**Sentiment Analysis System**

**WHAT**

1. **It is a Natural Language Processing Application which can analyze the sentiment on the text given by user either through Feedback,Survey,Forms etc.**
2. **This sentiment will predict the sentiment into three categories which are Positive, Negative, Neutral.**
3. **This application will access the data from a Google Form through Google sheets.**
4. **This will be a Web Application which can be utilized into a Local Area Network.**

**WHY**

**1.This project has wide range of application such as Product/Service Monitoring, Survey regarding a topic, event, personality and comments.**

**2.This project increases the opportunities for these who wants to improve their career as a data professional.**

**HOW**

**Backend: Google API**

**Google Sheets – Data Collection/Organization**

**Vadersentiment-Data Analysis**

**Plotly- Data Visualization**

**Frontend: Google Form**

**Streamlit**

**Google Sheets with Python**

1.Google Account - Google project -- Google Sheets API

2. visit <https://console.cloud.google.com>

3.Create a new Google project and select it

4.Enable the Google Sheets API

5.Configure consent screen

6.Create a key and download it.

pip install google\_auth\_oauthlib

pip install google\_api\_python\_client

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

import pandas as pd

#Permission

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

credentials=flow.run\_local\_server(port=0)

#Create a service

service=build('Sheets','v4',credentials=credentials).spreadsheets().values()

#Data Retrieval from Google Sheets

data=service.get(spreadsheetId='1JEI-3Ql7LV\_Kcf7H369X-AgQZOpCzV6Qyv3LV7OfSLA',range='A1:H').execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

pip install nltk

piip install vaderSentiment

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

import pandas as pd

#Permission

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

credentials=flow.run\_local\_server(port=0)

#Create a service

service=build('Sheets','v4',credentials=credentials).spreadsheets().values()

#Data Retrieval from Google Sheets

data=service.get(spreadsheetId='1JEI-3Ql7LV\_Kcf7H369X-AgQZOpCzV6Qyv3LV7OfSLA',range='A1:H').execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

for i in range(len(df)):

k=df.\_get\_value(i,'Please give your opinion regarding the Latest Product that you are using?')

print(k)

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

import pandas as pd

#Permission

mymodel=SentimentIntensityAnalyzer()

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

credentials=flow.run\_local\_server(port=0)

#Create a service

service=build('Sheets','v4',credentials=credentials).spreadsheets().values()

#Data Retrieval from Google Sheets

data=service.get(spreadsheetId='1JEI-3Ql7LV\_Kcf7H369X-AgQZOpCzV6Qyv3LV7OfSLA',range='A1:H').execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

for i in range(len(df)):

k=df.\_get\_value(i,'Opinion')

pred=mymodel.polarity\_scores(k)

if(pred['compound']>0.5):

print(k,"Positive sentiment")

elif(pred['compound']<-0.5):

print(k,"Negative sentiment")

else:

print("Neutral sentiment")

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

import pandas as pd

#Permission

mymodel=SentimentIntensityAnalyzer()

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

credentials=flow.run\_local\_server(port=0)

#Create a service

service=build('Sheets','v4',credentials=credentials).spreadsheets().values()

#Data Retrieval from Google Sheets

data=service.get(spreadsheetId='1aFU71bbT2QRo2zE1L8oFwW7vB88OkBLuX\_iQdqVujk4',range='A1:B').execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

l=[]

for i in range(len(df)):

k=df.\_get\_value(i,'Review')

pred=mymodel.polarity\_scores(k)

if(pred['compound']>0.5):

l.append("Positive")

elif(pred['compound']<-0.5):

l.append("Negative")

else:

l.append("Neutral")

df['Sentiment']=l

df.to\_csv("Review.csv",index=False)

pip install plotly

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

import plotly.express as px

import pandas as pd

#Permission

mymodel=SentimentIntensityAnalyzer()

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

credentials=flow.run\_local\_server(port=0)

#Create a service

service=build('Sheets','v4',credentials=credentials).spreadsheets().values()

#Data Retrieval from Google Sheets

data=service.get(spreadsheetId='1aFU71bbT2QRo2zE1L8oFwW7vB88OkBLuX\_iQdqVujk4',range='A1:B').execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

pos=0

neg=0

neu=0

for i in range(len(df)):

k=df.\_get\_value(i,'Review')

pred=mymodel.polarity\_scores(k)

if(pred['compound']>0.5):

pos+=1

elif(pred['compound']<-0.5):

neg+=1

else:

neu+=1

posper=(pos/len(df))\*100

negper=(neg/len(df))\*100

neuper=(neu/len(df))\*100

print("Positive %=",posper)

print("Negative %=",negper)

print("Neutral %=",neuper)

fig=px.pie(values=(posper,negper,neuper),names=['Positive','Negative','Neutral'])

fig.show()

main.py

import streamlit as st

from google\_auth\_oauthlib.flow import InstalledAppFlow

from googleapiclient.discovery import build

from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

import pandas as pd

import plotly.express as px

st.set\_page\_config("Sentiment Analysis Sysstem",page\_icon="https://static.thenounproject.com/png/3383100-200.png")

st.title("SENTIMENT ANALYSIS SYSTEM")

st.sidebar.image("https://www.shutterstock.com/image-vector/three-simple-emoticon-sad-neutral-260nw-1639203016.jpg")

choice=st.sidebar.selectbox("My Menu",("HOME","Analyze Sentiment","Visualize the Results","CSV File"))

if(choice=="HOME"):

st.image("https://miro.medium.com/v2/1\*\_JW1JaMpK\_fVGld8pd1\_JQ.gif")

st.markdown("<center><h1>WELCOME</h1></center>",unsafe\_allow\_html=True)

elif(choice=="Analyze Sentiment"):

url=st.text\_input("Enter Google sheet URL")

r=st.text\_input("Enter Range")

column=st.text\_input("Enter column")

btn=st.button("Analyze")

if btn:

if 'cred' not in st.session\_state:

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

st.session\_state['cred']=flow.run\_local\_server(port=0)

mymodel=SentimentIntensityAnalyzer()

service=build('Sheets','v4',credentials=st.session\_state['cred']).spreadsheets().values()

data=service.get(spreadsheetId=url,range=r).execute()

mycolumns=data['values'][0]

mydata=data['values'][1:]

df=pd.DataFrame(data=mydata,columns=mycolumns)

l=[]

for i in range(0,len(df)):

k=df.\_get\_value(i,column)

pred=mymodel.polarity\_scores(k)

if(pred['compound']>0.5):

l.append("Positive")

elif(pred['compound']<-0.5):

l.append("Negative")

else:

l.append("Neutral")

df['Sentiment']=l

st.dataframe(df)

df.to\_csv("Review.csv",index=False)

st.header("This data is been saved by the name of review.csv")

elif(choice=="Visualize the Results"):

choice2=st.selectbox("Choose Visualization",("None","Pie","Histogram"))

if choice2=="Pie":

df=pd.read\_csv("Review.csv")

posper=(len(df[df['Sentiment']=="Positive"])/len(df))\*100

negper=(len(df[df['Sentiment']=="Negative"])/len(df))\*100

neuper=(len(df[df['Sentiment']=="Neutral"])/len(df))\*100

fig=px.pie(values=[posper,negper,neuper],names=["Positive","Negative","Neutral"])

st.plotly\_chart(fig)

elif choice2=="Histogram":

t=st.text\_input("Choose any categorical Column")

if t:

df=pd.read\_csv("Review.csv")

fig=px.histogram(x=df['Sentiment'],color=df[t])

st.plotly\_chart(fig)

elif(choice=="CSV File"):

path=st.text\_input("Enter the file path")

column=st.text\_input("Enter column")

btn=st.button("Analyze")

if btn:

if 'cred' not in st.session\_state:

flow=InstalledAppFlow.from\_client\_secrets\_file('key.json',['https://www.googleapis.com/auth/spreadsheets'])

st.session\_state['cred']=flow.run\_local\_server(port=0)

mymodel=SentimentIntensityAnalyzer()

df=pd.read\_csv(path)

l=[]

for i in range(0,len(df)):

k=df.\_get\_value(i,column)

pred=mymodel.polarity\_scores(k)

if(pred['compound']>0.5):

l.append("Positive")

elif(pred['compound']<-0.5):

l.append("Negative")

else:

l.append("Neutral")

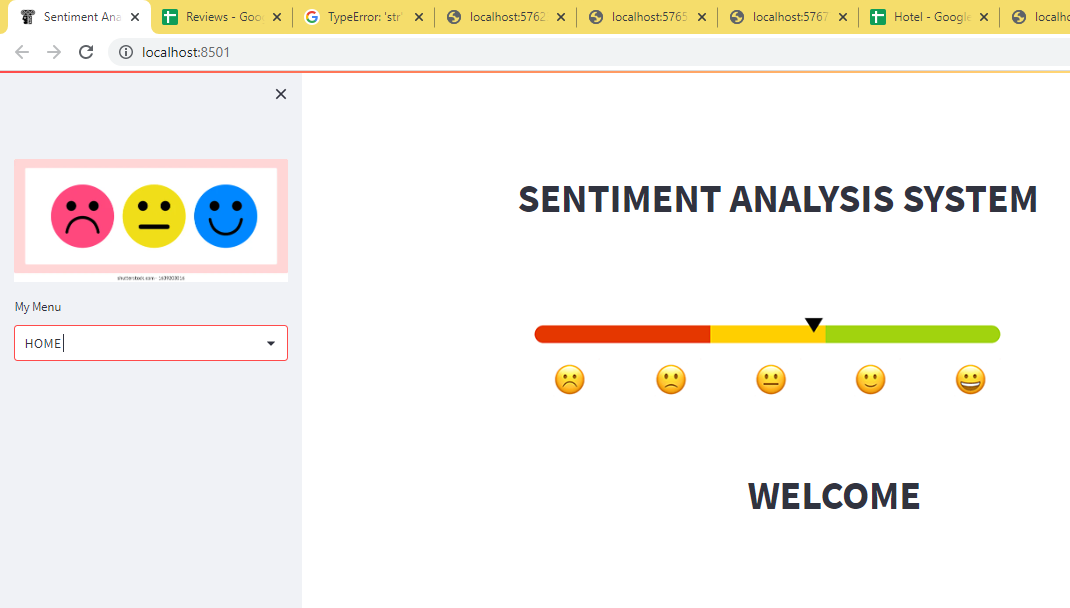
df['Sentiment']=l

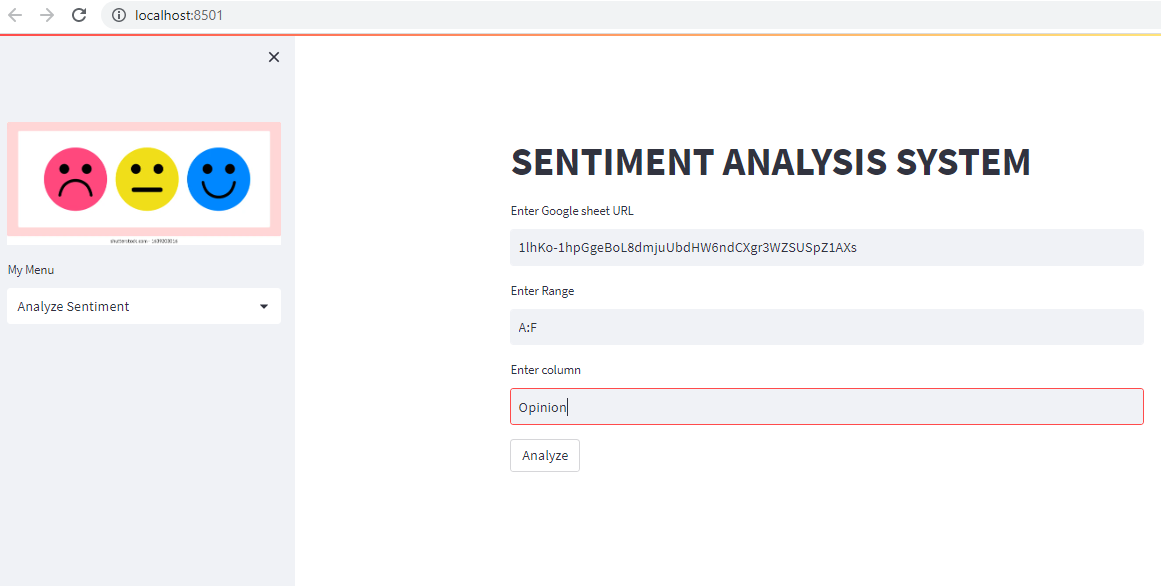
st.dataframe(df)

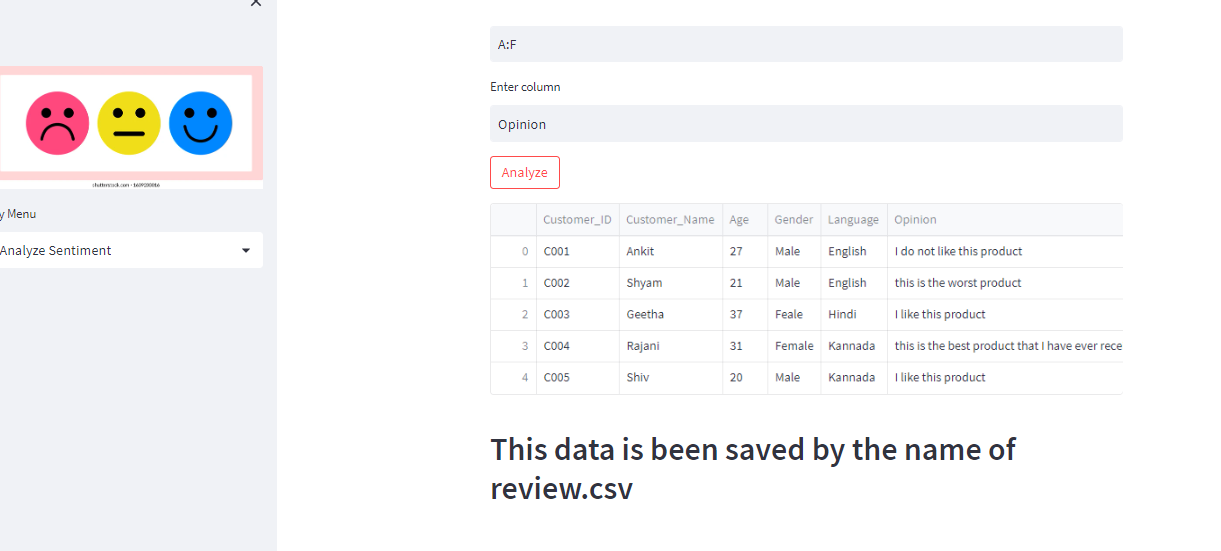
df.to\_csv("Review.csv",index=False)

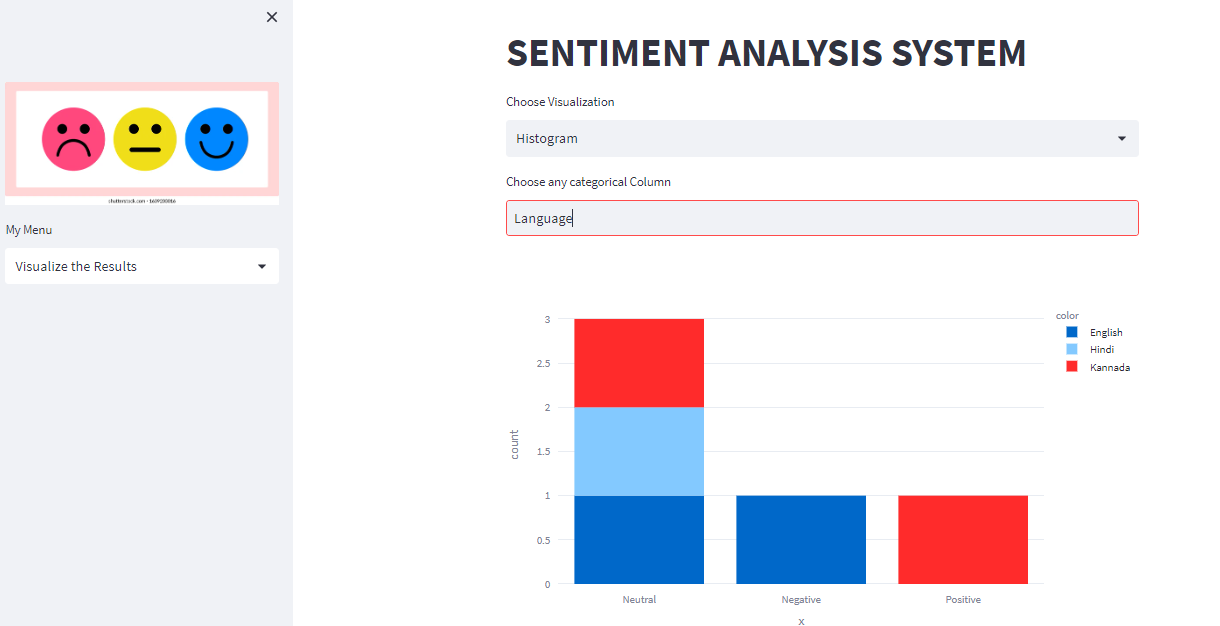
st.header("This data is been saved by the name of review.csv")

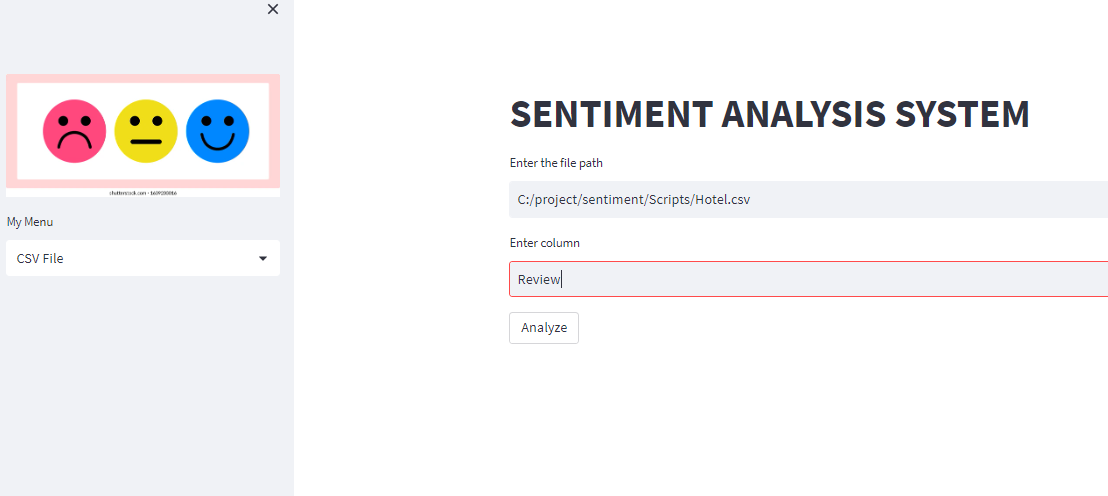
Screenshots:

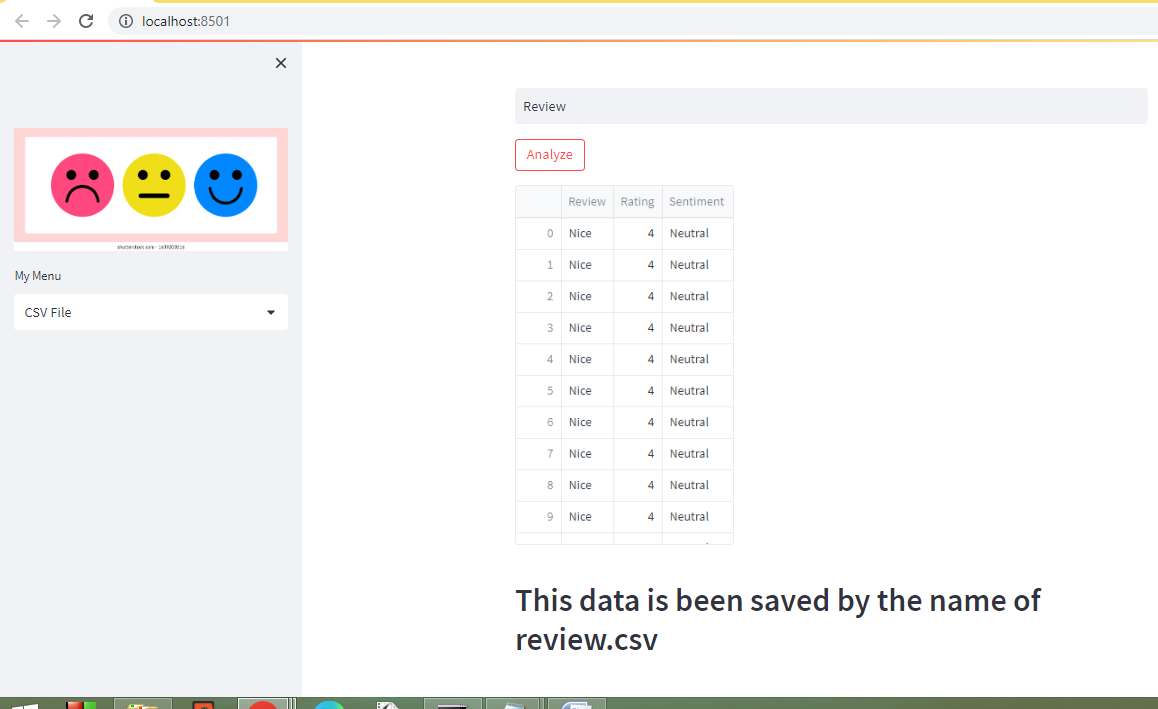


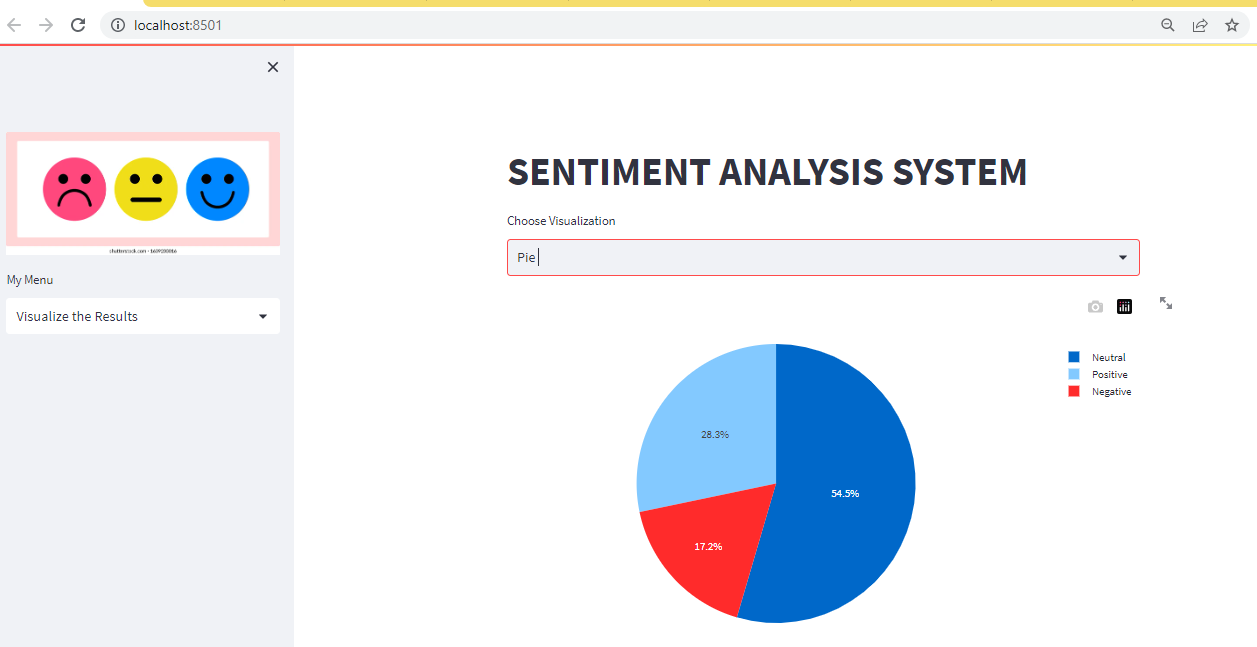


****

****

****

****

****