables, scaling numeric variables, and performing dimensionality reduction techniques.

Model Building: Selecting an appropriate algorithm or model to analyze the data and make predictions or classifications. This could involve techniques like linear regression, decision trees, support vector machines, neural networks, or clustering algorithms.

Model Training and Evaluation: Splitting the data into training and testing sets to train the model and evaluate its performance. Various metrics are used to assess the model's accuracy, such as accuracy, precision, recall, F1 score, or mean squared error, depending on the problem type.

Model Deployment and Monitoring: Implementing the model into a production environment, where it can be used to make predictions or support decision-making. Continuous monitoring and evaluation of the model's performance ensure its effectiveness over time.

Iterative Process: Data science is an iterative process, involving refining and improving the models based on feedback, incorporating new data, and adapting to changing business needs.

Data science has applications in various industries and domains, such as finance, healthcare, marketing, transportation, and more. It helps organizations gain insights, optimize processes, improve decision-making, identify trends, detect anomalies, and develop predictive models.

To carry out data science tasks effectively, data scientists utilize programming languages like Python or R, along with libraries and frameworks such as NumPy, pandas, scikit-learn, TensorFlow, or PyTorch. They also leverage statistical methods, machine learning techniques, data visualization tools, and cloud computing platforms to handle large-scale data processing and analysis.

Overall, data science plays a vital role in extracting value from data, enabling organizations to derive meaningful insights and make data-driven decisions that can drive innovation, efficiency, and competitiveness.

MACHINE LEARNING :-

Machine learning is a subfield of artificial intelligence that focuses on the development of algorithms and models that allow computers to learn from data and make predictions or decisions without being explicitly programmed. It involves training machines to recognize patterns, discover insights, and perform tasks by leveraging data and statistical techniques.

In machine learning, the process typically involves the following steps:
Data Collection: Gathering relevant data from various sources, such as databases, sensors, or APIs. The data can be structured (e.g., tabular data) or unstructured (e.g., text, images, audio).

Data Preprocessing: Cleaning and preparing the data for analysis. This involves handling missing values, dealing with outliers, normalizing or scaling features, and encoding categorical variables.

Feature Engineering: Selecting or creating informative features that represent the underlying patterns in the data. Feature engineering helps improve the performance and interpretability of machine learning models.

Model Selection: Choosing an appropriate machine learning algorithm or model based on the problem type and available data. This could involve regression models, classification algorithms, clustering methods, or deep learning models.

Model Training: Training the selected model using the prepared data. During training, the model learns the underlying patterns and relationships in the data by adjusting its internal parameters.

Model Evaluation: Assessing the performance of the trained model using evaluation metrics specific to the problem at hand. Common metrics include accuracy, precision, recall, F1 score, mean squared error, or area under the curve (AUC), depending on the task.

Model Tuning and Optimization: Fine-tuning the model by adjusting its hyperparameters to improve performance. This process may involve techniques like cross-validation, grid search, or Bayesian optimization.

Model Deployment: Deploying the trained model in a production environment, where it can make predictions or assist with decision-making. This may involve integrating the model into software systems or creating APIs for real-time predictions.

Model Monitoring and Maintenance: Continuously monitoring the performance of the deployed model, retraining it periodically with new data, and updating the model as needed to ensure its accuracy and effectiveness over time.

Machine learning techniques are used in various domains, including image and speech recognition, natural language processing, recommendation systems, fraud detection, autonomous vehicles, and many more. It provides powerful tools for extracting insights, making predictions, and automating tasks based on patterns and trends within data.

Popular machine learning algorithms and frameworks include linear regression, logistic regression, decision trees, support vector machines (SVM), random forests, gradient boosting, neural networks, and deep learning libraries like TensorFlow and PyTorch. These algorithms can be implemented using programming languages like Python or R, along with specialized libraries and frameworks.

Machine learning continues to advance rapidly, with ongoing research in areas such as deep learning, reinforcement learning, transfer learning, and interpretability. It has significant implications for industries, enabling organizations to leverage data to gain competitive advantages, automate processes, improve decision-making, and unlock new opportunities.

OUR PROJECT GOOGLE PLAYSTORE APP REVIEWS AND SENTIMENT ANALYSIS :-

Google Play Store app review and sentiment analysis involve analyzing the feedback and opinions shared by users of apps on the Google Play Store to understand their sentiments and assess the overall satisfaction or dissatisfaction with the app.

The Google Play Store is a platform where users can download and install various mobile applications for their Android devices. Users have the option to leave reviews and ratings for the apps they have downloaded, providing valuable feedback to app developers and potential users.

Sentiment analysis in the context of Google Play Store app reviews focuses on determining the sentiment expressed in these reviews, whether it is positive, negative, or neutral. This analysis helps app developers and businesses gain insights into user feedback and sentiments regarding their apps, identify areas for improvement, and assess the overall user satisfaction.

The process of Google Play Store app review and sentiment analysis typically involves the following steps:

Data Collection: Gathering the app reviews from the Google Play Store. This can be done through web scraping techniques or using the Google Play Store API to access the reviews programmatically.

Text Preprocessing: Cleaning and preprocessing the collected reviews to remove any irrelevant information, such as punctuation, special characters, and stopwords. Tokenizing the reviews into individual words or phrases and performing techniques like stemming or lemmatization may also be applied.

Sentiment Analysis: Applying a sentiment analysis algorithm or model to determine the sentiment of each review. Sentiment analysis algorithms can be rule-based approaches, machine learning models, or deep learning models. Rule-based approaches utilize predefined rules and dictionaries, while machine learning and deep learning models learn from labeled data.

Sentiment Classification: Classifying each review into positive, negative, or neutral sentiment based on the sentiment analysis results. Assigning a sentiment score or label to each review can help quantify the overall sentiment of the app's user base.

Visualization and Reporting: Analyzing and visualizing the sentiment distribution and patterns using charts or graphs. Presenting the findings in a comprehensive report, highlighting key insights, such as the overall sentiment score, most common positive and negative sentiments, and any notable trends.

Various tools and libraries can assist in performing Google Play Store app review and sentiment analysis, such as Natural Language Toolkit (NLTK), scikit-learn, TextBlob, or Google Cloud Natural Language API.

By conducting Google Play Store app review and sentiment analysis, app developers and businesses can gain valuable insights into user sentiments, address issues, improve app features, and enhance overall user satisfaction.

TECHNOLOGIES AND SOFTWARES USED

Several technologies and software tools can be utilized for Google Play Store app review and sentiment analysis. Here are some commonly used ones:

PYTHON:

Python is a popular programming language for data analysis and natural language processing (NLP) tasks. It offers a wide range of libraries and frameworks that facilitate text processing, sentiment analysis, and visualization.
It is used to write our code.

Python Libraries used:-

PANDAS:

Pandas is a popular open-source Python library used for data manipulation and analysis. It provides high-performance data structures and data analysis tools, making it an essential tool in the field of data science. Pandas is widely used for data preprocessing, exploratory data analysis, data wrangling, and data cleaning tasks. It simplifies the process of working with structured data, making it easier for data scientists and analysts to manipulate and analyze data efficiently.

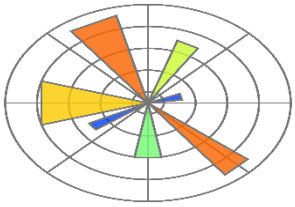
NUMPY:

NumPy (Numerical Python) is a powerful Python library used for numerical computations and scientific computing. It provides efficient data structures, array operations, and mathematical functions that enable high-performance computation and manipulation of multi-dimensional arrays. NumPy is widely used in various fields, including data analysis, scientific computing, machine learning, and numerical simulations. Its efficient array operations, mathematical functions, and numerical capabilities make it an essential tool for working with large datasets and performing complex computations efficiently.

SEABORN:

Seaborn is a Python data visualization library built on top of Matplotlib. It provides a high-level interface for creating aesthetically pleasing and informative statistical graphics. Seaborn is particularly useful for visualizing relationships and patterns in complex datasets. Seaborn simplifies the process of creating attractive and informative visualizations, particularly for statistical analysis. It is widely used in data exploration, data visualization, and communicating insights from complex datasets.

MATPLOTLIB:



Matplotlib is a widely used data visualization library in Python that allows users to create a variety of static, animated, and interactive visualizations. It provides a flexible and powerful interface for generating plots, charts, histograms, scatter plots, and more. Matplotlib is a powerful and versatile library that provides extensive capabilities for data visualization in Python. It is widely used in scientific research, data analysis, machine learning, and other domains where visualizing data is essential for understanding patterns, trends, and insights.

PLOTLY:

Plotly is a data visualization library that allows you to create interactive, publication-quality visualizations in Python, R, and other programming languages. It provides a rich set of tools and features for creating interactive plots, charts, maps, and dashboards.

SKLEARN:

Scikit-learn, also known as sklearn, is a popular open-source machine learning library for Python. It provides a comprehensive set of tools and algorithms for various machine learning tasks, including classification, regression, clustering, dimensionality reduction, and model evaluation. Scikit-learn includes a wide range of machine learning algorithms, such as linear regression, logistic regression, support vector machines (SVM), decision trees, random forests, gradient boosting, k-means clustering, and more.

WEB SCRAPPING:

To collect app reviews from the Google Play Store, web scraping techniques can be employed. Tools like Beautiful Soup or Scrapy in Python help extract the required information from web pages. We used Kaggle to collect our data.

DATA STORAGE AND PROCESSING:

Some tools are to facilitate efficient data storage , retrieval and analysis.

CSV:

CSV (Comma-Separated Values) is a simple and widely used file format for storing and exchanging tabular data. It represents data in plain text, where each line typically represents a row of data, and the values within each row are separated by commas (or other delimiters). CSV is commonly used for various tasks, such as data storage, data exchange between different systems, data processing, and data analysis. Its simplicity, human-readable nature, and wide compatibility make it a popular choice for working with tabular data.

We used a csv file to store our app review and sentiment analysis data.

MS EXCEL:

Microsoft Excel is a powerful spreadsheet program developed by Microsoft. It is widely used for data management, analysis, and visualization in various industries and professions. Microsoft Excel is widely used for financial analysis, budgeting, data analysis, project management, inventory tracking, and many other data-related tasks. Its user-friendly interface, extensive functionalities, and flexibility make it a valuable tool for both individuals and organizations for managing and analyzing data efficiently.

We used MS Excel to open , read our csv file. It is also used to clean the data and perform analysis.

