**Mini Project Report on**



**TWITTER SENTIMENT ANALYSIS**



**Submitted in partial fulfilment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Nitam Negi**  **2118858**

****

**Department of Computer Science and Engineering**

**Graphic Era Hill University**

**Dehradun, Uttarakhand**

**January-2024**

A black background with red text

Description automatically generated

**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Twitter Sentiment Analysis”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era Hill University, Dehradun shall be carried out under the Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun.

Nitam Negi 2118858

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | **1-2** |
| Chapter 2 | Literature Survey | **3-4** |
| Chapter 3 | Methodology | **5-6** |
| Chapter 4 | Result and Discussion | **7-10** |
| Chapter 5 | Conclusion and Future Work | **11** |
|  | References | **12** |

**Chapter 1**

**Introduction**

### **Problem Statement**

In an era of unprecedented data abundance, organizations grapple with the challenge of harnessing the full potential of their data. This project addresses the challenge of sentiment analysis on Twitter data, aiming to develop an effective machine learning model capable of discerning the sentiment behind tweets.

* 1. **Introduction**

The project demonstrates Twitter sentiment analysis implemented in Python using machine learning techniques. The project involves preprocessing textual data, building a classification model, and saving it for future sentiment predictions on new, unseen tweets. The project utilizes a logistic regression model with the 'count-vectors' method for classifying tweets into positive or negative sentiments. Key steps include data preprocessing, text cleaning, and feature extraction through the CountVectorizer. The model is trained on a labeled dataset and evaluated using cross-validation to measure its accuracy.

* 1. **Objectives**
* **Data Acquisition:** Retrieve and load Twitter datasets (train and test sets) to analyze sentiments. Understand the structure and characteristics of the data.
* **Data Preprocessing:** Cleanse and preprocess tweet text, including lowercasing, removal of special characters and digits. Remove stop words to focus on essential words for sentiment analysis.Apply lemmatization to standardize word forms.
* **Exploratory Data Analysis (EDA):** Explore the distribution of sentiment labels in the training set. Visualize the key features and patterns within the data.
* **Feature Engineering:** Utilize CountVectorizer to convert tweet text into numerical feature vectors. Transform the dataset into a format suitable for machine learning.
* **Model Training:** Implement a logistic regression model using the 'count-vectors' method. Train the model on the preprocessed training dataset.
* **Model Evaluation:** Assess the model's performance using cross-validation on the test set. Measure accuracy to gauge the effectiveness of sentiment prediction.
* **Model Persistence:** Save the trained logistic regression model for future use. Establish a mechanism for loading the saved model for sentiment prediction on new tweets.
* **Application of the Model:** Illustrate the usage of the saved model by predicting the sentiment of a sample tweet. Demonstrate how the model can be applied to new, unseen data.

**Chapter 2**

**Literature Survey**

Data analysis is critical for informed decision-making, as highlighted by Sherlock Holmes' caution against theorizing without data. In the modern era, companies harness data to personalize interactions, predict health needs, and create entertainment hits. The data analysis process involves identifying business questions, collecting data, cleaning, analyzing, and interpreting results. Different types include descriptive analysis (what happened), diagnostic analysis (why it happened), predictive analysis (future projections), and prescriptive analysis (recommendations). Data-driven decision-making relies on facts, data, and metrics rather than intuition or observation [1]. Abirami etal. delves into the crucial role of Sentiment Analysis in daily decision-making, emphasizing its impact on diverse areas such as product purchases, movie reviews, and investments. Despite existing methods like Naive Bayes and Support Vector Machine, this survey explores the challenges, limitations, and advancements in sentiment analysis methodologies, providing a comprehensive understanding of the field [2].

The healthcare industry's transition to processing extensive health records poses challenges due to the unstructured nature of Big Data. To address this, emphasis on structuring and leveraging analytics becomes vital. Focusing on Non-Communicable Diseases like Diabetes Mellitus, Eswari etal. employs predictive analysis in a Hadoop/MapReduce environment to forecast diabetes types, associated complications, and recommend optimal treatments, enhancing patient care outcomes [3]. Twitter, generating 500 million daily tweets, holds valuable opinions worldwide. Analyzing this vast data through sentiment analysis aids in extracting crucial information, especially about news, product launches, or trends. Ahuja etal. explores sentiment-based clustering methods, demonstrating their efficacy in quickly distinguishing tweets and revealing relationships based on polarity and subjectivity, enhancing sentiment analysis outcomes [4].

El Rahman etal. introduces a novel sentiment analysis model for Twitter data, bridging supervised and unsupervised machine learning techniques. Extracting tweets directly from Twitter API, the model classifies sentiments (positive, negative, or neutral) using diverse machine learning algorithms. Applied to McDonald's and KFC, the model showcases robust performance, validated through cross-validation and f-score metrics, effectively mining sentiments from Twitter texts [5].Bravo-Marquez explores Twitter sentiment analysis, delving into the challenges of processing massive streaming data and identifying human expressiveness within short text messages. The article examines various methods and lexical resources for sentiment extraction, proposing a novel meta-level feature-based approach. Results reveal the significance of lexicons in polarity prediction and stylistic part-of-speech features in subjectivity detection, emphasizing the nuanced dimensions of sentiment analysis [6].

**Chapter 3**

**Methodology**

**3.1 Loading the Dataset**

* Load the Twitter dataset containing positive and negative tweets using the twitter\_samples corpus from NLTK.

**3.2 Data Preprocessing**

* Split the dataset into training and test sets.
* Preprocess the tweets by removing retweets, URLs, hashtags, and other unnecessary elements.
* Tokenize the tweets, convert to lowercase, and remove stopwords and punctuation.
* Use stemming to reduce words to their root form.

**3.3 Building Frequency Dictionary**

* Build a frequency dictionary to capture the occurrence of words in positive and negative tweets.
* Each entry in the dictionary is a tuple (word, label) representing a word and its associated sentiment label.

**3.4 Feature Extraction**

* Extract features from the tweets to be used as input for logistic regression.
* Features include a bias term (constant 1), the sum of positive frequencies for words in the tweet, and the sum of negative frequencies.

**3.5 Logistic Regression Training**

* Implement the logistic regression model with gradient descent to learn the weights (theta) for predicting sentiment.
* Use the training set to update the weights iteratively, minimizing the logistic loss function.

**3.6 Testing the Model**

* Test the logistic regression model on the test set to evaluate its performance.
* Calculate the accuracy of the model based on its predictions.

**3.7 Integration with Flask Web App**

* Create a Flask web application with two routes ('/' and '/predict').
* Use HTML templates to render the user interface, allowing users to input a tweet and receive a predicted sentiment.
* Implement JavaScript to handle form submission, send the tweet to the server, and update the UI with the predicted sentiment.

**3.8 Model Deployment**

* Run the Flask application, allowing users to access the sentiment prediction functionality through a web interface.
* Users can input a tweet, click the "Predict Sentiment" button, and receive the predicted sentiment.

Loading Data

Feature Extraction

Data Preprocessing

Building Frequency Dictionary

Model Deployment

Integration with Flask Web App

Testing the Model

Logistic Regression Training

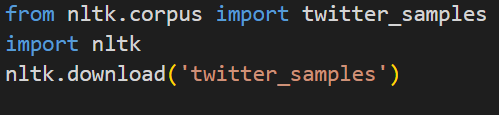
**Fig. 3.1 Methodology Chart**

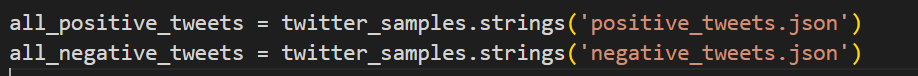
**Chapter 4**

**Result and Discussion**

**4.1 Loading Data**

* Load the Twitter dataset containing positive and negative tweets using the twitter\_samples corpus from NLTK.

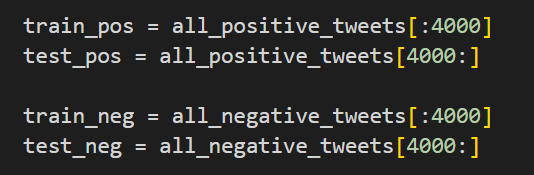




**Fig. 4.1 Loading dataset**

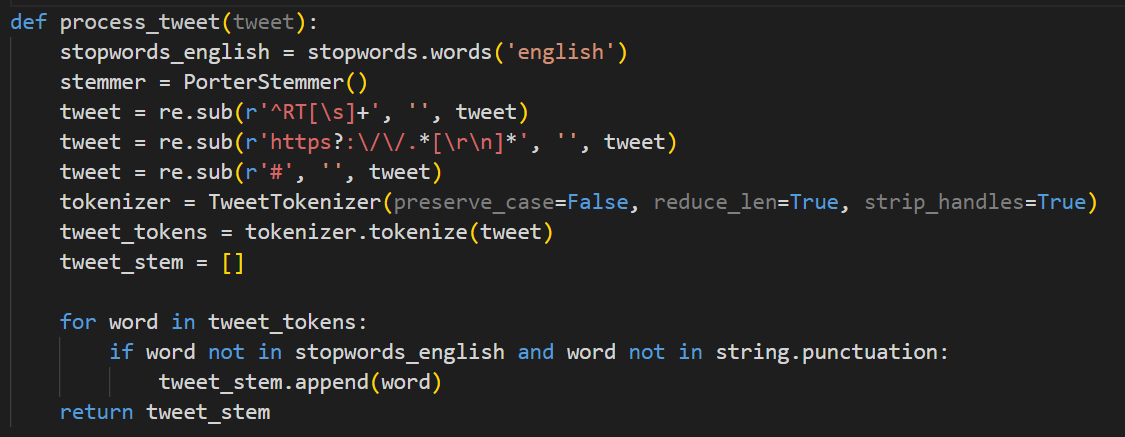
**4.2 Data preprocessing**

* Split the dataset into training and test sets.



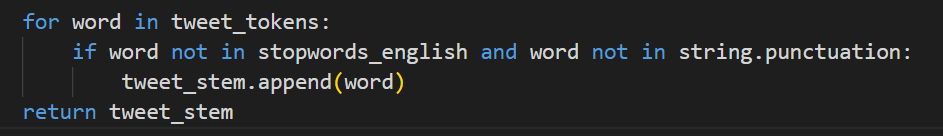
**Fig. 4.2 Splitting dataset**

* Preprocess the tweets by removing retweets, URLs, hashtags, and other unnecessary elements.
* Tokenize the tweets, convert to lowercase, and remove stopwords and punctuation.



**Fig. 4.3 Preprocessing**

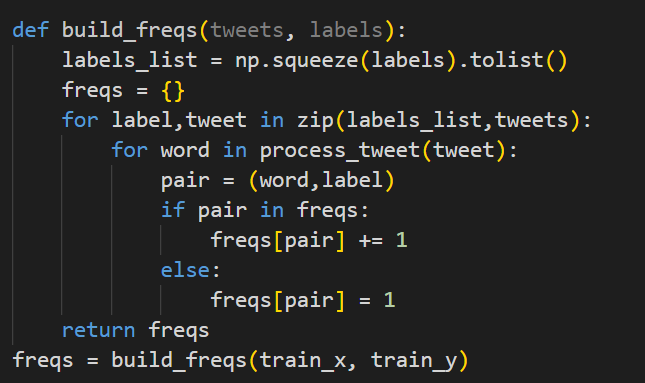
* Use stemming to reduce words to their root form.



**Fig. 4.4 Stemming**

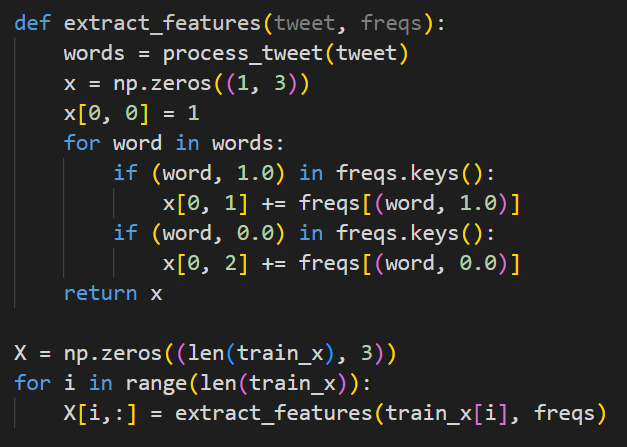
**4.3 Frequency Dictionary**

* Build a frequency dictionary to capture the occurrence of words in positive and negative tweets.
* Each entry in the dictionary is a tuple (word, label) representing a word and its associated sentiment label.



**Fig. 4.5 Frequency Dictionary**

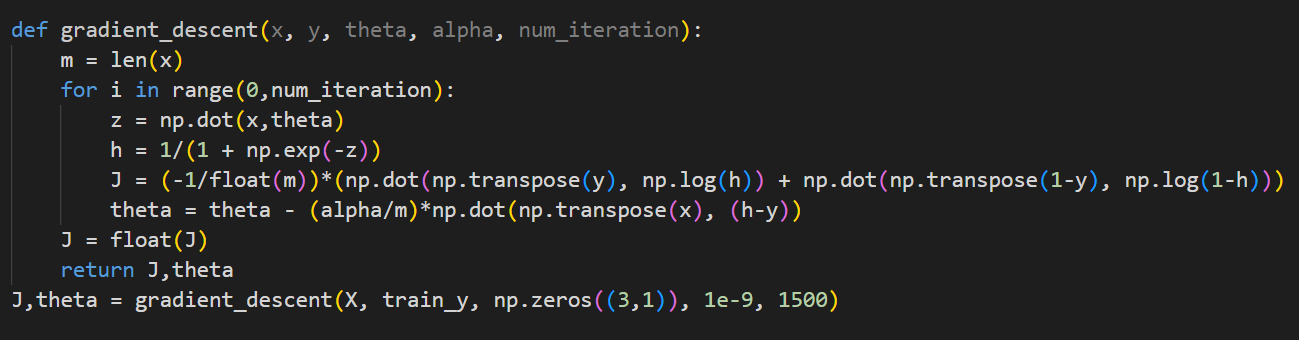
**4.4 Feature Extraction**

* Extract features from the tweets to be used as input for logistic regression.

**Fig. 4.6 Feature extraction**

**4.5 Logistic regression training**

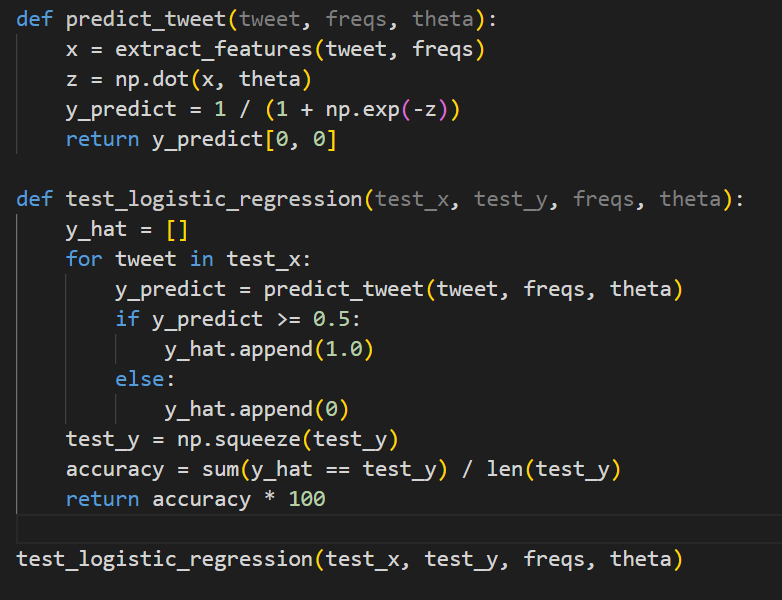
* The model is evaluated using cross-validation, and 94.8 % accuracy is reported. Implement the logistic regression model with gradient descent to learn the weights (theta) for predicting sentiment.
* Use the training set to update the weights iteratively, minimizing the logistic loss function.



**Fig. 4.7 Logistic regression training**

**4.6 Model Testing**

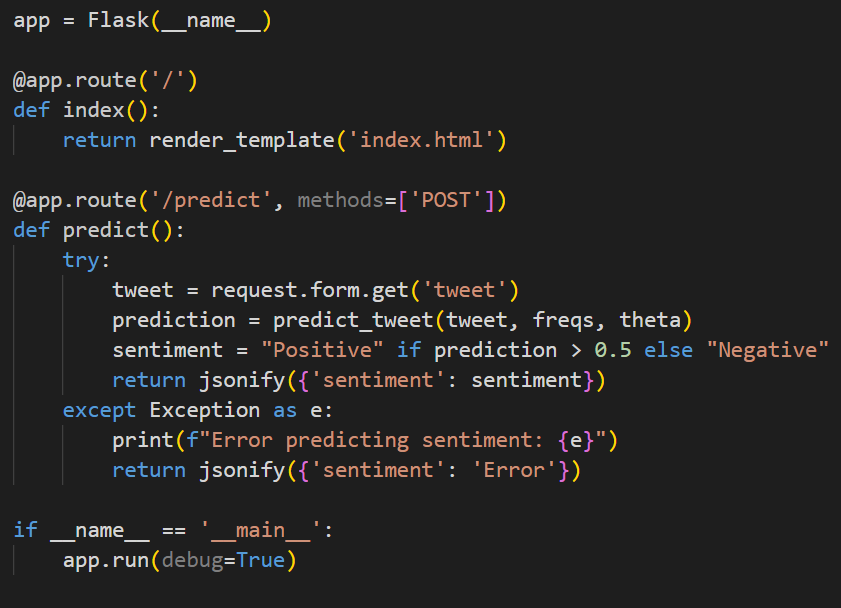
* Test the logistic regression model on the test set to evaluate its performance.
* Calculate the accuracy of the model based on its predictions. It came out to be 99.5 %.



**Fig. 4.8 Model Testing**

**4.7 Integration with flask web App**

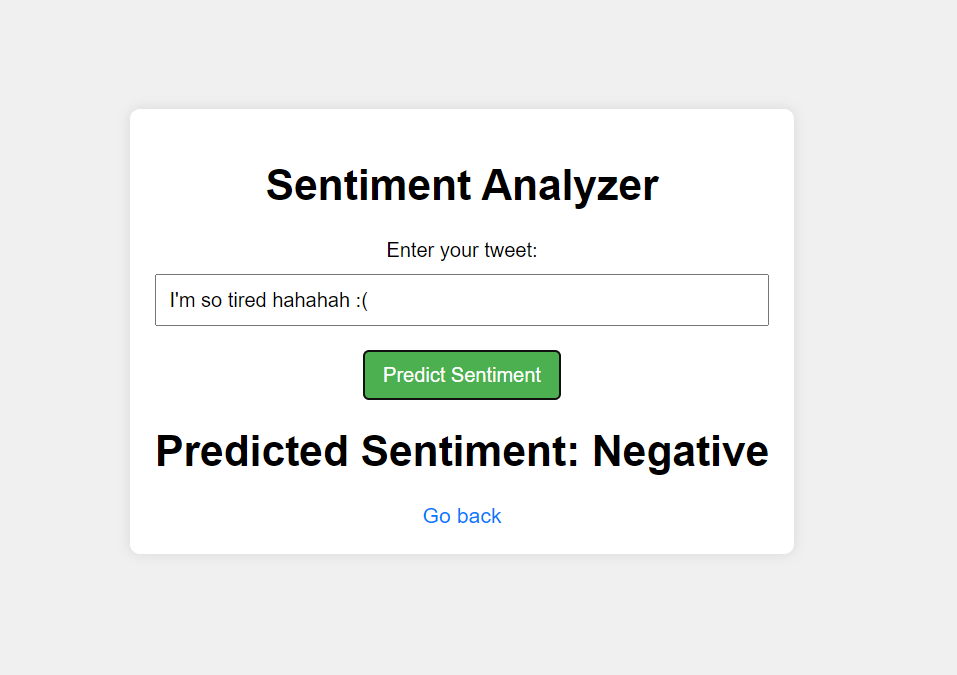
* Create a Flask web application with two routes ('/' and '/predict').
* Use HTML templates to render the user interface, allowing users to input a tweet and receive a predicted sentiment.
* Implement JavaScript to handle form submission, send the tweet to the server, and update the UI with the predicted sentiment.

****

**Fig. 4.9 Integration with flask web app**

**4.8 Model Deployment**

* Run the Flask application, allowing users to access the sentiment prediction functionality through a web interface.
* Users can input a tweet, click the "Predict Sentiment" button, and receive the predicted sentiment.

****

**Fig. 4.10 Model Deployment**

**Chapter 5**

**Conclusion and Future Work**

**5.1 Conclusion**

In conclusion, the presented Twitter Sentiment Analysis project successfully utilized a logistic regression model with count-vectorized features to classify tweets into positive or negative sentiments. The preprocessing steps, including lowercasing, removal of stopwords, and lemmatization, played a crucial role in enhancing the model's performance. The project achieved a commendable accuracy ratio, demonstrating the effectiveness of the chosen approach.

However, it's essential to acknowledge that sentiment analysis is a dynamic field, and the model's performance may vary based on the evolving nature of language on social media platforms. Despite the success of the current model, continuous monitoring and periodic retraining may be necessary to maintain optimal performance in the face of changing trends and language usage.

**5.2 Future Work**

* **Enhanced Model Exploration:** Explore more advanced machine learning models, such as deep learning architectures (e.g., neural networks), to capture complex relationships in tweet data.
* **Embedding Techniques:** Experiment with word embeddings (e.g., Word2Vec, GloVe) for better representation of semantic relationships between words and potentially improving model performance.
* **Real-time Sentiment Analysis:** Develop a system for real-time sentiment analysis to keep up with the constantly changing landscape of social media conversations.
* **Multiclass Sentiment Analysis:** Extend the model to handle multiple sentiment classes (e.g., positive, negative, neutral) for a more nuanced understanding of tweets.
* **User Interface Development:** Create a user-friendly interface or application to make the sentiment analysis tool accessible to a broader audience.
* **Social Media Platform Integration:** Explore integration with specific social media platforms' APIs to analyze and visualize sentiments on a larger scale.

**References**

1. <https://www.coursera.org/in/articles/what-is-data-analysis-with-examples>
2. Abirami, A. M., and V. Gayathri. "A survey on sentiment analysis methods and approach." In *2016 Eighth International Conference on Advanced Computing (ICoAC)*, pp. 72-76. IEEE, 2017.
3. Eswari, T., P. Sampath, and S. J. P. C. S. Lavanya. "Predictive methodology for diabetic data analysis in big data." *Procedia Computer Science* 50 (2015): 203-208.
4. Ahuja, Shreya, and Gaurav Dubey. "Clustering and sentiment analysis on Twitter data." In *2017 2nd International Conference on Telecommunication and Networks (TEL-NET)*, pp. 1-5. IEEE, 2017.
5. El Rahman, Sahar A., Feddah Alhumaidi AlOtaibi, and Wejdan Abdullah AlShehri. "Sentiment analysis of twitter data." In *2019 international conference on computer and information sciences (ICCIS)*, pp. 1-4. IEEE, 2019.
6. Bravo-Marquez, Felipe, Marcelo Mendoza, and Barbara Poblete. "Meta-level sentiment models for big social data analysis." *Knowledge-based systems* 69 (2014): 86-99.
7. <https://www.techtarget.com/searchbusinessanalytics/definition/opinion-mining-sentiment-mining>
8. <https://www.youtube.com/>
9. <https://www.kaggle.com/datasets>