**SENTIMENT ANALYSIS**

Sentiment Analysis involves classifying a text into various sentiments, such as positive or negative, Happy, Sad or Neutral, etc. Thus, the ultimate goal of sentiment analysis is to decipher the underlying mood, emotion, or sentiment of a text.

Sentiment analysis typically works by employing natural language processing (NLP) techniques to analyze and understand the sentiment expressed in text. The process involves several steps:

Import nltk and tokenize the word – from nltk.tokenize import word\_tokenize

To check the stopwords to be removed - From nltk.corpus import stopwords

To check if a sentiment is positive – from nltk.sentiment.vader import SentimentIntensityAnalyser

1. **Text Preprocessing:**The text data is cleaned by removing irrelevant information, such as special characters, punctuation, and stopwords.

**Example –**

1. first convert the text into lowercase by using lower() function.
2. Then clean the text by importing string and using translate(str.maketrans(“”,””,string.punctuation))
3. **Tokenization:** The text is divided into individual words or tokens to facilitate analysis.

**Example** –

1. To convert text into word we split the file contents by using word\_tokenize() function
2. Apply for loop to check the stop words and remove them and add only the meaningful words to the list
3. **Feature Extraction:** Relevant features are extracted from the text, such as words, n-grams, or even parts of speech.

**After applying the for loop these words are extracted in a list**

1. **Sentiment Classification:**Machine learning algorithms or pre-trained models are used to classify the sentiment of each text instance. This can be achieved through supervised learning, where models are trained on labeled data, or through pre-trained models that have learned sentiment patterns from large datasets.
2. Apply for loop in the emotion file and split the emotions and words
3. Check if the any word in the file is present in the list of emotions and then print those emotions as a list by IF condition
4. Show the count of the emotions – from collections import Counter then under Counter(give the list) to count the emotions.

After counting the emotions we use

SentimentIntensityAnalyser().polarity\_scores() functions to print the overall scores, amount of negative, neutral and positive sentiments.

Then we use basic If elif else condition to print neg or positive or neutral sentiments by comparing their respective scores

[**https://youtube.com/playlist?list=PLhTjy8cBISEoOtB5\_nwykvB9wfEDscuEo**](https://youtube.com/playlist?list=PLhTjy8cBISEoOtB5_nwykvB9wfEDscuEo)

[**https://www.youtube.com/live/HsI5YF6uxGs?feature=share**](https://www.youtube.com/live/HsI5YF6uxGs?feature=share)