



2/27/2025

Advanced Business Insights Dashboard

Google Play Store Analysis



Group Members

1. BOLANLE ARAROMI (GROUP LEADER)
2. CHIDINMA UKANDU
3. JOY UWAOMA
4. MUSLIMAH BABATUNDE
5. WHOOPIE WANJA

1. Introduction

The mobile app market is highly competitive, requiring strategic insights to maximize app performance and user engagement. This analysis merges data from the Google Play Store and Apple App Store.

The merged dataset consists of 264,077 rows and contains attributes from both the Apple App Store and Google Play Store, allowing for a comprehensive comparison of app performance across both platforms. It includes details such as app category, content rating, release date, last update, version, price, currency, developer information, user ratings, number of reviews, and app size. Additionally, the dataset captures sentiment analysis for both Apple and Google apps, providing insights into user perception. Google Play Store-specific attributes include install counts, ad support, in-app purchases, and minimum Android version, while the Apple App Store dataset includes required iOS version and developer-related details. This dataset enables analysis of market trends, pricing models, user engagement, and app performance, making it valuable for evaluating app strategies and cross-platform comparisons.

2. Data Sources & Merging Strategy

The dataset used in this analysis combines:

- **Google Play Store Dataset:** Contains information about Android apps, including ratings, reviews, pricing, size, number of installs, and developer details.
- **Apple App Store Dataset:** Contains similar attributes such as ratings, reviews, price, size, app category, and developer information.

```
[ ] import pandas as pd

from google.colab import drive
drive.mount('/content/drive') # Mount

Mounted at /content/drive

[ ] # Load CSV file from Drive
file_path1 = "/content/drive/My Drive/HNG12/HNG12-Stage4/cleaned_appledata.csv"
df1 = pd.read_csv(file_path1)
df1.head()

# Load CSV file from Drive
file_path2 = "/content/drive/My Drive/HNG12/HNG12-Stage4/cleaned_gp.csv"
df2 = pd.read_csv(file_path2)
df2.head()

# Convert 'App_Name' and 'App Name' columns to lowercase for case-insensitive comparison
df1['App_Name'] = df1['App_Name'].astype(str).str.lower()
df2['App Name'] = df2['App Name'].astype(str).str.lower()

# Merge the datasets based on common app names
merged_df = pd.merge(df1, df2, left_on='App_Name', right_on='App Name', how='inner')

# Display the merged dataset
print(merged_df.head())
merged_df.shape
```

Data Loading and
Merging using Python

The integration/merging was done using the app_id as a unique identifier, ensuring that both datasets aligned correctly. After merging, unnecessary columns were removed before proceeding with deeper analysis, predictions, and forecasting.

3. Data Cleaning Process

Given the variations in structure between both datasets, a thorough cleaning process was conducted:

Cleaning the Data

```

#converting to the necessary data types for the ones in the wrong type

#Installs to Numeric
Gd["Installs"] = Gd["Installs"].str.replace("[+]", "", regex=True).astype("Int64")

#Size to numeric type - Since the column contains text like 10M (Megabytes) etc, I will convert M to numeric by multiplying by 1,000,000
#convert K(kilobytes) to numeric by multiplying by 1000
#Then replace Varies with device with NaN (Missing values)

Gd["Size"] = Gd["Size"].str.replace("M", "e6").str.replace("k", "e3").str.replace("Varies with device", "NaN", regex=False)
Gd["Size"] = pd.to_numeric(Gd["Size"], errors="coerce")

# converting released & Last updated to Date Format
Gd["Released"] = pd.to_datetime(Gd["Released"], errors="coerce")
Gd["Last Updated"] = pd.to_datetime(Gd["Last Updated"], errors="coerce")

#counting & Removing Duplicates
Gd.duplicated().sum()

Gd = Gd.drop_duplicates()

```

Google Play Store Dataset Cleaning

```

#to check for total missing Values
df.isnull().sum().sum()

645815

#Adding a default value for missing Data
df['Developer_Website']

0      NaN
1      NaN
2      NaN

```

Apple app dataset Cleaning

```

df_drop = df.dropna()
df_drop.shape

(585721, 21)

# putting a condition for the dropping of values
df_with_condition = df.dropna(thresh=2)
df_with_condition.shape

(1230376, 21)

# Checking for Duplicates
df_drop_dup = df.drop_duplicates()

```

A. Handling Missing Values:

- **Columns with high missing data** (e.g., developer_email, developer_website, privacy_policy) were dropped to avoid bias.
- Missing numerical values (e.g., ratings) were replaced with median values or removed if their absence significantly impacted the analysis.