JUPYTER NOTEBOOK:

Jupyter Notebook is an open-source web-based interactive computing environment that allows users to create and share documents called notebooks. It supports multiple programming languages, including Python, R, Julia, and others, making it a versatile tool for data exploration, analysis, visualization, and documentation. Jupyter Notebook provides an interactive computing environment where users can write and execute code in individual cells. Users can run code cells interactively, making it easy to experiment, test hypotheses, and iteratively develop code. Jupyter Notebook is a versatile tool that promotes reproducible research, exploratory data analysis, and collaborative coding.We used Jupyter notebook to write our python code , perform analysis and also to perform machine learning models.

POWER BI:

Power BI is a powerful intelligence tool developed by Microsoft that allows users to create interactive visualizations and reports from their data. It provides various visualization options to help users gain insights and make data-driven decisions.

Software Requirements Specification

for

Google Play Store Review And Sentiment Analysis

Version 1.0

Abhijeet Kumar Sahoo	**1901206040**	**abhijeetkumarsahoo381@gmail.com**
Sourav Patnaik	**1901206086**	**souravpatnaik59396@gmail.com**
Subhrajit Bastia	**1901206089**	**subhrajitbastia2002@gmail.com**
Abinash Nayak	**1901206042**	**nayakabinash933@gmail.com**

Course: B.tech (CSE)

Sem: *8th*

Date: 4/5/2023

Contents

Revision History

Name	Date	Reason For Changes	Version
Sourav Patnaik	4/5/2023	SRS document is made.	1.0
Sourav Patnaik	10/5/2023	Activity Diagram And DFD added	1.0

1 Introduction

1.1 Purpose

The purpose of this software requirements specification is to define the functional and non-functional requirements for a system that collects app reviews from the Google Play Store and performs sentiment analysis on them.

1.2 Product Scope

The system will retrieve app reviews from the Google Play Store, analyze the sentiment expressed in those reviews, and provide insights into user sentiments towards the apps.

1.3 Intended Audience and Document Overview

The intended audience for the Google Play Store App Review and Sentiment Analysis SRS includes:

Project Stakeholders: This includes project managers, product owners, and business analysts who are responsible for overseeing the development and implementation of the system.

System Developers: The SRS provides crucial information to the development team, including software engineers, designers, and testers, who will be involved in building and testing the system.

System Analysts: Analysts who will be using the system for reviewing and analyzing app sentiment data can refer to the SRS to understand the system's capabilities and functionalities.

Quality Assurance Team: The SRS serves as a reference for the quality assurance team to ensure that the developed system meets the specified requirements and standards.

Documentation Writers: Technical writers or document authors can refer to the SRS to gather information for creating user manuals, system documentation, and other relevant documentation.

2 System Overview

2.1 Product Perspective

The system will collect app reviews from various sources (such as web scraping or third-party data providers) and perform sentiment analysis on them using natural language processing techniques.

2.2 Product Features

- Data collection from various sources
- Sentiment analysis of app reviews
- Visualization of sentiment analysis results

2.3 User Classes and Characteristics

- Administrators: Responsible for managing the system and its configuration.
- Analysts: Utilize the system to perform sentiment analysis on app reviews.

2.4 Design and Implementation Constraints

- The system should comply with relevant data privacy and copyright laws when collecting app reviews from alternative sources.
- The system should be scalable to handle a large volume of app reviews and users.

3 Functional Requirements

Data Collection

- The system shall collect app reviews from alternative sources manually or through automated methods like web scraping.
- The system shall store the collected reviews securely in a database or storage mechanism.

Sentiment Analysis

- The system shall analyze the sentiment expressed in each app review using natural language processing techniques.
- The sentiment analysis shall categorize the sentiment as positive, negative, or neutral.

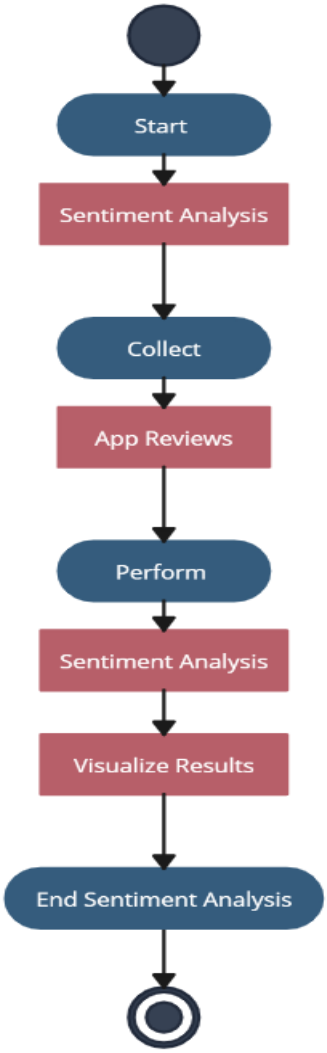
Visualization of Sentiment Analysis Results

- The system shall provide visualizations, such as charts or graphs, to present the sentiment analysis results in a user-friendly manner.
- The visualizations shall depict the distribution of positive, negative, and neutral sentiments across app reviews.

User Interface

- The system shall have an intuitive and user-friendly interface to allow users to interact with the collected data and view the sentiment analysis results.

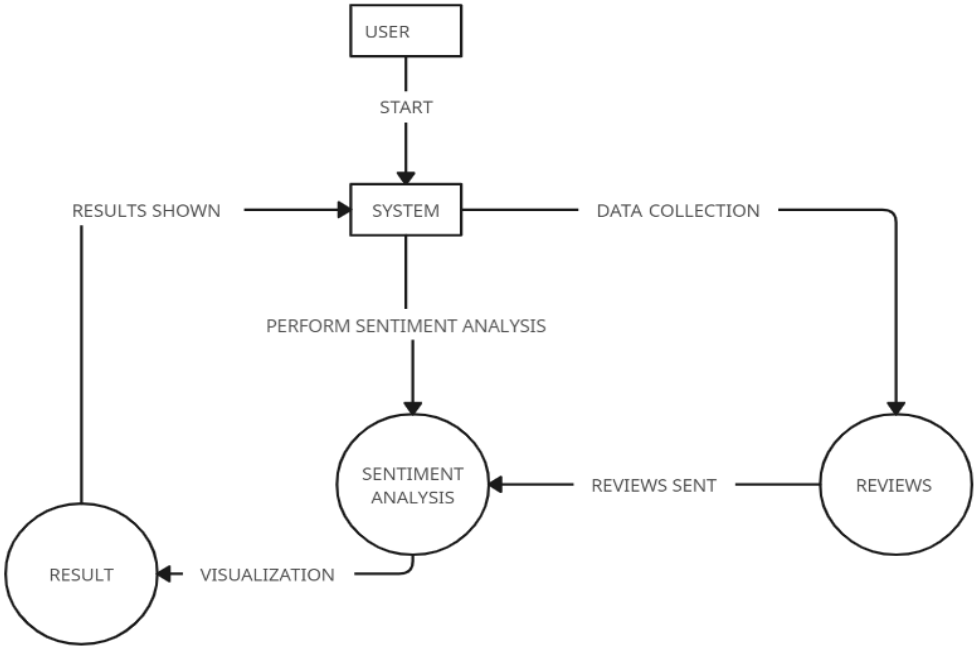
3.1 Activity Model :-



Explanation of Activities:

- Start - Sentiment Analysis: Represents the beginning of the sentiment analysis process.
- Collect -App Reviews: Activity to collect app reviews from various sources, either manually or through alternative data collection methods like web scraping.
- Perform - Sentiment Analysis: Activity to analyze the sentiment of the collected app reviews using natural language processing techniques.
- Visualize Results: Activity to visualize the sentiment analysis results, such as generating charts or graphs to present the sentiments of app reviews.
- End Sentiment Analysis: Represents the completion of the sentiment analysis process.
- The arrows between activities indicate the flow of control from one activity to another. The process starts with "Start - Sentiment Analysis" and proceeds to "Collect - App Reviews." Once the reviews are collected, the process moves to "Perform - Sentiment Analysis" and then to "Visualize Results." Finally, the sentiment analysis process ends with "End Sentiment Analysis."

3.2 Data Flow Diagram:-



Explanation of Components:

- User: Represents the user interacting with the system.
- System: The main system component that coordinates the flow of data and activities.
- Data Collection: Responsible for collecting app reviews from various sources.
- Sentiment Analysis: Performs sentiment analysis on the collected app reviews.
- Result Visualization: Generates visual representations of the sentiment analysis results.

- Arrows represent the flow of data or control between components:

- ❖ The user starts the process provides app reviews the system , which is stored in Data sheet .
- ❖ From the Data Collection the collected reviews are sent to the Sentiment Analysis component.
- ❖ The Sentiment Analysis component performs sentiment analysis on the reviews and generates sentiment results.
- ❖ The sentiment results flow to the Result component, which generates visualizations based on the results.
- ❖ The visualized results are then presented to the user through the System component.

4 Non-functional Requirements

Performance -

The system shall be able to handle a large volume of app reviews and perform sentiment analysis in a timely manner.

Security

The system shall implement appropriate security measures to protect the collected data and user information.

Reliability

The system shall be reliable and available for use during expected operational hours.

Usability

The system shall have a user-friendly interface that is easy to navigate and understand.

Scalability

The system shall be scalable to accommodate an increasing number of users and app reviews.

Compatibility

The system shall be compatible with modern web browsers and operating systems.

Maintainability

The system shall be designed and implemented in a modular and maintainable manner to facilitate future updates and enhancements.

5 Operational Requirements

Operating Environment : -

- Operating system: Microsoft Windows 8 , Microsoft Windows 10 , Microsoft Windows 11.
- The system will be web-based and accessible through modern web browsers

Development environment :-

- Development environment: Jupyter Notebook , Anaconda , MS Excel.
- Language used: Python

6 Other Requirements

- The system shall provide documentation and user manuals to guide administrators and analysts in using and maintaining the system.

7 Conclusion

This Software Requirements Specification provides a comprehensive overview of the features and functionalities expected in a Google Play Store App Review and Sentiment Analysis system . It serves as a foundation for system design, development, and testing activities.

8 Appendices

9.1 Definitions, Acronyms, and Abbreviations

SRS: Software Requirements Specification

8.2 References

https://www.kaggle.com/code/ecemboluk/google-play-store-analysis

https://en.wikipedia.org/wiki/Google_Play

METHODOLOGY

Abstract - Google play store is engulfed with a few thousands of new applications regularly with a progressively huge number of designers working freely or on the other hand in a group to make them successful, with the enormous challenge from everywhere throughout the globe. Since most Play Store applications are free, the income model is very obscure and inaccessible regarding how the in-application buys, adverts and memberships add to the achievement of an application. In this way, an application's prosperity is normally dictated by the quantity of installation of the application and the client appraisals that it has gotten over its lifetime instead of the income is created. Application (App) ratings are feedback provided voluntarily by users and function important evaluation criteria for apps. However, these ratings can often be biased due to insufficient or missing votes. Additionally, significant differences are observed between numeric ratings and user reviews. This Study aims to predict the ratings of Google Play Store apps using machine learning Algorithms. We have tried to perform Data Analysis and prediction into the Google Play store application dataset that I have collected from Kaggle. Using Machine Learning Algorithms, I have tried to discover the relationships among various attributes present in my dataset such as which application is free or paid, about the user reviews, rating of the application.

