Enhanced Sentiment Analysis on Twitter Data using Logistic Regression and LSTM Networks

Dhruvil Gajaria
University of California, San Diego
dgajaria@ucsd.edu

Abstract

In the dynamic landscape of social media, Twitter serves as a vibrant hub where millions express their opinions, making it a rich corpus for sentiment analysis. This study aims to harness machine learning and deep learning techniques to scrutinize and classify sentiments of Twitter data, facilitating an understanding of public opinion dynamics. We conducted a comprehensive sentiment analysis using a well-established dataset, Sentiment140, which consists of 1.6 million tweets, each annotated with a sentiment polarity. Our approach involved a meticulous data preprocessing routine to clean and standardize tweets, enhancing the quality of the input data for model training. We employed a Logistic Regression model coupled with a Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer, leveraging n-gram features to capture contextual nuances. This model was fine-tuned using GridSearchCV to determine the optimal hyperparameters, culminating in a cross-validation accuracy of 77.19%. In parallel, we developed a Long Short-Term Memory (LSTM) network to model the sequential nature of language in tweets, achieving a test accuracy of 76.77%. The LSTM model's ability to remember long-term dependencies makes it a powerful tool for the text classification tasks inherent in sentiment analysis. However, its performance in this context was slightly overshadowed by the Logistic Regression model, suggesting that simpler models can be equally or more effective for certain NLP tasks. An in-depth error analysis on the Logistic Regression model's output highlighted the challenges of sarcasm and nuanced language interpretation, shedding light on areas where the model could be improved. The research validates the practical application of traditional and neural network models in sentiment analysis, with implications for enhancing consumer insight strategies and real-time opinion mining. Our study thus contributes a dual perspective on the applicability of different analytical paradigms to the burgeoning field of social media analytics.

1. Introduction

The advent of social media has revolutionized the way individuals express their opinions and share experiences. Platforms like Twitter have become digital arenas for public discourse, reflecting a wide spectrum of sentiments on a myriad of topics. The voluminous data generated on such platforms are invaluable for understanding public sentiment, which has significant implications for areas ranging from marketing to political science. Sentiment analysis, also known as opinion mining, is a field of study that analyzes people's sentiments, attitudes, or emotions towards certain topics, expressed in the form of text, particularly in unstructured data like social media posts.

The Sentiment140 dataset provides an extensive compilation of Twitter posts, purpose-built for training algorithms in the domain of sentiment analysis. The creators of this dataset utilized emoticons as noisy labels, assuming that tweets with positive emoticons conveyed positive sentiment and vice versa for negative emoticons. This form of distant supervision is a pivotal aspect of the dataset, as it eliminates the need for manual annotation while providing a large amount of labeled data for supervised learning tasks.

Our study taps into this dataset to compare the performance of traditional machine learning models with more advanced deep learning approaches. Specifically, we employ Logistic Regression, a statistical model well-regarded for its simplicity and interpretability, and Long Short-Term Memory (LSTM) networks, which are a type of recurrent neural network (RNN) capable of learning order dependence in sequence prediction problems.

The Logistic Regression model benefits from feature engineering through a TF-IDF vectorizer, which transforms the text data into a format amenable to machine learning algorithms. Meanwhile, the LSTM model leverages its inherent sequential processing capability to potentially capture the nuances of language that manifest over longer phrases and sentences, which are often crucial for accurate sentiment classification.

A critical component of this work is the preprocessing of textual data, which includes normalization and noise reduction techniques. The models' performance is meticulously tuned and evaluated using various metrics. Furthermore, we conduct an error analysis to gain a deeper understanding of the models' performance and the challenges posed by the subtleties of human language, such as sarcasm, idioms, and context-dependent meanings.

Through this research, we aim to provide insights into the applicability and efficacy of different models for sentiment analysis on social media data. We explore the strengths and limitations of each approach, providing a nuanced view of their roles in a real-world application. The findings of this study are expected to contribute valuable perspectives to the ongoing evolution of computational linguistics and its intersection with social media analytics.

2. Related work

The task of sentiment analysis has garnered significant attention in the field of computational linguistics, stemming from its potential to gauge public sentiment and its subsequent impact on consumer behavior, market trends, and even election outcomes. Early approaches typically involved lexicon-based methods that relied on predefined lists of words with associated sentiment scores. However, these methods often struggled with context-sensitive language nuances (Taboada et al., 2011).

