Machine Learning 2024

Mini Project 2

Sentiment Analysis with the sentiment140 dataset

Project Report

Author: Halidu Abdulai

Student ID: 2301998

Contents

1 Introduction

In the realm of computational linguistics and natural language processing, sentiment analysis stands as a pivotal methodology for discerning subjective information within text data. This project is rooted in the exploration of sentiment analysis through the lens of machine learning, targeting the micro-blogging platform Twitter as a case study. The advent of social media has revolutionized the way individuals express opinions, emotions, and reactions to a myriad of topics, rendering platforms like Twitter rich repositories of real-time public sentiment. The Sentiment140 dataset, comprising 1.6 million tweets annotated for sentiment (of which only 10% was utilized for this project), serves as the empirical foundation for this analysis, offering a robust basis for model training and validation.

The objective of this project is two-fold: firstly, to implement Logistic Regression—a statistical model known for its efficacy in binary classification tasks—as a means to establish a baseline for sentiment analysis on Twitter data. This model's simplicity and interpretability make it an ideal candidate for initial explorations into sentiment classification, providing a benchmark against which more complex models can be measured. Secondly, the project aims to delve into the application of Long Short-Term Memory (LSTM) networks, a form of recurrent neural network specifically designed to address the challenges of sequence prediction problems. LSTMs are particularly adept at capturing long-range dependencies in text data, a characteristic essential for understanding the nuanced and context-dependent nature of sentiment in language.

This investigation is underpinned by a comprehensive methodology that encompasses data preprocessing, feature engineering, model training, and evaluation. A particular emphasis is placed on the preprocessing phase, recognizing the unique challenges posed by the informal and often idiosyncratic language used on Twitter. Techniques such as tokenization, stopword removal, and the application of word embeddings are employed to transform raw tweets into a format amenable to machine learning models.

The comparative analysis of Logistic Regression and LSTM networks aims to illuminate the strengths and limitations of each approach within the context of sentiment analysis. This includes an examination of model accuracy, precision, recall, and F1 scores, alongside a discussion on the interpretability and computational efficiency of each model. The study's broader implications for the field of natural language processing are also considered, with reflections on the potential for machine learning to enhance our understanding of public sentiment and its manifestations on social media platforms.

In summary, this project represents a nuanced inquiry into the application of machine learning techniques for sentiment analysis on Twitter, contributing to the ongoing discourse in the fields of artificial intelligence, computational linguistics, and social media analytics. Through this work, I aim to advance our comprehension of how machine learning can be harnessed to extract meaningful insights from the vast, unstructured datasets characteristic of social media, thereby bridging the gap between quantitative analysis and qualitative human expression.

2 Data Preprocessing

In the preprocessing phase of the analysis on the Sentiment140 dataset, I aimed to refine the raw tweet data into a format amenable to machine learning algorithms. This process involved several key steps:

1. **Cleaning Tweets:** Initially, I removed unnecessary noise from the tweets, including URLs, mentions of usernames (preceded by @), hashtags, and non-textual elements like emojis and special characters. This step was crucial to focus the analysis on the textual content, eliminating elements that could skew the interpretation of sentiment.
2. **Normalizing Text:** All texts were converted to lowercase to ensure uniformity across the dataset, mitigating the impact of case sensitivity on the analysis. Furthermore, I employed the removal of single character words, multiple spaces etc., standardizing the language for more accurate processing.
3. **Tokenizing Text:** The tweets were then broken down into individual components or tokens. This tokenization facilitated the analysis of word frequency and the application of word embeddings, preparing the data for the subsequent machine learning models.

Following these preprocessing steps, I generated word clouds for positively and negatively annotated tweets separately. The word clouds visually represented the most frequent words within each sentiment category, offering preliminary insights into the thematic distinctions between positive and negative sentiments in the dataset. This visual analysis served not only as a tool for exploratory data analysis but also as a foundation for the deeper sentiment analysis performed in later stages of the project.

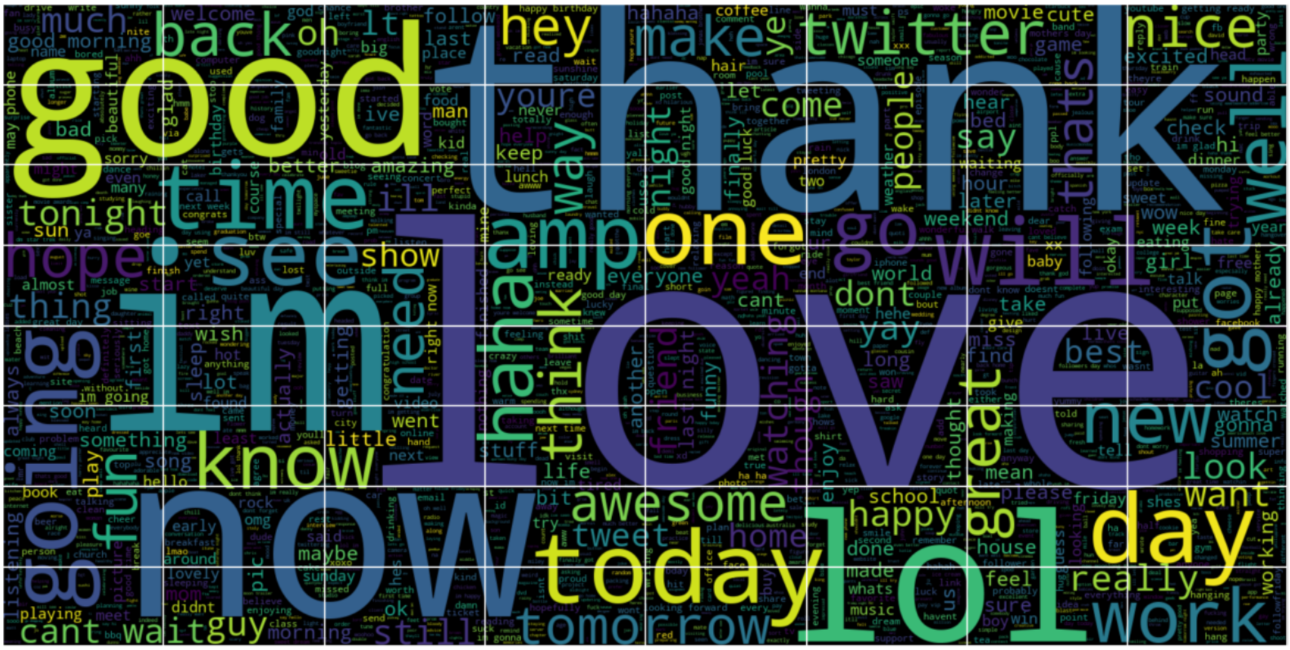


Figure 1: Word Cloud for the 'positive' annotated sentiments

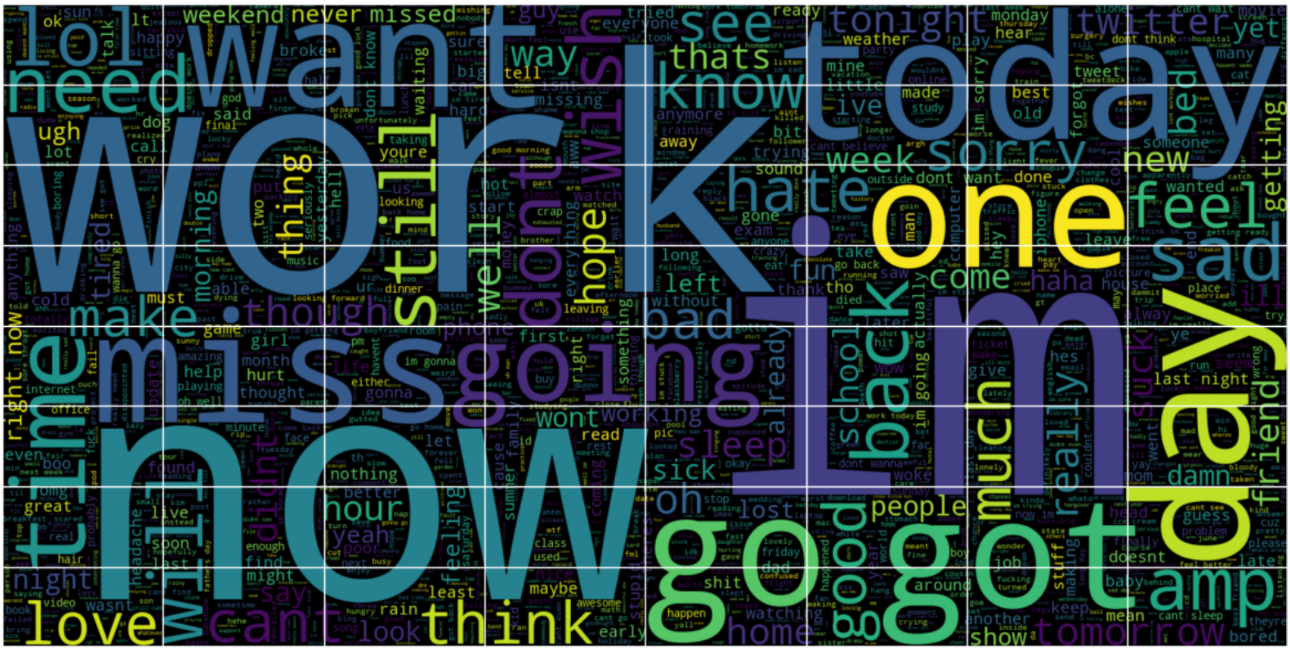


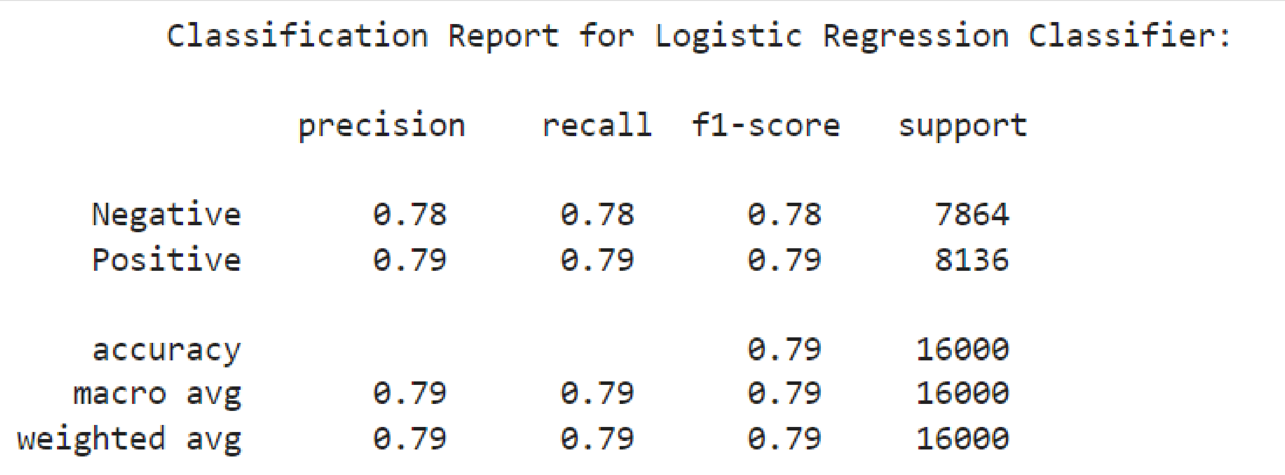
Figure 2: Word Cloud for the 'negative' annotated sentiments

3 Modelling

In the modeling phase of this project, I explored two distinct approaches to sentiment analysis: Logistic Regression using TF-IDF features and an LSTM model leveraging GloVe pretrained word embeddings.

3.1 Logistic Regression Classifier

The Logistic Regression model was implemented using a Count Vectorizer and a TFIDF Transformer. This approach transforms the tweets into a matrix of token counts followed by a weighting of these tokens to reflect their importance in the entire dataset. This model achieved an accuracy of 79%, demonstrating its capability to capture the linear relationships between word frequencies and tweet sentiments.

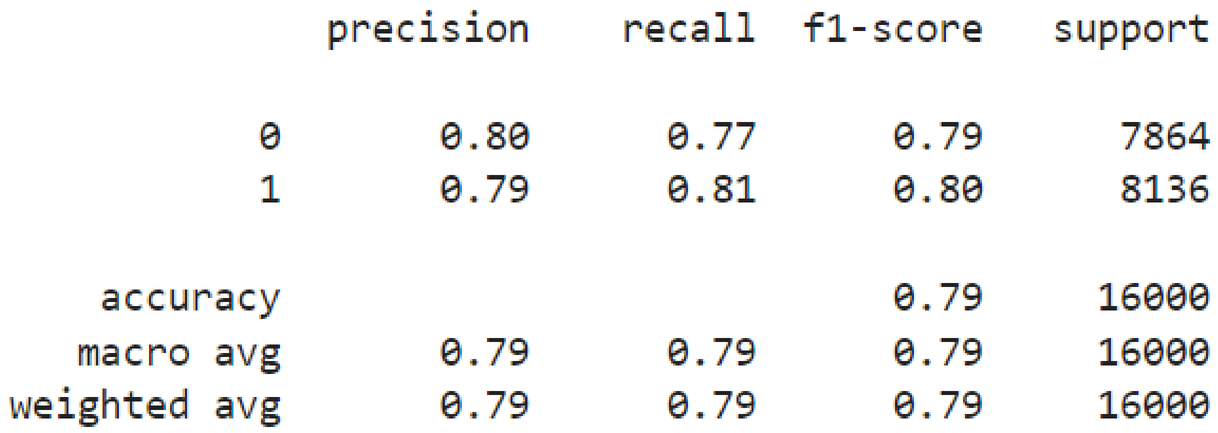


```
        Classification Report for Logistic Regression Classifier:

               precision    recall  f1-score   support

    Negative       0.78      0.78      0.78      7864
    Positive       0.79      0.79      0.79      8136

    accuracy                           0.79     16000
   macro avg       0.79      0.79      0.79     16000
weighted avg       0.79      0.79      0.79     16000
```

Figure 3: Performance of the Logistic Regression Classifier

3.2 LSTM Classifier

Conversely, the LSTM model was developed using GloVe's 6B.300d word embeddings to capture the contextual nuances of words in tweets. Despite the LSTM's advanced capability to understand sequential data and the richer representation of text through GloVe embeddings, it also achieved a 79% accuracy, on par with the Logistic Regression model.



```
              precision    recall  f1-score   support

           0       0.80      0.77      0.79      7864
           1       0.79      0.81      0.80      8136

    accuracy                           0.79     16000
   macro avg       0.79      0.79      0.79     16000
weighted avg       0.79      0.79      0.79     16000
```

Figure 4: Performance of the LSTM model

3.3 Comparing the performance of the two models

The comparable performance of these models suggests that while LSTM has the potential to capture complex patterns in sequential data, the relatively straightforward nature of the sentiment analysis task on the Sentiment140 dataset might not fully leverage LSTM's capabilities. Additionally, the pretrained GloVe embeddings, although powerful, may not perfectly align with the informal language and shorthand commonly found in tweets.

Improvements could include more advanced preprocessing techniques, hyperparameter tuning, and exploring model architectures for the LSTM. For the Logistic Regression model, experimenting with different n-grams and adjusting the TF-IDF parameters could potentially enhance its performance. Further exploration into hybrid models or ensemble methods might also offer a path to improved accuracy and insight into the sentiment analysis task.

4 Conclusion and Reflections

In concluding this project on sentiment analysis using Logistic Regression and LSTM models, several scientific challenges were encountered, primarily related to data preprocessing and model optimization. The informal and dynamic nature of language on Twitter posed significant preprocessing challenges, which were addressed through meticulous cleaning and normalization of the text data. Model optimization was another bottleneck, especially for the LSTM model, which required fine-tuning of hyperparameters to balance the trade-off between model complexity and overfitting. These challenges were overcome by employing rigorous cross-validation techniques and leveraging domain-specific adjustments to the preprocessing pipeline.

Reflecting on the project, the experience underscored the importance of thorough data preparation and the nuanced application of machine learning models to natural language processing tasks. Future work could explore more sophisticated models, alternative embedding techniques, and larger datasets to further refine the analysis and potentially improve accuracy beyond the current benchmark.