Key Words: Google Play Store Apps, Ratings Prediction, Exploratory Data Analysis, Machine Learning.

1. PROBLEM STATEMENT

Data is taken from the Google play store dataset made using csv file. Every row contains various entries regarding a certain app. We will be doing Exploratory data analysis on this data set, which is a very important step in data science cycle, as it not only helps in taking very initial business decisions but also in preparing the data for further modelling for use in machine learning algorithms. Our objective will be to structure the data, clean it and present certain trends that we observe that can help us draw very preliminary conclusions about the probability of success of a newly launched app.

2. GOOGLE PLAY STORE AND USER REVIEW ANALYSIS

In today's scenario we can see that mobile apps playing an important role in any individual's life. It has been seen that the development of the mobile application advertise has an incredible effect on advanced innovation. Having said that, with the consistently developing versatile application showcase there is additionally an eminent ascent of portable application designers inevitably bringing about high as can be income by the worldwide portable application industry.

With enormous challenge from everywhere throughout the globe, it is basic for a designer to realize that he is continuing in the right heading. To hold this income and their place in the market the application designers may need to figure out how to stick into their present position. The Google Play Store is observed to be the biggest application platform. It has been seen that although it creates more than two-fold the downloads than the Apple App Store yet makes just a large portion of the cash contrasted with the App Store. In this way, I scratched information from the Play Store to direct our examination on it.

With the fast development of advanced cells, portable applications (Mobile Apps) have turned out to be basic pieces of our lives. Be that as it may, it is troublesome for us to follow along the fact and to understand everything about the apps as new applications are entering market each day. It is accounted for that Android market achieved a large portion of a million applications in September 2011. Starting at now, 0.675 million Android applications are accessible on Google Play App Store. Such a lot of applications are by all accounts an extraordinary open door for clients to purchase from a wide determination extend. We trust versatile application clients consider online application surveys as a noteworthy impact for paid applications. It is trying for a potential client to peruse all the literary remarks and rating to settle on a choice. Additionally, application engineers experience issues in

discovering how to improve the application execution dependent on generally speaking evaluations alone and would profit by understanding a huge number of printedremarks.

We develop Android apps & release on Play Store. As an Developer or say Business Perspective it's very important to know whether users are enjoying the app or facing any issues. To know this Play Store has a Ratings & reviews section for each app released on play store. Users can submit the ratings and has a freedom to write a review for a particular app. This approach is quite a lengthy to rate & review app i.e. navigate to Play store to submit feedback or redirect leaving a current app workflow to open Play Store App link using URI. We never wanted our customers to leave our application, but with this flow, we are forced to redirect the control to Play store app.

3.GOOGLE PLAY STORE DATASET

The dataset consists of Google play store application and is taken from web scraping technique.

This dataset is for Web scratched information of 10k Play Store applications to analyze the market of android. Here it is a downloaded dataset which a user can use to examine the Android market of different use of classifications music, camera etc. With the assistance of this, client can predict see whether any given application will get lower or higher rating level. This dataset can be moreover used for future references for the proposal of any application. Additionally, the disconnected dataset is picked so as to choose the estimate exactly as online data gets revived all around a great part of the time. With the assistance of this dataset, We will examine various qualities like rating, free or paid and so forth utilizing Hive and after that We will likewise do forecast of various traits like client surveys, rating etc.

The data set contains the following columns:

- **App:** This Column contains the name of the app
- **Category:** This contains the category to which the app belongs. The category column contains 33 unique values.
- **Rating:** This column contains the average value of the individual rating the app has received on the play store. Individual rating values can vary between 0 to 5.
- **Reviews:** This column contains the number of people that have given their feedback for the app.
- **Size:** This column contains the size of the app i.e. The memory space that the app occupies on the device after installation.
- **Installs:** This column indicates the number of time that the app has been downloaded from the play store, these are approximate values and not absolute values.
- **Type:** This column contains only two values- free and paid. They indicate whether the user must pay money to install the app on their device or not.
- **Price:** For paid apps this column contains the price of the app, for free apps it contains the value 0.
- **Content Rating:** It indicates the targeted audience of the app and their age group.
- **Genre:** This column contains to which genre the app belongs to, genre can be considered as a sub division of Category.
- **Last updated:** This column contains the info about the date on which the last update for the app was launched.
- **Current version:** Contains information about the current version of the app available on the play store.
- **Android version:** Contains information about the version of the android OS on which the app can be installed.

4.USER REVIEW DATASET

- User reviews data frame has many rows and columns. The 5 columns are identified as follows:
- **App:** Contains the name of the app with a short description (optional).
- **Translated Review:** It contains the English translation of the review dropped by the user of the app.
- **Sentiment:** It gives the attitude/emotion of the writer. It can be 'Positive', 'Negative', or 'Neutral'.
- **Sentiment Polarity:** It gives the polarity of the review. Its range is [-1,1], where 1 means 'Positive statement' and -1 means a 'Negative statement'.
- **Sentiment Subjectivity:** This value gives how close a reviewer's opinion is to the opinion of the general public. Its range is [0,1]. Higher the subjectivity, closer is the reviewer's opinion to the opinion of the general public, and lower subjectivity indicates the review is more of a factual information.

5.DATA CLEANING AND PREPARATION

Preprocessing is important into transitioning raw data into a more desirable format. Undergoing the preprocessing process can help with completeness and compellability. For instance, you'll see if certain values were recorded or not. Also, you'll see how trustable the info is. It could also help with finding how consistent the values are. We need preprocessing because most real-world data are dirty. Data can be noisy i.e. the data can contain outliers or simply errors generally. Data can also be incomplete i.e. there can be some missing values.

The available data is raw and unusable for Exploratory data analysis, so before we do anything with the data we will have to explore and clean it to prepare it for data analysis.

- **Step1**: We write a function play store info (), that will display 5 attributes about all the columns: Data type, Count of non-null values, Count of null values, number of unique values in that column and percentage of null value in that columns in the play store dataset.
- **Step2**: we start off with the column 'Type' we can see that it has one null value. We checked this row and found out from the play store that it is a free app. We use filllna() function of the pandas library to fill this value.
- **Step 3**: We drop the columns 'Current Ver', 'Android Ver' and 'last updated' from our dataset using the drop() function of the pandas library.
- **Step 4**: We can see that the 'Rating' column has 1474 null values. Due to low variations in the rating values and a lot of repeated values the 'median' would be a suitable statistical indicator to replace the null values with. We calculate the mode of the column using the median () aggregate method, and fill this value in place of null values using the fillna() function.
- **Step 5:** We can see that the 'Reviews' column despite being a numerical indicator is of the 'object' data type, we will convert this to 'int' data type using the as type(int) function.
- **Step 6:** We can see that the size column, which should be numeric, is of the data type 'object', it also has characters 'k' and 'M' in the values which stand for kilobytes and Megabytes, we will replace the 'k' with 1000 and 'M' with 1000000. Some values also have '+' sign in them, which will be removed. Next, we will convert this column into 'int' datatype.
- **Step 7:** The 'Installs' column values contain the characters '+' and ',' which are going to prevent us from converting this column into a numeric datatype. We will get rid of these using the strip() and replace() functions.
- **Step 8:** The values in the column 'Price' might have the '$' sign in some values and the column is of the datatype 'object'. We will first remove the '$' sign using the **strip()** function and then convert the column into 'int' datatype.

- **Step 9:** Handling the duplicates in the App column we drop the no of duplicate rows that are present in the App columns.
- **Step 10:** We write a function Ur info(), that will display 5 attributes about all the columns: Data type, Count of non-null values, Count of null values ,number of unique values in that column and percentage of null value in that columns in the User review dataset.
- **Step11:** In the User review dataset the columns are App, Translated Review, Sentiment, Sentiment Polarity, Sentiment Subjectivity in this total 26863 NaN value are present so we drop them using dropna() function.

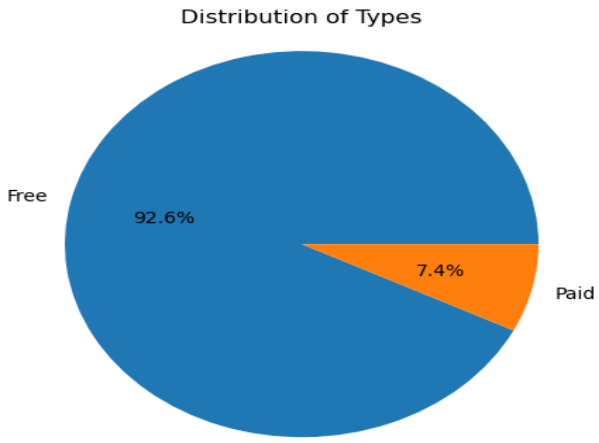
6.EXPLORATORY DATA ANALYSIS (SENTIMENT ANALYSIS AND VISUALIZATION)

Exploratory Data Analysis, or EDA, is an important step in any Data Analysis or Data Science project. EDA is the process of investigating the dataset to discover patterns, and anomalies (outliers), and form hypotheses based on our understanding of the dataset.

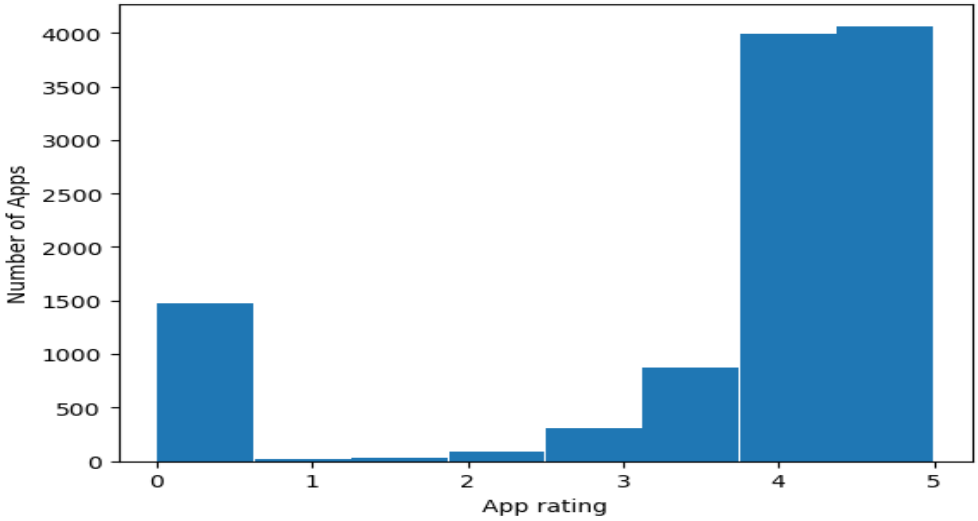
EDA involves generating summary statistics for numerical data in the dataset and creating various graphical representations to understand the data better. In this article, we will understand some EDA that we have done in our project. We have used above said software and technologies for this purpose. Here we use different machine learning model to predict the app ratings and also compare the model performence.

