

```
In [101... import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import warnings
warnings.filterwarnings(action='ignore')
```

In []:

```
In [7]: #Loading the data
```

```
data = pd.read_csv(r"C:\Users\punee\Downloads\twitter--- sentimeny\twitter.csv")
```

```
In [13]: data.head()
```

```
Out[13]:
```

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is s...
1	2	0	@user @user thanks for #lyft credit i can't us...
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in ...
4	5	0	factsguide: society now #motivation

In []:

```
In [29]: data.isnull().sum()
```

```
Out[29]: id      0
label    0
tweet    0
dtype: int64
```

```
In [31]: data.duplicated().sum()
```

```
Out[31]: 0
```

```
In [37]: data.tail()
```

	id	label	tweet
31957	31958	0	ate @user isz that youuu?ð□□□ð□□□ð□□□ð□□□ð□□□ð...
31958	31959	0	to see nina turner on the airwaves trying to...
31959	31960	0	listening to sad songs on a monday morning otw...
31960	31961	1	@user #sikh #temple vandalised in in #calgary,...
31961	31962	0	thank you @user for you follow

```
In [103... # describing the data
data.describe()
```

```
Out[103...

```

	id	label	Sentimente_score
count	31962.000000	31962.000000	31962.000000
mean	15981.500000	0.070146	0.141556
std	9226.778988	0.255397	0.414257
min	1.000000	0.000000	-0.970200
25%	7991.250000	0.000000	0.000000
50%	15981.500000	0.000000	0.000000
75%	23971.750000	0.000000	0.508000
max	31962.000000	1.000000	0.985100

```
In [ ]:
```

```
In [105... data['label'].value_counts()
```

```
Out[105... label
0      29720
1       2242
Name: count, dtype: int64
```

```
In [53]: from nltk.sentiment import SentimentIntensityAnalyzer
import nltk
```

```
In [109... nltk.download('vader_lexicon')
sia = SentimentIntensityAnalyzer()
```

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data] C:\Users\punee\AppData\Roaming\nltk_data...
[nltk_data] Package vader_lexicon is already up-to-date!
```

```
In [57]: sia
```

```
Out[57]: <nltk.sentiment.vader.SentimentIntensityAnalyzer at 0x2393a552e90>
```

```
In [ ]:
```

```
In [111... # analyzing the sentimental score

def Get_sentimental_score(text):
    return sia.polarity_scores(str(text))['compound']
```

```
In [113... data['Sentimente_score'] = data['tweet'].apply(Get_sentimental_score)
```

```
In [117... data.head()
```

Out[117...]

	id	label	tweet	Sentimente_score	sentement_label
0	1	0	@user when a father is dysfunctional and is s...	-0.8296	Negative
1	2	0	@user @user thanks for #lyft credit i can't us...	0.6705	Positive
2	3	0	bihday your majesty	0.0000	Nuetral
3	4	0	#model i love u take with u all the time in ...	0.7249	Positive
4	5	0	factsguide: society now #motivation	0.0000	Nuetral

In []:

In []: *# apply labeles for score*

In [119...]

```
def Label_setter(score):
    if score>0:
        return "Positive"
    elif score <0:
        return "Negitive"
    else:
        return "Nuetral"
```

In [121...]

```
data['sentement_label'] = data['Sentimente_score'].apply(Label_setter)
```

In [123...]

```
data.head()
```

Out[123...]

	id	label	tweet	Sentimente_score	sentement_label
0	1	0	@user when a father is dysfunctional and is s...	-0.8296	Negative
1	2	0	@user @user thanks for #lyft credit i can't us...	0.6705	Positive
2	3	0	bihday your majesty	0.0000	Nuetral
3	4	0	#model i love u take with u all the time in ...	0.7249	Positive
4	5	0	factsguide: society now #motivation	0.0000	Nuetral

In []:

In [125...]

```
# analysis based on labels

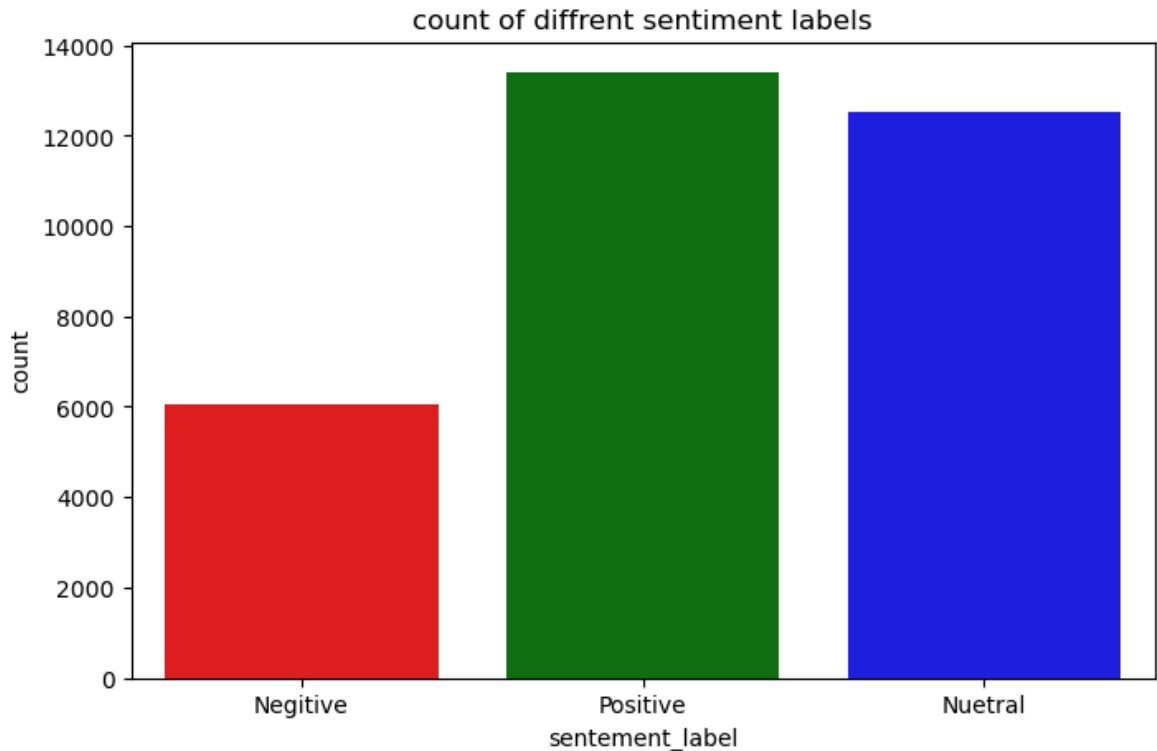
data['sentement_label'].value_counts()
```

Out[125...]

```
sentement_label
Positive      13404
Nuetral       12513
Negitive       6045
Name: count, dtype: int64
```

In []:

```
In [127... # visualizing the data
plt.figure(figsize=(8,5))
sns.countplot(x='sentement_label', data=data, palette=["red", "green", "blue"])
plt.title("count of diffrent sentiment labels")
plt.show()
```



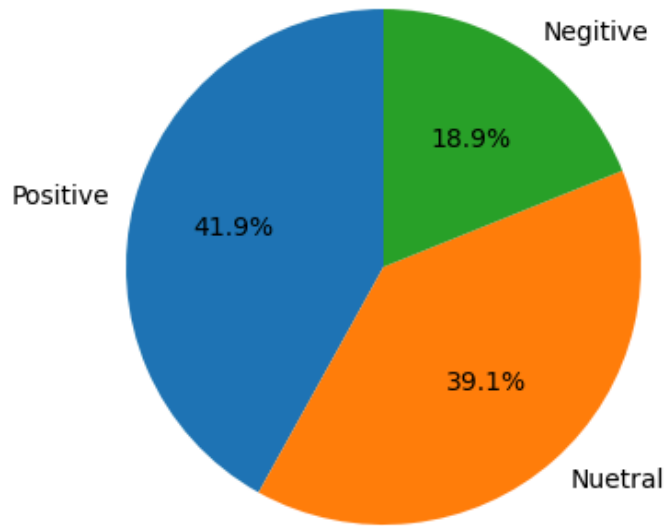
```
In [143... # piechart showing the distribution

sizes = data['sentement_label'].value_counts()

labels = sizes.index

plt.figure(figsize=(8,4))
plt.pie(sizes , labels=labels , autopct='%1.1f%%' , startangle=90)
plt.axis("equal")
plt.title("Distribution of sentiment labels")
plt.show()
```

Distribution of sentiment labels



In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: