Sentiment Analysis of Movie Reviews: A Comprehensive Machine Learning Study

Rashid Karimov

February 28, 2025

Abstract

This report presents a comprehensive machine learning study on classifying IMDB movie reviews as positive or negative, employing Logistic Regression, Long Short-Term Memory (LSTM) networks, and an ensemble of Random Forest and Gradient Boosting models. The IMDB dataset, comprising 50,000 reviews, was preprocessed and analyzed using advanced NLP and machine learning techniques. Results show accuracies of 87% (Logistic Regression), 88% (LSTM), and 90% (ensemble), with detailed error and feature analyses revealing insights into model performance and limitations. This study enhances understanding of sentiment analysis for academic and practical applications.

1 Introduction

Sentiment analysis is a pivotal application of natural language processing (NLP), enabling the automated determination of sentiment expressed in textual data, such as customer reviews, social media posts, and, in this case, movie reviews. This project focuses on the IMDB Movie Reviews dataset, which includes 50,000 reviews evenly split between positive and negative labels, sourced from http://ai.stanford.edu/ amaas/data/sentiment/. Initially, a baseline Logistic Regression model was developed, followed by advanced deep learning (LSTM) and ensemble techniques (Random Forest + Gradient Boosting) to explore performance improvements and provide deeper insights into text classification challenges. This report details the methodology, results, and analysis, offering a robust foundation for university-level research and future extensions.

2 Dataset and Methodology

2.1 Dataset Description

The IMDB dataset comprises 50,000 movie reviews, evenly divided into 25,000 training and 25,000 testing reviews, with each subset balanced between positive and negative sentiments (12,500 each). For this study, only the training set was used, sampled to 5,000 reviews for efficiency in advanced analyses, ensuring computational feasibility on a MacBook Air M2 while maintaining representative data.

2.2 Data Preprocessing

Reviews underwent preprocessing using the Natural Language Toolkit (NLTK) to ensure clean, standardized text:

- Converted all text to lowercase for consistency.
- Removed HTML tags (e.g., `<br />`) common in the dataset.
- Eliminated punctuation and special characters to focus on meaningful words.
- Removed stopwords (e.g., "the," "and") to reduce noise.

Additional features, such as review length (word count) and sentiment intensity (using TextBlob's polarity score), were engineered for advanced models.

2.3 Feature Engineering

Three distinct feature extraction methods were applied:

- **TF-IDF Vectorization:** Term Frequency-Inverse Document Frequency with 3,000 features for Logistic Regression and ensemble models, capturing word importance across reviews.
- **Word Embeddings:** Tokenization and padding for LSTM, using a 100-dimensional embedding layer (randomly initialized, with optional GloVe integration), representing semantic relationships between words.
- **Metadata Features:** Review length and sentiment polarity (TextBlob) for the ensemble model, enhancing feature richness.

2.4 Modeling Approaches

Three models were developed to classify sentiments:

- **Logistic Regression:** A baseline model trained on TF-IDF features, using an 80%/20% train/validation split, implemented in scikit-learn with 1,000 maximum iterations for convergence.
- **LSTM (Deep Learning):** A neural network with an embedding layer, two LSTM layers (128/64 units), dropout (0.2), and dense layers, trained on tokenized sequences using TensorFlow/Keras, with 5 epochs and batch size 64.
- **Ensemble (Random Forest + Gradient Boosting):** A voting ensemble combining Random Forest (50 estimators) and Gradient Boosting (50 estimators), trained on TF-IDF and metadata features, with 3-fold cross-validation for robustness, implemented in scikit-learn.

3 Results

3.1 Logistic Regression Results

The Logistic Regression model achieved an accuracy of 87% on the validation set (1,000 reviews, 20% of 5,000 sampled). The classification report showed balanced precision, recall, and F1-scores (0.87) for both positive and negative classes. The confusion matrix, visualized below, indicated few false positives and negatives but highlighted challenges with sarcastic or ambiguous reviews.

3.2 LSTM Results

The LSTM model, leveraging word embeddings, achieved an accuracy of 88% on the validation set. The classification report and confusion matrix (below) demonstrated improved handling of complex text patterns compared to Logistic Regression, though misclassifications persisted with ambiguous or short reviews.

3.3 Ensemble Results

The ensemble model (Random Forest + Gradient Boosting) achieved the highest accuracy of 90% on the validation set, with 3-fold cross-validation scores averaging 89% (±2%). The confusion matrix (below) and feature importance analysis revealed that review length, sentiment score, and key TF-IDF features (e.g., ”excellent,” ”terrible”) significantly influenced predictions.

4 Analysis

4.1 Model Performance Comparison

All models performed well, with the ensemble achieving the best accuracy (90%) due to its ability to leverage both text (TF-IDF) and metadata (length, sentiment) features. Logistic Regression (87%) provided a strong baseline, while LSTM (88%) improved on complex text patterns, though it required more computational resources. The ensemble's cross-validation robustness (89% mean) underscores its stability across folds.

4.2 Error Analysis

Misclassifications across models often involved:

- Sarcastic or ambiguous reviews, where sentiment cues were contradictory (e.g., “This film is surprisingly terrible”).
- Short reviews, lacking sufficient context for accurate classification.
- Reviews with mixed sentiments, challenging binary classification.

Sample misclassified reviews from each model (e.g., LSTM, ensemble) showed patterns like low sentiment scores or atypical lengths, suggesting areas for improvement.

4.3 Feature Importance

For the ensemble model, Random Forest feature importance analysis highlighted:

- TF-IDF features: ”excellent” (0.0152), ”terrible” (0.0148), ”great” (0.0135), indicating key sentiment indicators.

- Metadata features: ”review$_{l}ength$” (0.0123), $”sentiment_{s}core”$ (0.0119), $showingtheirpredictivepower.$

These insights guide future feature engineering and model refinement.

5 Conclusion and Future Work

This study demonstrated the efficacy of advanced machine learning techniques for sentiment analysis. Logistic Regression provided a solid baseline (87%), while LSTM (88%) and the ensemble (90%) offered marginal improvements, showcasing the value of deep learning and ensemble methods for nuanced text classification. Challenges like sarcasm and ambiguity suggest opportunities for:

- Rule-based preprocessing to handle sarcasm (e.g., negation detection).
- Integrating pretrained embeddings (e.g., GloVe, BERT) for LSTM to capture semantic nuances.
- Expanding to the test set (25,000 reviews) for final evaluation and deployment.
- Exploring multimodal features (e.g., user ratings, timestamps) or advanced neural architectures (e.g., Transformers).

This project significantly advances a university-level understanding of NLP and machine learning, offering a model for both academic research and practical applications.

6 Acknowledgments

This project was conducted using open-source tools (Python, TensorFlow, scikit-learn, NLTK) and the publicly available IMDB dataset, with computational support on a MacBook Air M2.