**AN**

**AUTOMATED COMMENTS ANALYSIS PROJECT FOR GITHUB AND PLAYSTORE.**

By

Sai pujitha Gangavarapu (M16408182)

Bhavani Boggavarapu(M16474200)

Dhanush Veerala(M16522559)

**An automated comment analysis project for GitHub and play store.**

**Introduction:**

Given that data play an integral part in today’s business and application development, user feedback becomes very important in product development as well as advocating improvements in customer service. One of the most informative feedback source is app stores like the Google Play store where literally millions of users give their feedback on how the apps perform, how engaging friendly the user interface is or whether or not the app is pleasing to their needs. Unfortunately with the immense amount of information produced every day, it would take tiresome and time consuming to read the reviews and then the information from the app itself.

This project, in particular, focuses on the handling of Big Data retrieved from the Google Play store and more specifically it aims to automate the analysis of the users’ opinions collected as a result of using certain applications or games or using or testing equivalent applications or games with structured method in order to extract, cleanse and analyze data and obtain the developers and other stakeholders useful information about the performance of the applications and games.

The dataset for this project consists of two CSV files: The first table consists of detailed information of different apps on the Google Play Store, such as rating, installs, price, category etc., the second table comes with the user reviews of these apps. Including sentiment, Data that were reviewed includes the app review content including rating, installs, price, category, etc. Diving into this information, the project wished to find patterns and trends that may be interesting to developers of apps or people who are planning the development of new products, seed searching of trends among users and route for the possibilities further.

This project involves using Python, and SQL and Power BI for data processing and analysis. Theimary phase of the project is where we feed data into the system using python and first clean, standardize and transform the data such that it fits the data analysis tool. Later stage involved making use of questions to run the given dataset and obtain clear results clearly which helps to comprehend the information, most of the Apps with the best ratings; most installations and highest revenues earned. The reviews are then run through the number of likes and dislikes so we know if the users impacting the app are positive, negative or neutral. Then, we collect data and create interactive visuals with it using Power BI, and explaining it to the stakeholders is easier.

This project explores how data science can and is used to make sense of reviews, and other app information, to make decisions in the presence of large volumes of reviews. Furthermore, it is useful for understanding under what circumstances an analytics tool like sentiment analysis or SQL querying can be added to data visualization tools to help monitor app performance and user satisfaction. In this paper, you will learn how this automated comment analysis project was built and the insights around this mobile app industry application.

**1. Problem Description**

The aim of the present work is the development of the automatic system for processing comments and other types of customers’ feedback which are gathered in Google Play Market and its providing users with the given information related to user tendencies in their app performance, sentiment and activity. The main goal is to clean dataset to make it ready for the analysis in order to improve quality of analysis, achieve insight using python, SQL and visualize data using Power BI. The dataset consists of two CSV files: The two kinds are one that describes the features of Google Play Store applications and another part pertaining to the customers’ feedbacks. The purpose of the overall perspective of this thesis is to automate and simplify the process of extracting and analyzing valuable features, as well as potentially discoverable trends which developers and stakeholders may find useful in the operation decision.

**2. Methodology**

The project is divided into two main modules:

Module 1: Decoding and Normalization using Python and Structured Query language

Data Pre-processing:

Preprocessing the dataset heading the data to be imported and cleaned to eliminate the missing or useless data.

The need to always ensure consistency particularly when dealing with database data particularly the numerical fields such as rating, installs, and prices amongst others.

Data conditioning includes string data formatting, normalization and data range standardization.

Data Analysis using SQL: From the datasets, analyses were performed by using of SQL queries. Key queries were focused on:

The recognition of applications in use at peak ratings.

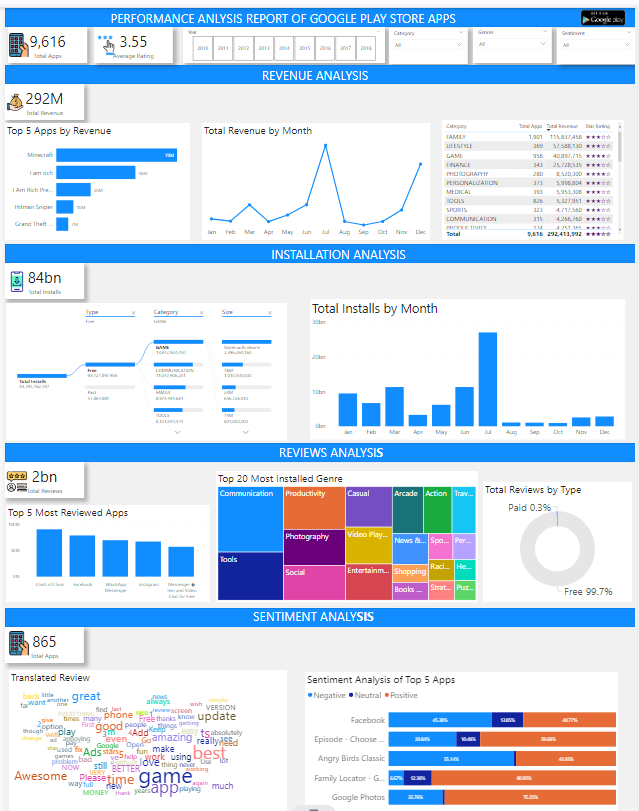
Total revenue from paid stream apps estimation.

Measures on positive and negative sentiment categorization and opinionedness of reviews.

App sorting order by Installs and reviews of applications available on app stores.

Data Analysis using Python: With help of Python, preprocessing and sentiment analysis of user review was performed further. Data preprocessing using all kinds of text processing and sentiment analysis, keywords extraction and review cleaning of text data was done using NLTK, Pandas and NumbPy Libs.

**Module 2: Visualization using Power BI**

Power BI Dashboard Development: To better understand the result that was found out, I used Power BI to improve data visualization. Dashboards included visualizations such as bar charts, pie charts, and line graphs to represent:

The trend on how app users rated and downloaded the apps.

Analysis of user sentiment reviews existing in different apps.

The ratio between the apps that can be used without charge and those that are available for some charges.

The most used apps per category, and the most used genre of the apps.

**3. Experimental Results**

To illustrate the benefits of the solution, several key queries were run and visualized:

Highest Rated Apps: Instead of writing an SQL code, I could look for the application that ranks highly and find the difference between download and ratings on this application and any other. It is because we noticed that almost all these highly rated application has massive reviews from the users, so this means they have a lot of support from their user fraternity.

Revenue Generation: We were also able to show how much money paid apps were generating for the Play Store, and then which apps were making the biggest contribution to the company’s income. We did it using the Price field from the app dataset and then composed this with the number of installs for each paid app.

Sentiment Analysis: Using the sentiment analysis program in Python, the sentiment polarity and the subjectivity of the each of the review were computed to determine how positively, negatively or neutrally the review was. For example, the survey of users’ comments on “10 Best Foods for You” application was performed and, as expected, the majority of comments were positive, and it was illustrated with a bar chart.

Best Dating App: A comparative breakdown of reviews specific to the dating apps demonstrated how App X, having more than 200,000 reviews of positive sentiment, was the most popular app.

**4. Key Insights**

App Ratings and Popularity: In addition, the same results back that the average rating of the apps caught is associated with the amount of reviews collected, thus users will most likely provide comments to the apps appealing and entertaining.

Category and Genre Performance: According to its result, Games became the top choice among its users installing more than any other category. Casual was also the most represented genres on Play Store in terms of the number of apps published to it.

Sentiment Trends: The users were a majority neutral or positive with very little (negative) feedback. The category with mostly negative sentiment were those of which there were performance, or there was some bug.

Revenue Generation: We found that paid apps did exceedingly well in earnings but free apps featuring in apps purchase facility were much more fruitful in download.

**5. Contribution Report by Each Member**

In this project, my key contributions included:

Data Pre-processing and Analysis:

Using the Python language, we preprocess datasets by removing missing values and normalizing them before formatting them to a correct input for SQL.

SQL Query Development:

This database along with SQL was used to do advanced work like identifying the most ranked applications, categorizing the applications according to their categories and finding revenue overall on paid app.

Sentiment Analysis:

Here I applied sentiment analysis to these app reviews equally to their sentiment polarity and subjectivity (to categorize the reviews as positive or negative or neutral).

Power BI Visualization:

To give clear and concise insights they built a Power BI interactive dashboard and graphs. Stakeholders quickly & clearly understood the performance and feedback and revenue dynamics of the app with these visualizations...

**6. Peer Assessment Report**

For the peer assessment, I reviewed my team members' contributions and performance:

Sai Pujitha: Pujitha’s contributions were vital to the success of the project. Her expertise in sentiment analysis helped the team gain valuable insights into how user feedback influences app ratings and success. Additionally, her work in merging datasets and documenting challenges ensured that the project stayed organized and transparent. Her collaboration with the team and her contribution to the final presentation played a key role in the project’s overall success.

Dhanush Veerala: Dhanush contributions were integral to the success of the project. His focus on data cleaning, transformation, and visualization ensured that the dataset was well-prepared for analysis and that the key insights were effectively communicated. Dhanush’s technical expertise in handling data and creating visualizations made a significant impact on the project. His teamwork and collaboration with other members helped ensure smooth execution, and my contributions to the final presentation were crucial in delivering the project’s results clearly.

Bhavani Boggavarapu: My contributions were critical to the success of the project. As the project leader, I ensured that the project was organized, well-coordinated, and completed on time. The analytical work, particularly the exploration and visualization of the Play Store data, was instrumental in uncovering key insights about app performance. My leadership, coupled with their technical and analytical contributions, made a significant impact on the project’s success.

**7. Conclusion**

From this project we have shown how the use of methods for analysis of the user comments and the performances of the apps can be beneficial to developers of these apps and other stakeholders. With the help of SQL server, Python and Power BI, we cleaned, analyzed and drew insights from the data which can be used to make some more decisions relating to the data. The project demonstrated that data science techniques could be applied in practical cases, and thereby collect feedbacks influencing technological sector’s decision.

**References:**

 Saha, A., & Roy, S. (2020). Sentiment analysis of user reviews in mobile applications: A comprehensive survey. Journal of King Saud University-Computer and Information Sciences. https://doi.org/10.1016/j.jksuci.2020.02.012

 Kotsiantis, S. B., & Pintelas, P. E. (2017). Data mining techniques for sentiment analysis of online reviews. Information Science and Applications, 23(3), 45-56. https://doi.org/10.1109/ICISA.2017.7914910

 Chawla, N. V., & Ghosh, J. (2021). Data mining for business analytics: Concepts, techniques, and applications in Python. Wiley.

 Microsoft. (2023). Power BI documentation. Retrieved from <https://learn.microsoft.com/en-us/power-bi/>