ChatGPT sentiment analysis using VADER

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
pip install vaderSentiment
```

```
    Collecting vaderSentiment
      Downloading vaderSentiment-3.3.2-py2.py3-none-any.whl (125 kB)
         ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 126.0/126.0 kB 2.5 MB/s eta 0:00:00
    Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from vaderSentiment) (2.31.0)
    Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->vaderSentiment)
    Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->vaderSentiment) (3.6)
    Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->vaderSentiment) (2.0.7
    Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->vaderSentiment) (2023.1
    Installing collected packages: vaderSentiment
    Successfully installed vaderSentiment-3.3.2
```

```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
```

Loading the dataset

Dataset source: https://www.kaggle.com/datasets/charunisa/chatgpt-sentiment-analysis

The dataset contains tweets about the recently developed AI system - ChatGPT. Different sentiments can be observed and the aim is to find which sentiment is the majority among them using VADER, a Natural Language Tool Kit Module.

```
data=pd.read_csv('/content/file.csv')
```

```
data.head(3) # first 3 rows
```

	tweets
0	ChatGPT: Optimizing Language Models for Dialog...
1	Try talking with ChatGPT, our new AI system wh...
2	ChatGPT: Optimizing Language Models for Dialog...

```
data.tail(3) # last 3 rows
```

	tweets
15115	So chatGPT has double the context length of th...
15116	Maybe I will go through ChatGPT instead googling.
15117	#ChatGPT needs to be

```
data.shape # dimensions
```

```
    (15118, 1)
```

```
snt=SentimentIntensityAnalyzer()
```

```
df=data['tweets']
l=[]
for i in df:
  pred=snt.polarity_scores(i)
  if(pred['compound']<-0.5):
    l.append('Negative')
  elif(pred['compound']>0.5):
    l.append('Positive')
  else:
    l.append('Neutral')
```

```
Df=pd.DataFrame() # creating a new dataframe that includes both tweets and their respective sentiments
Df['Tweet']=df
Df['Sentiment']=l
Df
```

	Tweet	Sentiment
0	ChatGPT: Optimizing Language Models for Dialog...	Neutral
1	Try talking with ChatGPT, our new AI system wh...	Positive
2	ChatGPT: Optimizing Language Models for Dialog...	Neutral
3	THRILLED to share that ChatGPT, our new model ...	Positive
4	As of 2 minutes ago, @OpenAI released their ne...	Neutral
...	...	...
15113	My reaction to chatGPT https://t.co/u5reRAghc1	Neutral
15114	I’m surprised no one has compared ChatGPT to P...	Neutral
15115	So chatGPT has double the context length of th...	Neutral
15116	Maybe I will go through ChatGPT instead googling.	Neutral
15117	#ChatGPT needs to be	Neutral

15118 rows × 2 columns

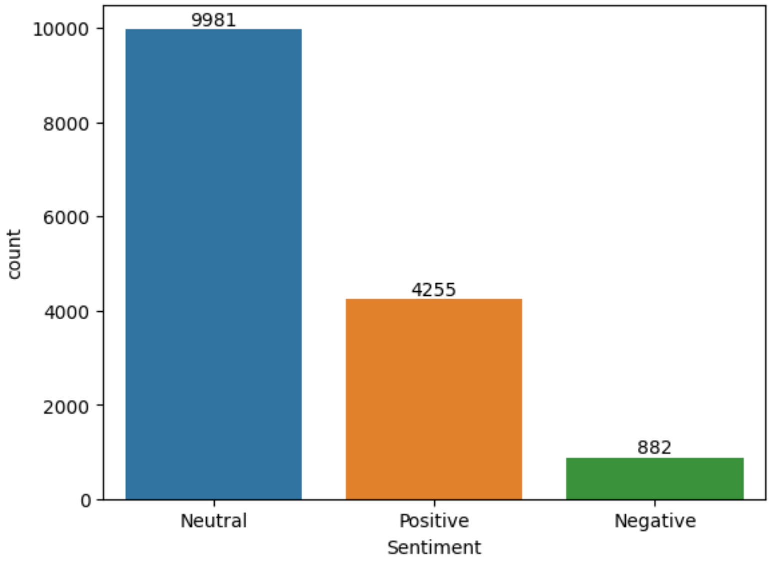
```
Df.groupby('Sentiment')['Sentiment'].count()
```

```
Sentiment
Negative     882
Neutral     9981
Positive    4255
Name: Sentiment, dtype: int64
```

```
Df['Sentiment'].value_counts()
```

```
Neutral     9981
Positive    4255
Negative     882
Name: Sentiment, dtype: int64
```

```
s=sns.countplot(data=Df,x='Sentiment',hue='Sentiment')
for x in s.containers:
  s.bar_label(x)
```



From the above chart, we can infer that, most of the tweets about ChatGPT are neutral ones.