PRODIGY INFOTECH

TASK 4

Analyze and visualize sentiment patterns in social media data to understand public opinion and attitudes towards specific topics or brands.

CHENNAM RUSHWANTH | Data Science Intern__

In [2]:
```
import numpy as np
import pandas as pd
```

In [4]:
```
data = pd.read_csv("sentimentdataset.csv")
data
```

Out[4]:

	Unnamed: 0.1	Unnamed: 0	Text	Sentiment	Timestamp	User
0	0	0	Enjoying a beautiful day at the park! ...	Positive	2023-01-15 12:30:00	User123
1	1	1	Traffic was terrible this morning. ...	Negative	2023-01-15 08:45:00	CommuterX
2	2	2	Just finished an amazing workout! 💪 ...	Positive	2023-01-15 15:45:00	FitnessFan
3	3	3	Excited about the upcoming weekend getaway! ...	Positive	2023-01-15 18:20:00	AdventureX
4	4	4	Trying out a new recipe for dinner tonight. ...	Neutral	2023-01-15 19:55:00	ChefCook
...	...	...	...	...	...	...
727	728	732	Collaborating on a science project that receiv...	Happy	2017-08-18 18:20:00	ScienceProjectSuccessHighSchool
728	729	733	Attending a surprise birthday party organized ...	Happy	2018-06-22 14:15:00	BirthdayPartyJoyHighSchool
729	730	734	Successfully fundraising for a school charity ...	Happy	2019-04-05 17:30:00	CharityFundraisingTriumphHighSchool
730	731	735	Participating in a multicultural festival, cel...	Happy	2020-02-29 20:45:00	MulticulturalFestivalJoyHighSchool
731	732	736	Organizing a virtual talent show during challe...	Happy	2020-11-15 15:15:00	VirtualTalentShowSuccessHighSchool

732 rows × 15 columns

In [6]:
```
data.head()
```

Out[6]:

	Unnamed: 0.1	Unnamed: 0	Text	Sentiment	Timestamp	User	Platform	Hashtags	Retw
0	0	0	Enjoying a beautiful day at the park! ...	Positive	2023-01-15 12:30:00	User123	Twitter	#Nature #Park	
1	1	1	Traffic was terrible this morning. ...	Negative	2023-01-15 08:45:00	CommuterX	Twitter	#Traffic #Morning	
2	2	2	Just finished an amazing workout! 💪 ...	Positive	2023-01-15 15:45:00	FitnessFan	Instagram	#Fitness #Workout	
3	3	3	Excited about the upcoming weekend getaway! ...	Positive	2023-01-15 18:20:00	AdventureX	Facebook	#Travel #Adventure	
4	4	4	Trying out a new recipe for dinner tonight. ...	Neutral	2023-01-15 19:55:00	ChefCook	Instagram	#Cooking #Food	

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

Out[8]:

	Unnamed: 0.1	Unnamed: 0	Text	Sentiment	Timestamp	User
727	728	732	Collaborating on a science project that receiv...	Happy	2017-08-18 18:20:00	ScienceProjectSuccessHighSchool
728	729	733	Attending a surprise birthday party organized ...	Happy	2018-06-22 14:15:00	BirthdayPartyJoyHighSchool
729	730	734	Successfully fundraising for a school charity ...	Happy	2019-04-05 17:30:00	CharityFundraisingTriumphHighSchool
730	731	735	Participating in a multicultural festival, cel...	Happy	2020-02-29 20:45:00	MulticulturalFestivalJoyHighSchool
731	732	736	Organizing a virtual talent show during challe...	Happy	2020-11-15 15:15:00	VirtualTalentShowSuccessHighSchool

In [10]:
```
data.dtypes
```

Out[10]:
```
Unnamed: 0.1      int64
Unnamed: 0        int64
Text             object
Sentiment        object
Timestamp        object
User             object
Platform         object
Hashtags         object
Retweets        float64
Likes           float64
Country          object
Year              int64
Month             int64
Day               int64
Hour              int64
dtype: object
```

In [12]:
```
data.describe()
```

Out[12]:

	Unnamed: 0.1	Unnamed: 0	Retweets	Likes	Year	Month	Day	Hou
count	732.000000	732.000000	732.000000	732.000000	732.000000	732.000000	732.000000	732.00000
mean	366.464481	369.740437	21.508197	42.901639	2020.471311	6.122951	15.497268	15.52185
std	211.513936	212.428936	7.061286	14.089848	2.802285	3.411763	8.474553	4.11341
min	0.000000	0.000000	5.000000	10.000000	2010.000000	1.000000	1.000000	0.00000
25%	183.750000	185.750000	17.750000	34.750000	2019.000000	3.000000	9.000000	13.00000
50%	366.500000	370.500000	22.000000	43.000000	2021.000000	6.000000	15.000000	16.00000
75%	549.250000	553.250000	25.000000	50.000000	2023.000000	9.000000	22.000000	19.00000
max	732.000000	736.000000	40.000000	80.000000	2023.000000	12.000000	31.000000	23.00000

In [14]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 732 entries, 0 to 731
Data columns (total 15 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Unnamed: 0.1  732 non-null    int64  
 1   Unnamed: 0    732 non-null    int64  
 2   Text          732 non-null    object 
 3   Sentiment     732 non-null    object 
 4   Timestamp     732 non-null    object 
 5   User          732 non-null    object 
 6   Platform      732 non-null    object 
 7   Hashtags      732 non-null    object 
 8   Retweets      732 non-null    float64
 9   Likes         732 non-null    float64
 10  Country       732 non-null    object 
 11  Year          732 non-null    int64  
 12  Month         732 non-null    int64  
 13  Day           732 non-null    int64  
 14  Hour          732 non-null    int64  
dtypes: float64(2), int64(6), object(7)
memory usage: 85.9+ KB
```

In [16]:
```
data.info
```

Out[16]:
```
<bound method DataFrame.info of      Unnamed: 0.1  Unnamed: 0  \
0               0           0
1               1           1
2               2           2
3               3           3
4               4           4
..            ...         ...
727           728         732
728           729         733
729           730         734
730           731         735
731           732         736

                                                  Text    Sentiment  \
0     Enjoying a beautiful day at the park!        ...   Positive
1     Traffic was terrible this morning.           ...   Negative
2     Just finished an amazing workout! 💪           ...   Positive
3     Excited about the upcoming weekend getaway!  ...   Positive
4     Trying out a new recipe for dinner tonight.  ...   Neutral
..                                                 ...         ...
727  Collaborating on a science project that receiv...       Happy
728  Attending a surprise birthday party organized ...       Happy
729  Successfully fundraising for a school charity ...       Happy
730  Participating in a multicultural festival, cel...       Happy
731  Organizing a virtual talent show during challe...       Happy

              Timestamp                                User     Platform  \
0    2023-01-15 12:30:00                           User123      Twitter
1    2023-01-15 08:45:00                           CommuterX    Twitter
2    2023-01-15 15:45:00                           FitnessFan   Instagram
3    2023-01-15 18:20:00                           AdventureX   Facebook
4    2023-01-15 19:55:00                           ChefCook     Instagram
..                   ...                                ...         ...
727  2017-08-18 18:20:00       ScienceProjectSuccessHighSchool   Facebook
728  2018-06-22 14:15:00            BirthdayPartyJoyHighSchool   Instagram
729  2019-04-05 17:30:00   CharityFundraisingTriumphHighSchool    Twitter
730  2020-02-29 20:45:00    MulticulturalFestivalJoyHighSchool   Facebook
731  2020-11-15 15:15:00    VirtualTalentShowSuccessHighSchool   Instagram

                                       Hashtags  Retweets  Likes  \
0        #Nature #Park                              15.0   30.0
1        #Traffic #Morning                           5.0   10.0
2        #Fitness #Workout                          20.0   40.0
3        #Travel #Adventure                          8.0   15.0
4        #Cooking #Food                             12.0   25.0
..                                          ...       ...    ...
727         #ScienceFairWinner #HighSchoolScience   20.0   39.0
728    #SurpriseCelebration #HighSchoolFriendship   25.0   48.0
729      #CommunityGiving #HighSchoolPhilanthropy   22.0   42.0
730         #CulturalCelebration #HighSchoolUnity   21.0   43.0
731   #VirtualEntertainment #HighSchoolPositivity   24.0   47.0

          Country  Year  Month  Day  Hour
0       USA        2023      1   15    12
1       Canada     2023      1   15     8
2     USA          2023      1   15    15
3       UK         2023      1   15    18
4      Australia   2023      1   15    19
..             ...   ...    ...  ...   ...
727            UK  2017      8   18    18
```

```
728           USA  2018      6   22    14
729        Canada  2019      4    5    17
730            UK  2020      2   29    20
731           USA  2020     11   15    15

[732 rows x 15 columns]>
```

In [18]:
```
data.isnull().sum()
```

Out[18]:
```
Unnamed: 0.1    0
Unnamed: 0      0
Text            0
Sentiment       0
Timestamp       0
User            0
Platform        0
Hashtags        0
Retweets        0
Likes           0
Country         0
Year            0
Month           0
Day             0
Hour            0
dtype: int64
```

In [20]:
```
data.duplicated().sum()
```

Out[20]:
```
0
```

In [22]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [24]:
```
temp = data.groupby('Sentiment').count()['Text'].reset_index().sort_values(by='Text',a
temp.style.background_gradient(cmap='Purples')
```

Out[24]:

	Sentiment	Text
214	Positive	44
172	Joy	42
110	Excitement	32
139	Happy	14
56	Contentment	14
196	Neutral	14
128	Gratitude	9
239	Sad	9
152	Hopeful	9
64	Curiosity	8
91	Embarrassed	8
178	Loneliness	7
89	Elation	6
211	Playful	6
143	Hate	6
28	Bad	6
72	Despair	6
187	Melancholy	5
158	Indifference	5
163	Inspired	5
32	Bitterness	5
121	Frustrated	5
1	Acceptance	5
68	Curiosity	5
132	Gratitude	5
53	Confusion	5
201	Nostalgia	5
203	Numbness	5
102	Enthusiasm	5
12	Ambivalence	5
57	Contentment	5
95	Empowerment	5
245	Serenity	5
246	Serenity	5

	Sentiment	Text
78	Determination	5
48	Compassionate	4
111	Excitement	4
133	Grief	4
122	Frustration	4
127	Grateful	4
109	Euphoria	4
160	Inspiration	4
71	Desolation	4
24	Awe	4
151	Hope	4
198	Nostalgia	4
197	Neutral	4
22	Arousal	4
195	Negative	4
243	Serenity	4
29	Betrayal	4
261	Tenderness	4
220	Proud	4
79	Devastated	3
94	Empathetic	3
256	Surprise	3
85	Dismissive	3
231	Resentment	3
225	Regret	3
135	Grief	3
218	Pride	3
217	Pride	3
103	Envious	3
116	Fearful	3
118	Free-spirited	3
176	Kind	3
168	Jealous	3
208	Overwhelmed	3

	Sentiment	Text
0	Acceptance	3
31	Bitter	3
2	Accomplishment	3
50	Confident	3
47	Compassion	3
8	Adventure	3
36	Boredom	3
38	Calmness	3
137	Happiness	2
136	Happiness	2
40	Captivation	2
134	Grief	2
240	Sadness	2
278	Zest	2
241	Satisfaction	2
6	Adoration	2
129	Gratitude	2
4	Admiration	2
125	Fulfillment	2
124	Fulfillment	2
123	Frustration	2
248	Shame	2
115	Fear	2
235	Reverence	2
145	Heartbreak	2
234	Reverence	2
228	Rejuvenation	2
192	Mischievous	2
25	Awe	2
184	Love	2
181	Loss	2
26	Awe	2
16	Anticipation	2
173	Joy	2

	Sentiment	Text
15	Anger	2
170	Jealousy	2
167	Isolation	2
13	Amusement	2
223	Reflection	2
224	Reflection	2
9	Affection	2
227	Regret	2
114	Exploration	2
21	Apprehensive	2
58	Coziness	2
73	Despair	2
77	Determination	2
75	Despair	2
96	Enchantment	2
97	Enchantment	2
262	Thrill	2
100	Enjoyment	2
101	Enthusiasm	2
55	Contemplation	2
80	Disappointed	2
92	Emotion	2
81	Disappointment	2
267	Tranquility	2
105	Envy	2
257	Surprise	2
270	Whimsy	2
84	Disgust	2
83	Disgust	2
52	Confusion	2
277	Yearning	2
60	Creativity	2
14	Amusement	1
215	Positivity	1

	Sentiment	Text
213	Positive	1
216	Pressure	1
271	Whispers of the Past	1
272	Winter Magic	1
212	PlayfulJoy	1
273	Wonder	1
254	Success	1
210	Pensive	1
209	Overwhelmed	1
20	Appreciation	1
19	Anxiety	1
199	Nostalgia	1
200	Nostalgia	1
276	Wonderment	1
18	Anxiety	1
202	Numbness	1
17	Anticipation	1
204	Obstacle	1
275	Wonder	1
205	Ocean's Freedom	1
219	Pride	1
206	Optimism	1
207	Overjoyed	1
274	Wonder	1
11	Ambivalence	1
269	Vibrancy	1
10	Amazement	1
221	Radiance	1
255	Suffering	1
252	Sorrow	1
251	Sorrow	1
3	Admiration	1
250	Solitude	1
249	Solace	1

	Sentiment	Text
247	Shame	1
258	Surprise	1
259	Suspense	1
260	Sympathy	1
5	Admiration	1
244	Serenity	1
23	ArtisticBurst	1
263	Thrill	1
7	Adrenaline	1
238	Runway Creativity	1
237	Ruins	1
236	Romance	1
264	Thrill	1
265	Thrilling Journey	1
233	Resilience	1
232	Resilience	1
266	Touched	1
230	Renewed Effort	1
229	Relief	1
268	Triumph	1
253	Spark	1
226	Regret	1
222	Radiance	1
242	Satisfaction	1
69	Darkness	1
194	Nature's Beauty	1
46	Compassion	1
108	Euphoria	1
51	Confusion	1
112	Excitement	1
113	Exhaustion	1
49	Confidence	1
117	FestiveJoy	1
119	Freedom	1

	Sentiment	Text
120	Friendship	1
45	Colorful	1
193	Motivation	1
126	Grandeur	1
44	Charm	1
43	Challenge	1
130	Gratitude	1
131	Gratitude	1
42	Celestial Wonder	1
41	Celebration	1
138	Happiness	1
107	Euphoria	1
106	Euphoria	1
104	Envisioning History	1
54	Connection	1
67	Curiosity	1
66	Curiosity	1
74	Despair	1
76	Desperation	1
65	Curiosity	1
63	CulinaryOdyssey	1
82	Disgust	1
62	Culinary Adventure	1
86	DreamChaser	1
87	Ecstasy	1
88	Elation	1
61	Creativity	1
90	Elegance	1
93	EmotionalStorm	1
59	Creative Inspiration	1
98	Energy	1
99	Engagement	1
39	Calmness	1
140	Harmony	1

	Sentiment	Text
141	Harmony	1
165	Intimidation	1
169	Jealousy	1
171	Journey	1
30	Betrayal	1
70	Dazzle	1
175	JoyfulReunion	1
27	Awe	1
177	Kindness	1
179	Loneliness	1
180	Loneliness	1
182	LostLove	1
183	Love	1
185	Marvel	1
186	Melancholy	1
188	Melodic	1
189	Mesmerizing	1
190	Mindfulness	1
191	Miscalculation	1
166	Intrigue	1
164	Intimidation	1
142	Harmony	1
33	Bittersweet	1
144	Heartache	1
146	Heartbreak	1
147	Heartwarming	1
148	Helplessness	1
149	Helplessness	1
150	Hope	1
37	Breakthrough	1
153	Hypnotic	1
154	Iconic	1
155	Imagination	1
156	Immersion	1

	Sentiment	Text
157	Indifference	1
35	Boredom	1
159	InnerJourney	1
34	Blessed	1
161	Inspiration	1
162	Inspiration	1
174	Joy in Baking	1

In [30]:
```
#Common words used on Text
from collections import Counter
from sklearn.model_selection import train_test_split

data['temp_list'] = data['Text'].apply(lambda x:str(x).split())
top = Counter([item for sublist in data['temp_list'] for item in sublist])
temp = pd.DataFrame(top.most_common(20))
temp.columns = ['Common_words','count']
temp.style.background_gradient(cmap='pink')
```

Out[30]:

	Common_words	count
0	the	808
1	of	623
2	a	621
3	in	259
4	to	133
5	and	111
6	with	107
7	for	99
8	on	91
9	by	69
10	through	51
11	an	49
12	my	41
13	at	40
14	new	39
15	each	39
16	In	39
17	that	37
18	as	36
19	A	33

In [32]:

```
#Bar plot for Common words on Text
import plotly.express as px
fig = px.bar(temp, x="count", y="Common_words", title='Commmon Words in Selected Text'
             width=400, height=400,color='Common_words')
fig.show()
```

In [34]:
```
top = Counter([item for sublist in data['temp_list'] for item in sublist])
temp = pd.DataFrame(top.most_common(20))
temp = temp.iloc[1:,:]
temp.columns = ['Common_words','count']
temp.style.background_gradient(cmap='Purples')
```

Out[34]:

	Common_words	count
1	of	623
2	a	621
3	in	259
4	to	133
5	and	111
6	with	107
7	for	99
8	on	91
9	by	69
10	through	51
11	an	49
12	my	41
13	at	40
14	new	39
15	each	39
16	In	39
17	that	37
18	as	36
19	A	33

In [36]:

```
#TreeMap of Most Common Words
import plotly.express as px
fig = px.treemap(temp, path=['Common_words'], values='count',width=400, height=400,tit
fig.show()
```

In [38]:
```
!pip install textblob
```

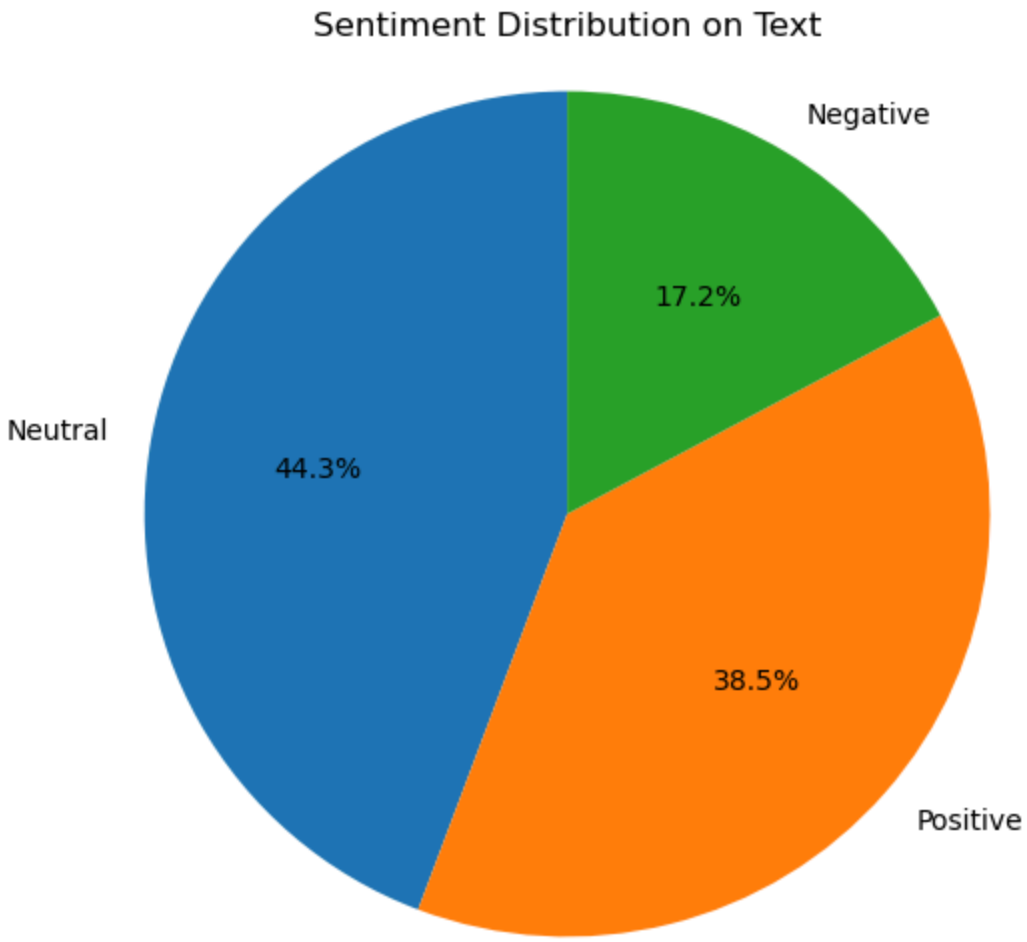
```
Requirement already satisfied: textblob in c:\users\dell\anaconda3\lib\site-packages
(0.18.0.post0)
Requirement already satisfied: nltk>=3.8 in c:\users\dell\anaconda3\lib\site-packages
(from textblob) (3.8.1)
Requirement already satisfied: click in c:\users\dell\anaconda3\lib\site-packages (fr
om nltk>=3.8->textblob) (8.0.4)
Requirement already satisfied: joblib in c:\users\dell\anaconda3\lib\site-packages (f
rom nltk>=3.8->textblob) (1.2.0)
Requirement already satisfied: regex>=2021.8.3 in c:\users\dell\anaconda3\lib\site-pa
ckages (from nltk>=3.8->textblob) (2022.7.9)
Requirement already satisfied: tqdm in c:\users\dell\anaconda3\lib\site-packages (fro
m nltk>=3.8->textblob) (4.65.0)
Requirement already satisfied: colorama in c:\users\dell\anaconda3\lib\site-packages
(from click->nltk>=3.8->textblob) (0.4.6)
```

In [40]:
```
# Perform sentiment analysis on text
import sys
from textblob import TextBlob
data['Sentiment'] = data['Text'].apply(lambda x: TextBlob(x).sentiment.polarity)

# Categorize sentiment into positive, negative, and neutral
data['Sentiment Category'] = data['Sentiment'].apply(lambda x: 'Positive' if x > 0 els

