Regression_and_Sentiment_Analysis

July 17, 2023

[88]:
```
import pandas as pd
```

1 Task 1 (Play Store Data Analysis)

1.1 Read CSV Data

[93]:
```
df = pd.read_csv("googleplaystore.csv")
df
```

[93]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0			
2	U Launcher Lite - FREE Live Cool Themes, Hide …	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0			
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0			
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0			
…	…	…	…	…	…	…	…	…	…	…	…
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0			
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0			
10838	Parkinson Exercices FR	MEDICAL	NaN	3	9.5M	1,000+	Free	0			
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	Varies with device	1,000+	Free	0			
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19M	10,000,000+	Free	0			

```
1           Everyone  Art & Design;Pretend Play  January 15, 2018
2           Everyone               Art & Design    August 1, 2018
3               Teen               Art & Design      June 8, 2018
4           Everyone    Art & Design;Creativity     June 20, 2018
…                ...                        …               …
10836       Everyone                 Education     July 25, 2017
10837       Everyone                 Education      July 6, 2018
10838       Everyone                   Medical  January 20, 2017
10839     Mature 17+          Books & Reference  January 19, 2015
10840       Everyone                 Lifestyle     July 25, 2018

              Current Ver         Android Ver
0                   1.0.0        4.0.3 and up
1                   2.0.0        4.0.3 and up
2                   1.2.4        4.0.3 and up
3      Varies with device          4.2 and up
4                     1.1          4.4 and up
…                     …                   …
10836                1.48          4.1 and up
10837                 1.0          4.1 and up
10838                 1.0          2.2 and up
10839  Varies with device  Varies with device
10840  Varies with device  Varies with device

[10841 rows x 13 columns]
```

[94]:
```
df.describe()
```

[94]:
```
             Rating
count  9367.000000
mean      4.193338
std       0.537431
min       1.000000
25%       4.000000
50%       4.300000
75%       4.500000
max      19.000000
```

[95]:
```
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         10841 non-null  object
 4   Size            10841 non-null  object
 5   Installs        10841 non-null  object
 6   Type            10840 non-null  object
 7   Price           10841 non-null  object
 8   Content Rating  10840 non-null  object
 9   Genres          10841 non-null  object
 10  Last Updated    10841 non-null  object
 11  Current Ver     10833 non-null  object
 12  Android Ver     10838 non-null  object
dtypes: float64(1), object(12)
memory usage: 1.1+ MB
```

[96]:
```
df.columns
```

[96]:
```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
       'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
       'Android Ver'],
      dtype='object')
```

1.2 Data Cleaning

[106]:
```
df.duplicated().sum()
```

[106]:
```
483
```

[107]:
```
df.drop_duplicates(inplace=True)
```

[108]:
```
df.duplicated().sum()
```

[108]:
```
0
```

[109]:
```
# data wrangling/cleaning
df.Rating.fillna(df.Rating.mean(),inplace=True)
df.Rating.isna().sum()
```

[109]:
```
0
```

[128]:
```
df.Reviews.fillna(df.Reviews.mean(),inplace=True)
df.Reviews.isna().sum()
```

[128]:
```
0
```

[98]:
```
df.loc[df.Rating>5] = df.Rating.mean()
```

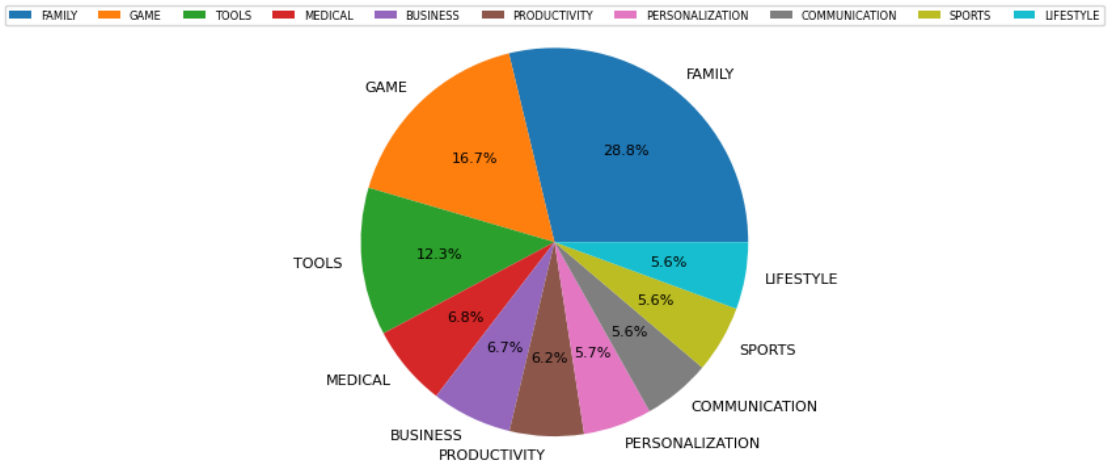
[99]:
```
df.isna().sum()
```

[99]:
```
App                0
Category           0
Rating             0
Reviews            0
Size               0
Installs           0
Type               1
Price              0
Content Rating     0
Genres             0
Last Updated       0
Current Ver        8
Android Ver        2
dtype: int64
```

1.3 Data Visualization

Top 10 Categories by Games

[100]:
```
import matplotlib.pyplot as plt

apps_count_by_category = df.groupby('Category')['Category'].count()
top_10_categories = apps_count_by_category.sort_values(ascending=False).head(10)

plt.pie(top_10_categories, labels=top_10_categories.index,autopct='%1.
↪1f%%',textprops={'fontsize': 8})
plt.legend(top_10_categories.index,loc="upper center", ncol=10,prop={'size': 6})
plt.show()
```



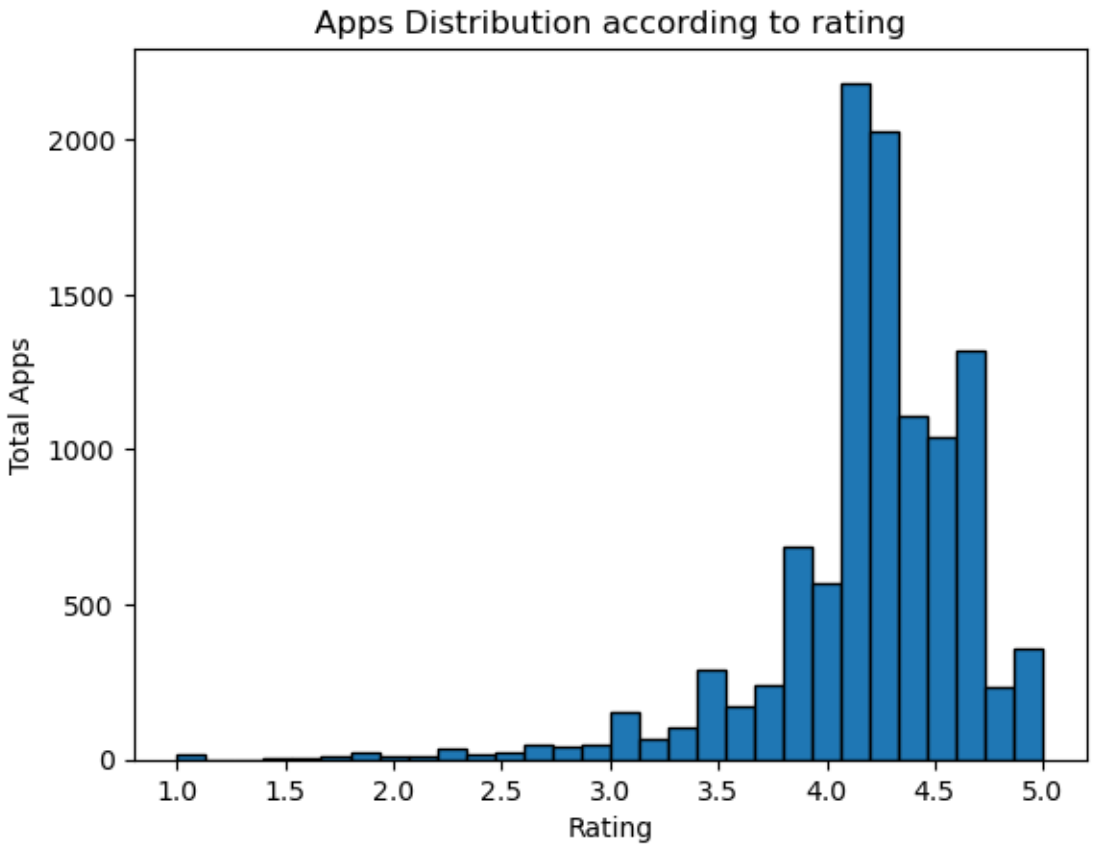
1.4 Basic Statistics

[101]:
```
# Descriptive Statistics:
print(df.Rating.mean())
print(df.Rating.median())
print(df.Rating.std())
```

```
4.191972513159116
4.2
0.4788850498067651
```

1.5 Apps Distribution according to rating

[102]:
```
category_avg_rating = df.groupby('Category')['Rating'].mean()
category_avg_rating

plt.hist(df.Rating, bins=30, edgecolor='black')
plt.xlabel('Rating')
plt.ylabel('Total Apps')
plt.title('Apps Distribution according to rating')
plt.show()
```



[103]:
```
df
```

[103]:

```
                                                 App             Category  \
0         Photo Editor & Candy Camera & Grid & ScrapBook       ART_AND_DESIGN
1                                    Coloring book moana       ART_AND_DESIGN
2      U Launcher Lite - FREE Live Cool Themes, Hide …       ART_AND_DESIGN
3                                    Sketch - Draw & Paint       ART_AND_DESIGN
4                  Pixel Draw - Number Art Coloring Book       ART_AND_DESIGN
…                                                    …                  …
10836                                  Sya9a Maroc - FR              FAMILY
10837                  Fr. Mike Schmitz Audio Teachings              FAMILY
10838                           Parkinson Exercices FR             MEDICAL
10839                     The SCP Foundation DB fr nn5n  BOOKS_AND_REFERENCE
10840      iHoroscope - 2018 Daily Horoscope & Astrology           LIFESTYLE

         Rating Reviews                Size     Installs  Type Price  \
0      4.100000     159                 19M      10,000+  Free     0
1      3.900000     967                 14M     500,000+  Free     0
2      4.700000   87510                8.7M   5,000,000+  Free     0
3      4.500000  215644                 25M  50,000,000+  Free     0
4      4.300000     967                2.8M     100,000+  Free     0
…           …       …                   …           …    …    …
10836  4.500000      38                 53M       5,000+  Free     0
10837  5.000000       4                3.6M         100+  Free     0
10838  4.193338       3                9.5M       1,000+  Free     0
10839  4.500000     114  Varies with device       1,000+  Free     0
10840  4.500000  398307                 19M  10,000,000+  Free     0

      Content Rating                     Genres      Last Updated  \
0           Everyone               Art & Design   January 7, 2018
1           Everyone  Art & Design;Pretend Play  January 15, 2018
2           Everyone               Art & Design    August 1, 2018
3               Teen               Art & Design      June 8, 2018
4           Everyone    Art & Design;Creativity     June 20, 2018
…              …                          …               …
10836       Everyone                  Education     July 25, 2017
10837       Everyone                  Education      July 6, 2018
10838       Everyone                    Medical  January 20, 2017
10839     Mature 17+          Books & Reference  January 19, 2015
10840       Everyone                  Lifestyle     July 25, 2018

             Current Ver         Android Ver
0                  1.0.0       4.0.3 and up
1                  2.0.0       4.0.3 and up
2                  1.2.4       4.0.3 and up
3      Varies with device         4.2 and up
4                    1.1         4.4 and up
```

```
…                    …                   …
10836                1.48          4.1 and up
10837                 1.0          4.1 and up
10838                 1.0          2.2 and up
10839  Varies with device  Varies with device
10840  Varies with device  Varies with device

[10841 rows x 13 columns]
```