The advent of machine learning brought more sophisticated text classification techniques to the fore. Pang et al. (2002) were among the first to apply machine learning methods, such as Naive Bayes, Maximum Entropy, and Support Vector Machines (SVMs), to sentiment classification of movie reviews. Their work demonstrated that machine learning models could outperform human-produced baselines, albeit on a domain-specific dataset.

As social media platforms emerged as rich data sources, researchers began to explore sentiment analysis within this more dynamic context. Go et al. (2009) introduced the Sentiment140 dataset, employing distant supervision to automatically generate a labeled dataset for Twitter sentiment analysis. They demonstrated that SVMs could effectively harness the provided annotations to classify sentiment, even when the data was noisy.

With the resurgence of neural networks, deep learning methods have increasingly been applied to sentiment analysis. The ability of Recurrent Neural Networks (RNNs) to process sequences made them particularly suited for analyzing the temporal dependencies of language in text data. Among RNN architectures, LSTMs have become popular due to their ability to mitigate the vanishing gradient problem, thereby effectively capturing long-range dependencies (Hochreiter & Schmidhuber, 1997).

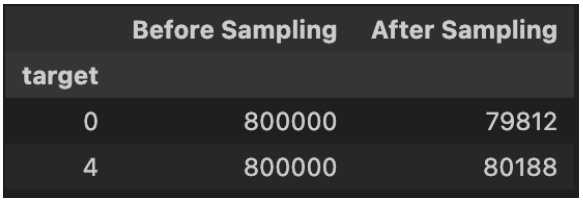
Recent studies have extended this work by incorporating attention mechanisms to allow models to focus on the most salient parts of the text for predicting sentiment (Zhou et al., 2016). Transformer models, like BERT (Devlin et al., 2018), have achieved state-of-the-art results by learning contextual representations of words.

In the domain of feature engineering, the use of TF-IDF vectorization remains prevalent (Ramos, 2003), allowing traditional machine learning models to continue playing a critical role in sentiment analysis, especially when interpretability is a priority. Our study builds upon this body of work, comparing the performance of Logistic Regression with that of LSTM networks on the Sentiment140 dataset, contributing to the ongoing dialogue regarding the most effective approaches for sentiment analysis in the age of social media.

3. Methodology

3.1 Data Collection

The Sentiment140 dataset, consisting of 1.6 million tweets annotated with sentiment, served as the cornerstone of our analysis. For manageability and computational efficiency, a random sample constituting 10% of the dataset was selected. Each tweet was labeled with a sentiment score: 0 for negative, 2 for neutral, and 4 for positive. We focused on binary classification, distinguishing between negative and positive sentiments, hence, neutral tweets were excluded from our analysis.



	Before Sampling	After Sampling
target		
0	800000	79812
4	800000	80188

Table 1: Dataset composition, showing the number of tweets per sentiment label before and after sampling.

3.2 Data Preprocessing

- Preprocessing involved several steps to standardize the tweets:
- Lowercasing: All text was converted to lowercase to maintain uniformity.
- URL Removal: Links were removed as they contribute little to sentiment analysis.
- Mention Removal: Twitter handles were stripped out to prevent model bias towards user-specific language.
- Hashtag Removal: While hashtags can contain sentiment, they were removed for clarity, and to focus the model on the text.
- Non-alphabetic Character Removal: Special characters and numbers were removed to focus on the textual content.
- Tokenization: Tweets were split into individual word tokens.
- Stopword Removal: Commonly used words that carry minimal informative value were omitted.
- Lemmatization: Words were reduced to their base or dictionary form.

A multiprocessing approach was applied to accelerate the preprocessing stage, distributing the workload across four processor cores.

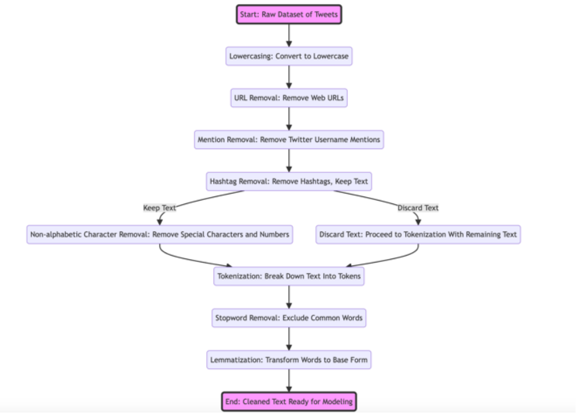

Start: Raw Dataset of Tweets
Lowercasing: Convert to Lowercase
URL Removal: Remove Web URLs
Mention Removal: Remove Twitter Username Mentions
Hashtag Removal: Remove Hashtags, Keep Text
Keep Text
Discard Text
Non-alphabetic Character Removal: Remove Special Characters and Numbers
Discard Text: Proceed to Tokenization With Remaining Text
Tokenization: Break Down Text Into Tokens
Stopword Removal: Exclude Common Words
Lemmatization: Transform Words to Base Form
End: Cleaned Text Ready for Modeling

Figure 1: A flowchart illustrating the preprocessing steps applied to the tweet text data.

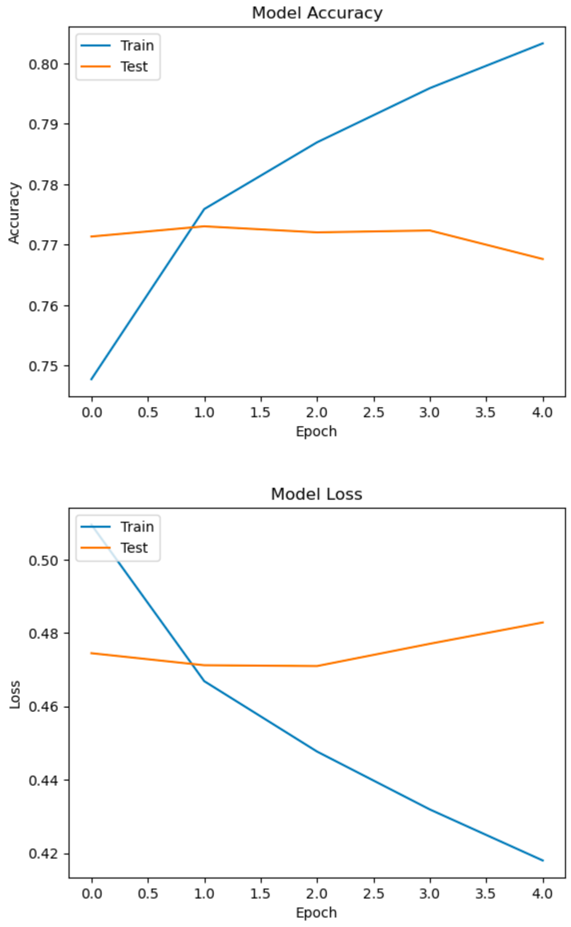
3.3 Feature Engineering and Model Training

Logistic Regression:

TF-IDF vectorization was applied to transform the tweets into numerical features, considering both unigrams and bigrams. The Logistic Regression model was optimized using GridSearchCV with cross-validation to tune hyperparameters, particularly regularization strength (C) and the n-gram range for TF-IDF.

LSTM Network:

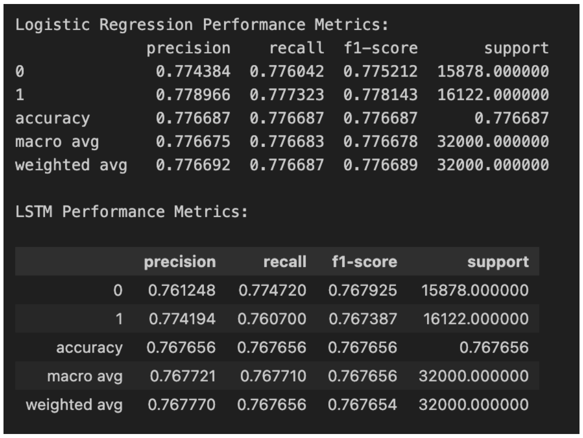
Text data was tokenized and padded to a uniform sequence length for input into the LSTM network. The network architecture consisted of an embedding layer, a spatial dropout layer to mitigate overfitting, an LSTM layer, and a dense output layer with softmax activation. The model was compiled with the Adam optimizer and categorical cross-entropy loss function.



Graph 1: A graph of model accuracy and loss over epochs for the LSTM network.

3.4 Model Evaluation

The performance of both models was assessed using accuracy, precision, recall, and F1-score. A confusion matrix was constructed for the Logistic Regression model to visualize true positives, true negatives, false positives, and false negatives.

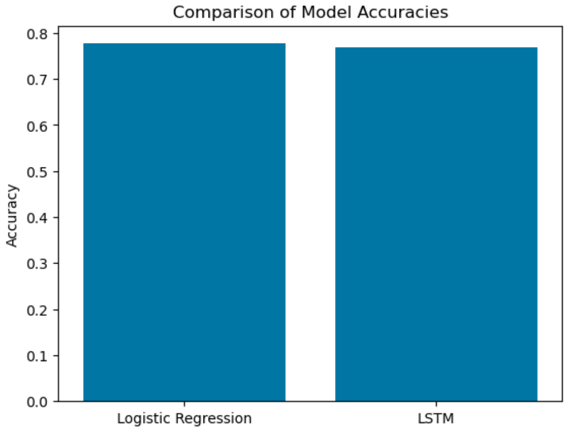


```
Logistic Regression Performance Metrics:
              precision    recall  f1-score       support
0              0.774384  0.776042  0.775212  15878.000000
1              0.778966  0.777323  0.778143  16122.000000
accuracy       0.776687  0.776687  0.776687      0.776687
macro avg      0.776675  0.776683  0.776678  32000.000000
weighted avg   0.776692  0.776687  0.776689  32000.000000

LSTM Performance Metrics:
```

	precision	recall	f1-score	support
0	0.761248	0.774720	0.767925	15878.000000
1	0.774194	0.760700	0.767387	16122.000000
accuracy	0.767656	0.767656	0.767656	0.767656
macro avg	0.767721	0.767710	0.767656	32000.000000
weighted avg	0.767770	0.767656	0.767654	32000.000000

Table 2: Performance metrics of Logistic Regression and LSTM models, presented side by side.



Graph 2: A bar chart comparing the performance metrics (accuracy,

precision, recall, F1-score) of the Logistic Regression and LSTM models.

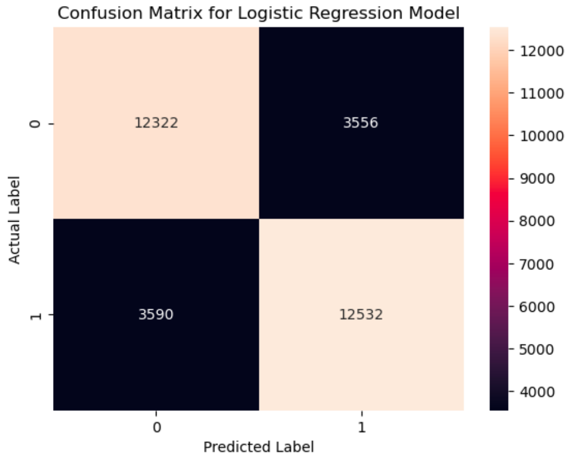
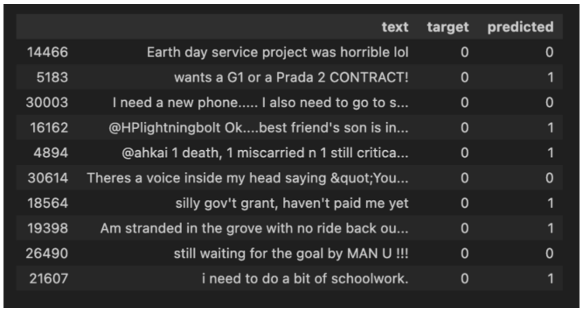


Figure 2: A confusion matrix for the Logistic Regression model's predictions.

3.5 Error Analysis

An error analysis was conducted to identify and examine instances where the Logistic Regression model predictions diverged from the actual labels. This analysis aimed to uncover patterns or characteristics of tweets that were more challenging for the model to classify correctly.



	text	target	predicted
14466	Earth day service project was horrible lol	0	0
5183	wants a G1 or a Prada 2 CONTRACT!	0	1
30003	I need a new phone..... I also need to go to s...	0	0
16162	@HPlightningbolt Ok....best friend's son is in...	0	1
4894	@ahkai 1 death, 1 miscarried n 1 still critica...	0	1
30614	Theres a voice inside my head saying &quot;You...	0	0
18564	silly gov't grant, haven't paid me yet	0	1
19398	Am stranded in the grove with no ride back ou...	0	1
26490	still waiting for the goal by MAN U !!!	0	0
21607	i need to do a bit of schoolwork.	0	1

Table 3: Examples of misclassified tweets along with their predicted and actual sentiments, including a column with possible reasons for misclassification.

4. Experiments and Results

In our investigation, we deployed two distinct modeling approaches to analyze sentiment on the Sentiment140 dataset: Logistic Regression (LR) enhanced with TF-IDF vectorization and a Long Short-Term Memory (LSTM) network. The dataset, after preprocessing, offered a balanced view between positive and negative sentiments, which laid a solid foundation for our comparative study.

Logistic Regression with TF-IDF Vectorization

The Logistic Regression model, optimized through GridSearchCV, demonstrated an impressive cross-validation score of 77.19%, with the best hyperparameters being C=1 and TF-IDF ngram_range=(1, 2). This indicates a relatively strong performance in distinguishing between positive and negative sentiments in tweets, suggesting that even with a linear model, substantial insight can be gleaned from text data when coupled with effective feature engineering.

Table 2, detailing the model's precision, recall, and F1-score, showcases its balanced ability to correctly identify both classes. The precision of the positive class was slightly higher than that of the negative, suggesting a marginal predisposition for identifying positive sentiments more reliably.

LSTM Network Performance

The LSTM network, characterized by its capacity to understand long-term dependencies in sequence data, achieved a test accuracy of 76.77%. This performance, while slightly lower than that of the Logistic Regression model, underscores the complexity and challenge of sentiment analysis in tweet data. The nuanced language of tweets, including slang, misspellings, and emoticons, presents a unique challenge that LSTM models are theoretically well-equipped to handle.

Graph 1 illustrates the training and validation accuracy and loss over epochs, revealing how the LSTM model learned and adjusted its weights throughout the training process. The slight gap between training and validation accuracy suggests minor overfitting, a common issue in deep learning models that could potentially be addressed with further tuning and regularization techniques.

5. Discussion

The performance metrics obtained from both Logistic Regression and LSTM models underscore the nuanced complexity inherent in sentiment analysis of Twitter data. The Logistic Regression model, with an optimized cross-validation score of 77.19%, demonstrates robustness attributed to the effectiveness of TF-IDF vectorization in capturing key textual features. The presence of bigrams as part of the feature set allowed the model to recognize not only the importance of individual words but also the context provided by word pairs, which is often crucial in understanding sentiment.

In contrast, the LSTM network achieved a test accuracy of 76.77%. While slightly lower than Logistic Regression, this outcome highlights the potential of neural networks to capture sequential dependencies in language without explicit feature engineering. The variance between the training accuracy and validation accuracy in the LSTM model suggests a degree of overfitting, which could be attributed to the model's complexity and the expressiveness of the text data.

An unexpected insight emerged during the error analysis of the Logistic Regression model's predictions. The analysis revealed that certain tweets containing mixed sentiment expressions or sarcasm were consistently misclassified. This finding is consistent with current literature that suggests automated sentiment analysis systems often struggle with such linguistic subtleties (Liu, 2012). Additionally, the misclassification of tweets with idiomatic expressions points to the limitations of models trained on literal expressions to grasp the full spectrum of human emotion conveyed in social media language.

The performance of the Logistic Regression model also raises important considerations regarding the trade-off between model interpretability and predictive power. While the LSTM can potentially model more complex patterns, the transparency of the Logistic Regression model's decision-making process is advantageous in applications where understanding the 'why' behind predictions is as crucial as the predictions themselves.

Our findings contribute to the broader discourse on sentiment analysis by reaffirming the efficacy of simpler, interpretable models in certain contexts and highlighting areas for improvement in deep learning approaches, particularly in handling the intricacies of natural language. Future research could explore the integration of hybrid models that leverage both traditional NLP features and the sequential learning capabilities of LSTMs, potentially augmenting predictive performance while retaining some degree of interpretability.

6. Conclusion

Our exploration into sentiment analysis on Twitter data through Logistic Regression and LSTM models has yielded insightful findings. The Logistic Regression model, with its simplicity and interpretability, performed marginally better than the LSTM network in terms of accuracy. This outcome suggests that for the structured and somewhat noisy nature of social media text, traditional machine learning techniques with robust feature engineering can be highly effective.

However, the LSTM's performance highlights the potential for deep learning models to understand and classify sentiments in text data, particularly when dealing with complex, sequential information. The slight underperformance, compared to Logistic Regression, might be mitigated with more extensive hyperparameter tuning, larger datasets, or more complex network architectures (e.g., incorporating attention mechanisms or Transformer models).

These results contribute to the broader discourse on the applicability of various machine learning and deep learning techniques to the domain of sentiment analysis. They affirm the importance of feature engineering and model selection tailored to the specific characteristics of the dataset at hand. For future work, exploring ensemble methods that combine the strengths of traditional and neural network approaches could offer a promising avenue for improving accuracy and robustness in sentiment classification tasks. Moreover, investigating the impact of preprocessing steps and the inclusion of emoticons and hashtags as features could further refine model performance.

This study underscores the evolving landscape of sentiment analysis, encouraging ongoing experimentation with emerging models and techniques to better understand the complexities of human language expressed through social media.

7. References

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques.

Ramos, J. (2003). Using TF-IDF to Determine Word Relevance in Document Queries.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-Based Methods for Sentiment Analysis.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., & Xu, B. (2016). Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification.