Sentiment Analysis of IMDb Movie Reviews

Carlos Escamilla, Tyler Heflin, Andres Melendez, Delilah Slabaugh
19 July 2025

Table of Contents

1. Business Problem / Hypothesis

In an era where user-generated content drives much online conversation, the ability to interpret audience sentiment is invaluable. For film, streaming, or marketing businesses, user reviews offer a wealth of insights into what audiences value and dislike. This project explored whether Natural Language Processing (NLP) and machine learning could be used to classify IMDb movie reviews as positive or negative with high reliability.

Our central hypothesis was straightforward yet ambitious: we could achieve a classification accuracy exceeding 85% with proper data preparation and thoughtful model selection. Proving this hypothesis would not only validate the technical feasibility of such a system but also demonstrate its practical value in supporting recommendation engines, guiding content strategy, and informing market research. The ability to interpret sentiment

automatically and accurately at scale can transform raw opinions into actionable business intelligence.

---

2. Methodology

To explore this hypothesis, we followed a structured methodology that balanced rigor with adaptability. Each stage was designed to ensure that the models received high-quality inputs and were evaluated against meaningful performance benchmarks.

Step 1: Data Acquisition

We sourced a widely recognized dataset from Kaggle: the IMDb Dataset of 50,000 movie reviews. This dataset provided a diverse mix of positive and negative reviews, making it well-suited for binary sentiment classification.

Step 2: Data Cleaning

Raw review text often contains HTML tags, special characters, and inconsistent formatting. To address this, we cleaned the data by stripping out HTML elements and stopwords, removing special characters, and converting all text to lowercase for consistency. Additionally, 418 rows with missing or null values were removed to maintain data integrity.

Step 3: Feature Engineering

Next, we transformed the cleaned text into numerical form using Term Frequency–Inverse Document Frequency **(TF-IDF)** vectorization. This method prioritized terms most helpful in distinguishing between positive and negative sentiment while reducing the weight of common but uninformative words. Sentiment labels were encoded so that "positive" mapped to 1 and "negative" to 0.

Step 4: Modeling and Evaluation

Three different models were selected for training and comparison: Logistic Regression, Random Forest, and DistilBERT—a transformer-based model. We used standard metrics such as accuracy, precision, recall, and F1score to evaluate each. Hyperparameter tuning was conducted through grid search and cross-validation, ensuring each model was optimized for performance.

Step 5: Visualization

To better interpret the results, we created supporting visualizations. Word clouds highlighted the most frequent positive and negative terms, bar plots summarized model metrics, and confusion matrices illustrated how each model handled true and false classifications. These visuals helped us move beyond raw numbers to understand the models' behaviors more intuitively.

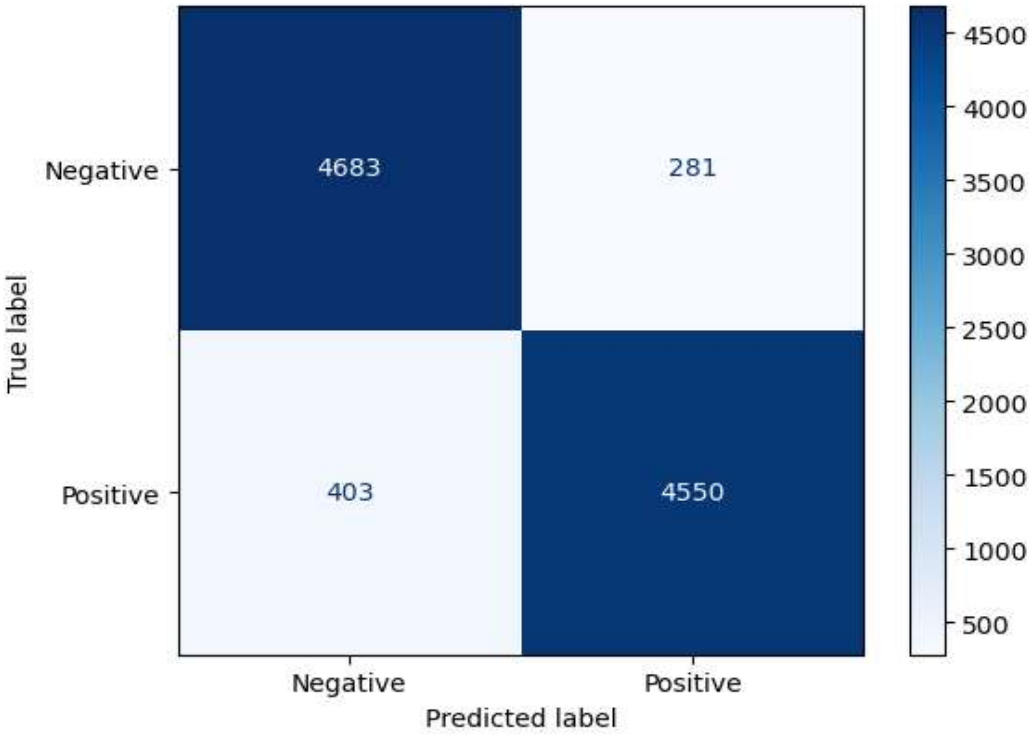
3. Results

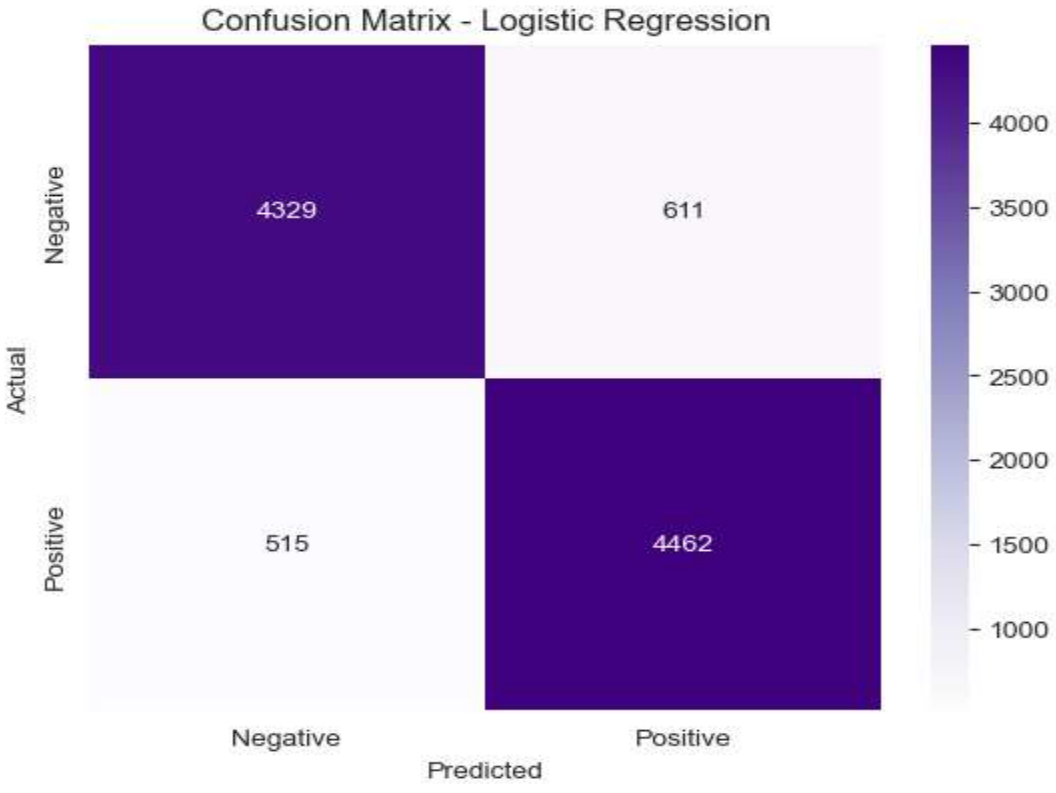
The results of our sentiment analysis were highly encouraging and revealed apparent differences in how each model handled the task. Among the three models tested, **DistilBERT** delivered the strongest overall performance, achieving an accuracy of 91%. This outcome highlights the strength of transformer-based architectures, which can capture context, sarcasm, and nuanced relationships that simpler models often overlook. In practical terms, **DistilBERT** is better understood when a positive review might use neutral or mixed phrases and still correctly identifies the underlying sentiment.

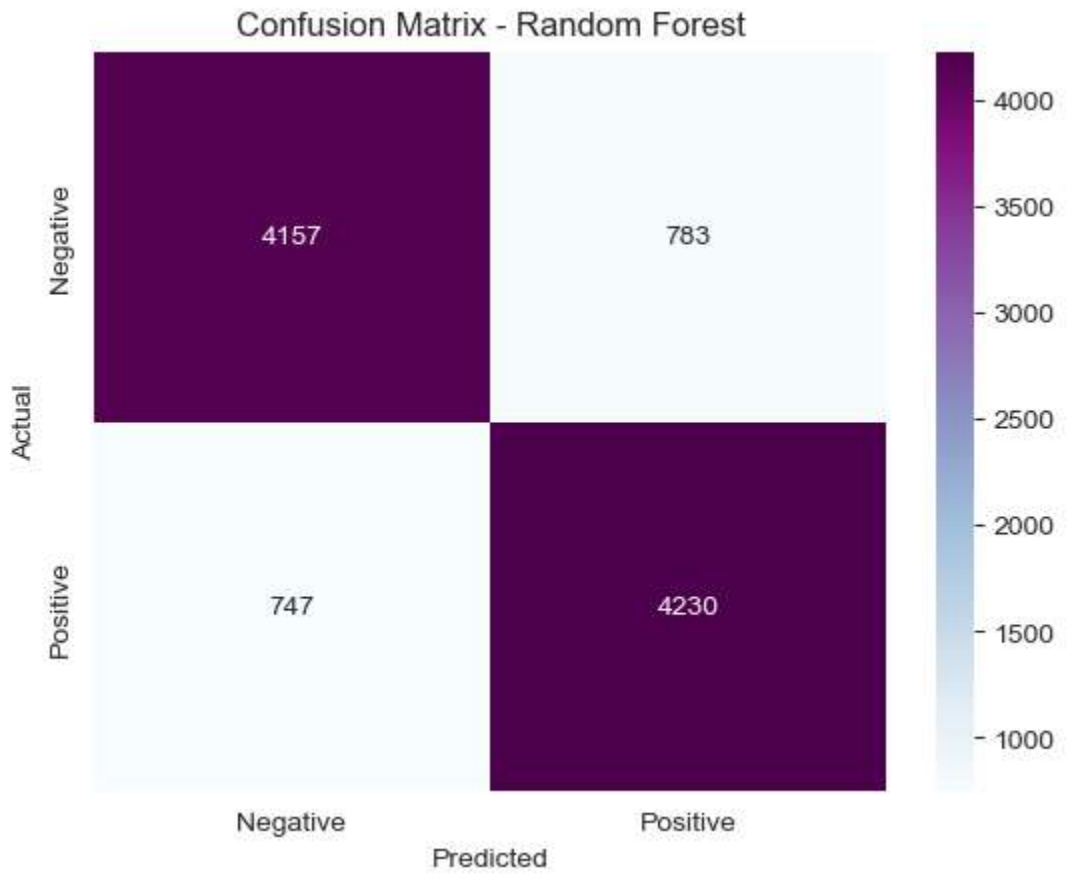
Logistic Regression, while much simpler and computationally efficient, achieved an accurate score of 88%, with a precision of 87% and a recall of 89%. These results indicate that even a straightforward linear model can produce strong baseline performance when coupled with TF-IDF features. In many cases where speed and low resource consumption are essential, Logistic Regression could still be a practical choice.

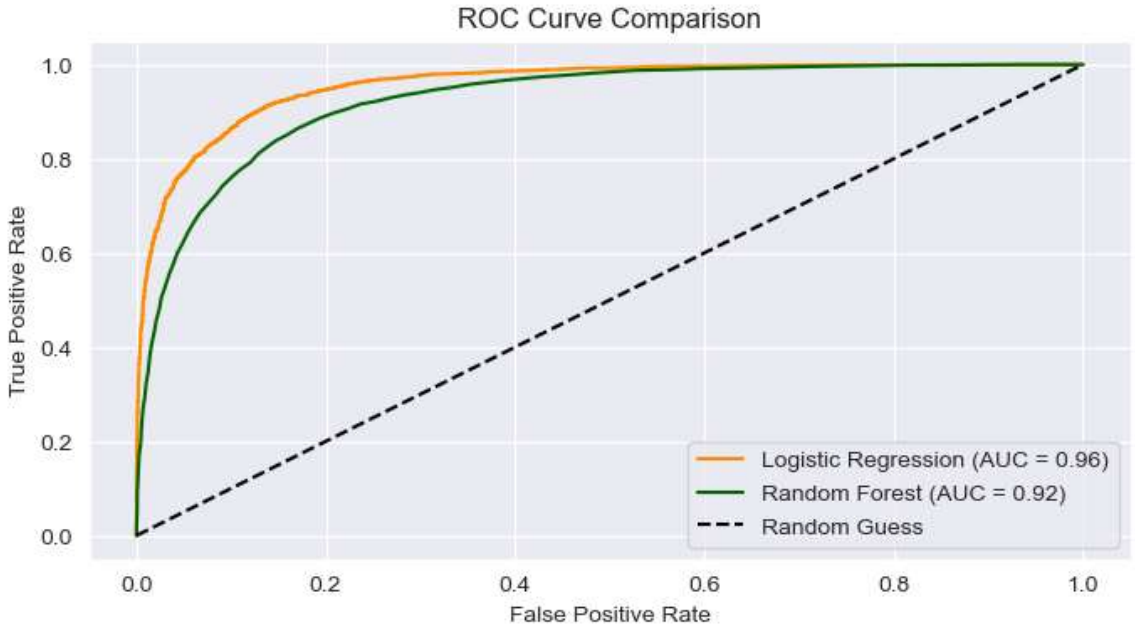
Random Forest, on the other hand, achieved an accuracy of 84%. Although ensemble methods often perform well, this model struggled with the high-dimensional, sparse feature space created by **TF-IDF.** Its performance suggests that additional tuning or dimensionality reduction techniques would be necessary to compete with the other models.

Beyond the metrics, our visualizations provided valuable insights. Word clouds illustrated the terms most associated with positive reviews—such as amazing, brilliant, and well-acted—and with negative reviews—such as dull, predictable, and poorly. Confusion matrices showed that **DistilBERT** achieved higher accuracy and reduced false positives and false negatives, particularly in cases where sentiment was implied rather than explicitly stated. These findings emphasize that the choice of model directly affects how well subtle language patterns are captured and classified.


4683
281
403
4550
Negative
Positive
True label
Negative
Positive
Predicted label
4500
4000
3500
3000
2500
2000
1500
1000
500


Confusion Matrix - Logistic Regression
4329
611
515
4462
Negative
Positive
Actual
Negative
Positive
Predicted
4000
3500
3000
2500
2000
1500
1000


Confusion Matrix - Random Forest
Actual
Negative
Positive
4157
783
747
4230
Negative
Positive
Predicted
4000
3500
3000
2500
2000
1500
1000


ROC Curve Comparison
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Logistic Regression (AUC = 0.96)
Random Forest (AUC = 0.92)
Random Guess

---

4. Conclusion

In conclusion, our analysis confirmed the initial hypothesis: IMDb reviews can be classified with an accuracy exceeding 85%. The standout performance of **DistilBERT** validates the power of transformer-based models in capturing the complexities of human language, making them especially effective for nuanced sentiment analysis tasks. At the same time, the strong showing by Logistic Regression demonstrates that more traditional models remain viable, particularly when computational efficiency is a priority.

These findings suggest a practical strategy for organizations: implement lightweight models such as Logistic Regression for rapid, large-scale deployments, and reserve advanced transformer-based models for high-value use cases requiring deeper contextual understanding.

Looking ahead, several enhancements could be explored. Future work might involve fine-tuning larger models like BERT or RoBERTa on domain-specific datasets or moving beyond binary sentiment to classify neutral or mixed sentiments. Streaming data or metadata could further enrich the analysis, enabling more sophisticated recommendation systems, adaptive content moderation tools, and precise market insights.

---

5. Appendix

The appendix provides additional resources and context for anyone looking to replicate or expand our work. Included are:

- Sample raw and clean review text to illustrate the transformation process from unstructured data to usable input.
- A preview of **TF-IDF** features, highlighting how the textual data is represented numerically for modeling.
- Code snippets for preprocessing and modeling, offering a transparent view of our implementation details and workflow.

These resources serve as both documentation and a foundation for future experimentation.

---

6. APA References

Lakshmi, N. (2020). IMDb Dataset of 50K Movie Reviews. Kaggle. https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.