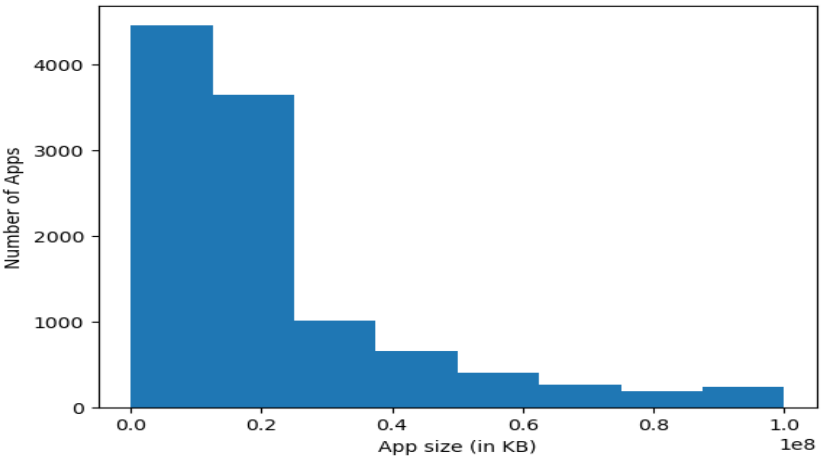
Uni-Variate Analysis:

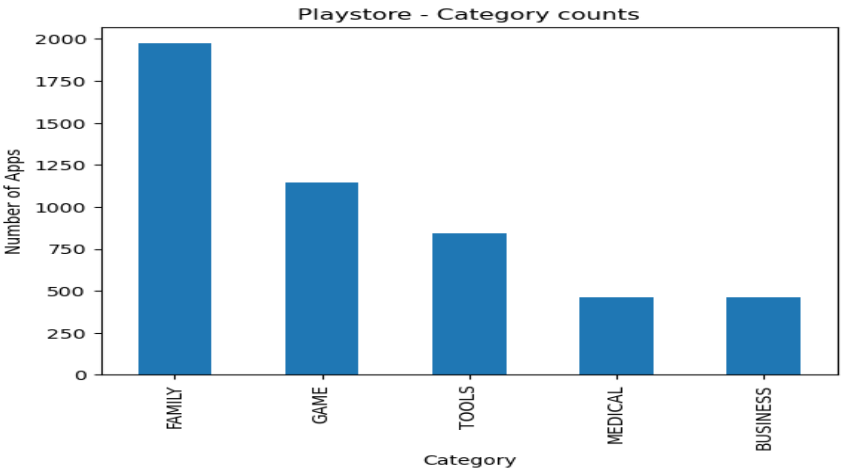
Univariate analysis explores each variable in a data set, separately. It looks at the range of values, as well as the central tendency of the values. It describes the pattern of response to the variable. It describes each variable on its own. Descriptive statistics describe and summarize data.

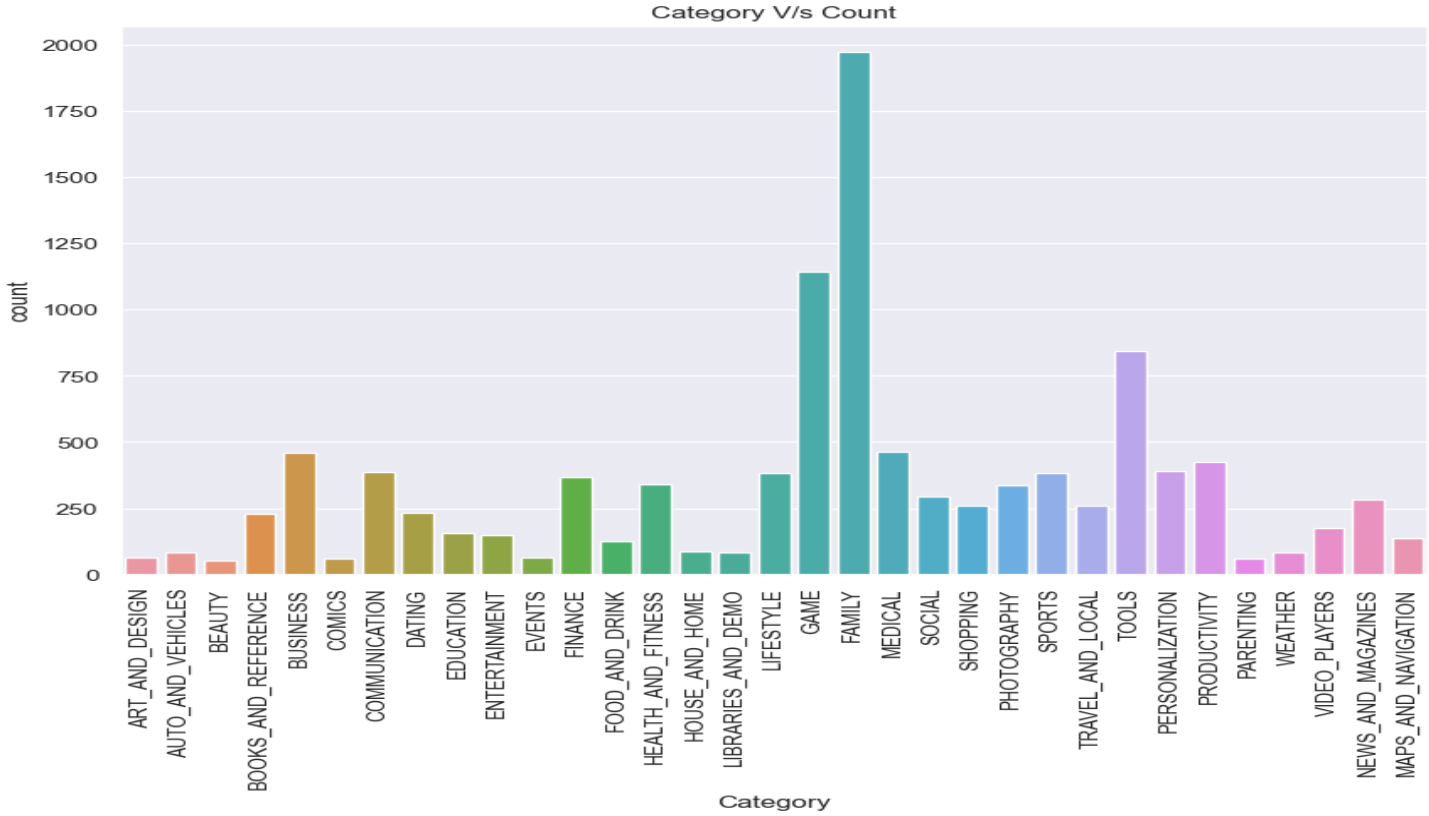


Here we see that 92.6% apps are free and 7.4% apps are paid on google play store. so we say that Most of the people love free services including us :) .

