

A person's hands are shown typing on a laptop keyboard. The laptop screen displays a Twitter interface. A semi-transparent grey overlay covers the right side of the image, containing the text 'Twitter Sentiment app Part - 1' in a yellow, handwritten-style font. The background is a blurred office setting.

Twitter Sentiment app Part - 1

What we will be building?

Twitter Sentiment Analyser

Keep your number of tweet less than
1000

Enter the topic..

Submit

Let us input some data....

Twitter Sentiment Analyser

Keep your number of tweet less than
1000

Enter the topic..

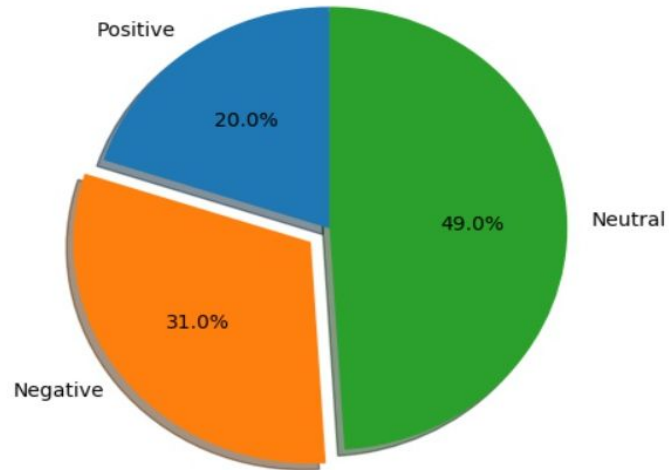
covid

100

Submit

The result....

Here is the sentiment of that topic...



So, how are we going to build that application...

The application that you just saw is a really big application (although it looks small). There are many small pieces attached to this app which we need to first understand before we actually can start building that app. This is why, we will building this application one by one. In this video, we will be learning about the backend and in the next video, we will build the frontend part. So, let us get started.

Backend vs Frontend

Backend means the part of application which is not visible to the user and it happens in the background like getting the tweets, doing some sentiment analysis etc.

The frontend is what is actually seen by the user. All the colors, look and feel of the app is decided by the frontend.

Agenda wise steps to build the whole app

1. Getting the required libraries.
 - a. Flask and Flask ngrok
 - b. Tweepy
 - c. Textblob
 - d. Matplotlib
 - e. Re
2. Getting the twitter credentials for accessing the tweets data. (consumer key, consumer secret, access token , access secret)
3. Connecting with the twitter for its data.
4. Cleaning the text of the twitter data
5. Obtaining the number of positive, negative and neutral sentiment tweets using textblob.
6. Plotting the results using Pie chart.

Agenda wise steps to build the whole app (Continued)

Building a simple “hello world” flask app.

1. Import Flask from flask
2. Import flask_ngrok (so that we can run on cloud)
3. Instantiate a new app variable
4. Set the route of the app
 - a. Return text “Hello world” from this route
5. Then, finally, run the app.



Thank you