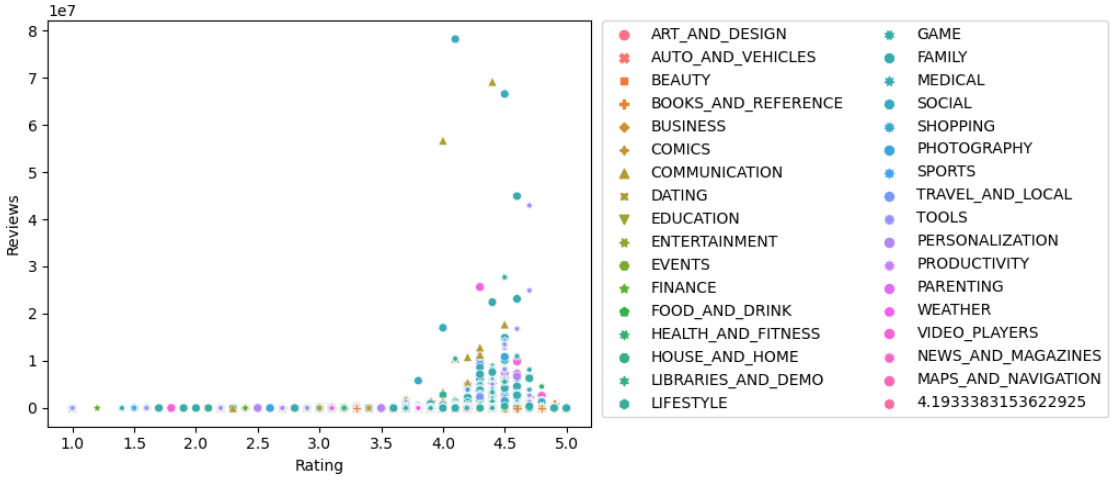
[104]:
```
# df['Reviews'] = df['Reviews'].str.replace('M+', '000000').astype(float)

# df['Reviews'].fillna(df['Reviews'].mode())
# df.loc[df.Reviews.isnull()]

df.loc[df.Reviews.isnull()].Reviews = df.Reviews.mode()

# df.loc[df.Reviews.isnull()]

# df.drop(labels=[10472], inplace=True)
```

```
C:\Users\Hp\AppData\Local\Temp\ipykernel_7368\2220076779.py:6:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.loc[df.Reviews.isnull()].Reviews = df.Reviews.mode()
```

1.6 Scatter Graph for Category Ratings

[113]:
```
import seaborn as sns

sns.scatterplot(data=df, x=df.Rating.astype(float), y=df.Reviews.astype(float),␣
 ↪hue="Category", style="Category")
plt.legend(bbox_to_anchor=(1.02, 1), loc='upper left', borderaxespad=0, ncol=2)
```

[113]:
```
<matplotlib.legend.Legend at 0x19f4c474640>
```



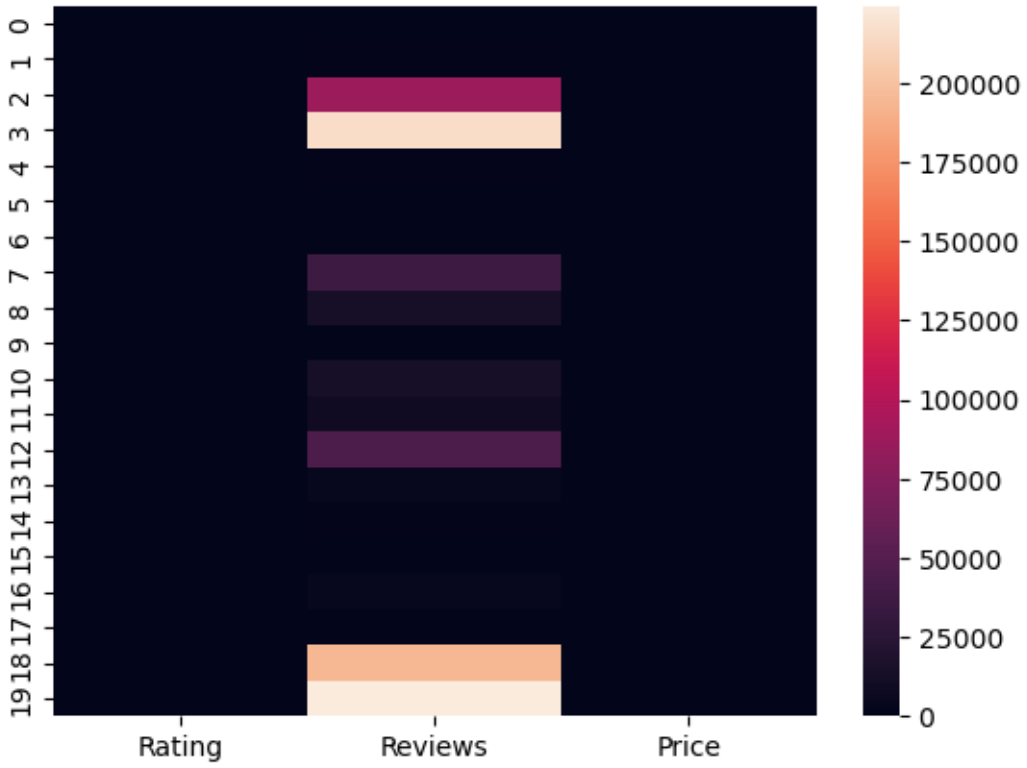
[251]:
```
df['Price'] = df['Price'].str.replace('$', '')
```

```
C:\Users\Hp\AppData\Local\Temp\ipykernel_16992\961214681.py:1: FutureWarning:
The default value of regex will change from True to False in a future version.
In addition, single character regular expressions will *not* be treated as
literal strings when regex=True.
  df['Price'] = df['Price'].str.replace('$', '')
```

1.7 Heat Map for Rating, Reviews and Pricing

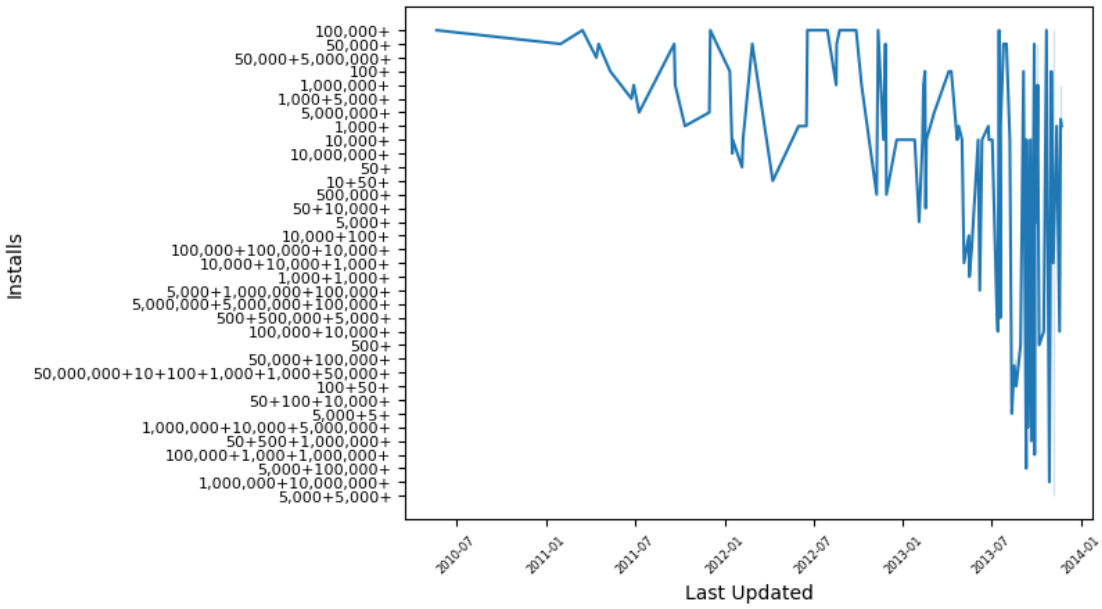
[111]:
```
rat_rev_df = pd.concat([df['Rating'].head(20).astype(float),df['Reviews'].
↪head(20).astype(float),
                        df['Price'].head(20).astype(float)], axis=1)
sns.heatmap(data=rat_rev_df)
```

[111]: <Axes: >



1.8 App Installs Throughtout Years

[322]:
```
df['Last Updated'] = pd.to_datetime(df['Last Updated'])
stats_by_date = df.groupby([df['Last Updated'].dt.year, df['Last Updated'].dt.
 ↪month, 'Category']).agg({'Rating': 'mean', 'Installs': 'sum', 'Category':
 ↪"count", "Last Updated": 'mean'}).head(100)

sns.lineplot(data=stats_by_date, x=stats_by_date['Last Updated'],␣
 ↪y=stats_by_date['Installs'])
plt.xticks(rotation=45, fontsize=6)
plt.yticks(fontsize=8)
plt.show()
```



[323]:
```
stats_by_date
```

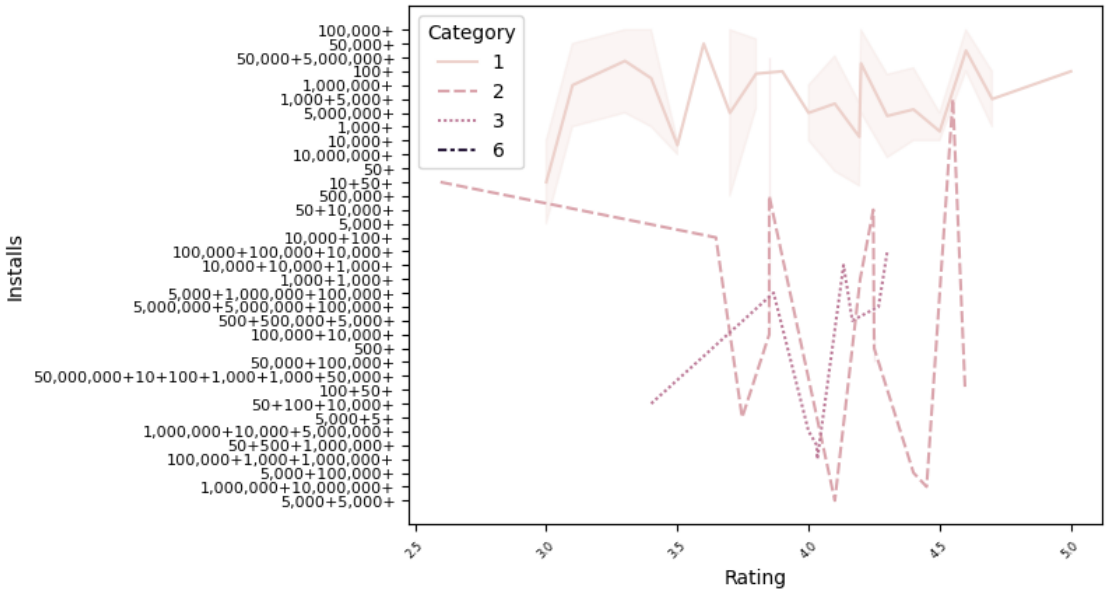
[323]:

			Rating	Installs
Last Updated	Last Updated	Category		
2010	5	FAMILY	4.20	100,000+
2011	1	GAME	4.10	50,000+
	3	TOOLS	4.10	100,000+
	4	FAMILY	3.60	50,000+
		GAME	3.85	50,000+5,000,000+
…			…	…
2013	11	HOUSE_AND_HOME	3.40	100,000+
		MEDICAL	4.10	5,000+5,000+
		PERSONALIZATION	3.85	100,000+10,000+
		SPORTS	4.00	1,000,000+
		TOOLS	4.10	1,000+

			Category	Last Updated
Last Updated	Last Updated	Category		
2010	5	FAMILY	1	2010-05-21 00:00:00
2011	1	GAME	1	2011-01-30 00:00:00
	3	TOOLS	1	2011-03-16 00:00:00
	4	FAMILY	1	2011-04-18 00:00:00
		GAME	2	2011-04-13 12:00:00
…			…	…
2013	11	HOUSE_AND_HOME	1	2013-11-05 00:00:00
		MEDICAL	2	2013-11-05 00:00:00
		PERSONALIZATION	2	2013-11-18 00:00:00

```
                              SPORTS                1 2013-11-20 00:00:00
                              TOOLS                 1 2013-11-12 00:00:00

[100 rows x 4 columns]
```

[320]:
```
df['Last Updated'] = pd.to_datetime(df['Last Updated'])
stats_by_date = df.groupby([df['Last Updated'].dt.year, df['Last Updated'].dt.
 ↪month, 'Category']).agg({'Rating': 'mean', 'Installs': 'sum', 'Category':
 ↪"count"}).head(100)

sns.lineplot(data=stats_by_date, x=stats_by_date['Rating'],␣
 ↪y=stats_by_date['Installs'],hue="Category", style="Category")
plt.xticks(rotation=45, fontsize=6)
plt.yticks(fontsize=8)
plt.show()
```



[324]:
```
stats_by_date
```

[324]:
```
                                                Rating           Installs  \
Last Updated Last Updated Category
2010         5            FAMILY                   4.20           100,000+
2011         1            GAME                     4.10            50,000+
             3            TOOLS                    4.10           100,000+
             4            FAMILY                   3.60            50,000+
                          GAME                     3.85  50,000+5,000,000+
…                                                  …                …
2013         11           HOUSE_AND_HOME           3.40           100,000+
```

```
                           MEDICAL            4.10       5,000+5,000+
                           PERSONALIZATION    3.85    100,000+10,000+
                           SPORTS             4.00         1,000,000+
                           TOOLS              4.10             1,000+

                                                 Category        Last Updated
Last Updated Last Updated Category
2010         5            FAMILY                      1 2010-05-21 00:00:00
2011         1            GAME                        1 2011-01-30 00:00:00
             3            TOOLS                       1 2011-03-16 00:00:00
             4            FAMILY                      1 2011-04-18 00:00:00
                          GAME                        2 2011-04-13 12:00:00
…                                                  …                   …
2013         11           HOUSE_AND_HOME              1 2013-11-05 00:00:00
                          MEDICAL                     2 2013-11-05 00:00:00
                          PERSONALIZATION             2 2013-11-18 00:00:00
                          SPORTS                      1 2013-11-20 00:00:00
                          TOOLS                       1 2013-11-12 00:00:00

[100 rows x 4 columns]
```

[336]:
```
df
```

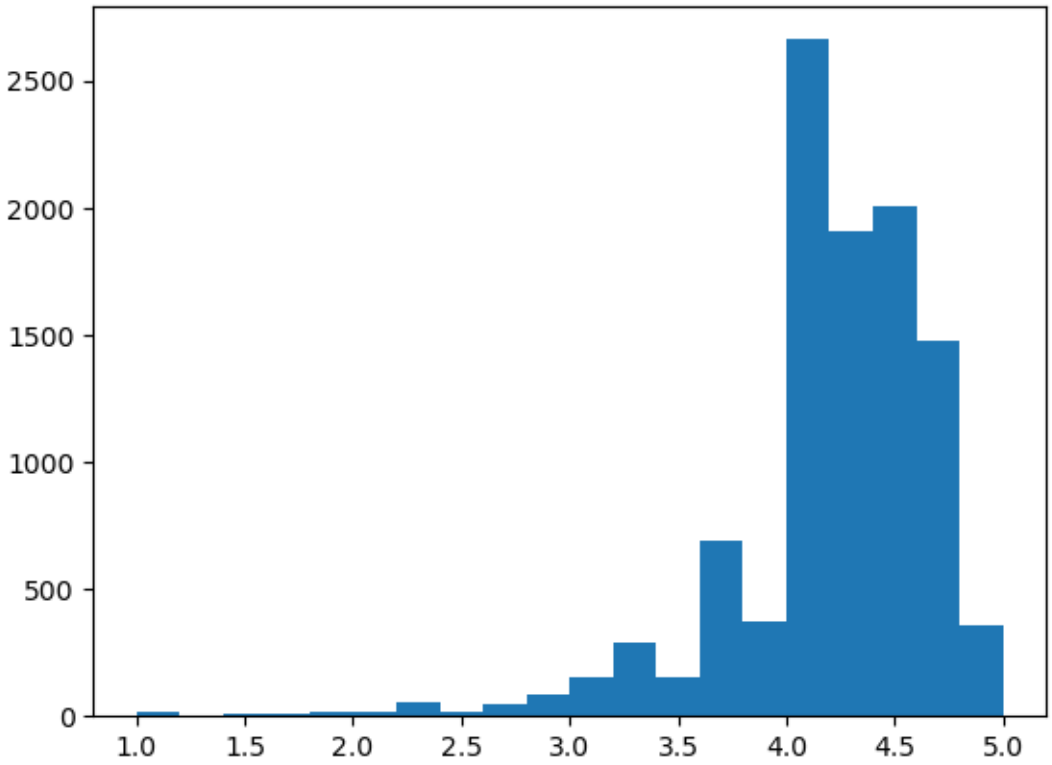
[336]:
```
                                                  App             Category  \
0         Photo Editor & Candy Camera & Grid & ScrapBook       ART_AND_DESIGN
1                                     Coloring book moana       ART_AND_DESIGN
2      U Launcher Lite - FREE Live Cool Themes, Hide …      ART_AND_DESIGN
3                                   Sketch - Draw & Paint       ART_AND_DESIGN
4                  Pixel Draw - Number Art Coloring Book       ART_AND_DESIGN
…                                                    …                   …
10836                                   Sya9a Maroc - FR              FAMILY
10837                   Fr. Mike Schmitz Audio Teachings              FAMILY
10838                            Parkinson Exercices FR             MEDICAL
10839                      The SCP Foundation DB fr nn5n  BOOKS_AND_REFERENCE
10840      iHoroscope - 2018 Daily Horoscope & Astrology            LIFESTYLE

         Rating   Reviews                Size     Installs  Type Price  \
0      4.100000     159.0                 19M      10,000+  Free     0
1      3.900000     967.0                 14M     500,000+  Free     0
2      4.700000   87510.0                8.7M   5,000,000+  Free     0
3      4.500000  215644.0                 25M  50,000,000+  Free     0
4      4.300000     967.0                2.8M     100,000+  Free     0
…           …          …                   …           …    …    …
10836  4.500000      38.0                 53M       5,000+  Free     0
10837  5.000000       4.0                3.6M         100+  Free     0
10838  4.193338       3.0                9.5M       1,000+  Free     0
10839  4.500000     114.0  Varies with device       1,000+  Free     0
```

```
10840  4.500000  398307.0                19M  10,000,000+  Free     0

      Content Rating                     Genres Last Updated  \
0           Everyone               Art & Design   2018-01-07
1           Everyone  Art & Design;Pretend Play   2018-01-15
2           Everyone               Art & Design   2018-08-01
3               Teen               Art & Design   2018-06-08
4           Everyone    Art & Design;Creativity   2018-06-20
…              …                        …           …
10836       Everyone                 Education   2017-07-25
10837       Everyone                 Education   2018-07-06
10838       Everyone                   Medical   2017-01-20
10839     Mature 17+          Books & Reference   2015-01-19
10840       Everyone                 Lifestyle   2018-07-25

              Current Ver         Android Ver
0                   1.0.0        4.0.3 and up
1                   2.0.0        4.0.3 and up
2                   1.2.4        4.0.3 and up
3      Varies with device          4.2 and up
4                     1.1          4.4 and up
…                     …                …
10836                1.48          4.1 and up
10837                 1.0          4.1 and up
10838                 1.0          2.2 and up
10839  Varies with device  Varies with device
10840  Varies with device  Varies with device

[10357 rows x 13 columns]
```

[338]:
```
df.describe()[['Rating', 'Reviews']]
```

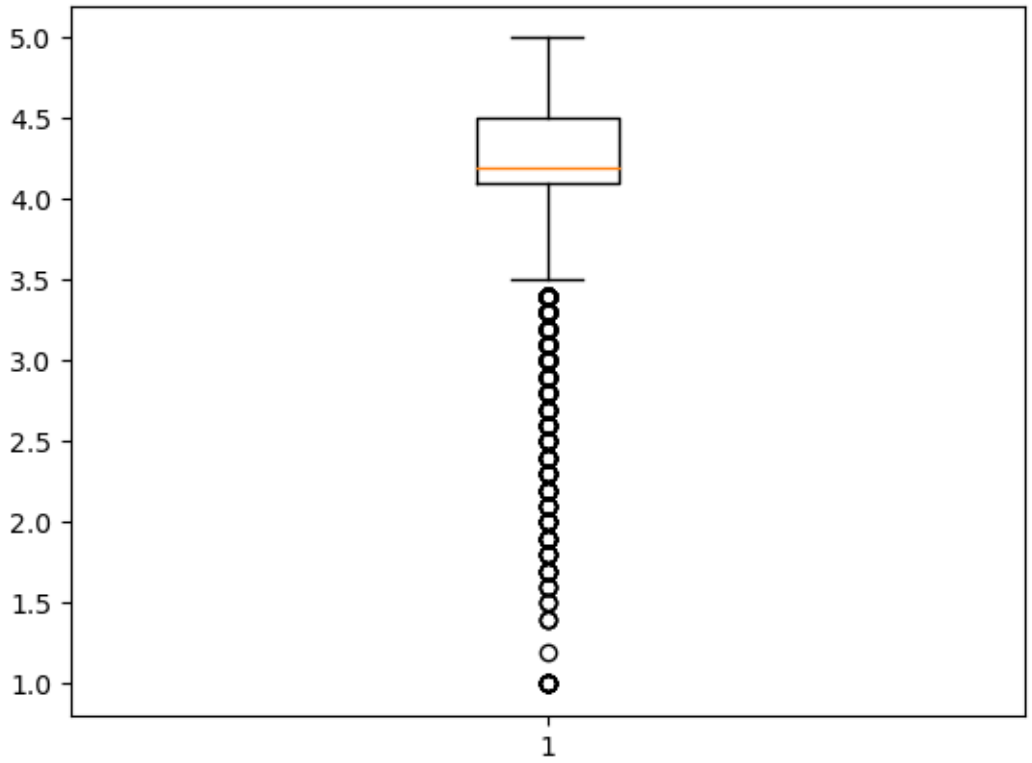
[338]:
```
             Rating       Reviews
count  10357.000000  1.035700e+04
mean       4.188649  4.059046e+05
std        0.484023  2.696778e+06
min        1.000000  0.000000e+00
25%        4.100000  3.200000e+01
50%        4.200000  1.680000e+03
75%        4.500000  4.641600e+04
max        5.000000  7.815831e+07
```

[344]:
```
plt.hist(data=df, x='Rating', bins=20)
plt.show()
```



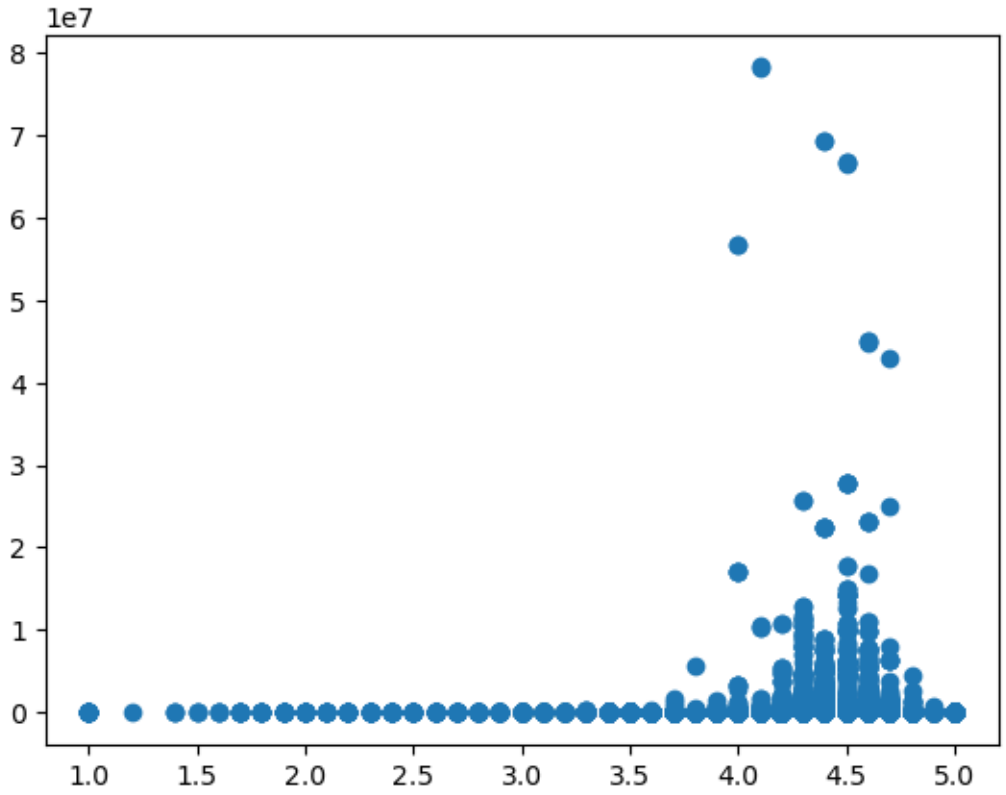
1.9 Handling Outliers

[357]:
```
plt.boxplot(df.Rating)
plt.show()
```



[358]:
```
plt.scatter(df.Rating, df.Reviews)
```

[358]:
```
<matplotlib.collections.PathCollection at 0x1b4c8327fd0>
```



[373]:
```
Q1 = df.Rating.quantile(0.25)
Q3 = df.Rating.quantile(0.75)

IQR = Q3 - Q1
IQR

outliers = df[(df.Rating < (Q1 - 1.5 * IQR)) | (df.Rating > (Q3 + 1.5 * IQR))]


outliers
```

[373]:

	App	Category	Rating	Reviews	\
15	Learn To Draw Kawaii Characters	ART_AND_DESIGN	3.2	55.0	
87	RST - Sale of cars on the PCT	AUTO_AND_VEHICLES	3.2	250.0	
159	Cloud of Books	BOOKS_AND_REFERENCE	3.3	1862.0	
176	Free Book Reader	BOOKS_AND_REFERENCE	3.4	1680.0	
209	Plugin:AOT v5.0	BUSINESS	3.1	4034.0	
…	…	…	…	…	
10757	Fisher-Price® Smart Connect	TOOLS	2.7	422.0	
10765	Chat For Strangers - Video Chat	SOCIAL	3.4	622.0	
10766	FreedomPop Diagnostics	TOOLS	2.9	452.0	

```
10819                        Fanfic-FR  BOOKS_AND_REFERENCE     3.3     52.0
10828          Manga-FR - Anime Vostfr               COMICS     3.4    291.0

                    Size     Installs  Type Price Content Rating  \
15                  2.7M       5,000+  Free     0       Everyone
87                  1.1M     100,000+  Free     0       Everyone
159                  19M   1,000,000+  Free     0       Everyone
176                 4.0M     100,000+  Free     0       Everyone
209                  23k     100,000+  Free     0       Everyone
…                    …          …    …    …             …
10757                72M      50,000+  Free     0       Everyone
10765  Varies with device     100,000+  Free     0     Mature 17+
10766                7.0M     100,000+  Free     0       Everyone
10819                3.6M       5,000+  Free     0           Teen
10828                 13M      10,000+  Free     0       Everyone

                 Genres Last Updated           Current Ver  \
15          Art & Design   2018-06-06                   NaN
87       Auto & Vehicles   2018-04-27                   1.4
159    Books & Reference   2018-04-27                 2.2.5
176    Books & Reference   2016-08-20                  3.05
209             Business   2015-09-11  3.0.1.11 (Build 311)
…                    …          …                     …
10757              Tools   2018-02-23                 2.4.1
10765             Social   2018-05-23   Varies with device
10766              Tools   2017-07-17       1.03.123.0713
10819  Books & Reference   2017-08-05                 0.3.4
10828             Comics   2017-05-15                 2.0.1

              Android Ver
15              4.2 and up
87            4.0.3 and up
159             4.1 and up
176           4.0.3 and up
209             2.2 and up
…                    …
10757           4.4 and up
10765  Varies with device
10766         4.0.3 and up
10819           4.1 and up
10828           4.0 and up

[723 rows x 13 columns]
```

[377]:
```
df.groupby("Category")['Rating'].mean().sort_values(ascending=False)
```

[377]:
```
Category
EDUCATION              4.374564
EVENTS                 4.363647
ART_AND_DESIGN         4.350462
BOOKS_AND_REFERENCE    4.311943
PERSONALIZATION        4.305620
PARENTING              4.282223
GAME                   4.277598
BEAUTY                 4.260882
HEALTH_AND_FITNESS     4.251656
SOCIAL                 4.247001
SHOPPING               4.245774
WEATHER                4.239675
SPORTS                 4.219279
PRODUCTIVITY           4.200279
FAMILY                 4.191406
AUTO_AND_VEHICLES      4.190824
MEDICAL                4.185279
PHOTOGRAPHY            4.183479
LIBRARIES_AND_DEMO     4.181962
HOUSE_AND_HOME         4.169001
FOOD_AND_DRINK         4.168388
COMMUNICATION          4.158216
COMICS                 4.156445
NEWS_AND_MAGAZINES     4.140784
ENTERTAINMENT          4.136036
BUSINESS               4.135958
FINANCE                4.135315
LIFESTYLE              4.113799
TRAVEL_AND_LOCAL       4.107539
VIDEO_PLAYERS          4.074858
TOOLS                  4.066280
MAPS_AND_NAVIGATION    4.065061
DATING                 4.013538
Name: Rating, dtype: float64
```

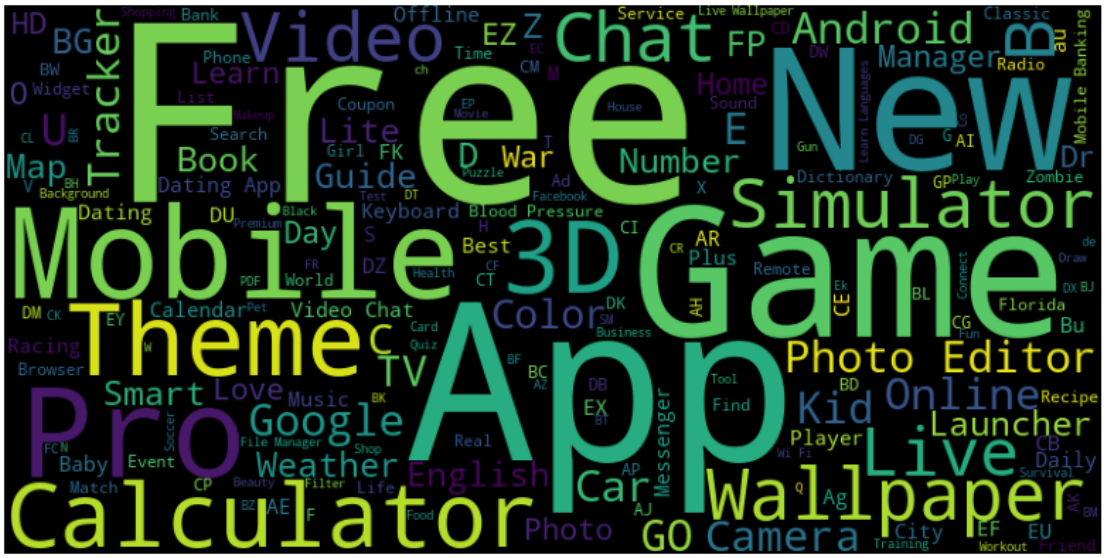
1.10 Word Cloud for Apps names

[16]:
```
# word cloud
!pip install wordcloud
```

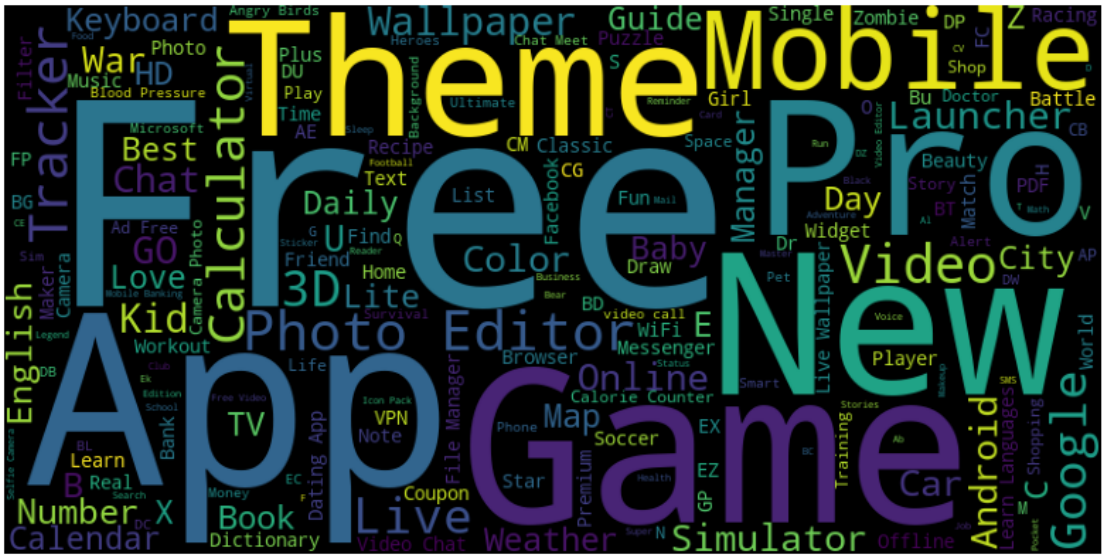
```
Collecting wordcloud
  Downloading wordcloud-1.9.2-cp39-cp39-win_amd64.whl (153 kB)
                                              0.0/153.3 kB ? eta -:--:--
     ---------                               41.0/153.3 kB 960.0 kB/s eta 0:00:01
     ----------------------------------      143.4/153.3 kB 1.7 MB/s eta 0:00:01
     ---------------------------------------- 153.3/153.3 kB 1.3 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.6.1 in d:\apps\anaconda\files\lib\site-
```

```
packages (from wordcloud) (1.23.5)
Requirement already satisfied: pillow in d:\apps\anaconda\files\lib\site-
packages (from wordcloud) (9.4.0)
Requirement already satisfied: matplotlib in d:\apps\anaconda\files\lib\site-
packages (from wordcloud) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (1.0.5)
Requirement already satisfied: cycler>=0.10 in d:\apps\anaconda\files\lib\site-
packages (from matplotlib->wordcloud) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (1.4.4)
Requirement already satisfied: packaging>=20.0 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (23.0)
Requirement already satisfied: pyparsing>=2.3.1 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (2.8.2)
Requirement already satisfied: importlib-resources>=3.2.0 in
d:\apps\anaconda\files\lib\site-packages (from matplotlib->wordcloud) (5.2.0)
Requirement already satisfied: zipp>=3.1.0 in d:\apps\anaconda\files\lib\site-
packages (from importlib-resources>=3.2.0->matplotlib->wordcloud) (3.11.0)
Requirement already satisfied: six>=1.5 in d:\apps\anaconda\files\lib\site-
packages (from python-dateutil>=2.7->matplotlib->wordcloud) (1.12.0)
Installing collected packages: wordcloud
Successfully installed wordcloud-1.9.2
```

[44]:
```
from wordcloud import WordCloud

all_app_names = ' '.join(df.App.astype(str))
all_app_names

wordCloud = WordCloud(width=800, height=400, background_color='black').
 ↪generate_from_text(all_app_names)

plt.figure(figsize=(10, 6))
plt.imshow(wordCloud)
plt.axis('off')
plt.show()
```



[54]:
```
apps_above_4 = ' '.join(df.loc[df.Rating>4]['App'].astype(str))
wordCloud = WordCloud(width=800, height=400, background_color='black').
 ↪generate_from_text(apps_above_4)

plt.figure(figsize=(10, 6))
plt.imshow(wordCloud)
plt.axis('off')
plt.show()
```



[ ]:

2 TASK 2 (Urdu Dataset Sentiment Analysis)

Implement following sequence based deep learning models for the same task of sentiment analysis. Perform binary text classification.

- **RNN**
- **GRU**
- **LSTM**
- **BiLSTM**

You can implement these models in Keras or Pytorch. Split the data into train and test set. Use 75% for training and 25% for testing.

For each of these models, try following hyper parameters and report the best results with parameter values.

- Number of layers = 2 or 3.
- Dropout rate, 0.3 or 0.7

So you will have 2 *2 = 4 different sets of parameters. Calculate accuracy, Precision, Recall and F-score for all classifiers and report the results in table.

2.1 Load Data

[3]:
```
urdu_dataframe = pd.read_table("urdu-sentiment-corpus-v1.tsv")
urdu_dataframe
```

[3]:

	Tweet	Class
0	…	P
1	…	N
2		O
3	130,000…	P
4	:	P
..	…	…
995		P
996	…	P
997	! …	P
998	: …	P
999	…	P

1000] rows x 2 columns]

2.2 Data Cleaning

[12]:
```
urdu_dataframe.isna().sum()
```

[12]:
```
Tweet    0
Class    1
dtype: int64
```

[22]:
```
urdu_dataframe.Class.fillna(urdu_dataframe.Class.mode()[0], inplace=True)
```

[24]:
```
urdu_dataframe.isna().sum()
```

[24]:
```
Tweet    0
Class    0
dtype: int64
```

[39]:
```
urdu_dataframe.drop(urdu_dataframe[urdu_dataframe.Class=='O'].index,␣
↪inplace=True)
```

[41]:
```
urdu_dataframe.describe()
```

[41]:
```
                                                Tweet Class
count                                             980   980
unique                                            979     2
top                                   …             N
freq                                                2   500
```

[43]:
```
urdu_dataframe.groupby("Class").describe().T
```

[43]:
```
Class                                                    N  \
Tweet count                                            500
      unique                                           500
      top                                    …
      freq                                               1

Class                                                    P
Tweet count                                            480
      unique                                           480
      top                                     …
      freq                                               1
```

2.3 Word Cloud for Tweets

[50]:
```
all_tweets = " ".join(urdu_dataframe.Tweet.astype(str))
tweets_cloud = WordCloud(width=520, height=260, background_color='black',␣
↪font_path="Noto_Naskh_Arabic/static/NotoNaskhArabic-Regular.ttf").
↪generate(all_tweets)

plt.figure(figsize=(8,6))
plt.imshow(tweets_cloud, interpolation = 'bilinear')
plt.axis("off")
plt.show()
```

[4]:
```
urdu_dataframe['text_length'] =  urdu_dataframe.Tweet.apply(len)

urdu_dataframe['tweet_type'] = urdu_dataframe['Class'].map({'P': 0, 'N': 1})
```

[5]:
```
urdu_dataframe
```

[5]:

	Tweet	Class	text_length	tweet_type
0	…	P	114	0.0
1	…	N	52	1.0
2		0	25	NaN
3	130,000…	P	114	0.0
4	:	P	48	0.0
..	…	…	…	…
995		P	47	0.0
996	…	P	86	0.0
997	! …	P	61	
998	: …	P	79	
999	…	P	73	

```
997         0.0
998         0.0
999         0.0

[1000 rows x 4 columns]
```

2.4 Seniment Analysis

[6]:
```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(urdu_dataframe.Tweet,␣
 ↪urdu_dataframe.tweet_type.values,
                                                    test_size=0.25,␣
 ↪random_state=0)
```

[15]:
```
max_len = 50
trunc_type = 'post'
padding_type = 'post'
oov_tok = '<oov>'
vocab_size = 500
embedding_dim = 16
```

[13]:
```
pip install tensorflow
```

```
Requirement already satisfied: tensorflow in d:\apps\anaconda\files\lib\site-
packages (2.11.0)
Collecting tensorflow-intel==2.11.0 (from tensorflow)
  Using cached tensorflow_intel-2.11.0-cp39-cp39-win_amd64.whl (266.3 MB)
Requirement already satisfied: absl-py>=1.0.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=2.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (23.5.26)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (0.4.0)
Requirement already satisfied: google-pasta>=0.1.1 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (3.7.0)
Requirement already satisfied: libclang>=13.0.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (15.0.6.1)
```

```
Requirement already satisfied: numpy>=1.20 in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (1.23.5)
Requirement already satisfied: opt-einsum>=2.3.2 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (3.3.0)
Requirement already satisfied: packaging in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (23.0)
Collecting protobuf<3.20,>=3.9.2 (from tensorflow-intel==2.11.0->tensorflow)
  Using cached protobuf-3.19.6-cp39-cp39-win_amd64.whl (895 kB)
Requirement already satisfied: setuptools in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (65.6.3)
Requirement already satisfied: six>=1.12.0 in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (1.15.0)
Requirement already satisfied: termcolor>=1.1.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (1.1.0)
Requirement already satisfied: typing-extensions>=3.6.6 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (3.7.4.3)
Requirement already satisfied: wrapt>=1.11.0 in d:\apps\anaconda\files\lib\site-
packages (from tensorflow-intel==2.11.0->tensorflow) (1.12.1)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (1.56.0)
Collecting tensorboard<2.12,>=2.11 (from tensorflow-intel==2.11.0->tensorflow)
  Using cached tensorboard-2.11.2-py3-none-any.whl (6.0 MB)
Requirement already satisfied: tensorflow-estimator<2.12,>=2.11.0 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (2.11.0)
Collecting keras<2.12,>=2.11.0 (from tensorflow-intel==2.11.0->tensorflow)
  Using cached keras-2.11.0-py2.py3-none-any.whl (1.7 MB)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in
d:\apps\anaconda\files\lib\site-packages (from tensorflow-
intel==2.11.0->tensorflow) (0.31.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in
d:\apps\anaconda\files\lib\site-packages (from astunparse>=1.6.0->tensorflow-
intel==2.11.0->tensorflow) (0.38.4)
Requirement already satisfied: google-auth<3,>=1.6.3 in
d:\apps\anaconda\files\lib\site-packages (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (2.22.0)
Collecting google-auth-oauthlib<0.5,>=0.4.1 (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow)
  Using cached google_auth_oauthlib-0.4.6-py2.py3-none-any.whl (18 kB)
Requirement already satisfied: markdown>=2.6.8 in
d:\apps\anaconda\files\lib\site-packages (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (3.4.1)
Requirement already satisfied: requests<3,>=2.21.0 in
d:\apps\anaconda\files\lib\site-packages (from
```

```
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (2.22.0)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in
d:\apps\anaconda\files\lib\site-packages (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (0.6.1)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in
d:\apps\anaconda\files\lib\site-packages (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (1.8.1)
Requirement already satisfied: werkzeug>=1.0.1 in
d:\apps\anaconda\files\lib\site-packages (from
tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow) (2.0.3)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in
d:\apps\anaconda\files\lib\site-packages (from google-
auth<3,>=1.6.3->tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow)
(5.3.1)
Requirement already satisfied: pyasn1-modules>=0.2.1 in
d:\apps\anaconda\files\lib\site-packages (from google-
auth<3,>=1.6.3->tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow)
(0.2.8)
Requirement already satisfied: rsa<5,>=3.1.4 in d:\apps\anaconda\files\lib\site-
packages (from google-auth<3,>=1.6.3->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (4.9)
Requirement already satisfied: urllib3<2.0 in d:\apps\anaconda\files\lib\site-
packages (from google-auth<3,>=1.6.3->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (1.25.11)
Requirement already satisfied: requests-oauthlib>=0.7.0 in
d:\apps\anaconda\files\lib\site-packages (from google-auth-
oauthlib<0.5,>=0.4.1->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (1.3.1)
Requirement already satisfied: importlib-metadata>=4.4 in
d:\apps\anaconda\files\lib\site-packages (from
markdown>=2.6.8->tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow)
(6.0.0)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in
d:\apps\anaconda\files\lib\site-packages (from
requests<3,>=2.21.0->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (3.0.4)
Requirement already satisfied: idna<2.9,>=2.5 in
d:\apps\anaconda\files\lib\site-packages (from
requests<3,>=2.21.0->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (2.8)
Requirement already satisfied: certifi>=2017.4.17 in
d:\apps\anaconda\files\lib\site-packages (from
requests<3,>=2.21.0->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (2022.12.7)
Requirement already satisfied: zipp>=0.5 in d:\apps\anaconda\files\lib\site-
packages (from importlib-
metadata>=4.4->markdown>=2.6.8->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (3.11.0)
```

```
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in
d:\apps\anaconda\files\lib\site-packages (from pyasn1-modules>=0.2.1->google-
auth<3,>=1.6.3->tensorboard<2.12,>=2.11->tensorflow-intel==2.11.0->tensorflow)
(0.4.8)
Requirement already satisfied: oauthlib>=3.0.0 in
d:\apps\anaconda\files\lib\site-packages (from requests-oauthlib>=0.7.0->google-
auth-oauthlib<0.5,>=0.4.1->tensorboard<2.12,>=2.11->tensorflow-
intel==2.11.0->tensorflow) (3.2.2)
Installing collected packages: protobuf, keras, google-auth-oauthlib,
tensorboard, tensorflow-intel
  Attempting uninstall: protobuf
    Found existing installation: protobuf 4.21.0
    Uninstalling protobuf-4.21.0:
      Successfully uninstalled protobuf-4.21.0
Note: you may need to restart the kernel to use updated packages.

WARNING: Ignoring invalid distribution - (d:\apps\anaconda\files\lib\site-
packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution - (d:\apps\anaconda\files\lib\site-
packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
ERROR: Could not install packages due to an OSError: [WinError 5] Access is
denied: 'D:\\apps\\anaconda\\files\\Lib\\site-
packages\\google\\~~pb\\_message.cp39-win_amd64.pyd'
Consider using the `--user` option or check the permissions.
```

[28]:
```
pip install keras
```

```
Requirement already satisfied: keras in d:\apps\anaconda\files\lib\site-packages
(2.3.1)
Requirement already satisfied: numpy>=1.9.1 in d:\apps\anaconda\files\lib\site-
packages (from keras) (1.23.5)
Requirement already satisfied: scipy>=0.14 in d:\apps\anaconda\files\lib\site-
packages (from keras) (1.10.1)
Requirement already satisfied: six>=1.9.0 in d:\apps\anaconda\files\lib\site-
packages (from keras) (1.15.0)
Requirement already satisfied: pyyaml in d:\apps\anaconda\files\lib\site-
packages (from keras) (6.0)
Requirement already satisfied: h5py in d:\apps\anaconda\files\lib\site-packages
(from keras) (3.7.0)
Requirement already satisfied: keras-applications>=1.0.6 in
```

```
d:\apps\anaconda\files\lib\site-packages (from keras) (1.0.8)
Requirement already satisfied: keras-preprocessing>=1.0.5 in
d:\apps\anaconda\files\lib\site-packages (from keras) (1.1.2)
Note: you may need to restart the kernel to use updated packages.

WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
```

[7]:
```
pip install --upgrade protobuf
```

```
Requirement already satisfied: protobuf in d:\apps\anaconda\files\lib\site-
packages (3.19.6)
Collecting protobuf
  Using cached protobuf-4.23.4-cp39-cp39-win_amd64.whl (422 kB)
Installing collected packages: protobuf
  Attempting uninstall: protobuf
    Found existing installation: protobuf 3.19.6
    Uninstalling protobuf-3.19.6:
      Successfully uninstalled protobuf-3.19.6
Successfully installed protobuf-4.23.4
Note: you may need to restart the kernel to use updated packages.

WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
ERROR: pip's dependency resolver does not currently take into account all the
packages that are installed. This behaviour is the source of the following
dependency conflicts.
apache-beam 2.46.0 requires crcmod<2.0,>=1.7, which is not installed.
apache-beam 2.46.0 requires fastavro<2,>=0.23.6, which is not installed.
apache-beam 2.46.0 requires fasteners<1.0,>=0.3, which is not installed.
apache-beam 2.46.0 requires hdfs<3.0.0,>=2.1.0, which is not installed.
apache-beam 2.46.0 requires httplib2<0.22.0,>=0.8, which is not installed.
apache-beam 2.46.0 requires objsize<0.7.0,>=0.6.1, which is not installed.
apache-beam 2.46.0 requires orjson<4.0, which is not installed.
apache-beam 2.46.0 requires proto-plus<2,>=1.7.1, which is not installed.
apache-beam 2.46.0 requires pyarrow<10.0.0,>=3.0.0, which is not installed.
apache-beam 2.46.0 requires pydot<2,>=1.2.0, which is not installed.
```

```
apache-beam 2.46.0 requires pymongo<4.0.0,>=3.8.0, which is not installed.
tensorboard 2.13.0 requires tensorboard-data-server<0.8.0,>=0.7.0, but you have
tensorboard-data-server 0.6.1 which is incompatible.
tensorflow-intel 2.13.0 requires keras<2.14,>=2.13.1, but you have keras 2.3.1
which is incompatible.
tensorflow-intel 2.13.0 requires tensorflow-estimator<2.14,>=2.13.0, but you
have tensorflow-estimator 2.11.0 which is incompatible.
apache-beam 2.46.0 requires cloudpickle~=2.2.1, but you have cloudpickle 2.0.0
which is incompatible.
apache-beam 2.46.0 requires dill<0.3.2,>=0.3.1.1, but you have dill 0.3.6 which
is incompatible.
apache-beam 2.46.0 requires protobuf<4,>3.12.2, but you have protobuf 4.23.4
which is incompatible.
apache-beam 2.46.0 requires requests<3.0.0,>=2.24.0, but you have requests
2.22.0 which is incompatible.
```

[12]:
```
pip install protobuf==4.21
```

```
Collecting protobuf==4.21
  Downloading protobuf-4.21.0-cp39-cp39-win_amd64.whl (524 kB)
                                              0.0/524.8 kB ? eta -:--:--
     -----                                   71.7/524.8 kB 2.0 MB/s eta 0:00:01
     -----------------                      235.5/524.8 kB 2.9 MB/s eta 0:00:01
     ----------------------------------     501.8/524.8 kB 2.9 MB/s eta 0:00:01
     ---------------------------------------- 524.8/524.8 kB 2.4 MB/s eta 0:00:00
Installing collected packages: protobuf
  Attempting uninstall: protobuf
    Found existing installation: protobuf 4.23.4
    Uninstalling protobuf-4.23.4:
      Successfully uninstalled protobuf-4.23.4
Note: you may need to restart the kernel to use updated packages.
```

```
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
WARNING: The candidate selected for download or install is a yanked version:
'protobuf' candidate (version 4.21.0 at https://files.pythonhosted.org/packages/
c8/5a/5f0c1b55f4f248b0e7a2db67cdd12d9be9bdfa552161884f58ffbdea6f6d/protobuf-4.21
.0-cp39-cp39-win_amd64.whl (from https://pypi.org/simple/protobuf/))
Reason for being yanked: Required python version not configured correctly
(https://github.com/protocolbuffers/protobuf/issues/10076)
WARNING: Ignoring invalid distribution -ensorflow-intel
(d:\apps\anaconda\files\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf
(d:\apps\anaconda\files\lib\site-packages)
ERROR: Could not install packages due to an OSError: [WinError 5] Access is
denied: 'D:\\apps\\anaconda\\files\\Lib\\site-
packages\\google\\~-pb\\_message.cp39-win_amd64.pyd'
```

```
Consider using the `--user` option or check the permissions.
```

[9]:
```
import tensorflow as tf
from keras_preprocessing.text import Tokenizer

tokenizer = Tokenizer(num_words = vocab_size, char_level = False, oov_token =␣
 ↪oov_tok)
tokenizer.fit_on_texts(X_train)
```

[10]:
```
# get the word index
word_index = tokenizer.word_index
word_index
total_words = len(word_index)
total_words
```

[10]:
```
4588
```

[11]:
```
from keras_preprocessing.sequence import pad_sequences

training_sequences = tokenizer.texts_to_sequences(X_train)
training_padding = pad_sequences(training_sequences, maxlen=max_len,␣
 ↪padding=padding_type, truncating=trunc_type)

training_padding
```

[11]:
```
array([[ 94, 276,   2, …,   0,   0,   0],
       [ 24,   3,   1, …,   0,   0,   0],
       [  1,   1,   1, …,   0,   0,   0],
       …,
       [  1,   1, 192, …,   0,   0,   0],
       [  1, 125,  19, …,   0,   0,   0],
       [  1, 196, 453, …,   0,   0,   0]])
```

[12]:
```
testing_sequences = tokenizer.texts_to_sequences(X_test)
testing_padding = pad_sequences(testing_sequences, maxlen=max_len,␣
 ↪padding=padding_type, truncating=trunc_type)

testing_padding
# testing_sequences
```

[12]:
```
array([[ 59,   5,  37, …,   0,   0,   0],
       [  1,  60,   1, …,   0,   0,   0],
       [444, 148,   1, …,   0,   0,   0],
       …,
       [  1,   7,   1, …,   0,   0,   0],
       [  1,   1,   1, …,   0,   0,   0],
       [ 93, 122,   9, …,   0,   0,   0]])
```

[13]:
```
print("Training tensor shape: {}".format(training_padding.shape))
print("Testing tensor shape: {}".format(testing_padding.shape))
```

```
Training tensor shape: (750, 50)
Testing tensor shape: (250, 50)
```

[51]:
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import SimpleRNN, LSTM, GRU, Dense, Embedding,␣
↪Dropout,GlobalAveragePooling1D, Flatten, SpatialDropout1D, Bidirectional
```

2.5 1. LSTM (Long Short Term Memory)

[67]:
```
# LSTM
n_lstm = 128
dropout_lstm = 0.2

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim, input_length=max_len))
model.add(SpatialDropout1D(dropout_lstm))
model.add(LSTM(n_lstm, return_sequences=False))
model.add(Dropout(dropout_lstm))
model.add(Dense(1, activation="sigmoid"))
```

[68]:
```
model.summary()
```

```
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_10 (Embedding)    (None, 50, 16)            8000

 spatial_dropout1d_4 (Spatia  (None, 50, 16)           0
 lDropout1D)

 lstm_4 (LSTM)               (None, 128)               74240

 dropout_9 (Dropout)         (None, 128)               0

 dense_6 (Dense)             (None, 1)                 129

=================================================================
Total params: 82,369
Trainable params: 82,369
Non-trainable params: 0
_________________________________________________________________
```

[69]:
```
model.compile(loss='binary_crossentropy', optimizer='adam',␣
↪metrics=['accuracy'])
```

[70]:
```
from tensorflow.keras.callbacks import EarlyStopping

num_epochs = 30
early_stop = EarlyStopping(monitor='val_loss', patience=2)
history = model.fit(training_padding, y_train, epochs=num_epochs,
                    validation_data=(testing_padding, y_test),
                    callbacks=[early_stop],
                    verbose=2)
```

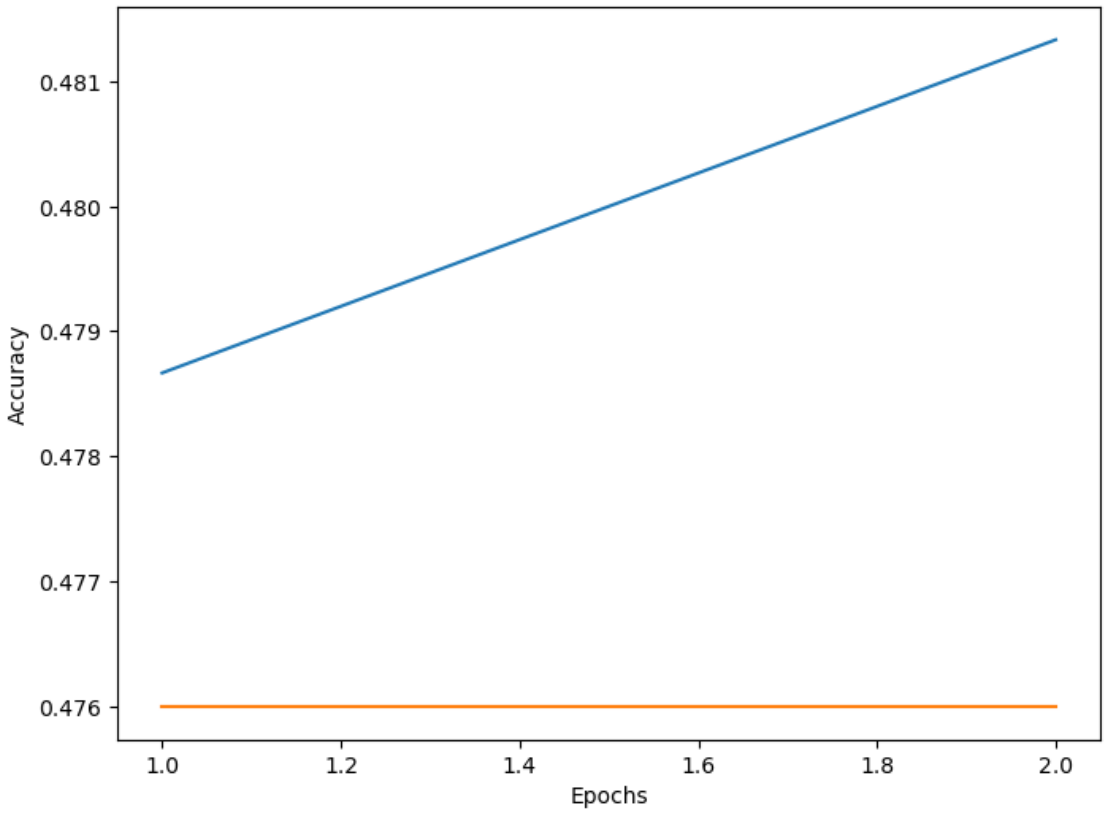
```
Epoch 1/30
24/24 - 7s - loss: nan - accuracy: 0.4787 - val_loss: nan - val_accuracy: 0.4760
- 7s/epoch - 311ms/step
Epoch 2/30
24/24 - 2s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 2s/epoch - 64ms/step
```

[71]:
```
model.evaluate(testing_padding, y_test)
```

```
8/8 [==============================] - 0s 30ms/step - loss: nan - accuracy:
0.4760
```

[71]:
```
[nan, 0.47600001096725464]
```

[72]:
```
history_dict = history.history

accuracy = history_dict['accuracy']
val_accuracy = history_dict['val_accuracy']
loss = history_dict['loss']
val_loss = history_dict['val_loss']

plt.figure(figsize=(8,6))
plt.plot(range(1, len(accuracy) + 1), accuracy, label = 'Training Accuracy')
plt.plot(range(1, len(val_accuracy) + 1), val_accuracy, label = 'Validation␣
 ↪Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.show()
```



2.6 2. Bi-LSTM (Bi Long Short Term Memory)

[30]:
```
# Bi-LSTM

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim, input_length=max_len))
model.add(Bidirectional(LSTM(n_lstm, return_sequences=False)))
model.add(Dropout(dropout_lstm))
model.add(Dense(1, activation='sigmoid'))
```

[31]:
```
model.summary()
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_3 (Embedding)     (None, 50, 16)            8000

 bidirectional (Bidirectiona  (None, 256)              148480
 l)
```

```
 dropout_2 (Dropout)         (None, 256)               0

 dense_1 (Dense)             (None, 1)                 257

=================================================================
Total params: 156,737
Trainable params: 156,737
Non-trainable params: 0
_________________________________________________________________
```

[32]:
```
model.compile(loss='binary_crossentropy', optimizer='adam',␣
↪metrics=['accuracy'])
```

[33]:
```
num_epochs = 30
early_stop = EarlyStopping(monitor='val_loss', patience=2)
history = model.fit(training_padding,
                    y_train,
                    epochs=num_epochs,
                    validation_data=(testing_padding, y_test),
                    callbacks=[early_stop],
                    verbose = 2)
```

```
Epoch 1/30
24/24 - 13s - loss: nan - accuracy: 0.4747 - val_loss: nan - val_accuracy:
0.4760 - 13s/epoch - 560ms/step
Epoch 2/30
24/24 - 3s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 3s/epoch - 115ms/step
```

[37]:
```
model.evaluate(testing_padding, y_test)
```

```
8/8 [==============================] - 0s 41ms/step - loss: nan - accuracy:
0.4760
```

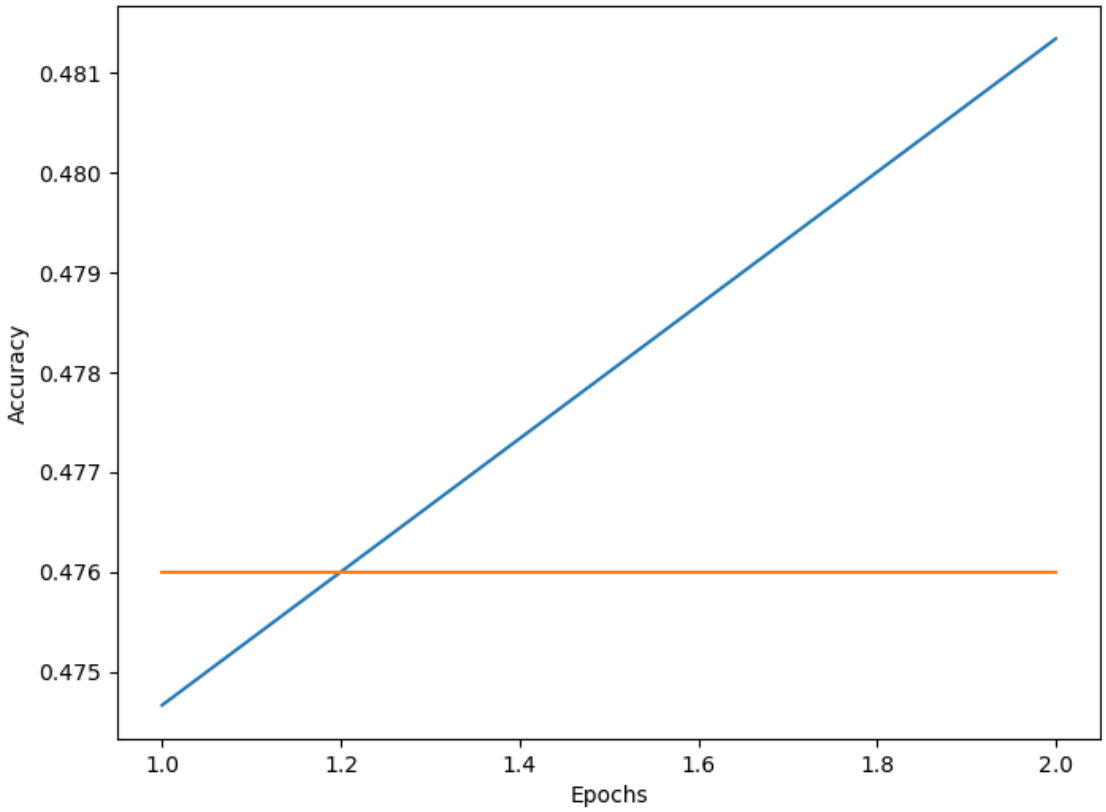
[37]:
```
[nan, 0.47600001096725464]
```

[38]:
```
history.history
```

[38]:
```
{'loss': [nan, nan],
 'accuracy': [0.47466665506362915, 0.48133334517478943],
 'val_loss': [nan, nan],
 'val_accuracy': [0.47600001096725464, 0.47600001096725464]}
```

[36]:
```
history_dict = history.history

accuracy = history_dict['accuracy']
val_accuracy = history_dict['val_accuracy']
loss = history_dict['loss']
val_loss = history_dict['val_loss']
```

```
plt.figure(figsize=(8,6))
plt.plot(range(1, len(accuracy) + 1), accuracy, label = 'Training Accuracy')
plt.plot(range(1, len(val_accuracy) + 1), val_accuracy, label = 'Validation␣
↪Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.show()
```



2.7 3. GRU

[39]:
```
# GRU
model = Sequential()
model.add(Embedding(vocab_size, embedding_dim, input_length=max_len))
model.add(SpatialDropout1D(0.2))
model.add(GRU(128, return_sequences=False))
model.add(Dropout(0.2))
model.add(Dense(1, activation='sigmoid'))
```

[40]:
```
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_4 (Embedding)     (None, 50, 16)            8000

 spatial_dropout1d_2 (Spatia  (None, 50, 16)           0
 lDropout1D)

 gru (GRU)                   (None, 128)               56064

 dropout_3 (Dropout)         (None, 128)               0

 dense_2 (Dense)             (None, 1)                 129

=================================================================
Total params: 64,193
Trainable params: 64,193
Non-trainable params: 0
_________________________________________________________________
```

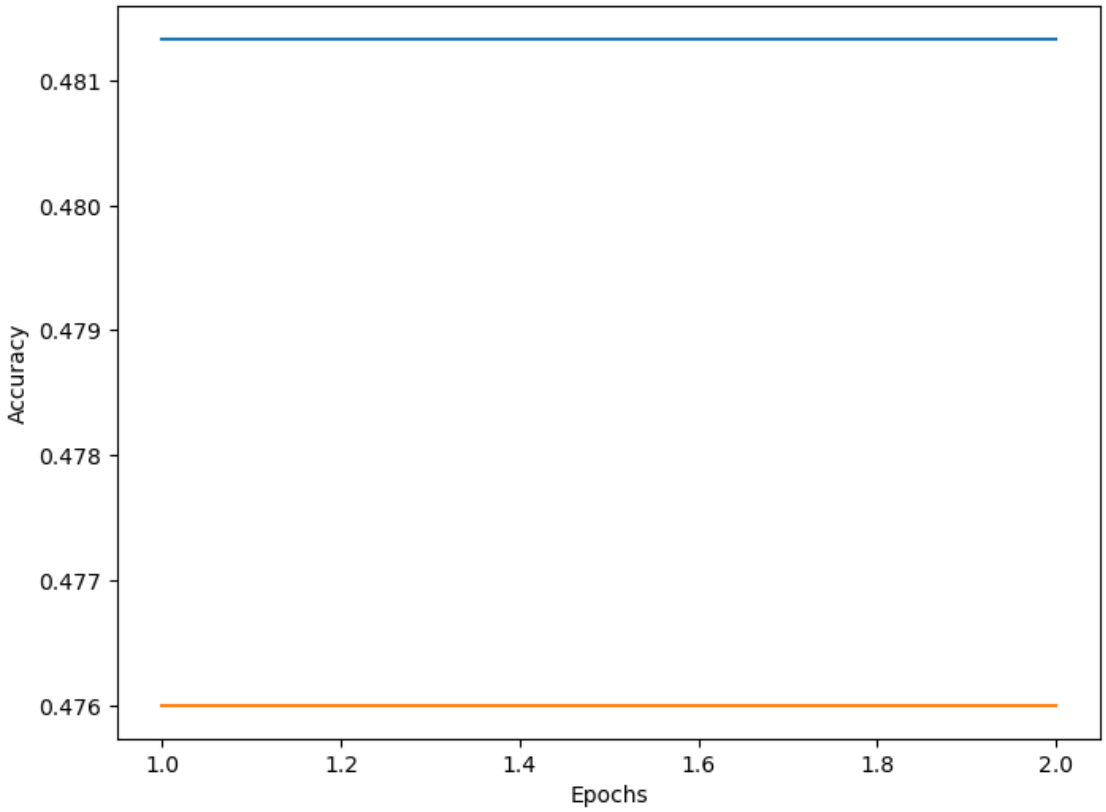
[41]:
```
model.compile(loss = 'binary_crossentropy',
                      optimizer = 'adam',
                      metrics=['accuracy'])
```

[45]:
```
num_epochs = 30
early_stop = EarlyStopping(monitor='val_loss', patience=2)
history = model.fit(training_padding,
                    y_train,
                    epochs=num_epochs,
                    validation_data=(testing_padding, y_test),
                    callbacks =[early_stop],
                    verbose=2)
```

```
Epoch 1/30
24/24 - 6s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 6s/epoch - 239ms/step
Epoch 2/30
24/24 - 2s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 2s/epoch - 74ms/step
```

[46]:
```
history_dict = history.history

accuracy = history_dict['accuracy']
val_accuracy = history_dict['val_accuracy']
loss = history_dict['loss']
val_loss = history_dict['val_loss']
```

```
plt.figure(figsize=(8,6))
plt.plot(range(1, len(accuracy) + 1), accuracy, label = 'Training Accuracy')
plt.plot(range(1, len(val_accuracy) + 1), val_accuracy, label = 'Validation␣
↪Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.show()
```



2.8 4. RNN

[74]:
```
# RNN
model = Sequential()
model.add(Embedding(vocab_size, embedding_dim, input_length=max_len))
model.add(Dropout(dropout_lstm))
model.add(SimpleRNN(n_lstm))
model.add(Dense(1, activation='sigmoid'))
```

[75]:
```
model.summary()
```

```
Model: "sequential_11"
_________________________________________________________________
```

```
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_11 (Embedding)    (None, 50, 16)            8000

 dropout_10 (Dropout)        (None, 50, 16)            0

 simple_rnn_4 (SimpleRNN)    (None, 128)               18560

 dense_7 (Dense)             (None, 1)                 129

=================================================================
Total params: 26,689
Trainable params: 26,689
Non-trainable params: 0
_________________________________________________________________
```

[76]:
```
model.compile(loss='binary_crossentropy', optimizer='adam',␣
 ↪metrics=['accuracy'])
```

[77]:
```
history = model.fit(training_padding, y_train,
                    epochs=num_epochs,
                    validation_data=(testing_padding, y_test),
                    callbacks=[early_stop], verbose=2)
```

```
Epoch 1/30
24/24 - 3s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 3s/epoch - 118ms/step
Epoch 2/30
24/24 - 1s - loss: nan - accuracy: 0.4813 - val_loss: nan - val_accuracy: 0.4760
- 509ms/epoch - 21ms/step
```

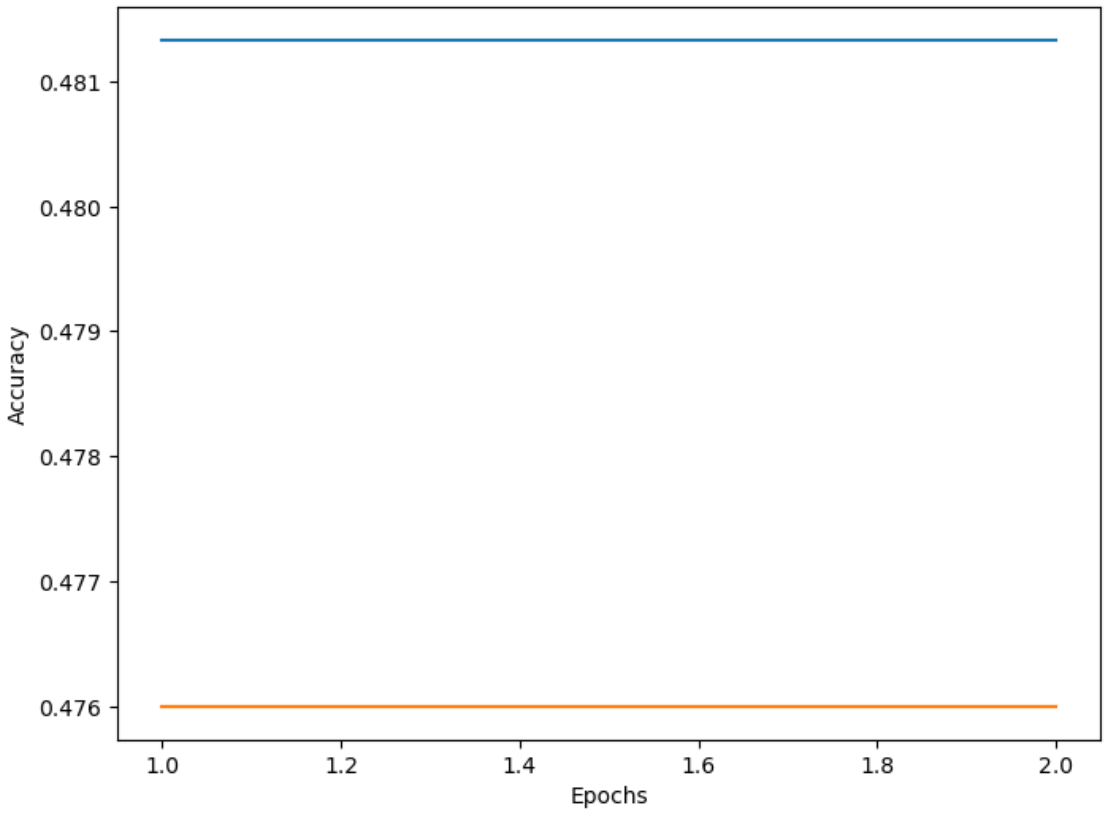
[78]:
```
model.evaluate(testing_padding, y_test)
```

```
8/8 [==============================] - 0s 9ms/step - loss: nan - accuracy:
0.4760
```

[78]:
```
[nan, 0.47600001096725464]
```

[79]:
```
history_dict = history.history

accuracy = history_dict['accuracy']
val_accuracy = history_dict['val_accuracy']
loss = history_dict['loss']
val_loss = history_dict['val_loss']

plt.figure(figsize=(8,6))
plt.plot(range(1, len(accuracy) + 1), accuracy, label = 'Training Accuracy')
plt.plot(range(1, len(val_accuracy) + 1), val_accuracy, label = 'Validation␣
 ↪Accuracy')
```

```
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.show()
```



[86]:
```
# testing/predicting
tweets=["                ", "                          "]

def predict_tweet(tweets):
    new_seq = tokenizer.texts_to_sequences(tweets)
    padded = pad_sequences(new_seq, maxlen=max_len, padding=padding_type,␣
 ↪truncating=trunc_type)
    return (model.predict(padded))

print(predict_tweet(tweets))
```

```
1/1 [==============================] - 0s 38ms/step
[[nan]
 [nan]]
```

[ ]:
```

```

[ ]:

[ ]: