Twitter Sentiment Analysis

Submitted for

Natural Language Processing CBCA275/CBSC360

Submitted by:

Shuchita Sinh (E23BCAU0027)

Manasvi Bhati (E23BCAU0028)

Saatvik Atreya (E23BCAU0014)

Submitted to:

Dr. Sakshi Sharma

Jan-May 2025

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

INDEX

ABSTRACT

Sentiment analysis of social media platforms, particularly Twitter, has gained immense importance in understanding public opinion on various topics. This paper presents a research review on sentiment analysis, discussing previous works, methodologies, and approaches.

Furthermore, it provides an overview of the model used in this study, its architecture, dataset, and evaluation metrics. The implementation of a sentiment analysis model using machine learning and deep learning techniques is demonstrated, followed by a discussion of the results and conclusions drawn.

INTRODUCTION

Sentiment analysis, also known as opinion mining, refers to the use of natural language processing (NLP), text analysis, and computational linguistics to identify and extract subjective information from text. With Twitter's rapid growth as a microblogging platform, researchers have leveraged its data to analyse public sentiments toward different topics, including politics, product reviews, and social issues.

RELATED WORK

Numerous studies have been conducted in the domain of Twitter sentiment analysis:

- Pak and Paroubek (2010) implemented sentiment classification using a Naïve Bayes classifier trained on Twitter data.
- Go et al. (2009) introduced distant supervision by using emoticons as labels for training data in sentiment classification.
- Saif et al. (2012) incorporated semantic features using ontologies for improved sentiment detection.

- Recent deep learning models, such as those proposed by Severyn and Moschitti (2015), have demonstrated superior performance using Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).

METHODOLOGY

1. **Data Collection**
   - Collected Twitter data (from a pre-downloaded dataset or through Twitter API).
   - Dataset includes tweet text and sentiment labels (positive, negative, neutral).

2. **Data Preprocessing**

   Before we input the tweets into the model, we preprocess them:
   - Strip punctuation, URLs, mentions, hashtags, and emojis.
   - Convert text to lowercase (standardization).
   - Remove stopwords (such as "the", "is", "and").
   - Tokenization: Split tweets into individual words.
   - Stemming/Lemmatization: Normalize words to their root form (e.g., "running" → "run").

3. **Feature Extraction (Text Vectorization)**
   - Machine learning algorithms can't read text, so we transform it into numbers.
   - Utilized TF-IDF Vectorizer to transform text into numerical format (feature vectors).

4. **Model Training**

- Trained machine learning models using the cleaned and vectorized tweets.
- Models used:
  - Logistic Regression
  - Naïve Bayes

5. **Evaluation:** Accuracy, Precision, Recall, F1-Score

6. **Deployment:** Streamlit web application hosted via Streamlit Cloud

HARDWARE/SOFTWARE REQUIRED

Hardware: Any modern laptop/PC with internet access

Software:

- Python (Google Colab)
- NLTK / SpaCy / Scikit-learn
- Streamlit
- GitHub
- Web browser

EXPERIMENTAL RESULTS

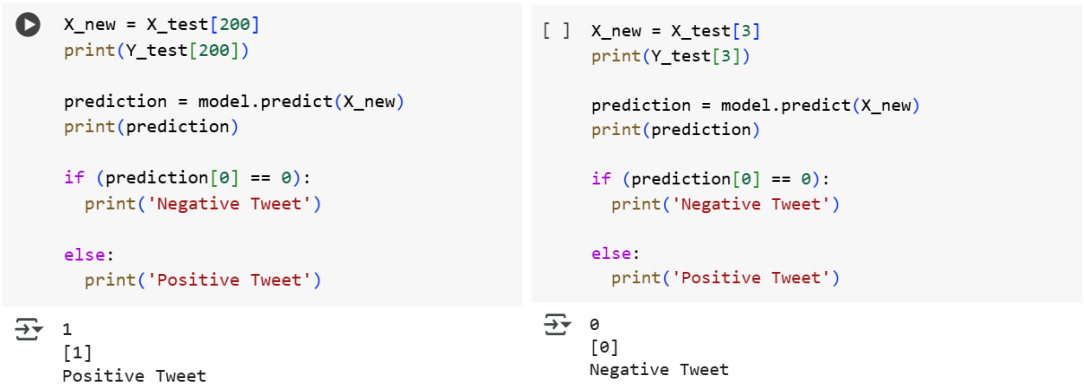
Model Accuracy: ~77% on test data

Sample Inputs:

- "I love this product!" → Positive

- “This is terrible.” → Negative

Real-time sentiment predictions via web UI



```
X_new = X_test[200]
print(Y_test[200])

prediction = model.predict(X_new)
print(prediction)

if (prediction[0] == 0):
  print('Negative Tweet')

else:
  print('Positive Tweet')
```

```
1
[1]
Positive Tweet
```

```
[ ] X_new = X_test[3]
print(Y_test[3])

prediction = model.predict(X_new)
print(prediction)

if (prediction[0] == 0):
  print('Negative Tweet')

else:
  print('Positive Tweet')
```

```
0
[0]
Negative Tweet
```

CONCLUSION

Twitter sentiment analysis is an evolving field with continuous advancements in NLP techniques. The proposed model demonstrates promising results by leveraging deep learning architectures. Future work may focus on multilingual sentiment analysis and real-time implementation for dynamic Twitter streams.

FUTURE SCOPE

- Use live tweet scraping through the Twitter API
- Apply complex models such as BERT for better accuracy
- Include multilingual sentiment analysis
- Enhance data visualization within the web application

GITHUB LINK

https://github.com/Manasvibhati/twitter-sentiment-analysis