**SENTIMENT ANALYSIS USING NLP**

**COURSE PROJECT REPORT**

**18CSE398J -Machine Learning - Core Concepts with Applications**

**(2018 Regulation)**

**III Year/ VI Semester**

**Academic Year: 2022 -2023 (EVEN)**

By

**ADARSH ANAND-RA2011003010792**

**AISHWARYA C-** **RA2011030010164**

Under the guidance of

**Dr. G. VADIVU**

**Professor**

**Department of Data Science and Business Systems**



**DEPARTMENT OF DATA SCIENCE AND BUSINESS SYSTEMS**

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**Kattankulathur, Kancheepuram**

**MAY 2023**

**SENTIMENT ANALYSIS USING NLP**

**TEAM MEMBERS**

Adarsh Anand (aa7819@srmist.edu.in)

Aishwarya C (ac4097@srmist.edu.in)

**1.ABSTRACT**

This project presents a Python sentiment analysis system utilizing Natural Language Processing (NLP) techniques, specifically the RoBERTa method. Sentiment analysis involves extracting emotions, opinions, and attitudes from textual data to classify them as positive, negative, or neutral. The project leverages the power of Python and the RoBERTa method to create an accurate sentiment analysis system.

The system begins by pre-processing textual data, removing noise, special characters, and stopwords to clean and normalize the input. Tokenization techniques are then applied to split the text into individual words or phrases, facilitating the analysis process.

The project employs the RoBERTa method, which is a state-of-the-art transformer-based model, to perform sentiment analysis. The RoBERTa model is fine-tuned on a sentiment analysis dataset, learning to associate textual features with sentiment categories. This process enhances the model's ability to accurately classify sentiment in new, unseen text.

The system allows users to input text for sentiment analysis through a user-friendly interface. Upon analysis, it provides sentiment polarity results, indicating whether the text is positive, negative, or neutral. Additionally, it offers sentiment scores to quantify the strength of the sentiment expressed in the text.

To improve the system's performance, the project incorporates various NLP techniques. These include feature engineering approaches like n-gram models and word embeddings, which capture the contextual and semantic meaning of words, enabling a better understanding of human language nuances.

The Python sentiment analysis system with the RoBERTa method finds applications in diverse domains such as social media monitoring, customer feedback analysis, and market research. It offers valuable insights into public opinion, enabling businesses to make data-driven decisions and understand user sentiment effectively.

The system's versatility is further enhanced by its capability to handle real-time data streams, allowing for instant sentiment analysis on live text feeds. It empowers users to gain valuable insights and make informed decisions based on up-to-date sentiment trends.

In summary, this Python sentiment analysis project with the RoBERTa method showcases the utilization of NLP techniques to accurately classify sentiment in textual data. It provides a user-friendly interface, robust sentiment analysis capabilities, and real-time analysis possibilities, making it a valuable tool for understanding and interpreting sentiment in the modern digital landscape.

**2.INTODUCTION**

The sentiment analysing model will be created with the help of hugging face’s pre-trained RoBERTa model.

RoBERTa is a state-of-the-art pre-trained language model developed by Facebook AI Research (FAIR) and made available by Hugging Face. It is based on the same architecture as BERT (Bidirectional Encoder Representations from Transformers), but uses a larger and optimized pre-training scheme. RoBERTa is pre-trained on a large corpus of text data and can be fine-tuned on various downstream natural language processing (NLP) tasks, such as text classification, question answering, and named entity recognition.

The implementation includes creating a function which calculates the polarity scores of the given text and then store these values into a dictionary. These polarity scores include the negative polarity, the neutral polarity, and the positive polarity.

We will begin by tokenization of the given data and running the model on the encoded text, which will be followed by storing the data into a form convenient to store and manage.

Sentiment analysis has numerous applications, including market research, social media monitoring, customer feedback analysis, and political analysis. It can help businesses and organizations understand customer opinions and preferences, improve products and services, and make informed decisions based on the feedback received from their customers. It can also be used in social media monitoring to track brand reputation and public opinion on certain topics, as well as in political analysis to understand the sentiment of voters towards political candidates or policies.

**3.DATASET**

Link for the dataset:

**<https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews>**

**About this Dataset**

**Context**

This dataset consists of reviews of fine foods from amazon. The data span a period of more than 10 years, including all ~500,000 reviews up to October 2012. Reviews include product and user information, ratings, and a plain text review. It also includes reviews from all other Amazon categories.

**Contents**

* Reviews.csv: Pulled from the corresponding SQLite table named Reviews in database.sqlite
* database.sqlite: Contains the table 'Reviews'

Data includes:

* Reviews from Oct 1999 - Oct 2012
* 568,454 reviews
* 256,059 users
* 74,258 products
* 260 users with > 50 reviews

**4.METHODS**

This project focuses on sentiment analysis using multiple methods, including Vader and RoBERTa, and compares their results to evaluate their effectiveness. In addition to these methods, the project explores other commonly used approaches in sentiment analysis.

The system starts by pre-processing textual data, performing tasks such as noise removal, special character handling, and stopword removal. The processed data is then fed into various sentiment analysis methods for evaluation.

Vader, a rule-based method, utilizes sentiment lexicons to assign sentiment scores to words in the text. By aggregating these scores, Vader calculates an overall sentiment score for the text.

RoBERTa, a transformer-based model, leverages deep learning techniques and fine-tuning on a sentiment analysis dataset to predict sentiment categories for input text. It captures contextual information and learns to associate textual features with sentiment.

In addition to Vader and RoBERTa, the project incorporates other popular sentiment analysis methods. These may include Naive Bayes, Support Vector Machines, or other machine learning algorithms trained on sentiment-labeled datasets.

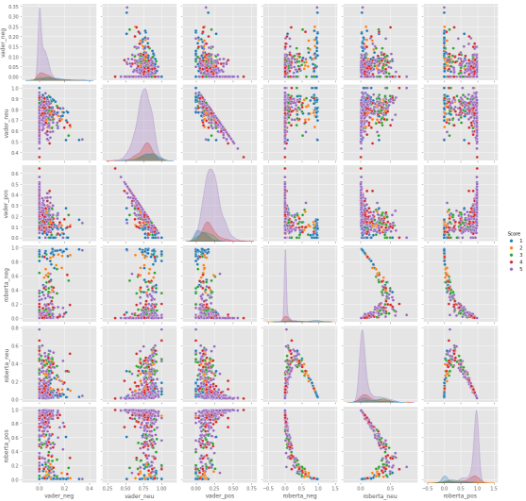
The system collects the sentiment predictions from each method and compares the results. It evaluates the performance of each method in terms of accuracy, precision, recall, and F1-score to assess their effectiveness in sentiment classification.

Furthermore, the project explores the strengths and weaknesses of each method. It examines cases where different methods provide consistent results and cases where they differ, aiming to identify scenarios where certain methods excel or struggle in sentiment analysis.

The project presents the comparison results in a user-friendly interface, allowing users to input text and view the sentiment analysis outcomes from multiple methods. Visualizations and metrics are provided to facilitate understanding and interpretation of the comparative results.

By comparing the results of different sentiment analysis methods, this project provides insights into the suitability and performance of various approaches. It helps users understand the trade-offs between different methods and make informed decisions about selecting the most appropriate method for their specific sentiment analysis tasks.

**5.EXPERIMENTS AND RESULTS**

****

The Roberta Model is more accurate when compared to Vader Model.

**6.CONCLUSIONS AND FUTURE WORK**

**Positive 1-Star and Negative 5-Star Reviews**

Lets look at some examples where the model scoring(by the commentator) and review score(by Roberta model) differ the most.

In [27]:

results\_df.query('Score == 1') \

.sort\_values('roberta\_pos', ascending=False)['Text'].values[0]

Out[27]:

'I felt energized within five minutes, but it lasted for about 45 minutes. I paid $3.99 for this drink. I could have just drunk a cup of coffee and saved my money.'

In [28]:

results\_df.query('Score == 1') \

.sort\_values('vader\_pos', ascending=False)['Text'].values[0]

Out[28]:

'So we cancelled the order. It was cancelled without any problem. That is a positive note...'

In [29]:

*# nevative sentiment 5-Star view*

In [30]:

results\_df.query('Score == 5') \

.sort\_values('roberta\_neg', ascending=False)['Text'].values[0]

Out[30]:

'this was sooooo deliscious but too bad i ate em too fast and gained 2 pds! my fault'

In [31]:

linkcode

results\_df.query('Score == 5') \

.sort\_values('vader\_neg', ascending=False)['Text'].values[0]

Out[31]:

'this was sooooo deliscious but too bad i ate em too fast and gained 2 pds! my fault'

These are some nuanced comments which confused our model. In future we have to work on this model to make it more accurate.

**7.REFERENCES**

* J. McAuley and J. Leskovec. [From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. WWW, 2013.](http://i.stanford.edu/~julian/pdfs/www13.pdf)



##### [Amazon Fine Food Reviews](https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews" \t "_blank)

* **Analyze ~500,000 food reviews from Amazon**

<https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews>