**REPORT**

**ON**

**Sentimental Analysis in Python**

Introduction

Sentiment analysis is a popular Natural Language Processing (NLP) technique used to determine the sentiment or emotional tone expressed in a given text. In this report, we will cover the full details of performing sentiment analysis using machine learning techniques in Python.

**But why do we need sentiment analysis?**

Sentiment analysis serves as a fundamental aspect of dealing with customers on online portals and websites for the companies. They do this all the time to classify a comment as a query, complaint, suggestion, opinion, or just love for a product. This way they can easily sort through the comments or questions and prioritize what they need to handle first and even order them in a way that looks better. Companies sometimes even try to delete content that has a negative sentiment attached to it.

It is an easy way to understand and analyse public reception and perception of different ideas and concepts, or a newly launched product, maybe an event or a government policy.

Emotion understanding and sentiment analysis play a huge role in collaborative filtering-based recommendation systems. Grouping together people who have similar reactions to a certain product and showing them related products. Like recommending movies to people by grouping them with others that have similar perceptions for a certain show or movie.

**How does sentiment analysis work?**

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

1. **Text Pre-processing:**The text data is cleaned by removing irrelevant information, such as special characters, punctuation, and stop words.
2. **Tokenization:** The text is divided into individual words or tokens to facilitate analysis.
3. **Feature Extraction:** Relevant features are extracted from the text, such as words, n-grams, or even parts of speech.
4. **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 labelled data, or through pre-trained models that have learned sentiment patterns from large datasets.
5. **Post-processing:**The sentiment analysis results may undergo additional processing, such as aggregating sentiment scores or applying threshold rules to classify sentiments as positive, negative, or neutral.
6. **Evaluation:**The performance of the sentiment analysis model is assessed using evaluation metrics, such as accuracy, precision, recall, or F1 score.

## Types of Sentiment Analysis

Various types of sentiment analysis can be performed, depending on the specific focus and objective of the analysis. Some common types include:

1. **Document-Level Sentiment Analysis**: This type of analysis determines the overall sentiment expressed in a document, such as a review or an article. It aims to classify the entire text as positive, negative, or neutral.
2. **Sentence-Level Sentiment Analysis:**Here, the sentiment of each sentence within a document is analysed. This type provides a more granular understanding of the sentiment expressed in different text parts.
3. **Aspect-Based Sentiment Analysis:**This approach focuses on identifying and extracting the sentiment associated with specific aspects or entities mentioned in the text. For example, in a product review, the sentiment towards different features of the product (e.g., performance, design, usability) can be analysed separately.
4. **Entity-Level Sentiment Analysis:** This type of analysis identifies the sentiment expressed towards specific entities or targets mentioned in the text, such as people, companies, or products. It helps understand the sentiment associated with different entities within the same document.
5. **Comparative Sentiment Analysis:** This approach involves comparing the sentiment between different entities or aspects mentioned in the text. It aims to identify the relative sentiment or preferences expressed towards various entities or features.

### **About Sentiment Analysis Project:**

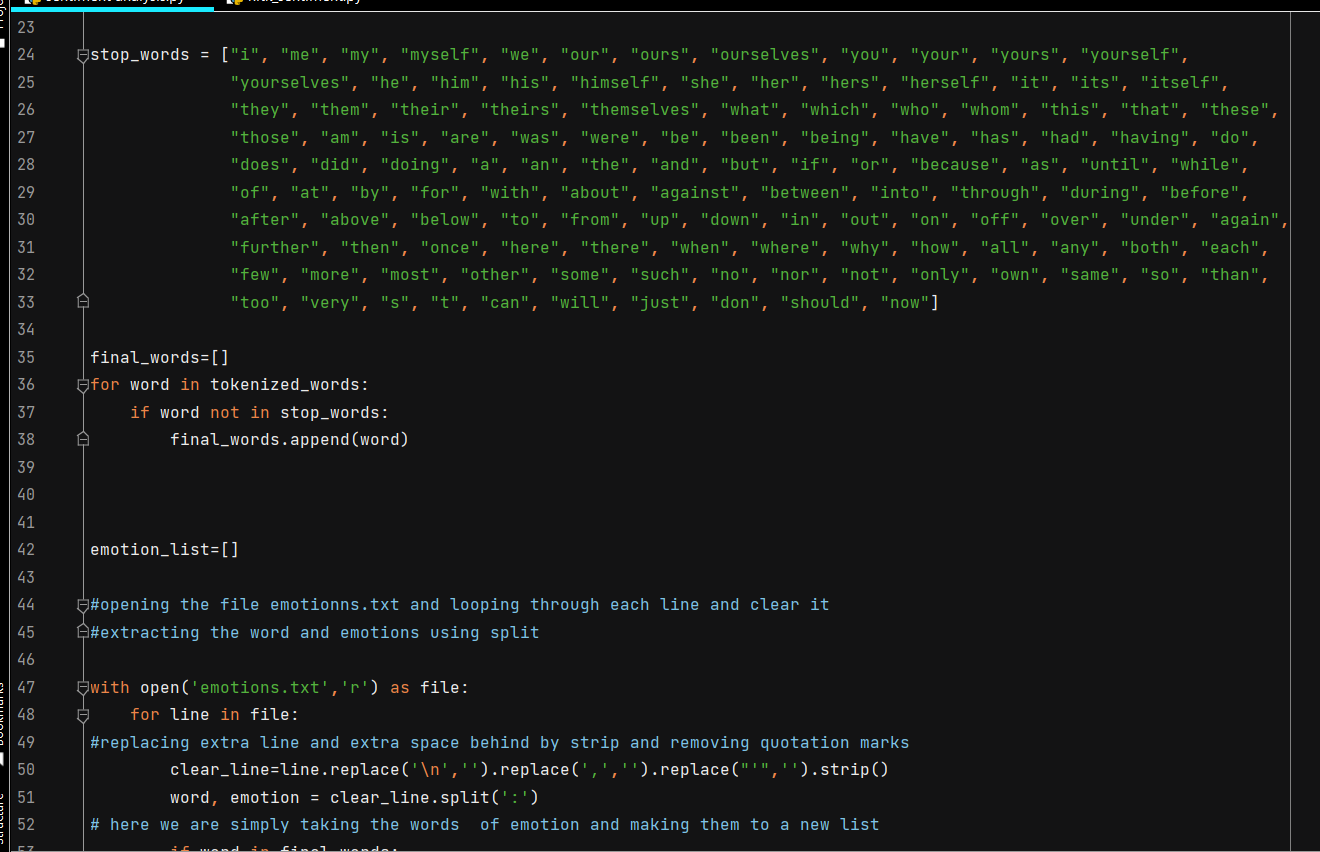
This is a Machine Learning project, in which with the help of machine learning algorithms and techniques we will classify the sentiment of text that is positive, negative, or neutral.

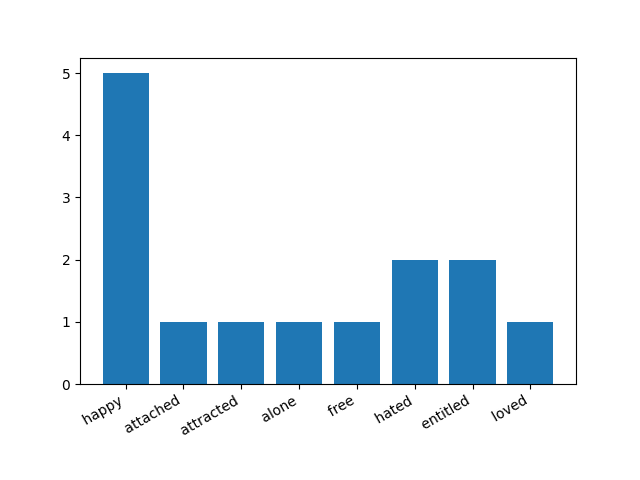
### **Required Libraries:**

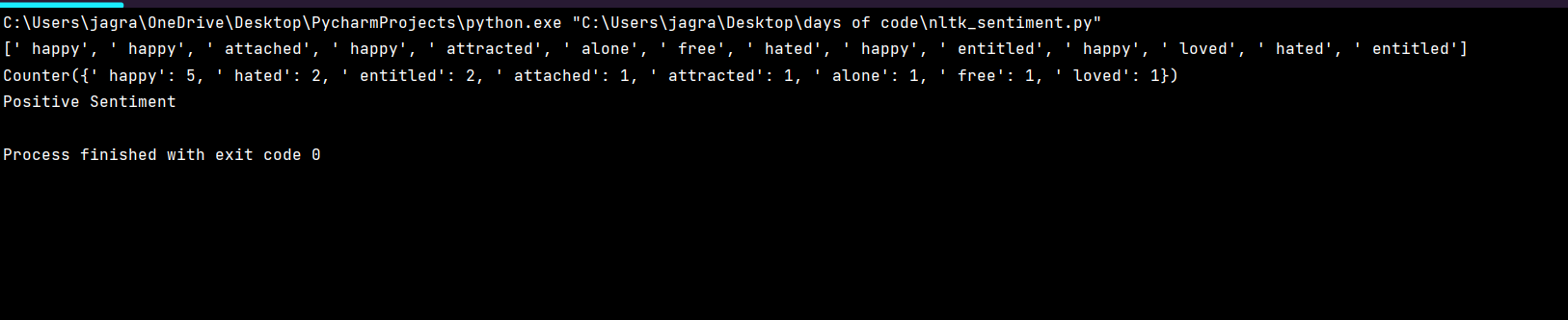
You need to install certain libraries in your system to implement the python sentiment analysis project. The required libraries are:

* NumPy (pip install NumPy)
* Pandas (pip install pandas)
* Matplotlib (pip install matplotlib)
* Natural language Processing toolkit (NLTK) (pip install nltk)
* Sklearn (pip install sklearn)

**Some source code pictures**







### **Python Sentiment Analysis Output**

**Conclusion-**

We have successfully developed python sentiment analysis model based on lstm techniques that is pretty robust and highly accurate. As discussed earlier, sentiment analysis has many use-cases based on requirements we can use it. We can similarly train it on any other kind of data just by changing the dataset according to our needs. We can use this sentiment analysis model in all different ways possible.

THANK YOU