In [ ]:
```
# Intenship project

Title: Unveiling the Android App Market

Subtitle: “A Data-Driven Exploration of App Performance and User Sentiment”

Presented by: Faleye Doyin Opeyemi

Date: 23-8-2025
```

1. Data Preparation

In [47]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import re
from textblob import TextBlob
from sklearn.feature_extraction.text import TfidfVectorizer
```

In [94]:
```
# Load datasets
apps_df = pd.read_csv("C:\\Users\\FALEYE DOYINSOLA\\OneDrive\\Desktop\\project  8 Unveli
reviews_df = pd.read_csv("C:\\Users\\FALEYE DOYINSOLA\\OneDrive\\Desktop\\project  8 Unv
```

In [95]:
```
# preveiw the App data
apps_df.head()
```

Out[95]:

	Unnamed: 0	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10,000+	Free	0	Everyone	
1	1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500,000+	Free	0	Everyone	De
2	2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5,000,000+	Free	0	Everyone	
3	3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50,000,000+	Free	0	Teen	
4	4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100,000+	Free	0	Everyone	Desi

In [96]:
```
# preveiw the User reveiw data
reviews_df.head()
```

Out[96]:

	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

In [97]:
```
# Basic inspection on the App data information
apps_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9659 entries, 0 to 9658
Data columns (total 14 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Unnamed: 0      9659 non-null   int64  
 1   App             9659 non-null   object 
 2   Category        9659 non-null   object 
 3   Rating          8196 non-null   float64
 4   Reviews         9659 non-null   int64  
 5   Size            8432 non-null   float64
 6   Installs        9659 non-null   object 
 7   Type            9659 non-null   object 
 8   Price           9659 non-null   object 
 9   Content Rating  9659 non-null   object 
 10  Genres          9659 non-null   object 
 11  Last Updated    9659 non-null   object 
 12  Current Ver     9651 non-null   object 
 13  Android Ver     9657 non-null   object 
dtypes: float64(2), int64(2), object(10)
memory usage: 1.0+ MB
```

In [98]:
```
# Basic inspection on the User Reveiw data information
reviews_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64295 entries, 0 to 64294
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   App                     64295 non-null  object 
 1   Translated_Review       37427 non-null  object 
 2   Sentiment               37432 non-null  object 
 3   Sentiment_Polarity      37432 non-null  float64
 4   Sentiment_Subjectivity  37432 non-null  float64
dtypes: float64(2), object(3)
memory usage: 2.5+ MB
```

In [99]:
```
# Convert price to float in the App Dataset
apps_df['Price'] = apps_df['Price'].str.replace('$', '').astype(float)
```

```
C:\Users\FALEYE DOYINSOLA\AppData\Local\Temp\ipykernel_20556\3571078355.py:2: FutureWar
ning:

The default value of regex will change from True to False in a future version. In addit
ion, single character regular expressions will *not* be treated as literal strings when
regex=True.
```

In [100]:
```
# preveiw the App datatype information to check if the Price covertion work
apps_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9659 entries, 0 to 9658
Data columns (total 14 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Unnamed: 0      9659 non-null   int64  
 1   App             9659 non-null   object 
 2   Category        9659 non-null   object 
 3   Rating          8196 non-null   float64
 4   Reviews         9659 non-null   int64  
 5   Size            8432 non-null   float64
 6   Installs        9659 non-null   object 
 7   Type            9659 non-null   object 
 8   Price           9659 non-null   float64
 9   Content Rating  9659 non-null   object 
 10  Genres          9659 non-null   object 
 11  Last Updated    9659 non-null   object 
 12  Current Ver     9651 non-null   object 
 13  Android Ver     9657 non-null   object 
dtypes: float64(3), int64(2), object(9)
memory usage: 1.0+ MB
```

In [101]:
```
# Clean installs column
apps_df['Installs'] = apps_df['Installs'].str.replace('[+,]', '', regex=True).astype(int
```

In [102]:
```
# preview the clean column- Installs
apps_df['Installs']
```

Out[102]:
```
0          10000
1         500000
2        5000000
3       50000000
4         100000
           ...   
9654        5000
9655         100
9656        1000
9657        1000
9658    10000000
Name: Installs, Length: 9659, dtype: int32
```

Text Preprocessing (NLP)

In [103]:
```
# Define preprocessing function,
# we are using this function for the User Review dataset coz the Translated_Review colum
# it also consist of upper and lower case word in a sentence
import string
def clean_review(text):
    text = text.lower() # lowercase
    text= re.sub(r"http\S+|www\S+https\S+", '',text, flags=re.MULTILINE) # regre express
    text= text.translate(str.maketrans('','',string.punctuation)) # removing all punctua
    text = re.sub(r'\d+', '', text) # remove all numbers
    return text
```

In [104]:
```
# let run the clean review function we created
reviews_df['Clean_review']= reviews_df['Translated_Review'].astype(str).apply(clean_revi
```

In [105]:
```
# let preview the dataset again to see if the function worked
reviews_df.head()
```

Out[105]:

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

In [106]:
```
# previewing 10 list of the 'clean_review' column to verify well if the functions create
reviews_df['Clean_review']. iloc[:10]
```

Out[106]:
```
0    i like eat delicious food thats im cooking foo...
1      this help eating healthy exercise regular basis
2                                                  nan
3           works great especially going grocery store
4                                        best idea us
5                                            best way
6                                             amazing
7                                                 nan
8                                looking forward app
9                  it helpful site  it help foods get
Name: Clean_review, dtype: object
```

In [107]:
```
# checking for the Missing values in each colum for App dataset
apps_df.isnull().sum()
```

Out[107]:
```
Unnamed: 0          0
App                 0
Category            0
Rating           1463
Reviews             0
Size             1227
Installs            0
Type                0
Price               0
Content Rating      0
Genres              0
Last Updated        0
Current Ver         8
Android Ver         2
dtype: int64
```

In [108]:
```
# checking for the Missing values in each colum for User Reviews  dataset
reviews_df.isnull().sum()
```

Out[108]:
```
App                        0
Translated_Review      26868
Sentiment              26863
Sentiment_Polarity     26863
Sentiment_Subjectivity 26863
Clean_review               0
dtype: int64
```

In [109]:
```
# Drop rows with missing critical values for App dataset
apps_df.dropna(subset=['Rating', 'Installs'], inplace=True)
```

In [110]:
```
apps_df.head()
```

Out[110]:

	Unnamed: 0	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	
0	0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19.0	10000	Free	0.0	Everyone	Art
1	1	Coloring book moana	ART_AND_DESIGN	3.9	967	14.0	500000	Free	0.0	Everyone	Desig
2	2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7	5000000	Free	0.0	Everyone	Art
3	3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25.0	50000000	Free	0.0	Teen	Art
4	4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8	100000	Free	0.0	Everyone	Design

In [111]:
```
# Drop rows with missing critical values for User Review dataset
reviews_df.dropna(subset=['Sentiment', 'Sentiment_Polarity', 'Sentiment_Subjectivity'],
```

In [112]:
```
reviews_df.head()
```

Out[112]:

	App	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity	Clean_review
0	10 Best Foods for You	I like eat delicious food. That's I'm cooking ...	Positive	1.00	0.533333	i like eat delicious food thats im cooking foo...
1	10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462	this help eating healthy exercise regular basis
3	10 Best Foods for You	Works great especially going grocery store	Positive	0.40	0.875000	works great especially going grocery store
4	10 Best Foods for You	Best idea us	Positive	1.00	0.300000	best idea us
5	10 Best Foods for You	Best way	Positive	1.00	0.300000	best way

In [113]:
```
# Drop unwanted Column
# let drop Translated column because its not useful anymore
reviews_df.drop(['Translated_Review'],axis =1, inplace=True)
```

In [114]:
```
# let preview the User Reviews data
reviews_df.head()
```

Out[114]:

	App	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity	Clean_review
0	10 Best Foods for You	Positive	1.00	0.533333	i like eat delicious food thats im cooking foo...
1	10 Best Foods for You	Positive	0.25	0.288462	this help eating healthy exercise regular basis
3	10 Best Foods for You	Positive	0.40	0.875000	works great especially going grocery store
4	10 Best Foods for You	Positive	1.00	0.300000	best idea us
5	10 Best Foods for You	Positive	1.00	0.300000	best way

In [117]:
```
# checking for duplicates in Apps data
apps_df.duplicated()
```

Out[117]:
```
0       False
1       False
2       False
3       False
4       False
        ...
9652    False
9654    False
9655    False
9657    False
9658    False
Length: 8196, dtype: bool
```

In [118]:
```
# checking for duplicate in User Reviews data
reviews_df.duplicated()
```

Out[118]:
```
0        False
1        False
3        False
4        False
5        False
         ...
64222    False
64223    False
64226    False
64227    False
64230    False
Length: 37432, dtype: bool
```

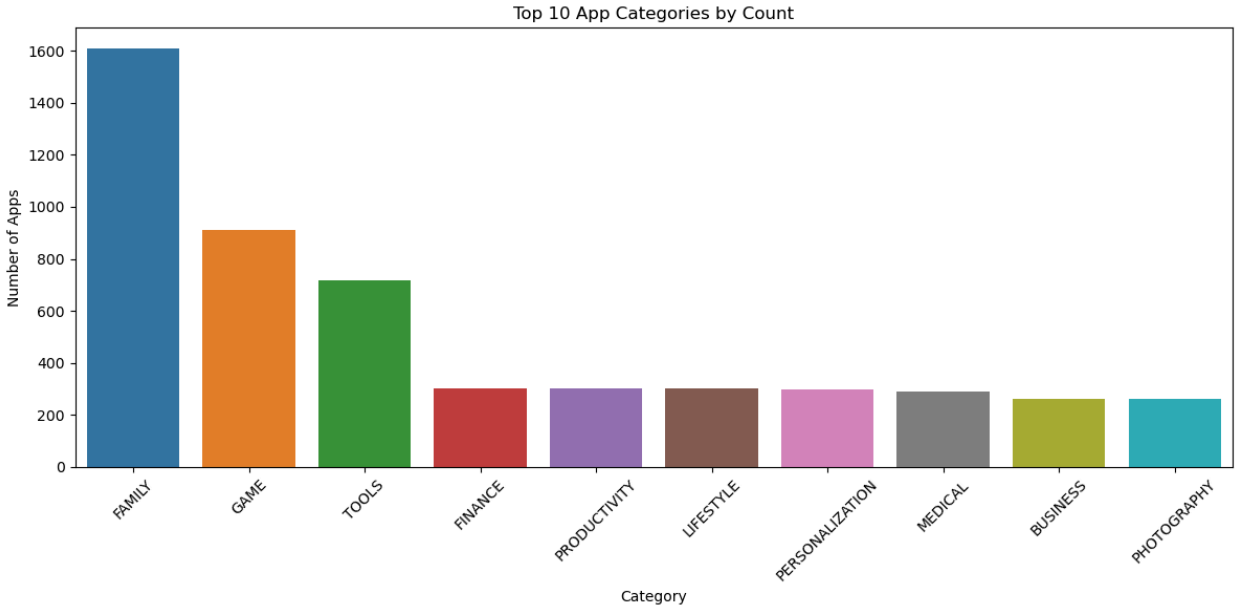
2. Category Exploration

In [76]:
```
# Count of apps per category
category_counts = apps_df['Category'].value_counts()

print(category_counts)
```

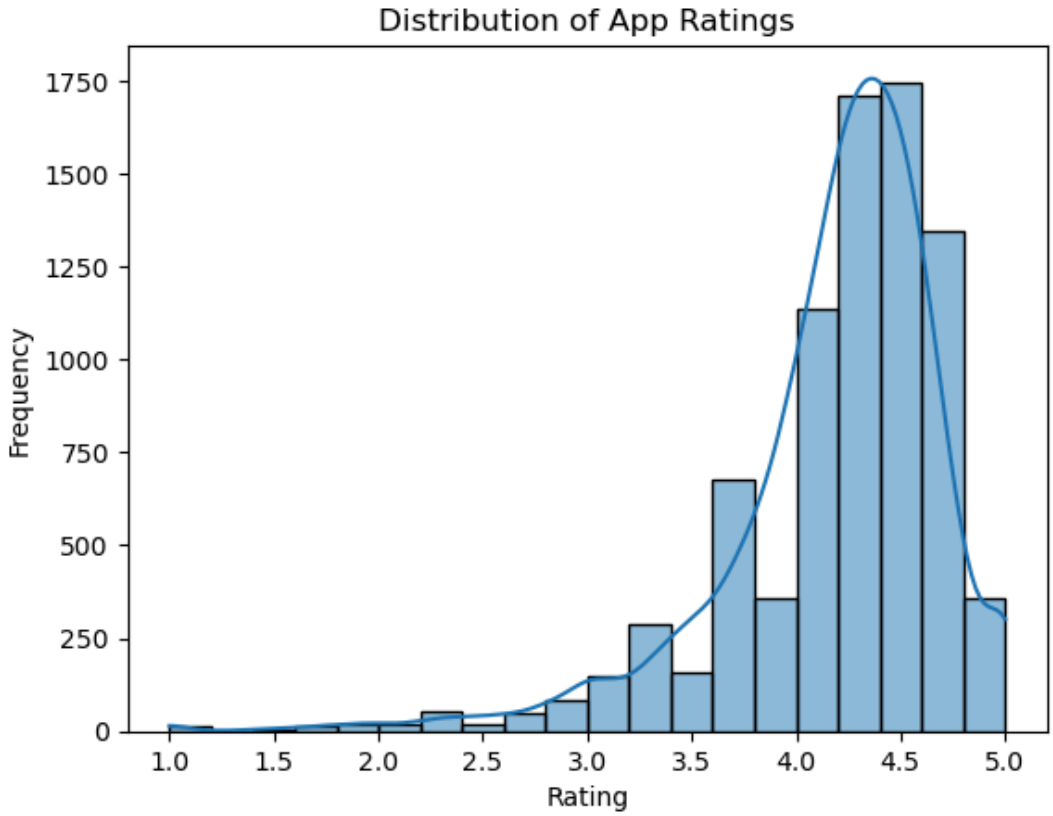
```
FAMILY                 1608
GAME                    912
TOOLS                   718
FINANCE                 302
PRODUCTIVITY            301
LIFESTYLE               301
PERSONALIZATION         298
MEDICAL                 290
BUSINESS                263
PHOTOGRAPHY             263
SPORTS                  260
COMMUNICATION           256
HEALTH_AND_FITNESS      244
NEWS_AND_MAGAZINES      204
SOCIAL                  203
TRAVEL_AND_LOCAL        187
SHOPPING                180
BOOKS_AND_REFERENCE     169
VIDEO_PLAYERS           148
DATING                  134
MAPS_AND_NAVIGATION     118
EDUCATION               118
ENTERTAINMENT           102
FOOD_AND_DRINK           94
AUTO_AND_VEHICLES        73
WEATHER                  72
LIBRARIES_AND_DEMO       64
HOUSE_AND_HOME           62
ART_AND_DESIGN           61
COMICS                   54
PARENTING                50
EVENTS                   45
BEAUTY                   42
Name: Category, dtype: int64
```

In [90]:
```
# Plot Count for apps dataset per category
plt.figure(figsize=(12,6))
sns.barplot(x=category_counts.index[:10], y=category_counts.values[:10])
plt.xticks(rotation=45)
plt.title('Top 10 App Categories by Count')
plt.ylabel('Number of Apps')
plt.xlabel('Category')
plt.tight_layout()
plt.show()
```

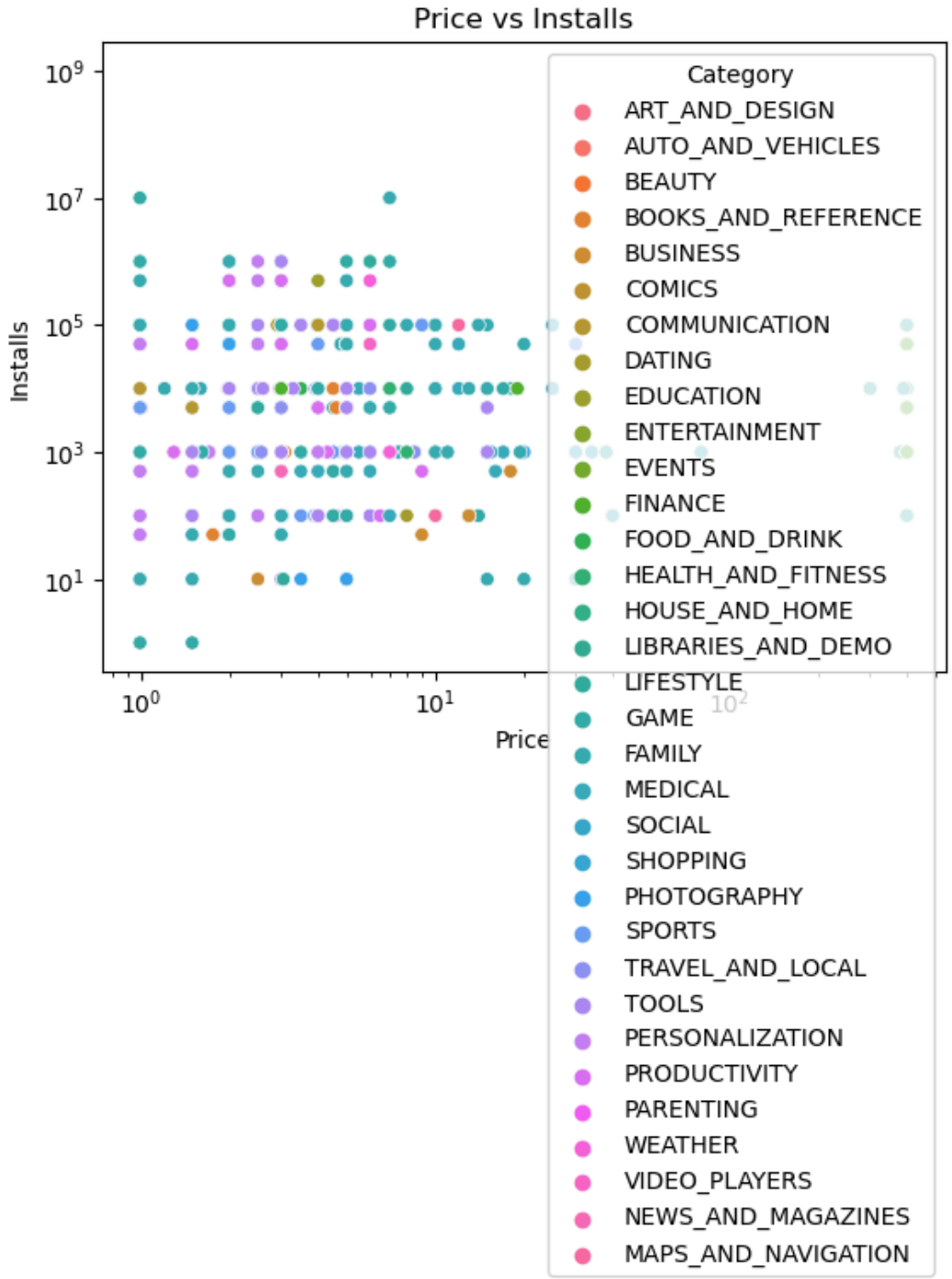


3. Metrics Analysis

In [78]:
```
# let plot a historical plot to check the Rating distribution for Apps data
sns.histplot(apps_df['Rating'], bins=20, kde=True)
plt.title('Distribution of App Ratings')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.show()
```



In [79]:
```
# let plot a scatter plot for Price vs Installs for Apps Data
sns.scatterplot(data=apps_df, x='Price', y='Installs', hue='Category')
plt.title('Price vs Installs')
plt.xscale('log')
plt.yscale('log')
plt.show()
```



4. Sentiment Analysis (User Reviews)

In [85]:
```
# from textblob import TextBlob

# Apply sentiment polarity
reviews_df['Sentiment'] = reviews_df['Clean_review'].apply(lambda x: TextBlob(str(x)).se

print(reviews_df['Sentiment'])
```

```
0        1.000000
1        0.250000
3        0.400000
4        1.000000
5        1.000000
           ...
64222    0.113333
64223    0.225000
64226   -0.287500
64227    0.800000
64230   -0.316667
Name: Sentiment, Length: 37432, dtype: float64
```

In [86]:
```
# Merge both Apps data and User Reviews data as one dataset(merged_df)
# also we are using 'inner join' to merged both coz App column is present in both datase

merged_df = pd.merge(reviews_df, apps_df, on='App', how='inner')
```

In [87]:
```
# let preview our new dataset called - merged_df
merged_df.head()
```

Out[87]:

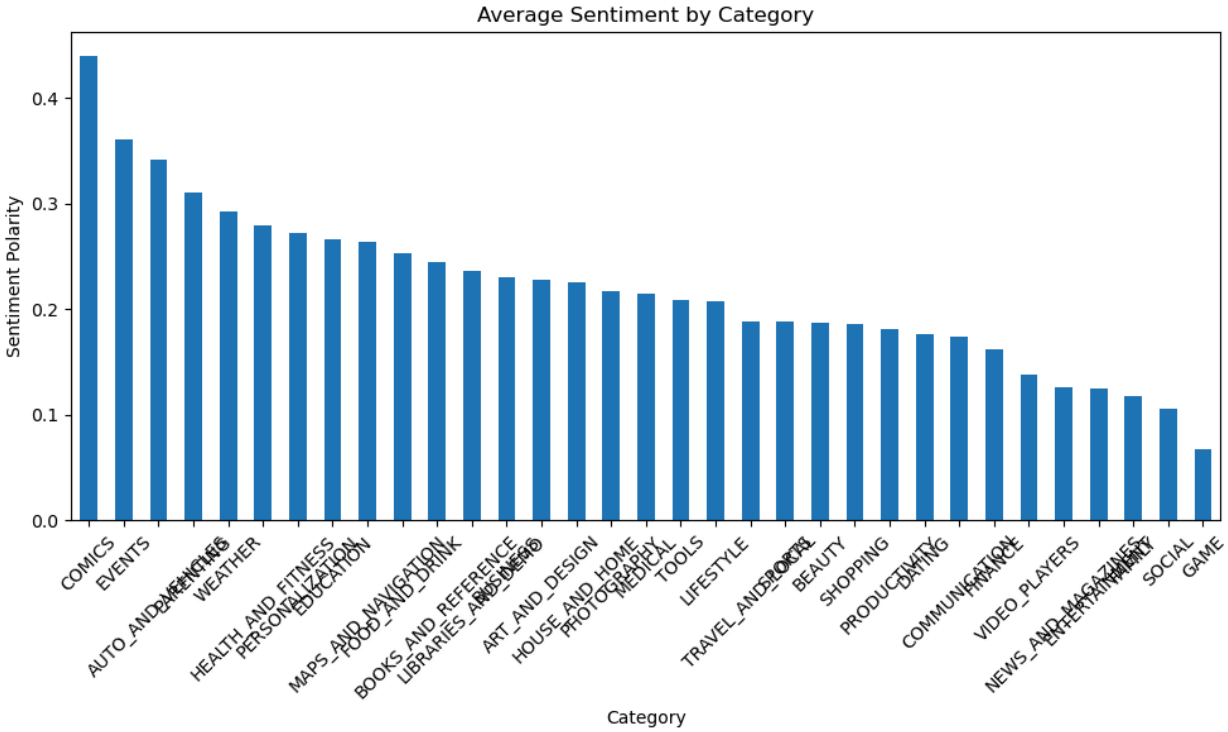
	App	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity	Clean_review	Unnamed: 0	Categ
0	10 Best Foods for You	1.00	1.00	0.533333	i like eat delicious food thats im cooking foo...	1393	HEALTH_AND_FITNE
1	10 Best Foods for You	0.25	0.25	0.288462	this help eating healthy exercise regular basis	1393	HEALTH_AND_FITNE
2	10 Best Foods for You	0.40	0.40	0.875000	works great especially going grocery store	1393	HEALTH_AND_FITNE
3	10 Best Foods for You	1.00	1.00	0.300000	best idea us	1393	HEALTH_AND_FITNE
4	10 Best Foods for You	1.00	1.00	0.300000	best way	1393	HEALTH_AND_FITNE

In [89]:
```
# let check for the Average sentiment per category using our new merged_df dataset
sentiment_by_category = merged_df.groupby('Category')['Sentiment'].mean().sort_values(as

print(sentiment_by_category)
```

```
Category
COMICS                0.439895
EVENTS                0.360715
AUTO_AND_VEHICLES     0.341639
PARENTING             0.309719
WEATHER               0.292839
HEALTH_AND_FITNESS    0.279238
PERSONALIZATION       0.271846
EDUCATION             0.266334
MAPS_AND_NAVIGATION   0.263547
FOOD_AND_DRINK        0.252680
BOOKS_AND_REFERENCE   0.243845
LIBRARIES_AND_DEMO    0.235713
BUSINESS              0.229680
ART_AND_DESIGN        0.227809
HOUSE_AND_HOME        0.224875
PHOTOGRAPHY           0.217018
MEDICAL               0.214010
TOOLS                 0.208840
LIFESTYLE             0.207424
TRAVEL_AND_LOCAL      0.188564
SPORTS                0.188059
BEAUTY                0.187338
SHOPPING              0.185283
PRODUCTIVITY          0.181230
DATING                0.175638
COMMUNICATION         0.173498
FINANCE               0.162423
VIDEO_PLAYERS         0.137586
NEWS_AND_MAGAZINES    0.125409
ENTERTAINMENT         0.125081
FAMILY                0.118058
SOCIAL                0.105194
GAME                  0.067222
Name: Sentiment, dtype: float64
```

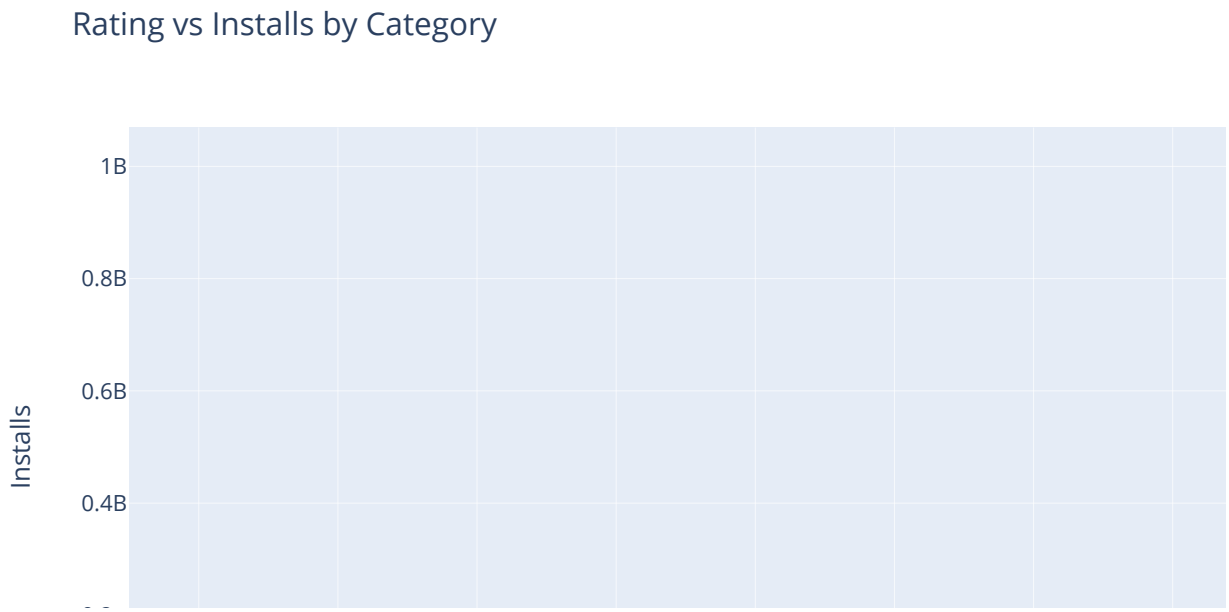
In [91]:
```
# Plot Average sentiment per category on the merged_df data
plt.figure(figsize=(10,6))
sentiment_by_category.plot(kind='bar')
plt.title('Average Sentiment by Category')
plt.ylabel('Sentiment Polarity')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



5. Interactive Visualization using Plotly

In [93]:
```
# import plotly.express as px
# let plot a scatter plot to check for Rating VS Installs by Category

fig = px.scatter(apps_df, x='Rating', y='Installs', color='Category',
                 hover_data=['App', 'Price'], title='Rating vs Installs by Category')
fig.show()
```



6. Skill Enhancement Summary

Skills Practiced

- Data Cleaning & Preprocessing
- Exploratory Data Analysis (EDA)
- Natural Language Processing (NLP)
- Data Visualization (Matplotlib, Seaborn, Plotly)
- Merging and Aggregating Data
- Sentiment Analysis

In [ ]: