Report on Sentiment Analysis of IMDB Reviews Using Naive Bayes and Logistic Regression

Advanced Visual Text Analytics

Merve Pakcan Tufenk

Introduction

With the rapid growth of user-generated content on the internet, understanding public opinion has become more important than ever. Among various forms of online feedback, movie reviews serve as a rich source of data, offering valuable insights into audience preferences and reactions. Sentiment analysis, a subfield of natural language processing (NLP), focuses on automatically detecting and classifying emotions or opinions expressed in text. In this context, the IMDB movie review dataset is a widely used benchmark for evaluating the effectiveness of sentiment classification models.

This report presents a comparative study of two machine learning algorithms Naive Bayes and Logistic Regression applied to the task of classifying IMDB reviews as either positive or negative. While both models are well-established in the field of text classification, their performance can differ based on how they process and learn from textual patterns. The goal is to highlight their strengths, limitations, and overall effectiveness in sentiment analysis.

Dataset Description

This study utilizes the "IMDB Dataset of 50K Movie Reviews", available on Kaggle. The dataset contains 50,000 movie reviews, each labeled with a sentiment value: positive or negative. It is structured into two columns: review (the text of the movie review) and sentiment (the corresponding sentiment label). The dataset is equally divided into 25,000 training and 25,000 testing samples, with an even distribution of sentiments. The reviews are highly polarized, making the dataset ideal for evaluating sentiment classification models. During preprocessing, one problematic row was identified and removed to ensure data quality.

The dataset was originally introduced as part of the Large Movie Review Dataset v1.0, created by Andrew L. Maas et al., and was first published in the ACL 2011 paper Learning Word Vectors for Sentiment Analysis.It is a balanced and trusted dataset commonly used in sentiment analysis research.

Train-Test Split and Label Encoding

To prepare the dataset for model training, the data was randomly shuffled and split into 80% training and 20% testing sets. Sentiment labels were encoded as binary values, assigning 1 to positive reviews and 0 to negative ones. This encoding aligns with the binary classification objective and simplifies the evaluation process using metrics such as accuracy. The resulting shapes of the training and testing sets confirm a clean and balanced split. These data preparation steps were applied identically for both models to ensure a fair comparison.

After the split, the training and testing sets had the following shapes:
Train X: (39,999,) **Test X:** (10,000,)
Train Y: (39,999, 1) **Test Y:** (10,000, 1)
These dimensions confirm that the data was properly formatted for input into the models.

Text Preprocessing

To ensure the text data was clean and consistent for model training, a comprehensive preprocessing pipeline was applied. This included converting all text to lowercase, removing hyperlinks, HTML tags, and punctuation marks. Tokenization was performed using NLTK's TweetTokenizer, which efficiently handles informal language. Stopwords were then removed, and stemming was applied using the PorterStemmer to reduce words to their root forms. These preprocessing steps were consistently applied across both the Naive Bayes and Logistic Regression models to maintain fairness in evaluation.

Furthermore, a frequency dictionary was created by pairing each word in the preprocessed reviews with its corresponding sentiment label. The number of occurrences for each (word, label) pair was counted to help capture the association between specific words and sentiment classes.

Model descriptions:

Naive Bayes is a simple probabilistic model that uses word frequencies to estimate the likelihood of a review being positive or negative, assuming independence between words.
Logistic Regression is a linear model that calculates sentiment probability based on a weighted combination of all input features, without assuming feature independence.

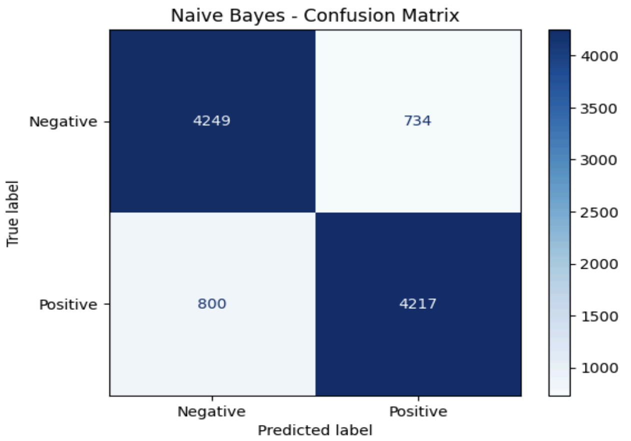
Results and Evaluation

The Naive Bayes model reached 84.66% accuracy on the test set, which shows that it performed quite well in classifying the reviews correctly. I used a custom implementation where the model makes predictions based on word-level log probabilities. Even though the model assumes that words are independent from each other, it still worked well overall.When I tested it with neutral or mixed-tone reviews, it usually predicted them as negative. This might be because it focuses only on word frequencies and can’t really understand the full meaning of a sentence. So while it's very effective on clear examples, it may not always capture more complex or balanced meanings.

On the other hand, the Logistic Regression model had a lower test accuracy of 62.08%. However, its behavior was a bit different with mixed reviews. It sometimes predicted positive sentiment even when the sentence had both good and bad parts. This means it might be more influenced by strong positive words in a sentence. Since it doesn’t assume that words are independent, it can be more flexible — but it also needs more careful tuning and more data to perform well, especially with more complex inputs.

When comparing the two, Naive Bayes clearly performed better in this project, both in terms of accuracy and consistency. While Logistic Regression has more potential in more advanced data and tuning., Naive Bayes gave more reliable results for this specific task using a simple approach.

For fast and basic sentiment analysis tasks like this one, Naive Bayes still proves to be a very useful baseline model.

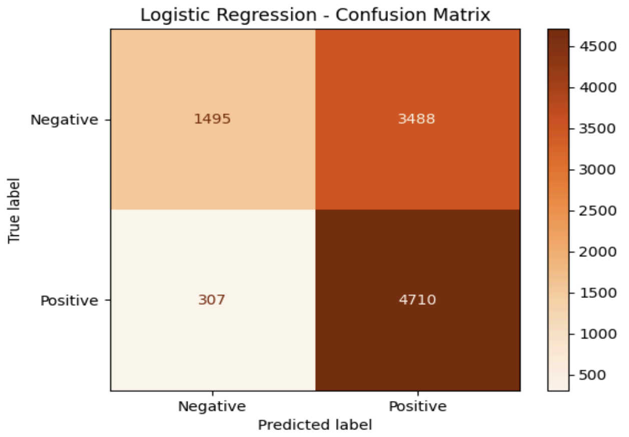


```
Naive Bayes - Classification Report
              precision    recall  f1-score   support

    Negative       0.84      0.85      0.85      4983
    Positive       0.85      0.84      0.85      5017

    accuracy                           0.85     10000
   macro avg       0.85      0.85      0.85     10000
weighted avg       0.85      0.85      0.85     10000
```

The Naive Bayes model achieved 85% accuracy, correctly classifying 4,249 negative and 4,217 positive reviews out of 10,000. Precision and recall scores were balanced at 0.85 for both classes which shows that the model works well despite being simple and assuming that words are independent.



```
Logistic Regression - Classification Report
              precision    recall  f1-score   support

    Negative       0.83      0.30      0.44      4983
    Positive       0.57      0.94      0.71      5017

    accuracy                           0.62     10000
   macro avg       0.70      0.62      0.58     10000
weighted avg       0.70      0.62      0.58     10000
```

The Logistic Regression model reached 62% accuracy, but it mostly predicted reviews as positive. It had high recall for the positive class (0.94), but did poorly on negative ones with only 0.30 recall.This suggests the model needs improvements in features or training to perform better.