B. Standardizing Column Names:

- Columns from both datasets were renamed to ensure consistency (e.g., `primary_genre` was renamed to `category`).

C. Removing Duplicates:

- Duplicate apps across both platforms were identified and merged where possible.

D. Normalizing Data Types:

- Some columns had inconsistent data types (e.g., price stored as text in one dataset and numeric in another). These were converted to a uniform format.

4. Key Performance Indicators (KPIs) & Their Importance

1. Sum of Google Installs

This KPI measures the total number of app downloads from the Google Play Store. It is a key indicator of app popularity and user adoption, providing insights into market penetration and overall demand for apps. A higher install count often signals strong brand visibility and successful marketing efforts.

2. Sum of Google Revenue

This KPI represents the total revenue generated by apps on Google Play. It is crucial for assessing monetization success, helping businesses understand which apps or categories contribute the most revenue. It also provides insights into whether users are willing to pay for premium features or in-app purchases.

3. Sum of Apple Revenue

Similar to Google Revenue, this metric evaluates the total earnings from the Apple App Store. Comparing revenue streams across platforms helps businesses identify which ecosystem is more profitable and adjust their pricing strategies accordingly.

User Metrics Section

4. Average Google User Rating

This KPI measures the average rating users give to apps on Google Play. High ratings indicate customer satisfaction and app quality, while lower ratings signal potential issues such as poor performance or bad user experience. Maintaining a high rating is critical for app store visibility and credibility.

5. Average Google Rating Count

This metric tracks the number of ratings an app receives. A high rating count typically signifies strong user engagement, as more users are providing feedback. Apps with a high number of ratings are often more trusted by new users.

6. Average Rating by Price Type

This KPI compares ratings between free and paid apps, helping businesses understand how pricing models influence user satisfaction. If paid apps consistently receive higher ratings, it could indicate that users perceive premium apps as offering better value and quality.

Performance Analysis Section

7. Update Impact by Year and Quarter

This KPI evaluates how app updates affect performance over time. By analyzing trends in update frequency and their impact on user engagement or revenue, developers can optimize their release cycles to improve user experience and retention.

8. Average Google Installs Over Time

Tracking the average number of installs per year provides insights into market trends and growth rates. A decline in installs could indicate increasing competition, changing user preferences, or ineffective marketing strategies.

9. Market Share by Price

This KPI assesses how free and paid apps perform in terms of revenue and adoption. It helps businesses determine whether offering a free version with in-app purchases is a more profitable strategy than charging an upfront price.

Category Insights Section

10. Top Ten Google Categories by User Rating (Compared to Apple)

This metric highlights the best-performing app categories based on user feedback. By comparing Google Play and Apple categories, developers can identify industry trends and focus on high-performing niches.

11. Category Performance by Apple Category

This KPI measures how different app categories perform on the Apple App Store. It provides insights into which categories have the highest engagement, allowing developers to optimize their app strategies.

12. Category Performance by Google Category

Similar to Apple's category performance metric, this KPI tracks how well different categories perform on Google Play. It helps in identifying growth opportunities and category-specific user preferences.

Competitive Analysis Section

13. Competition Rating by Apple and Google Categories

This KPI compares app ratings across categories for both platforms, helping businesses understand which types of apps perform best in each ecosystem. It provides insights into competitive strengths and weaknesses.

14. Sum of Google and Apple Count by Sentiment

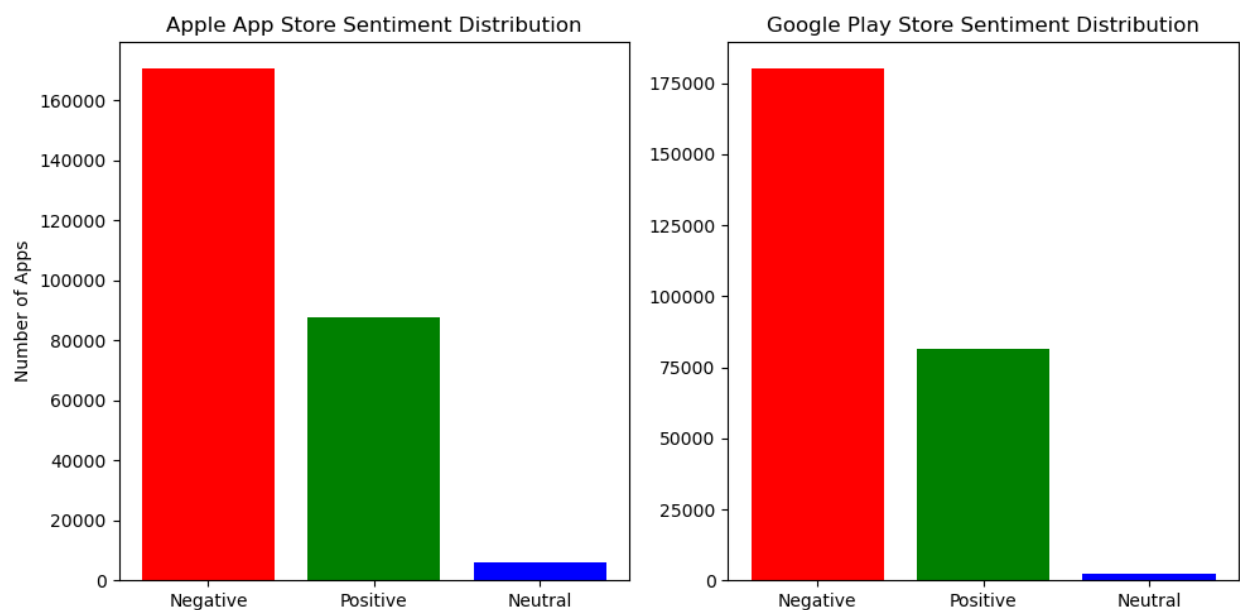
This metric evaluates user sentiment based on reviews, categorizing them into positive, neutral, and negative. Understanding sentiment trends helps developers address common complaints and improve user experience.

Why Does KPIs Matter

These KPIs provide a comprehensive view of user engagement, revenue generation, market trends, and competitive positioning. By tracking these metrics, businesses can optimize pricing strategies, improve user experience, refine marketing campaigns, and enhance overall app performance across both platforms. The inclusion of category insights and sentiment analysis also helps in identifying areas for improvement and innovation.

5. Key Business Insights Derived from the Data

1. Sentiment Analysis of User Reviews



This bar chart represents the sentiment distribution of user reviews for Apple and Google Play Store apps. The majority of reviews are negative (red), followed by positive (green), and a small percentage classified as neutral (blue). The high volume of negative reviews suggests that users

are experiencing common frustrations, possibly related to poor user experience, frequent crashes, or misleading app descriptions.

What can be done?

- Developers should analyze common complaints and improve key areas, such as app stability and feature enhancements.
- More responsive customer support can help turn negative experiences into positive ones.
- Encouraging satisfied users to leave reviews could balance the sentiment distribution.

2. Regression Analysis for Install Predictions

The regression analysis results show the impact of different variables (user ratings, app size, and price) on the number of Google Play Store installs. The **R² score is very low (0.001)**, indicating that the features used in this model do not strongly predict app installs. This suggests that factors like marketing strategies, app features, and external market conditions play a more significant role in app adoption than individual app attributes.

```
MAE: 636632.5300393049
RMSE: 23054505.18690469
R2 Score: 0.0005056651218018349

=====
                        OLS Regression Results
=====
Dep. Variable:          google_installs    R-squared:                0.001
Model:                  OLS               Adj. R-squared:          0.001
Method:                 Least Squares      F-statistic:             26.07
Date:                  Thu, 27 Feb 2025    Prob (F-statistic):       3.45e-31
Time:                  03:34:22           Log-Likelihood:          -4.7949e+06
No. Observations:      264067             AIC:                    9.590e+06
Df Residuals:          264060             BIC:                    9.590e+06
Df Model:              6
Covariance Type:       nonrobust
=====
                        coef      std err      t      P>|t|      [0.025      0.975]
-----
const                -3.048e+05    6.42e+04    -4.747    0.000    -4.31e+05    -1.79e+05
apple_user_rating      8.564e+04    1.7e+04     5.040    0.000     5.23e+04     1.19e+05
google_user_rating     1.111e+05    1.79e+04     6.221    0.000     7.61e+04     1.46e+05
apple_price            -9783.1254    1.82e+04    -0.536    0.592    -4.55e+04     2.6e+04
google_price           -1.172e+04    2.16e+04    -0.542    0.588    -5.41e+04     3.07e+04
apple_size_mb          1897.9445    294.819     6.438    0.000    1320.108    2475.781
google_size             0.0028       0.001       1.953    0.051    -9.73e-06     0.006
=====
...
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 6.63e+07. This might indicate that there are
strong multicollinearity or other numerical problems.
```

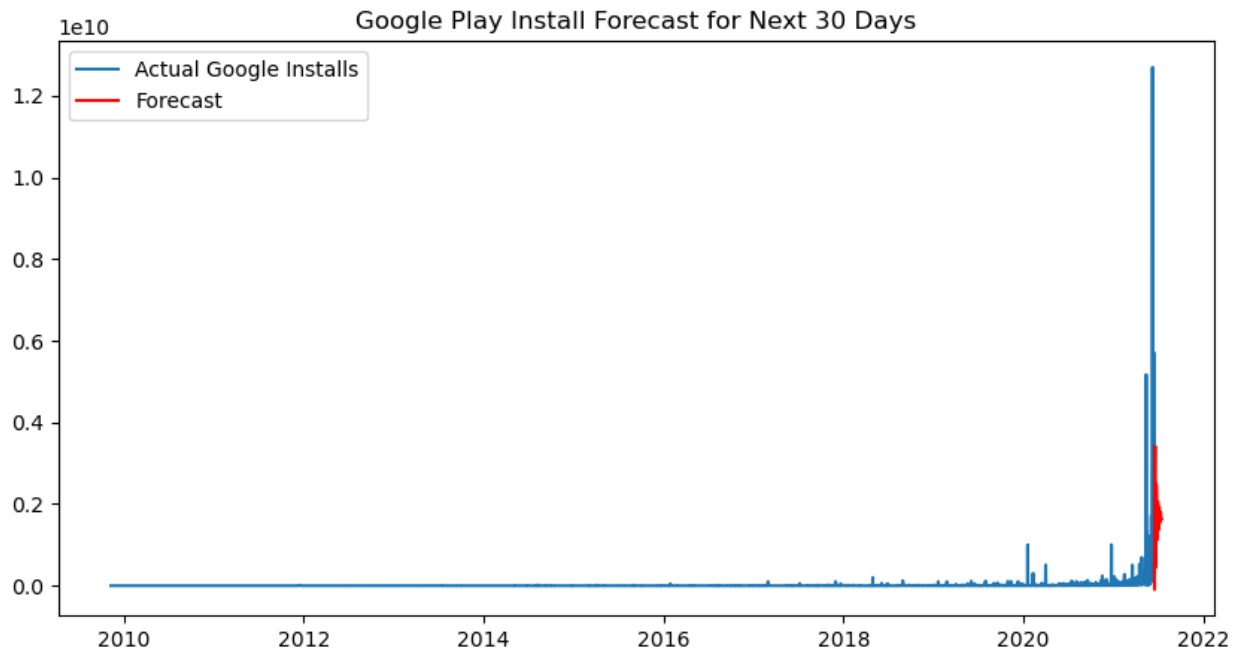
Insights:

- **User ratings significantly influence installs**, with both Apple and Google ratings showing a positive correlation.
- **App price has a minimal effect**, implying that users may not base their installation decision solely on cost.
- **App size has a minor impact**, suggesting that storage constraints do not heavily influence downloads.

3. Forecasting Google Play Store Installs

This time-series forecasting chart predicts Google Play Store installs for the next 30 days. The blue line represents historical data, while the red section shows predicted future installs.

- The forecast suggests a decline in the number of installs compared to previous peaks, possibly due to market saturation or changing user preferences.
- Past spikes in installs (2020-2022) were likely influenced by global events such as the pandemic, where digital adoption surged.



What does this mean?

- Google Play Store should focus on strategies to sustain growth, such as targeted marketing campaigns.
- Developers should align their release schedules with expected peaks in demand to maximize downloads.
- Analyzing external trends (e.g., emerging app categories) could help identify new growth opportunities.

RECOMMENDATIONS

Based on the insights gained, we recommend the following strategies for improving app performance on Google Play Store:

1. Enhance User Engagement & Reviews:

- Developers should actively respond to user reviews, as engaged developers tend to have higher-rated apps.
- Encouraging users to leave reviews through incentives or app reminders can boost ratings.

2. Optimize Pricing Strategy:

- Consider adopting a freemium model with optional in-app purchases, as this is the most successful monetization method.
- Subscription-based pricing should be promoted in premium app categories.

3. Improve App Store Optimization (ASO):

- Focus on high-ranking keywords in app descriptions.
- Regularly update app visuals (screenshots, videos) to attract new users.
- Ensure a clear and concise app description with bullet points highlighting key features.

4. Leverage Category-Specific Insights:

- Target growing categories like Finance, Health & Fitness, and Productivity for better long-term performance.
- For gaming apps, a strong user retention strategy (e.g., daily rewards, social engagement) is crucial.

CONCLUSION

This analysis provides an in-depth comparison of app performance across the Google Play Store and Apple App Store. By understanding KPIs, business insights, and optimization strategies, app developers and businesses can make data-driven decisions to enhance user experience and revenue potential.

Implementing the recommended strategies will help improve visibility, engagement, and monetization, leading to a stronger presence in the highly competitive app market.