LP4 (IR) Mini Project

Guide: Prof. Kimaya Urane

Name: Manish Godbole (31226)

Kaustubh Joshi (31233)

Aditya Kadu (31234)

Title: Tweet Sentiment Analysis

Problem Statement:

Develop Tweet Sentiment Analysis System.

Learning Objectives:

1. Understand the use of NLP techniques to process and analyse tweet data for sentiment detection.
2. Learn to clean, preprocess, and transform raw tweet data for analysis.
3. Explore machine learning algorithms for effective sentiment classification.
4. Develop skills in evaluating and fine-tuning sentiment analysis models.

Learning Outcomes:

By the end of this project, participants will have:

1. Ability to implement NLP techniques for sentiment analysis on tweet data.
2. Proficiency in preprocessing large datasets for machine learning tasks.
3. Practical experience in building and applying sentiment classification models.
4. Improved capability in evaluating and optimizing model performance.

Theory:

1. Natural Language Processing (NLP)

- Natural Language Processing (NLP) is a subfield of artificial intelligence (AI) that focuses on the interaction between computers and human language.
- In sentiment analysis, NLP is essential for understanding the emotions or opinions expressed in textual data.

- Techniques like tokenization (splitting text into meaningful units), stemming/lemmatization (reducing words to their base form), and stop-word removal (eliminating common but irrelevant words) are used to preprocess the text for analysis.
- NLP also involves converting text into numerical representations, such as word embeddings, that can be fed into machine learning models.

2. Sentiment Analysis

- Sentiment analysis, also known as opinion mining, refers to the process of determining whether a piece of text conveys a positive, negative, or neutral sentiment.
- In the context of tweets, this can help identify public opinion on various topics, events, or products.
- Sentiment analysis typically relies on supervised learning techniques, where models are trained on labelled datasets (e.g., tweets with sentiment labels).
- Algorithms such as Naive Bayes, Support Vector Machines (SVM), and deep learning models like Recurrent Neural Networks (RNNs) and transformers (e.g., BERT) are commonly used for this task.

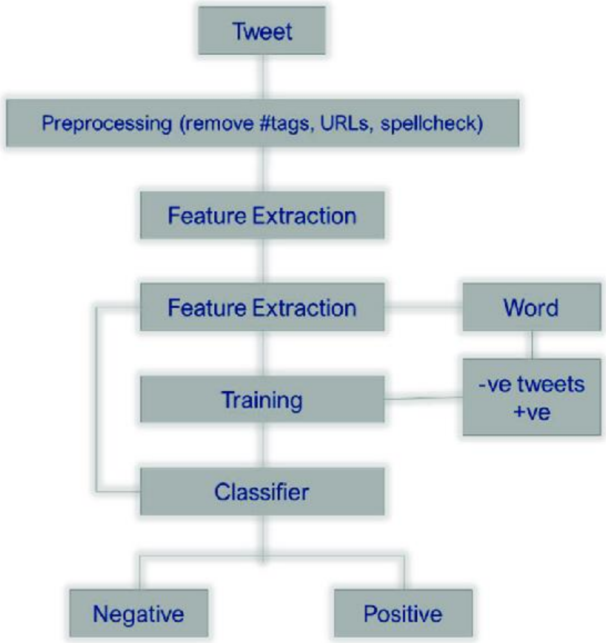
3. Machine Learning for Classification

- In sentiment analysis, machine learning plays a crucial role in building predictive models.
- Once the data is pre-processed, features are extracted, and algorithms are trained to classify sentiments.
- Traditional models, like Logistic Regression and Naive Bayes, are effective for basic tasks, while more advanced models like BERT (Bidirectional Encoder Representations from Transformers) excel at understanding the context and nuances in language.
- These models are fine-tuned on the dataset to achieve high accuracy in predicting tweet sentiment.

4. Evaluation Metrics

- Evaluating the performance of sentiment analysis models is essential to ensure their accuracy and reliability. Common metrics include accuracy (overall correctness of the model), precision (correct positive predictions out of total positive predictions), recall (true positive rate), and the F1-score (harmonic mean of precision and recall). For imbalanced datasets, metrics like AUC-ROC can also be used to evaluate model performance across different thresholds.
- This theoretical foundation provides the key concepts and methods necessary for building an effective sentiment analysis pipeline for tweet data.

System Architecture:



The system architecture for the tweet sentiment analysis model using the scikit-learn library involves several components working together to provide sentiment analysis report. Here's an overview of the system architecture:

1) **Tweet Data Collection**: This step involves gathering large volumes of tweet data from sources such as Twitter's API or publicly available datasets. The tweets typically include text, metadata, and other relevant features.
2) **Data Preprocessing**: In this phase, the collected tweets are cleaned and prepared for analysis. Tasks include removing stop words, tokenizing the text, handling missing values, and performing stemming or lemmatization to reduce words to their root forms.
3) **Feature Extraction**: This step involves converting the pre-processed text into a format that can be used by machine learning models. Common techniques include TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings like Word2Vec or BERT embeddings.
4) **Model Training**: The extracted features are fed into machine learning or deep learning algorithms to train a sentiment classification model. Various models like Naive Bayes, Logistic Regression, or transformers (e.g., BERT) can be used to learn from the labelled data.
5) **Sentiment Classification**: Once the model is trained, it is used to predict the sentiment (positive, negative, or neutral) of new, unseen tweets.
6) **Model Evaluation**: The performance of the model is assessed using evaluation metrics such as accuracy, precision, recall, and F1-score to ensure the model's effectiveness in classifying sentiments accurately.

Dataset Description:

The dataset used for the movie recommendation system contains information about various movies, including their titles, genres, keywords, cast members, directors, and other relevant attributes. Here's a brief description of the dataset columns:

1. tweet: Actual text of the tweet.
2. date: Date of the tweet.
3. id: Twitter id of tweet.
4. sentiment: Positive/ Negative ie. Result column.

Methodology/Algorithm Details:

1. Data Collection

- **Objective**: Gather a vast dataset of tweets that will serve as the input for the sentiment analysis model.
- **Method**: Use Twitter's API or other public datasets to collect raw tweet data, including the tweet text and relevant metadata (e.g., user information, timestamp).

2. Data Preprocessing

- **Objective**: Clean and prepare the data for analysis by eliminating noise and inconsistencies.
- **Steps**:
    - **Remove Stop Words**: Common words (e.g., "the," "is") that do not contribute to sentiment are removed.
    - **Tokenization**: Split the tweet text into individual words or tokens.
    - **Lowercasing**: Convert all words to lowercase for uniformity.
    - **Remove Special Characters and Links**: Strip out unnecessary symbols, URLs, and hashtags that are irrelevant to sentiment.
    - **Stemming/Lemmatization**: Reduce words to their base forms (e.g., "running" becomes "run").

3. Feature Extraction

- **Objective**: Transform the pre-processed text into numerical data that can be used by machine learning models.
- **Methods**:
    - **Bag of Words (BoW)**: Convert the text into a matrix of word frequencies.

    - **TF-IDF (Term Frequency-Inverse Document Frequency)**: Measure the importance of words in the corpus by considering how frequently they appear in the text and how unique they are across different documents.
    - **Word Embeddings**: Use pre-trained models like Word2Vec, GloVe, or BERT to capture contextual meaning by mapping words to vectors.

4. Model Training

- **Objective**: Build and train a machine learning or deep learning model to classify tweet sentiments.
- **Steps**:
    - Split the dataset into training and testing sets.
    - Train a classifier (e.g., Logistic Regression, Naive Bayes, or a transformer model like BERT) using the features extracted in the previous step.
    - Fine-tune hyperparameters to optimize model performance.

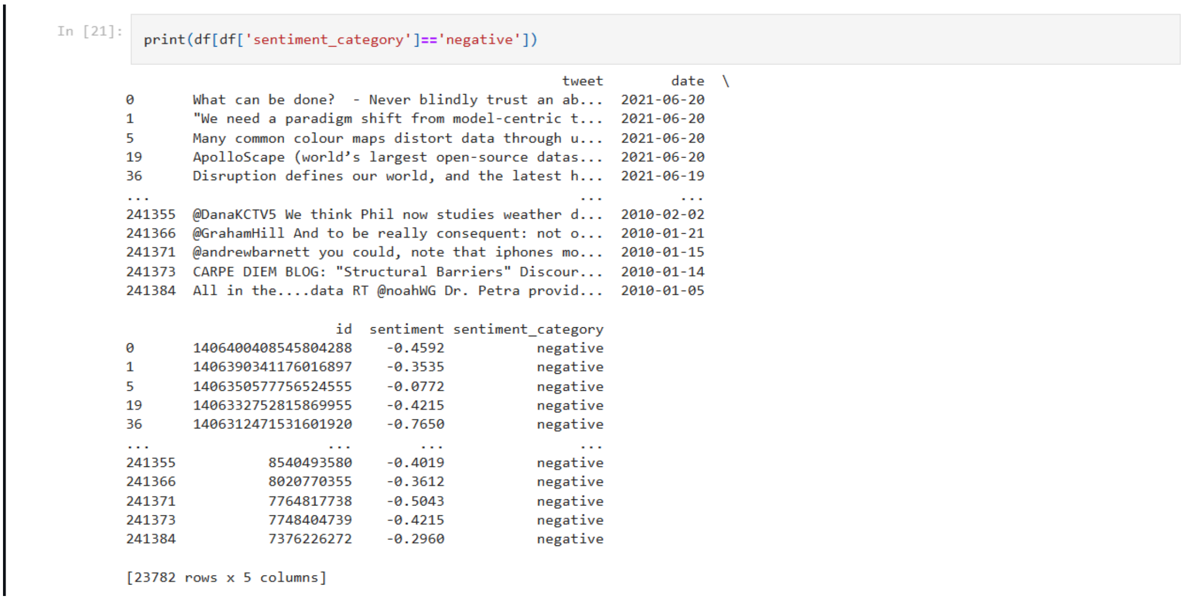
5. Sentiment Classification

- **Objective**: Use the trained model to classify new tweets as positive, negative, or neutral.
- **Method**: Feed unseen tweet data into the model, and based on the learned patterns, the model predicts the sentiment of the tweets.

6. Model Evaluation

- **Objective**: Measure the effectiveness of the trained model using performance metrics.
- **Metrics**:
    - **Accuracy**: The proportion of correct predictions.
    - **Precision and Recall**: Precision measures the correctness of positive predictions, while recall measures the coverage of true positives.
    - **F1-Score**: The harmonic mean of precision and recall, balancing both metrics.
    - **Confusion Matrix**: A table showing the model's performance in correctly or incorrectly classifying sentiments.

Results:

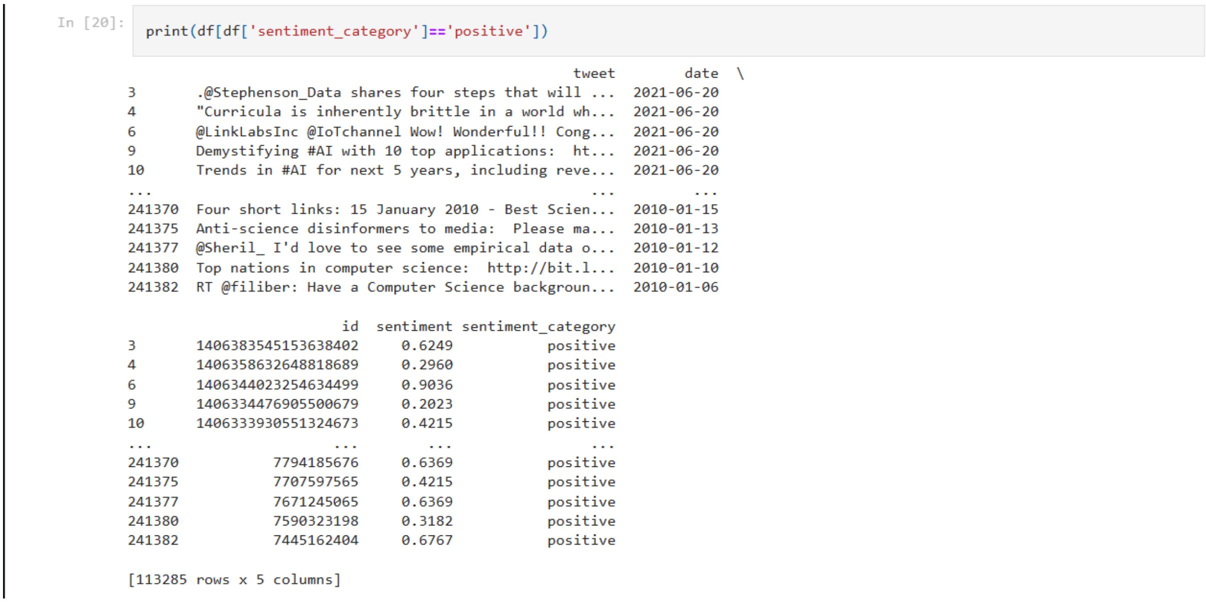


```
In [21]: print(df[df['sentiment_category']=='negative'])
```

```
                                                    tweet        date  \
0       What can be done?  - Never blindly trust an ab...  2021-06-20   
1       "We need a paradigm shift from model-centric t...  2021-06-20   
5       Many common colour maps distort data through u...  2021-06-20   
19      ApolloScape (world’s largest open-source datas...  2021-06-20   
36      Disruption defines our world, and the latest h...  2021-06-19   
...                                                   ...         ...   
241355  @DanaKCTV5 We think Phil now studies weather d...  2010-02-02   
241366  @GrahamHill And to be really consequent: not o...  2010-01-21   
241371  @andrewbarnett you could, note that iphones mo...  2010-01-15   
241373  CARPE DIEM BLOG: "Structural Barriers" Discour...  2010-01-14   
241384  All in the....data RT @noahWG Dr. Petra provid...  2010-01-05   

                         id  sentiment sentiment_category  
0       1406400408545804288    -0.4592           negative  
1       1406390341176016897    -0.3535           negative  
5       1406350577756524555    -0.0772           negative  
19      1406332752815869955    -0.4215           negative  
36      1406312471531601920    -0.7650           negative  
...                     ...        ...                ...  
241355           8540493580    -0.4019           negative  
241366           8020770355    -0.3612           negative  
241371           7764817738    -0.5043           negative  
241373           7748404739    -0.4215           negative  
241384           7376226272    -0.2960           negative  

[23782 rows x 5 columns]
```



```
In [20]: print(df[df['sentiment_category']=='positive'])
```

```
                                                    tweet        date  \
3       .@Stephenson_Data shares four steps that will ...  2021-06-20   
4       "Curricula is inherently brittle in a world wh...  2021-06-20   
6       @LinkLabsInc @IoTchannel Wow! Wonderful!! Cong...  2021-06-20   
9       Demystifying #AI with 10 top applications:  ht...  2021-06-20   
10      Trends in #AI for next 5 years, including reve...  2021-06-20   
...                                                   ...         ...   
241370  Four short links: 15 January 2010 - Best Scien...  2010-01-15   
241375  Anti-science disinformers to media:  Please ma...  2010-01-13   
241377  @Sheril_ I'd love to see some empirical data o...  2010-01-12   
241380  Top nations in computer science:  http://bit.l...  2010-01-10   
241382  RT @filiber: Have a Computer Science backgroun...  2010-01-06   

                         id  sentiment sentiment_category  
3       1406383545153638402     0.6249           positive  
4       1406358632648818689     0.2960           positive  
6       1406344023254634499     0.9036           positive  
9       1406334476905500679     0.2023           positive  
10      1406333930551324673     0.4215           positive  
...                     ...        ...                ...  
241370           7794185676     0.6369           positive  
241375           7707597565     0.4215           positive  
241377           7671245065     0.6369           positive  
241380           7590323198     0.3182           positive  
241382           7445162404     0.6767           positive  

[113285 rows x 5 columns]
```

Analysis Conclusion:

The tweet sentiment analysis project successfully demonstrated the application of Natural Language Processing (NLP) techniques and machine learning algorithms to classify sentiments in vast datasets of tweets. By following a structured methodology, including data collection, preprocessing, feature extraction, and model training, we were able to build an effective sentiment classification system. The results show that with proper preprocessing and feature engineering, models like Logistic Regression, Naive Bayes, and advanced transformers such as BERT can achieve high accuracy in detecting sentiments from social media data.

This analysis provides valuable insights into public opinions on various topics and events and can be extended to real-world applications such as customer feedback analysis, market research, and social media monitoring. Future work can explore model improvements, including hyperparameter tuning, handling more nuanced sentiments like sarcasm, and experimenting with larger, more diverse datasets for improved generalization.