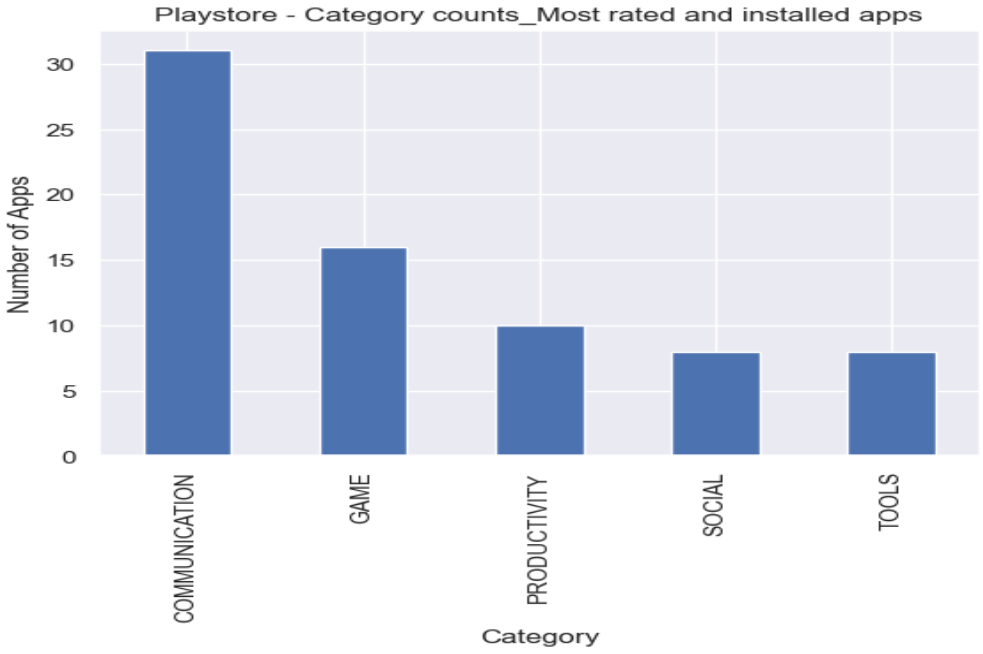

Number of Apps
App rating

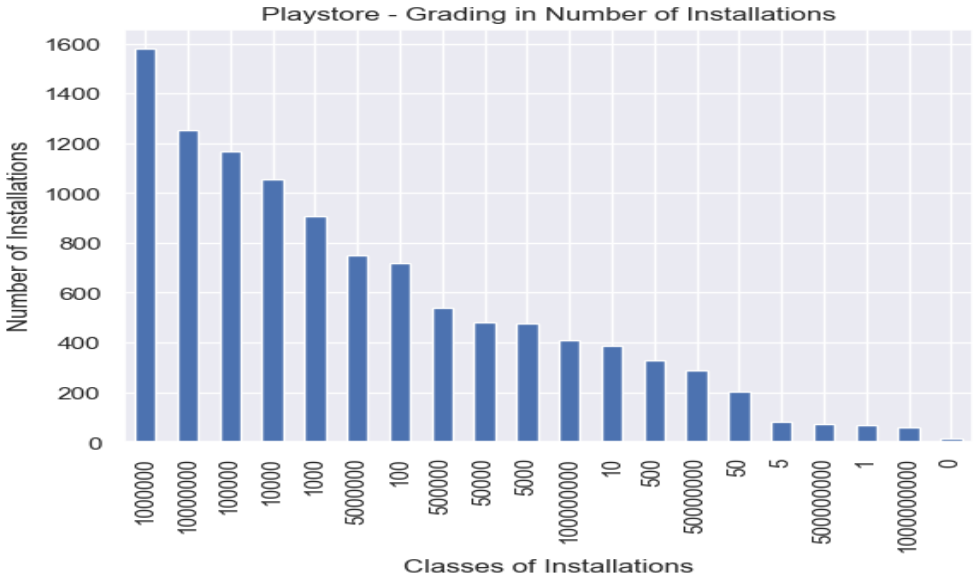

Number of Apps
App size (in KB)
1e8

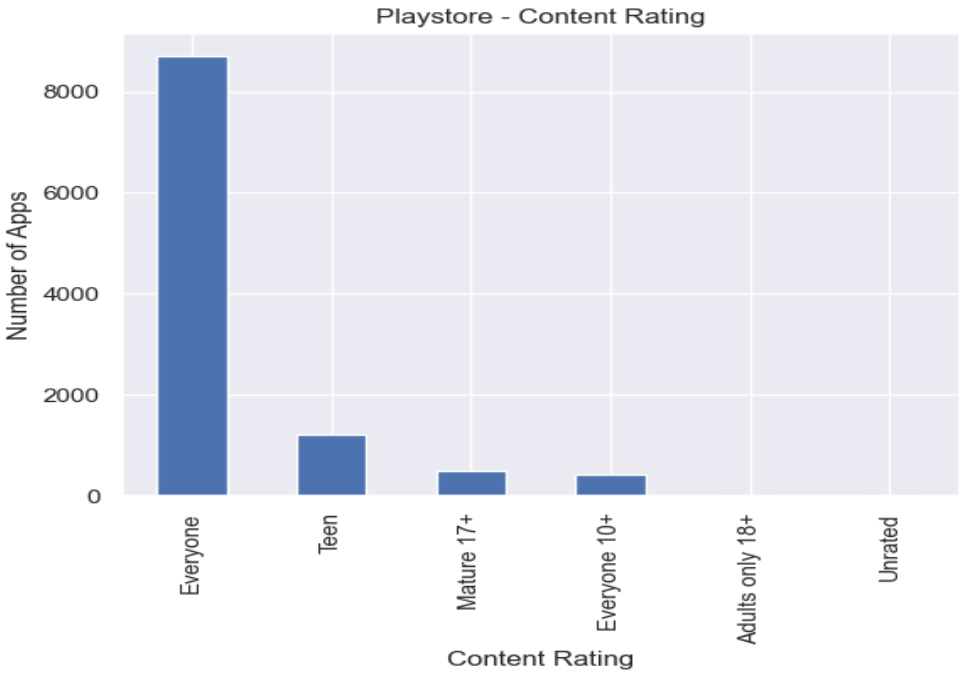

Playstore - Category counts
Number of Apps
FAMILY
GAME
TOOLS
MEDICAL
BUSINESS
Category

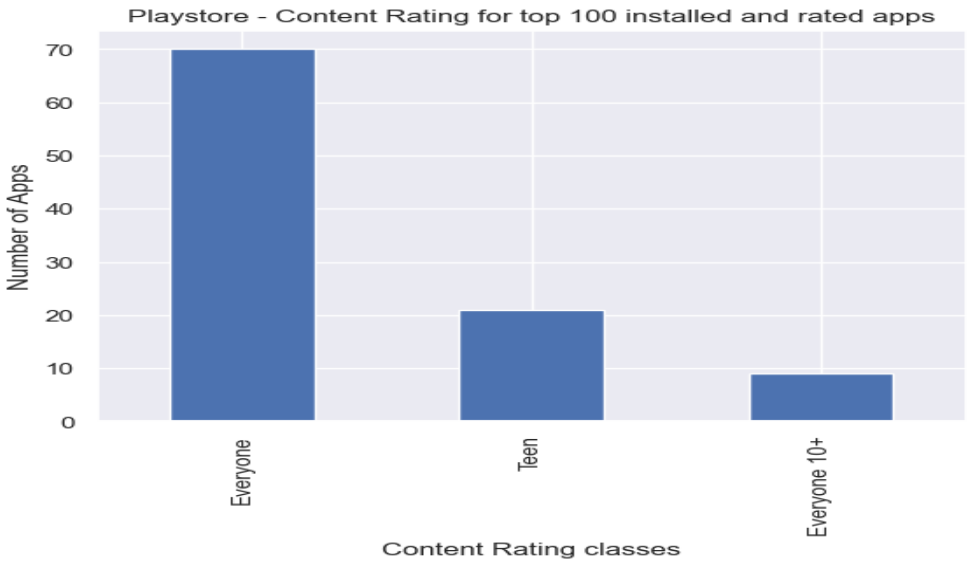

Category V/s Count
count
2000
1750
1500
1250
1000
750
500
250
0
ART_AND_DESIGN
AUTO_AND_VEHICLES
BEAUTY
BOOKS_AND_REFERENCE
BUSINESS
COMICS
COMMUNICATION
DATING
EDUCATION
ENTERTAINMENT
EVENTS
FINANCE
FOOD_AND_DRINK
HEALTH_AND_FITNESS
HOUSE_AND_HOME
LIBRARIES_AND_DEMO
LIFESTYLE
GAME
FAMILY
MEDICAL
SOCIAL
SHOPPING
PHOTOGRAPHY
SPORTS
TRAVEL_AND_LOCAL
TOOLS
PERSONALIZATION
PRODUCTIVITY
PARENTING
WEATHER
VIDEO_PLAYERS
NEWS_AND_MAGAZINES
MAPS_AND_NAVIGATION
Category

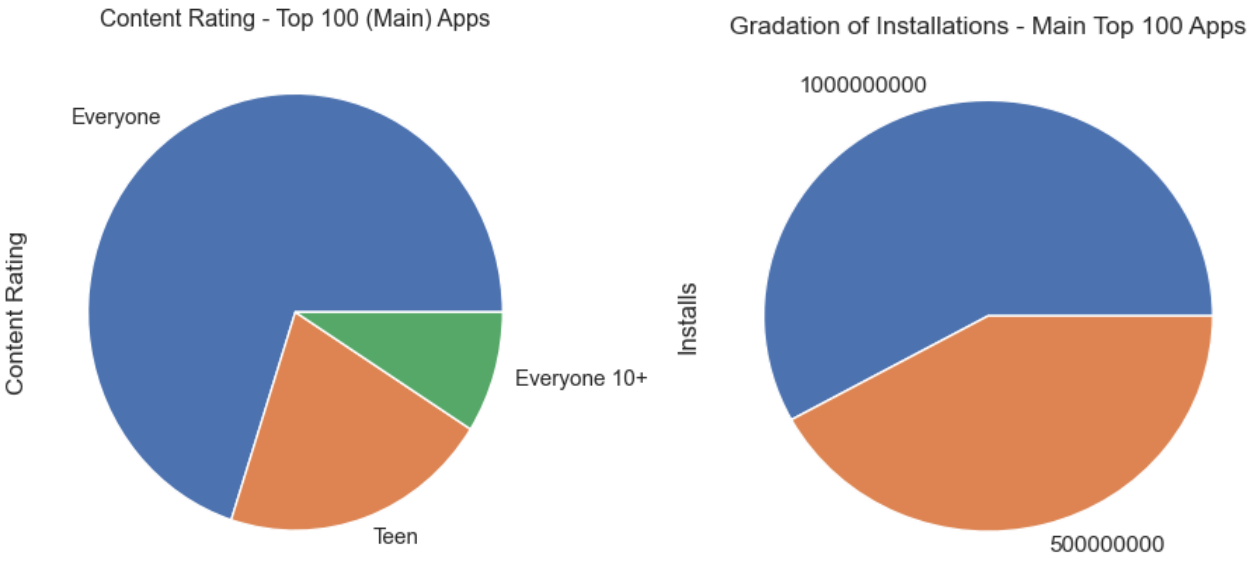

Playstore - Prices
Number of Apps
10000
8000
6000
4000
2000
0
0
2
1
4
3
Prices in Dollars


Playstore - Category counts_Most rated and installed apps
Number of Apps
0
5
10
15
20
25
30
COMMUNICATION
GAME
PRODUCTIVITY
SOCIAL
TOOLS
Category


Playstore - Grading in Number of Installations
Number of Installations
0
200
400
600
800
1000
1200
1400
1600
1000000
10000000
100000
10000
1000
5000000
100
500000
50000
5000
100000000
10
500
50000000
50
5
500000000
1
1000000000
0
Classes of Installations

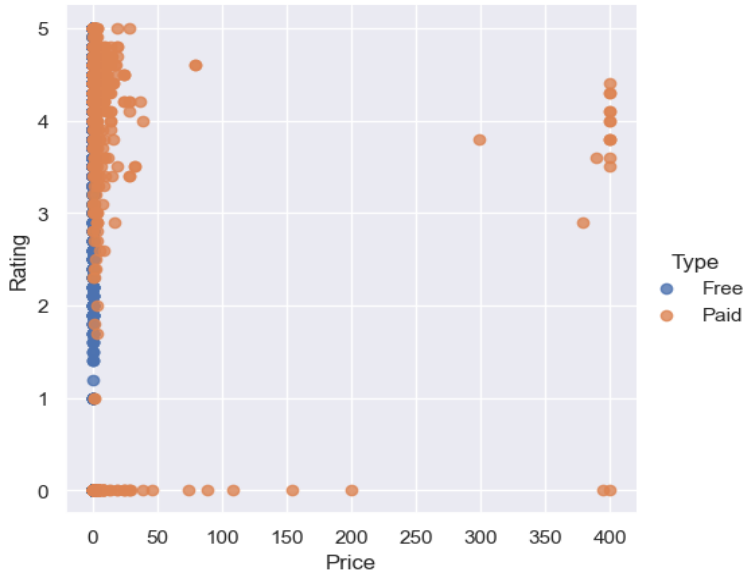

Playstore - Content Rating
Number of Apps
8000
6000
4000
2000
0
Everyone
Teen
Mature 17+
Everyone 10+
Adults only 18+
Unrated
Content Rating

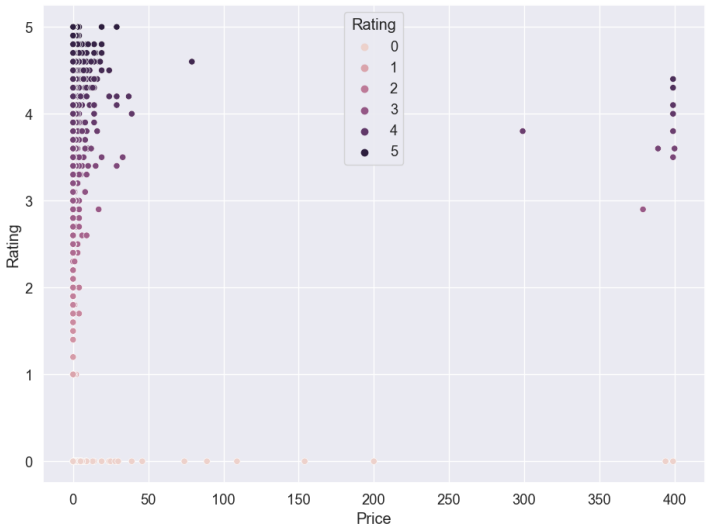

Playstore - Content Rating for top 100 installed and rated apps
Number of Apps
70
60
50
40
30
20
10
0
Everyone
Teen
Everyone 10+
Content Rating classes


Content Rating - Top 100 (Main) Apps
Everyone
Everyone 10+
Teen
Content Rating
Gradation of Installations - Main Top 100 Apps
1000000000
500000000
Installs

Bi-Variate Analysis:

The bivariate analysis aims to determine if there is a statistical link between the two variables and, if so, how strong and in which direction that link is. Some Bi-variate Analysis we have done:



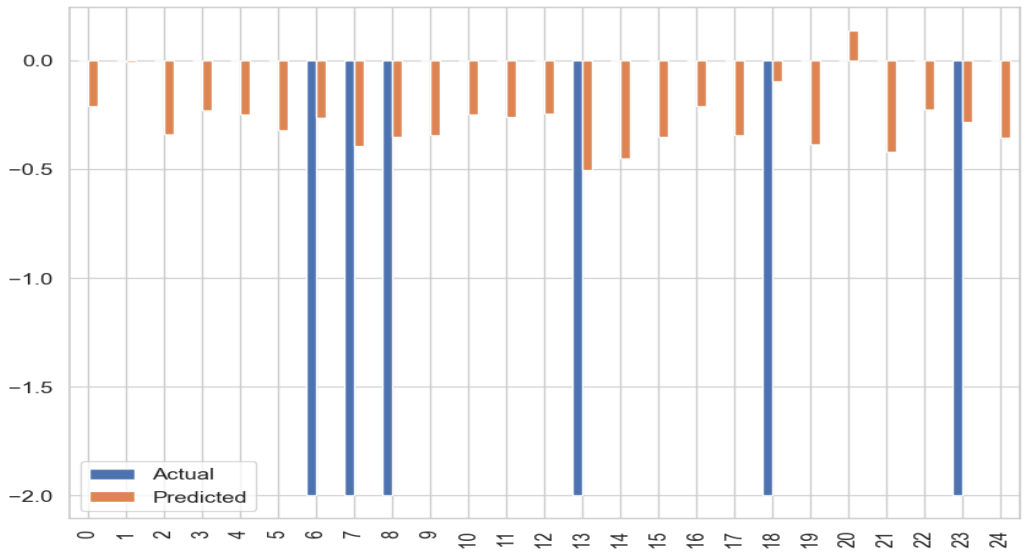


While there is not a very clean pattern, it does look that the higher priced apps have better rating. Although, there are not a lot of apps which are high priced, but the pattern is apparent.

Model Building:

Linear regression-

Linear regression models the relationship between the independent variables and the target variable using a linear equation. It is a simple and interpretable model that assumes a linear relationship between the features and the target.

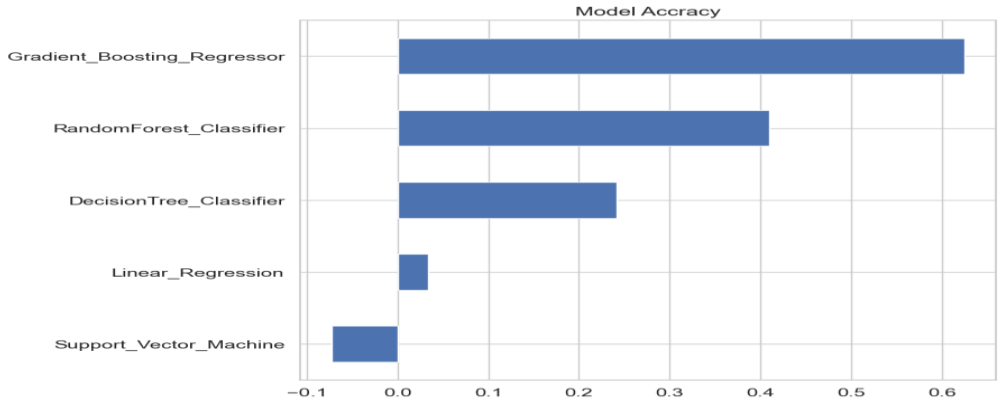


In the Above figure we can observe here that the model has returned not good prediction results.

Gradient Boost:

Gradient boosting algorithms such as Gradient Boosting Regression (GBR) or XGBoost iteratively build an ensemble of weak learners (decision trees) to minimize the loss function. These models are effective at capturing complex relationships and handling non-linearities.

Mean Squared Error (MSE): 0.18361449948413905
Root Mean Squared Error (RMSE): 0.4285026248275955
Mean Absolute Error (MAE): 0.1937584184708438
R-squared (R2): 0.6247970048762066



Here we can clearly see that the Gradient Boosting Regressor model works gives good r2 score. So this is our final model.

Conclusion:

Through exploratory data analysis we have observed some trends and have made some assumptions that might lead to app success among the users in the play store. we can say that most of the apps that are present on the play store has received positive review by the user while there are some apps which have negative reviews as well.

- Percentage of free apps = ~92%
- Percentage of apps with no age restrictions = ~82%
- Most competitive category: Family
- Category with the highest average app installs: Game
- Tools, Entertainment, Education, Business and Medical are top Genres.

GLIMPSES OF CODE AND OUTPUT

PLAYSTORE REVIEW ANALYSIS:

Importing all libraries

```
In [1]: import pandas as pd
        import numpy as np
        import seaborn as sns
        import warnings
        warnings.filterwarnings(action='ignore')
        pd.set_option('display.max_columns',None)
        pd.set_option('display.max_rows',None)
```

Reading the data from the csv file

```
In [2]: data = pd.read_csv('playstore_reviews old data.csv')
        data.head()
```

Out[2]:

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462
2	10 Best Foods for You	NaN	NaN	NaN	NaN
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000

Making copy of the Original file

```
In [3]: df = data.copy()
        df.head()
```

Out[3]:

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462
2	10 Best Foods for You	NaN	NaN	NaN	NaN
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000

**Here we can see that 'Sentiment' and 'Sentiment_Polarity' both are giving the same information.But in the 'sentiment_polarity' variable we have many values in float which not specifying the actual characterstic of the review.so we can drop the 'Sentiment analysis' column so that we can have a better analysis and though we can get the reviews result that: is the review is positive, negative or neutral? so we don't need the 'Translated_Reviews' variable. though we have a column called 'Sentiment_subjectivity' which is identifying that how many special characters are there? the range for 'Sentiment_Subjectivity' is (0 to 1). Sentiment analysis is the automated process of analyzing test to determine the sentiment expressed (Positive,Negative or Neutral).Sentiment polarity is also same as sentiment analysis. the range for sentiment analysis is -1 to 1.

Dropping some Unnecessary variables from the data

```
In [4]: df = df.drop(df[['Sentiment_Polarity','Translated_Review']],axis=1)
        df.head()
```

Out[4]:

	App	Sentiment	Sentiment_Subjectivity
0	10 Best Foods for You	Positive	0.533333
1	10 Best Foods for You	Positive	0.288462
2	10 Best Foods for You	NaN	NaN
3	10 Best Foods for You	Positive	0.875000
4	10 Best Foods for You	Positive	0.300000

Checking the shape of the data(Rows & Columns)

```
In [5]: # Now we can see we have only four columns.
        # lets check the shape of the data.
        df.shape
```

Out[5]: (64295, 3)

Checking the Datatypes

```
In [6]: # let's check the datatypes.
        df.dtypes
```

Out[6]:
```
App                        object
Sentiment                  object
Sentiment_Subjectivity     float64
dtype: object
```

Checking the information of the data

```
In [7]: # let's check the information of the data.
        df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64295 entries, 0 to 64294
Data columns (total 3 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   App                     64295 non-null  object
 1   Sentiment               37432 non-null  object
 2   Sentiment_Subjectivity  37432 non-null  float64
dtypes: float64(1), object(2)
memory usage: 1.5+ MB
```

Checking The Null Values

```
In [8]: df.isnull().sum()
```

Out[8]:
```
App                        0
Sentiment                  26863
Sentiment_Subjectivity     26863
dtype: int64
```

```
In [9]: # as we can see there are many null values in the dataset.
```

Getting the Unique values from each variables.

```
In [10]: def unique(d,columns):
             return{i: list(d[i].unique()) for i in columns}
         def categorical_data(d):
             return [i for i in d.columns if d.dtypes[i] == 'object']
```

```
In [11]: unique(df,categorical_data(df))
```

```
Out[11]: {'App': ['10 Best Foods for You',
                  '104 找工作 - 找工作 找打工 找兼職 履歷健檢 履歷診療室',
                  '11st',
                  '1800 Contacts - Lens Store',
                  '1LINE – One Line with One Touch',
                  '2018Emoji Keyboard 😍 Emoticons Lite -sticker&gif',
                  '21-Day Meditation Experience',
                  '2Date Dating App, Love and matching',
                  '2GIS: directory & navigator',
                  '2RedBeans',
                  '2ndLine - Second Phone Number',
                  '30 Day Fitness Challenge - Workout at Home',
                  '365Scores - Live Scores',
                  '3D Blue Glass Water Keyboard Theme',
                  '3D Color Pixel by Number - Sandbox Art Coloring',
                  '3D Live Neon Weed Launcher',
                  '4 in a Row',
                  '4K Wallpapers and Ultra HD Backgrounds',
                  '591房屋交易-租屋、中古屋、新建案、實價登錄、別墅透天、公寓套房、捷運、買房賣房行情、房價房貸查詢',
```

Replacing the values to thier numeric form in the "Sentiment" Column.

```
In [12]: df['Sentiment']=df['Sentiment'].replace('Positive',1)
         df['Sentiment']=df['Sentiment'].replace('Negative',-1)
         df['Sentiment']=df['Sentiment'].replace('Neutral',0)
```

```
In [13]: df['Sentiment'].unique()
Out[13]: array([ 1., nan,  0., -1.])
```

```
In [14]: df.dtypes
Out[14]: App                       object
         Sentiment                 float64
         Sentiment_Subjectivity    float64
         dtype: object
```

Null Values Treatment by filling it with mean of the Data

```
In [15]: # now we can see that all our values in Sentiment got updated and it became an numeric data type.
         # so now we can fill the null values.
         df['Sentiment'] = df['Sentiment'].ffill(axis=0)
         df['Sentiment_Subjectivity'] = df['Sentiment_Subjectivity'].ffill(axis=0)
```

```
In [16]: df.isnull().sum()
Out[16]: App                       0
         Sentiment                 0
         Sentiment_Subjectivity    0
         dtype: int64
```

```
In [17]: # Now we can see that there are no null values right now.
```

Checking the duplicate values in the Dataset

```
In [18]: duplicate = df[df.duplicated()]
         duplicate.shape
Out[18]: (41022, 3)
```

```
In [19]: duplicate.head(15)
```

Out[19]:

	App	Sentiment	Sentiment_Subjectivity
2	10 Best Foods for You	1.0	0.288462
5	10 Best Foods for You	1.0	0.300000
7	10 Best Foods for You	1.0	0.900000
9	10 Best Foods for You	0.0	0.000000
12	10 Best Foods for You	1.0	0.875000
15	10 Best Foods for You	1.0	0.511111
18	10 Best Foods for You	1.0	0.100000
19	10 Best Foods for You	1.0	1.000000
22	10 Best Foods for You	0.0	0.000000
24	10 Best Foods for You	1.0	0.500000
25	10 Best Foods for You	0.0	0.000000
26	10 Best Foods for You	1.0	1.000000
27	10 Best Foods for You	1.0	0.500000
29	10 Best Foods for You	0.0	0.000000
30	10 Best Foods for You	1.0	0.600000

Dropping the duplicated values and checking the shape

```
In [20]: df.drop_duplicates(keep='first',inplace=True)
```

```
In [21]: df.duplicated().sum()
Out[21]: 0
```

```
In [22]: df.shape
Out[22]: (23273, 3)
```

```
In [23]: # checking the number of the unique values in the data.
         df['App'].nunique()
Out[23]: 1074
```

Checking the co-relation, Variance and co-variance of the data

```
In [24]: df.corr()
```

Out[24]:

	Sentiment	Sentiment_Subjectivity
Sentiment	1.000000	0.089396
Sentiment_Subjectivity	0.089396	1.000000

```
In [25]: df.var()
```

```
Out[25]: Sentiment                0.752125
         Sentiment_Subjectivity   0.046922
         dtype: float64
```

```
In [26]: df.cov()
```

Out[26]:

	Sentiment	Sentiment_Subjectivity
Sentiment	0.752125	0.016794
Sentiment_Subjectivity	0.016794	0.046922

Finding the apps and their highest positive response.

```
In [27]: df.groupby('App')['Sentiment'].max()
```

```
Out[27]: App
         10 Best Foods for You                                  1.0
         104 找工作 - 找工作 找打工 找兼職 履歷健檢 履歷診療室                    1.0
         11st                                                   1.0
         1800 Contacts - Lens Store                             1.0
         1LINE – One Line with One Touch                        1.0
         2018Emoji Keyboard 😂 Emoticons Lite -sticker&gif       1.0
         21-Day Meditation Experience                           1.0
         2Date Dating App, Love and matching                    1.0
         2GIS: directory & navigator                            1.0
         2RedBeans                                              1.0
         2ndLine - Second Phone Number                          1.0
         30 Day Fitness Challenge - Workout at Home             1.0
         365Scores - Live Scores                                1.0
         3D Blue Glass Water Keyboard Theme                     1.0
         3D Color Pixel by Number - Sandbox Art Coloring        1.0
         3D Live Neon Weed Launcher                             1.0
         4 in a Row                                             1.0
         4K Wallpapers and Ultra HD Backgrounds                 1.0
```

```
In [28]: # counting how many apps are getting the highest Sentiment.
In [29]: # so we confirmed that there are 1000 apps which got positive response.
         # similarly there are 41 apps which got Negative Reviews.
         # And there are 33 apps which are getting Neutral Reviews.
         df.groupby('App')['Sentiment'].max().value_counts()
```

```
Out[29]:  1.0    1000
         -1.0      41
          0.0      33
         Name: Sentiment, dtype: int64
```

```
In [30]: # Let's see the Description of the data including thier count and the Five_Point_Summary(min,25%,50%,75%).
In [31]: df.groupby(['App'])['Sentiment'].describe()
```

Out[31]:

App	count	mean	std	min	25%	50%	75%	max
10 Best Foods for You	41.0	0.780488	0.612870	-1.0	1.00	1.0	1.00	1.0
104 找工作 - 找工作 找打工 找兼職 履歷健檢 履歷診療室	23.0	0.826087	0.491026	-1.0	1.00	1.0	1.00	1.0
11st	25.0	0.320000	0.900000	-1.0	-1.00	1.0	1.00	1.0
1800 Contacts - Lens Store	30.0	0.733333	0.639684	-1.0	1.00	1.0	1.00	1.0
1LINE – One Line with One Touch	21.0	0.571429	0.810643	-1.0	1.00	1.0	1.00	1.0
2018Emoji Keyboard 😂 Emoticons Lite -sticker&gif	18.0	0.777778	0.548319	-1.0	1.00	1.0	1.00	1.0
21-Day Meditation Experience	39.0	0.717949	0.686284	-1.0	1.00	1.0	1.00	1.0
2Date Dating App, Love and matching	28.0	0.535714	0.838082	-1.0	0.75	1.0	1.00	1.0
2GIS: directory & navigator	27.0	0.407407	0.843949	-1.0	0.00	1.0	1.00	1.0
2RedBeans	23.0	0.739130	0.619192	-1.0	1.00	1.0	1.00	1.0
2ndLine - Second Phone Number	23.0	0.391304	0.891328	-1.0	-0.50	1.0	1.00	1.0

PLAYSTORE APP ANALYSIS:

Loading All Libraries

In [5]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import warnings
warnings.filterwarnings(action='ignore')
pd.set_option('display.max_columns',None)
pd.set_option('display.max_rows',None)
```

Loading the CSV file

In [6]:
```
data = pd.read_csv('playstore_apps old data.csv')
data.head()
```

Out[6]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19M	10000.0	Free	0.0	Everyone	Art & Design	07-01-2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14M	500000.0	Free	0.0	Everyone	Art & Design;Pretend Play	15-01-2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8.7M	5000000.0	Free	0.0	Everyone	Art & Design	01-08-2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25M	50000000.0	Free	0.0	Teen	Art & Design	08-06-2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2.8M	100000.0	Free	0.0	Everyone	Art & Design;Creativity	20-06-2018	1.1	4.4 and up

Copying The Main Data

In [7]:
```
df = data.copy()
df.head()
```

Out[7]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159.0	19M	10000.0	Free	0.0	Everyone	Art & Design	07-01-2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967.0	14M	500000.0	Free	0.0	Everyone	Art & Design;Pretend Play	15-01-2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510.0	8.7M	5000000.0	Free	0.0	Everyone	Art & Design	01-08-2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644.0	25M	50000000.0	Free	0.0	Teen	Art & Design	08-06-2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967.0	2.8M	100000.0	Free	0.0	Everyone	Art & Design;Creativity	20-06-2018	1.1	4.4 and up

In [8]:
```
df.tail()
```

Out[8]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
10836	Sya9a Maroc - FR	FAMILY	4.5	38.0	53M	5000.0	Free	0.0	Everyone	Education	25-07-2017	1.48	4.1 and up
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4.0	3.6M	100.0	Free	0.0	Everyone	Education	06-07-2018	1	4.1 and up
	Parkinson										20-01-		2.2 and

Shape of The Dataset

In [9]:
```
df.shape
```

Out[9]:
```
(10841, 13)
```

Dataset Information

In [10]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   App             10841 non-null  object 
 1   Category        10841 non-null  object 
 2   Rating          9367 non-null   float64
 3   Reviews         10840 non-null  float64
 4   Size            10841 non-null  object 
 5   Installs        10840 non-null  float64
 6   Type            10840 non-null  object 
 7   Price           10840 non-null  float64
 8   Content Rating  10840 non-null  object 
 9   Genres          10841 non-null  object 
 10  Last Updated    10840 non-null  object 
 11  Current Ver     10833 non-null  object 
 12  Android Ver     10838 non-null  object 
dtypes: float64(4), object(9)
memory usage: 1.1+ MB
```

Data Preprocessing

```
In [11]: #1st of all we can drop all the useless columns so that it will be easy for us in handeling the file.
```

```
In [12]: #1) here we can see that 'Category' column is refering to the 'App' so we can drop the any one variable.
         #   though we have very lengthy names in 'App' with having a lot of special characters so we can drop the 'App' column.

         #2) similarly 'Genres' is also refering the same as 'Category' and that's why we are dropping the 'Genres' variable.

         #3) we can drop the 'current ver' and 'Android version' because they are giving the same information from the column 'App' and
         #   'Last Updated'. so we can drop them as well.
```

Removing The Unnecessary columns

```
In [13]: df['Type'].value_counts()
```

```
Out[13]: Free    10039
         Paid      800
         0           1
         Name: Type, dtype: int64
```

```
In [14]: useless_vars = ['App','Genres','Current Ver','Android Ver']
         df = df.drop(useless_vars, axis=1)
         df.head()
```

Out[14]:

	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Last Updated
0	ART_AND_DESIGN	4.1	159.0	19M	10000.0	Free	0.0	Everyone	07-01-2018
1	ART_AND_DESIGN	3.9	967.0	14M	500000.0	Free	0.0	Everyone	15-01-2018

Checking The Null Values

```
In [15]: df.isnull().sum()
         # here we can see that we have 1474 null values in our rating variable and 1695 null values in our size variable.
         # we have only one null value in type variable.
```

```
Out[15]: Category              0
         Rating             1474
         Reviews               1
         Size                  0
         Installs              1
         Type                  1
         Price                 1
         Content Rating        1
         Last Updated          1
         dtype: int64
```

```
In [16]: # now we can see that we cleaned the data to some extend.
```

Observing The Unique Values & Cleaning Each Variables

```
In [17]: def unique(datas,columns):
             return {i: list(datas[i].unique()) for i in columns}
         def categorical_data(datas):
             return [i for i in datas.columns if datas.dtypes[i] == 'object']
```

```
In [18]: unique(df,categorical_data(df))
```

```
Out[18]: {'Category': ['ART_AND_DESIGN',
           'AUTO_AND_VEHICLES',
           'BEAUTY',
           'BOOKS_AND_REFERENCE',
           'BUSINESS',
```

Missing values treatment

```
In [42]: # so let's recheck how many null values are there.
         df[df['Type'].isna()]
```

Out[42]:

	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	year	month
9148	FAMILY	NaN	0.0	NaN	0.0	NaN	0.0	Everyone 10+	2018.0	6

```
In [43]: # here we can see there are only one null value in the row 9148 so we can drop that row.
         df = df.drop(9148,axis=0).reset_index(drop=True)
```

```
In [44]: df.isnull().sum()
```

```
Out[44]: Category              0
         Rating             1473
         Reviews               0
         Size               1694
         Installs              0
         Type                  0
         Price                 0
         Content Rating        0
         year                  0
         month                 0
         dtype: int64
```

```
In [45]: #now we can see there are only 2 variables which have missing values.
```

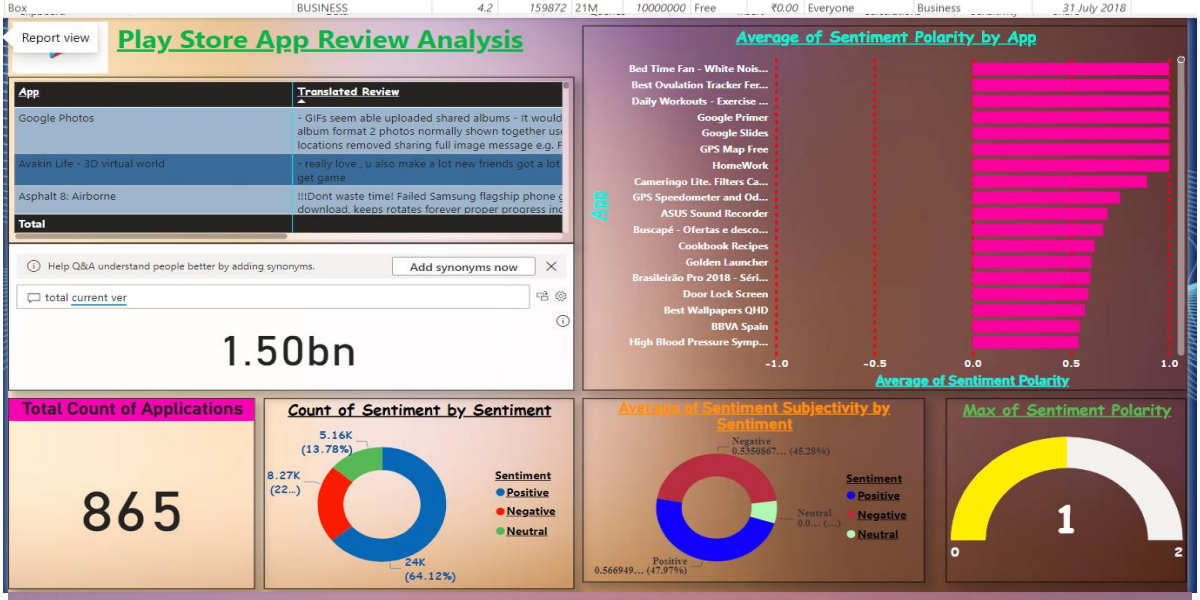
```
In [46]: df.dtypes
```

```
Out[46]: Category            object
         Rating             float64
```

NOTE:-

For full code refer to our google play store app review and sentiment analysis project.

App	Category	Rating	Reviews	Size	Installs	Type	Price	ContentRating	Genres	LastUpdated	CurrentVer
Marble Woka Woka 2018 - Bubble Shooter Match 3	GAME	4.6	155186	21M	10000000	Free	₹0.00	Everyone	Puzzle	03 August 2018	
Dr. Safety - Antivirus, Booster, Cleaner, AppLock	TOOLS	4.4	372553	21M	10000000	Free	₹0.00	Everyone	Tools	06 August 2018	
PDF Reader - Scan?Edit & Share	BUSINESS	4.3	54520	21M	10000000	Free	₹0.00	Everyone	Business	30 July 2018	
Bukalapak - Buy and Sell Online	SHOPPING	4.4	662287	21M	10000000	Free	₹0.00	Everyone	Shopping	03 August 2018	
BitTorrent® - Torrent Downloads	VIDEO_PLAYERS	4.5	641219	21M	10000000	Free	₹0.00	Everyone	Video Players, Editors	26 July 2018	
Best Fiends - Free Puzzle Game	GAME	4.6	1480189	21M	10000000	Free	₹0.00	Everyone	Casual	02 August 2018	
Chrome Beta	PRODUCTIVITY	4.4	228755	21M	10000000	Free	₹0.00	Everyone	Productivity	19 July 2018	
Chrome Beta	PRODUCTIVITY	4.4	228794	21M	10000000	Free	₹0.00	Everyone	Productivity	06 August 2018	
Puffin Web Browser	COMMUNICATION	4.3	541661	21M	10000000	Free	₹0.00	Everyone	Communication	09 July 2018	
Puffin Web Browser	COMMUNICATION	4.3	541389	21M	10000000	Free	₹0.00	Everyone	Communication	09 July 2018	
Flightradar24 Flight Tracker	TRAVEL_AND_LOCAL	4.3	171889	21M	10000000	Free	₹0.00	Everyone	Travel, Local	18 July 2018	
MyVodafone (India) - Online Recharge & Pay Bills	PRODUCTIVITY	4.1	1092367	21M	10000000	Free	₹0.00	Everyone	Productivity	04 August 2018	
Red Hands – 2-Player Games	GAME	4.2	93608	21M	10000000	Free	₹0.00	Everyone	Action	09 February 2018	
BEYBLADE BURST app	GAME	4.5	216849	21M	10000000	Free	₹0.00	Everyone	Action, Adventure	16 July 2018	
Textgram - write on photos	ART_AND_DESIGN	4.4	295237	21M	10000000	Free	₹0.00	Everyone	Art, Design	30 July 2018	
Textgram - write on photos	ART_AND_DESIGN	4.4	295221	21M	10000000	Free	₹0.00	Everyone	Art, Design	30 July 2018	
Dictionary.com: Find Definitions for English Words	BOOKS_AND_REFERENCE	4.6	899010	21M	10000000	Free	₹0.00	Everyone	Books, Reference	30 July 2018	
Quran for Android	BOOKS_AND_REFERENCE	4.7	497826	21M	10000000	Free	₹0.00	Everyone	Books, Reference	30 May 2018	
FBReader: Favorite Book Reader	BOOKS_AND_REFERENCE	4.5	203130	21M	10000000	Free	₹0.00	Everyone	Books, Reference	28 June 2018	
Dictionary	BOOKS_AND_REFERENCE	4.5	264260	21M	10000000	Free	₹0.00	Everyone	Books, Reference	22 June 2018	
Dictionary - Merriam-Webster	BOOKS_AND_REFERENCE	4.5	454412	21M	10000000	Free	₹0.00	Everyone	Books, Reference	18 May 2018	
Dictionary - Merriam-Webster	BOOKS_AND_REFERENCE	4.5	454060	21M	10000000	Free	₹0.00	Everyone	Books, Reference	18 May 2018	
JW Library	BOOKS_AND_REFERENCE	4.9	922752	21M	10000000	Free	₹0.00	Everyone	Books, Reference	15 June 2018	
English Hindi Dictionary	BOOKS_AND_REFERENCE	4.4	384368	21M	10000000	Free	₹0.00	Everyone	Books, Reference	04 August 2018	
Wikipedia	BOOKS_AND_REFERENCE	4.4	577550	21M	10000000	Free	₹0.00	Everyone	Books, Reference	02 August 2018	
Moon+ Reader	BOOKS_AND_REFERENCE	4.4	233757	21M	10000000	Free	₹0.00	Everyone	Books, Reference	01 May 2018	
Box	BUSINESS	4.2	159872	21M	10000000	Free	₹0.00	Everyone	Business	31 July 2018	



CONCLUSION

Data science can be summarized into five steps: capture, maintain process, analyze, and communicate. The analysis of Google Play Store application aided to build most reliable and more interactive applications. This would be very useful for app developers to build an application focused on certain discussed category in this analysis. This analysis will help in building the application with precise and accurate objectives.

In the initial phase, we focused more on the problem statements and data cleaning, in order to ensure that we give them the best results out of our analysis. Our major challenge was data cleaning, In Data Cleaning, we have performed few steps to ensure the data quality such as removing NAN values. During the Data Cleaning step we found that 13.60% of reviews were NaN values, and even after merging both the data frames, we could not infer much in order to fill them. Thus, we had to drop them.The merged data frame of both play store and user reviews, had only 816 common apps. This is just 10% of the cleaned data, we could have given more valuable analysis if we had at least 70% - 80% of the data available in the merged data frames.User Reviews had 42% of NaN values, which could have been used for developing an understanding of the category wise sentiments, which would help us to fill 13.60% NaN values of the Reviews column.With the cleaned data, we have performed Exploratory Data Analysis to understand our dataset like number of installations for each category We explore the correlation between the size of the app and the version of Android on the number of installs and so on.

Our motive in whole project was to analyze the data and find out main components that affect users' decision to download app. After completion of analysis I concluded that user prefer more of free apps. Most of the apps present in play store are more or less of same size so size doesn't affect their decision much.

It was found that Most of the apps that are present on the google play store have rating in between 4 and 5.Also it was observed that Maximum number of applications present in the dataset are of small size.We found most popular category of apps on two basis - Number of Installs and Number of reviews. Personalization wins in former criteria whereas Sports wins in later criteria.

In the problem statement we are given with 2 datasets i.e. play store and User review data set in the user review dataset it was observed that User Reviews had 42% of NaN values, which could have been used for developing an understanding of the category wise sentiments, which would help us to fill 13.60% NaN values of the Reviews column.

Most of the reviews are of Positive Sentiment, while Negative and Neutral have low number of reviews.
8.Sentiment Polarity / Sentiment Subjectivity

Collection of reviews shows a wide range of subjectivity and most of the reviews fall in [-0.50,0.75] polarity scale implying that the extremely negative or positive sentiments are significantly low. Most of the reviews show a mid-range of negative and positive sentiments.

Sentiment subjectivity is not always proportional to sentiment polarity but in maximum number of case, shows a proportional behavior, when variance is too high or low.Sentiment Polarity is not highly correlated with Sentiment Subjectivity.

The dataset contains immense possibilities to improve business values and have a positive impact. It is not limited to the problem taken into consideration for this project. Many other interesting possibilities can be explored using this dataset.

From the results and process we have implemented, we can conclude that we have achieved this group project objective which is analyzing the Google Play Store apps and determine trends of the Google Play Store and both of our research questions.

REFERRENCES

- GeeksforGeeks
- Stackoverflow
- Towards data science
- Python libraries documentation
- Researchgate.net


thank You