# Calculate the count of each sentiment category
sentiment_counts = data['Sentiment Category'].value_counts()

# Plot a pie chart of sentiment distribution
plt.figure(figsize=(6, 6))
plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', startangle
plt.axis('equal')
plt.title('Sentiment Distribution on Text')
plt.show()
```

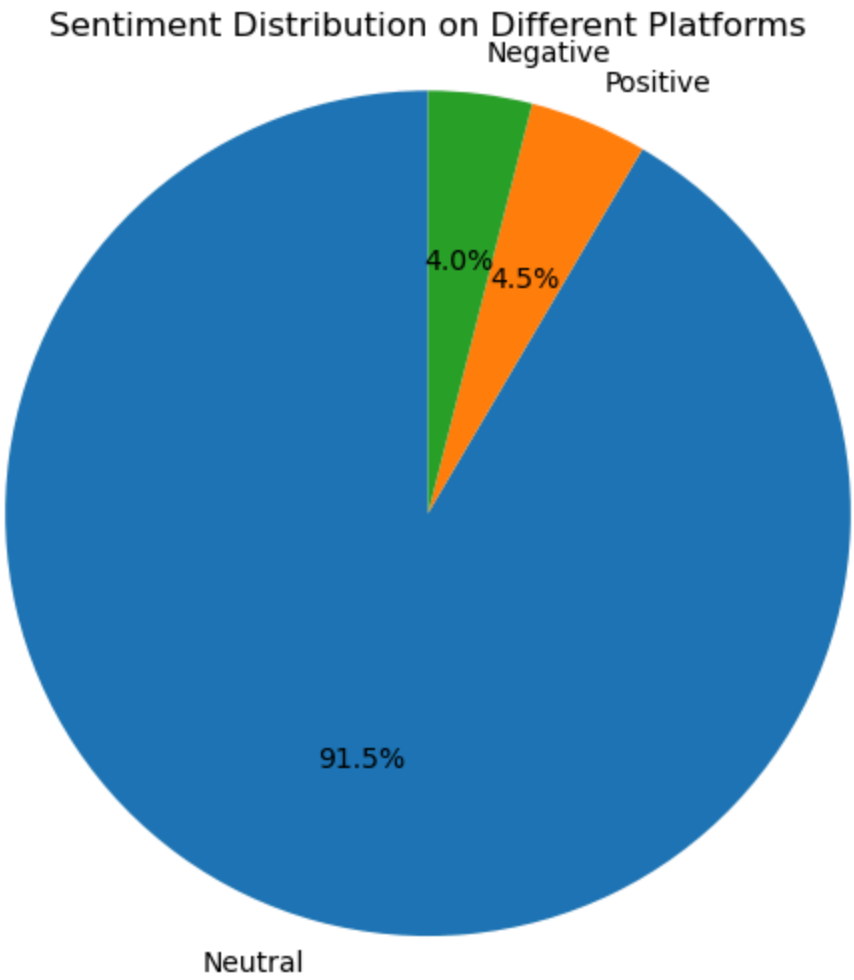


In [42]:
```
# Perform sentiment analysis on platform
import sys
from textblob import TextBlob
data['Sentiment'] = data['Hashtags'].apply(lambda x: TextBlob(x).sentiment.polarity)

# Categorize sentiment into positive, negative, and neutral
data['Sentiment Category'] = data['Sentiment'].apply(lambda x: 'Positive' if x > 0 els

# Calculate the count of each sentiment category
sentiment_counts = data['Sentiment Category'].value_counts()

# Plot a pie chart of sentiment distribution
plt.figure(figsize=(6, 6))
plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', startangle
plt.axis('equal')
plt.title('Sentiment Distribution on Different Platforms')
plt.show()
```

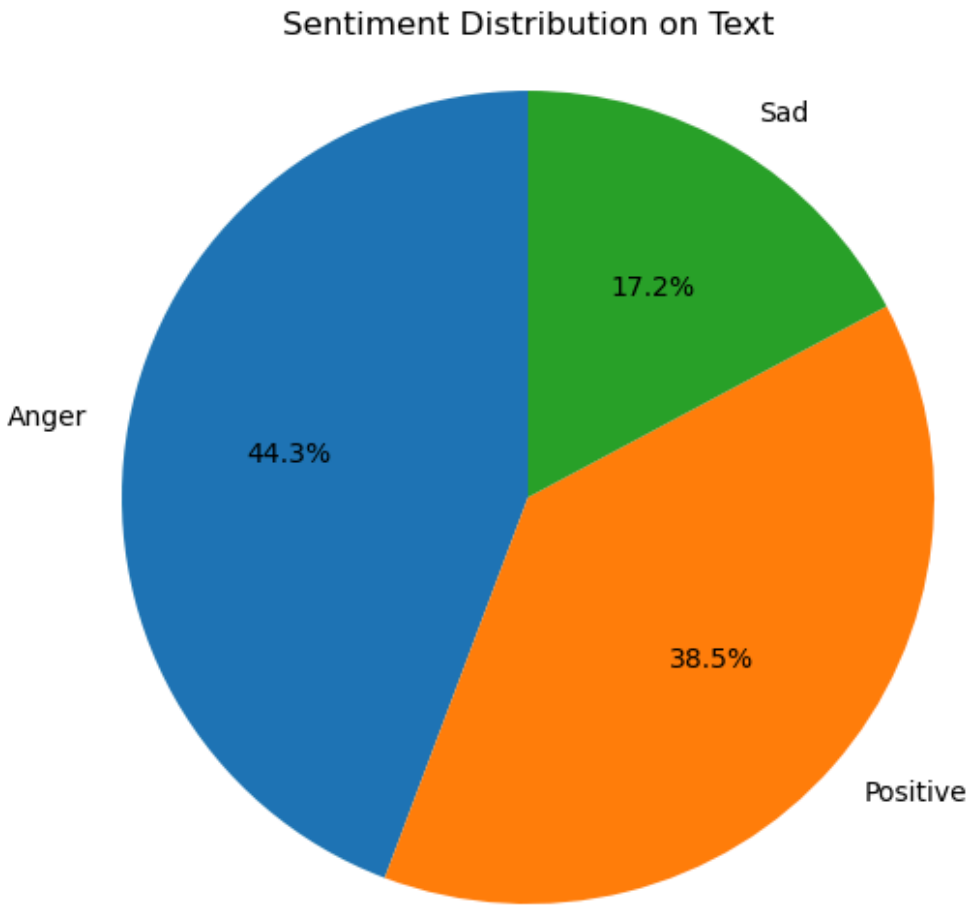


In [44]:
```
# Perform sentiment analysis on text
import sys
from textblob import TextBlob
data['Sentiment'] = data['Text'].apply(lambda x: TextBlob(x).sentiment.polarity)

# Categorize sentiment into positive, sad, and anger
data['Sentiment Category'] = data['Sentiment'].apply(lambda x: 'Positive' if x > 0 els

# Calculate the count of each sentiment category
sentiment_counts = data['Sentiment Category'].value_counts()

# Plot a pie chart of sentiment distribution
plt.figure(figsize=(8, 6))
plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', startangle
plt.axis('equal')
plt.title('Sentiment Distribution on Text')
plt.show()
```



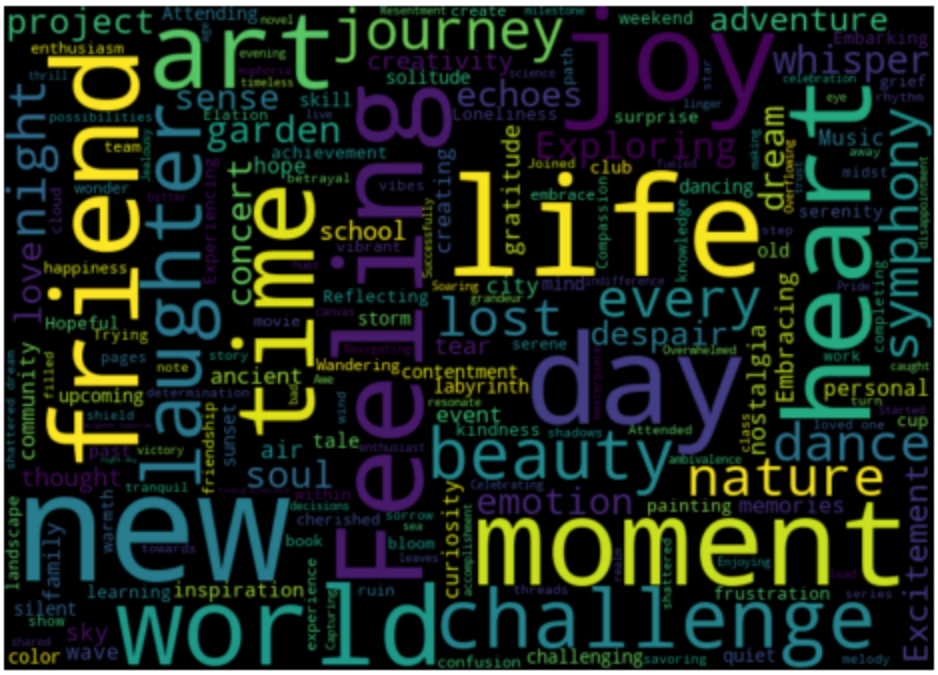
In [46]:
```
#Distribution of various Likes
plt.figure(figsize=(6,5))
p=data.groupby(['Likes']).size().reset_index(name='counts')
piechart=px.pie(p,values='counts',names='Likes',title='Likes count')
piechart.show()
```

```
<Figure size 600x500 with 0 Axes>
```

In [48]:
```
#Top 5 Hashtags Analysis
Hashtags_name=pd.DataFrame(data)
Hashtags_name=data['Hashtags'].str.split(',',expand=True).stack()
Hashtags_name=Hashtags_name.to_frame()
Hashtags_name.columns=['Hashtags']
Hashtags=Hashtags_name.groupby(['Hashtags']).size().reset_index(name='Total Content')
Hashtags=Hashtags[Hashtags.Hashtags !='Not specified']
Hashtags=Hashtags.sort_values(by=['Total Content'],ascending=False)
HashtagsTop5=Hashtags.head()
HashtagsTop5=HashtagsTop5.sort_values(by=['Total Content'])
fig1=px.bar(HashtagsTop5,x='Total Content',y='Hashtags',width=800, height=500,title='T
fig1.show()
```

In [50]:
```
#Top 5 Text Analysis
Text_name=pd.DataFrame(data)
Text_name=data['Text'].str.split(',',expand=True).stack()
Text_name=Text_name.to_frame()
Text_name.columns=['Text']
Text=Text_name.groupby(['Text']).size().reset_index(name='Total Content')
Text=Text[Text.Text !='Not specified']
Text=Text.sort_values(by=['Total Content'],ascending=False)
TextTop5=Text.head()
TextTop5=TextTop5.sort_values(by=['Total Content'])
fig1=px.bar(TextTop5,x='Total Content',y='Text',width=800, height=500,title='Top 5 Tex
fig1.show()
```

In [52]:
```
# Creating word Cloud for all Words in all platform Text
from wordcloud import WordCloud
allWords = ' '.join([text for text in data['Text']])
wordcloud = WordCloud(width=700, height=500, random_state=21, max_font_size=115).gener
plt.figure(figsize=(6, 6))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```

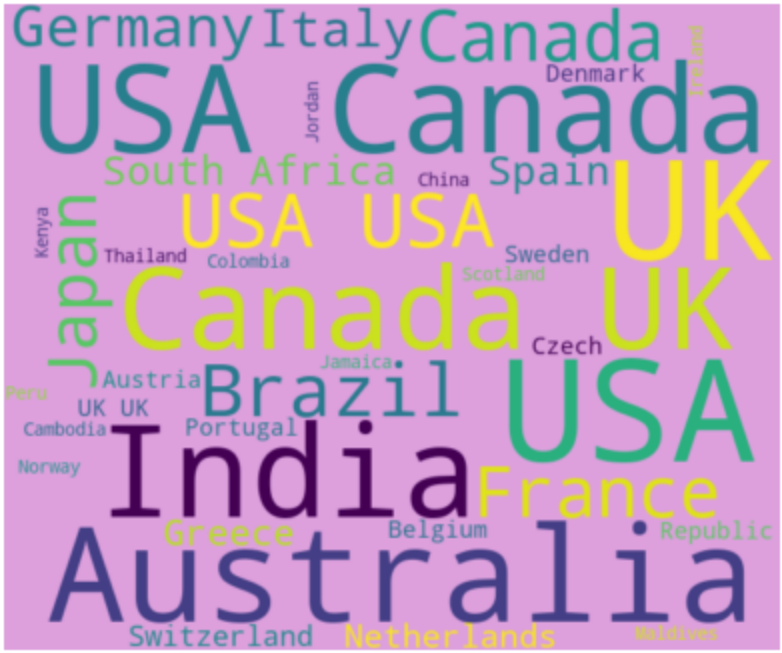


In [54]:
```
# Creating word Cloud for all Words in Platform
from wordcloud import WordCloud
allWords = ' '.join([text for text in data['Platform']])
wordcloud = WordCloud(width=800, height=500, random_state=21,background_color='green',
plt.figure(figsize=(10, 10))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```

In [56]:
```
# Creating word Cloud for all Words in Users
from wordcloud import WordCloud
allWords = ' '.join([text for text in data['User']])
```

```
wordcloud = WordCloud(width=800, height=500, random_state=21,background_color='orange'
plt.figure(figsize=(10, 10))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```

In [58]:
```
# Creating word Cloud for all Words in Country
from wordcloud import WordCloud
allWords = ' '.join([text for text in data['Country']])
wordcloud = WordCloud(width=600, height=500, random_state=21,background_color='plum',
plt.figure(figsize=(5, 5))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```


Germany
Italy
Canada
USA
Canada
Denmark
Ireland
Jordan
South Africa
China
Spain
UK
USA
USA
Kenya
Thailand
Colombia
Sweden
Scotland
Japan
Canada
UK
Czech
Jamaica
Austria
Brazil
USA
Peru
UK UK
Cambodia
Portugal
India
Norway
France
Greece
Belgium
Republic
Australia
Switzerland
Netherlands
Maldives

In [ ]: