SENTIMENT ANALYSIS OF IMDB REVIEWS

Advanced Text Analytics Project

Merve Pakcan

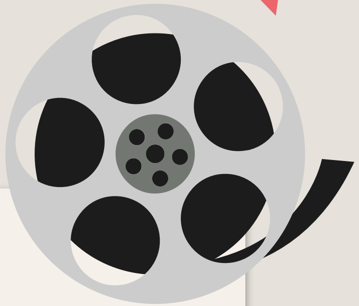


TABLE OF CONTENTS

INTRODUCTION

User-generated content is growing. Movie reviews are a great source to understand audience sentiment.

Objective:

To compare the performance of three machine learning models **Naive Bayes**, **Logistic Regression**, and a **Neural Network** for classifying IMDB movie reviews as **positive** or **negative**.

What is Sentiment Analysis?

A subfield of NLP, detects and classifies **emotions or opinions** in text.

DATASET DESCRIPTION

"IMDB Dataset of 50K Movie Reviews", available on **Kaggle**

50,000 REVIEWS, 2 COLUMNS

80% training, 20% testing

REVIEW

SENTIMENT

Source of the data:

Name: Large Movie Review Dataset (IMDB)
Source: Stanford AI Lab
Authors: Andrew L. Maas et al., ACL 2011
Size: 50,000 reviews (25K train / 25K test)

Balanced dataset (50% positive, 50% negative)

DATA PREPROCESSING

Naive Bayes & Logistic Regression

- Lowercasing
- Remove HTML, URLs, punctuation
- Tokenization (TweetTokenizer)
- Stopword removal
- Stemming (Porter)
- (NB only) Frequency dictionary creation

Neural Network

- Tokenization (Keras Tokenizer, vocab=10,000)
- Convert to sequences
- Padding to fixed length
- No stemming or stopword removal
- Word representation via Embedding Layer

Why Different Preprocessing?

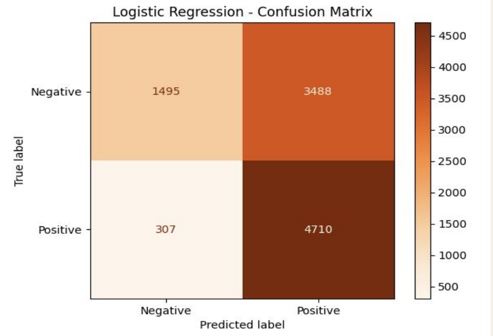
★ ★ ★ ★ ★ ★ ★

MODEL STRUCTURES & IMPLEMENTATION

1. LOGISTIC REGRESSION

A supervised binary classifier to predict sentiment(positive/negative),
Built with Scikit-learn's LogisticRegression (default parameters)
Input: Preprocessed text converted into numeric features
Output: Probability mapped to sentiment label (0 or 1)
Initial test accuracy: 62.08%

- Model predicted **positive** for mixed/neutral reviews
- "I expected a great movie, but it was boring and full of clichés." → **Positive (0.62)**
- **Biased toward positive words**



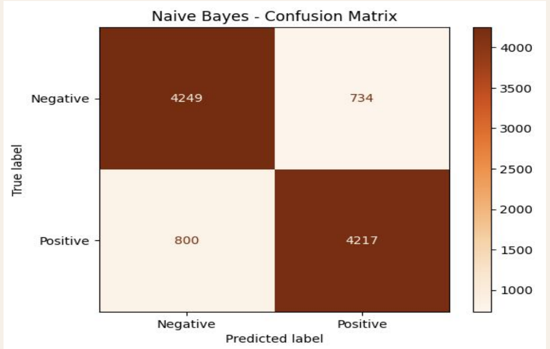
```
Logistic Regression – Classification Report
              precision    recall  f1-score   support

    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
```

• **Recall imbalance:** Positive = 0.94, Negative = 0.30
(3488 was negative but model says positive)

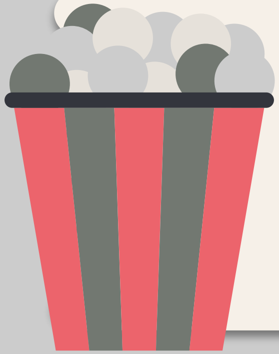
2. NAIVE BAYES

A probabilistic classifier that applies Bayes' Theorem, assuming word independence.
Built with NLTK's NaiveBayesClassifier

Input: Cleaned, stemmed tokens with (word, label) frequency pairs
Output: Class label based on word-level probability estimates
Initial test accuracy: 85.0%



```
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
```

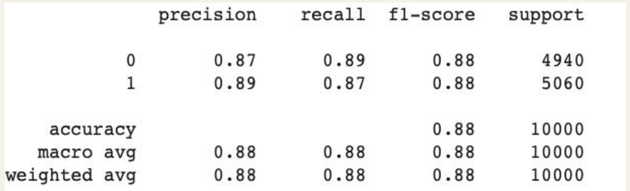


Balanced recall for both classes (≈0.85)
Correctly predicted:
• 4,249 negative, 4,217 positive
Strength: Effective on clear sentiment
The movie had some good moments, but overall it felt flat.
Limitation: Tended to label mixed-tone reviews as negative

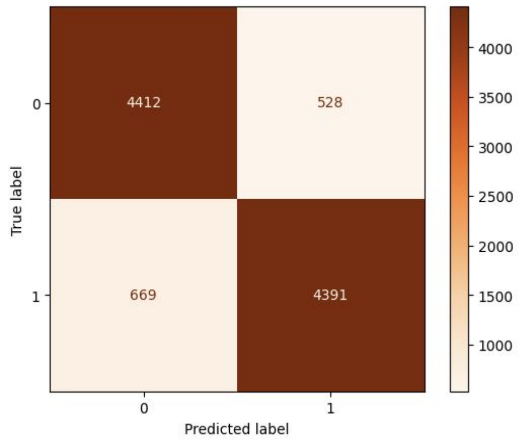
3. NEURAL NETWORK

A neural model built using embedding and dense layers to classify review sentiment.
Built with: Keras Sequential model
Input: Padded token sequences
Output: Sentiment label (0 or 1) via sigmoid
Test Accuracy: 88.00%

Balanced **precision & recall**: both **0.88**
Correctly classified **neutral/mixed** reviews
“The movie had good moments, but felt flat.” → **Negative (0.33)**
Model learned word patterns through **embedding layer**
Used **EarlyStopping** to avoid overfitting and boost generalization



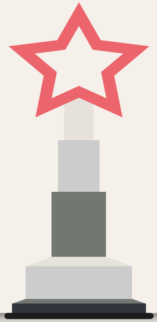
```
              precision    recall  f1-score   support

           0       0.87      0.89      0.88      4940
           1       0.89      0.87      0.88      5060

    accuracy                           0.88     10000
   macro avg       0.88      0.88      0.88     10000
weighted avg       0.88      0.88      0.88     10000
```

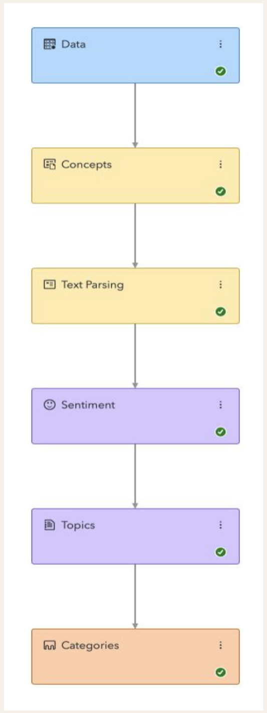


SENTIMENT ANALYSIS IN SAS

Pipeline completed successfully.

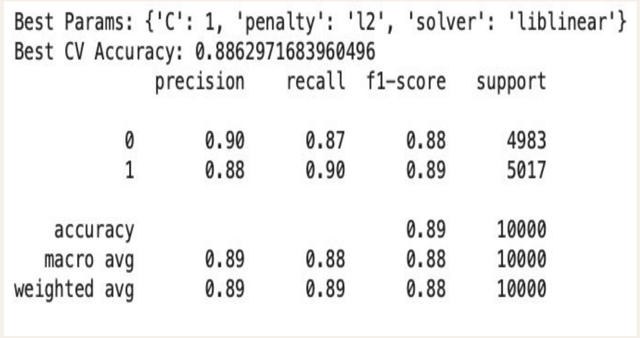
- **Text Parsing**: 64,477 unique terms extracted with word types (noun, verb)
- **Sentiment Analysis**: Automatically labeled 49,999 reviews as *positive*, *negative*, or *neutral*
- **Topic Modeling**: Clustered reviews into key review topics
- **Categories**: Organized output into clear, interpretable groups





Hyperparameter Tuning

Logistic Regression

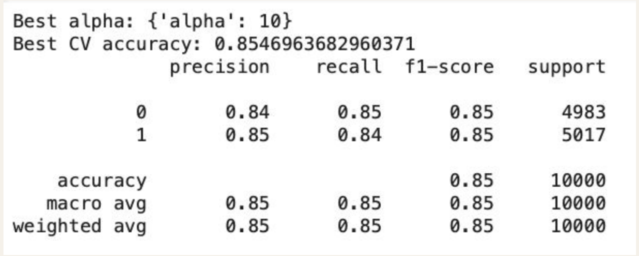


```
Best Params: {'C': 1, 'penalty': 'l2', 'solver': 'liblinear'}
Best CV Accuracy: 0.8862971683960496
              precision    recall  f1-score   support

           0       0.90      0.87      0.88      4983
           1       0.88      0.90      0.89      5017

    accuracy                           0.89     10000
   macro avg       0.89      0.88      0.88     10000
weighted avg       0.89      0.89      0.88     10000
```

Method: GridSearchCV (CV=5, scoring='accuracy')
Parameters Tuned: C, penalty, solver
Before tuning: 62% accuracy, low recall on negative class
After tuning:Accuracy: 88.0%,Recall (Neg): ↑ from 0.30 to 0.87,
Recall (Pos): stayed high at 0.90
Model became more balanced and consistent

Naive Bayes

Already in optimum
Before: 85% After: 85%



```
Best alpha: {'alpha': 10}
Best CV accuracy: 0.8546963682960371
              precision    recall  f1-score   support

           0       0.84      0.85      0.85      4983
           1       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
```

Neural Network

88% accuracy

Tuned **vocab size((10k)**, **max length(200)**, and **embedding dim(16)**

Added **EarlyStopping** to prevent overfitting

Trained with **batch data + validation set**

Tested on **ambiguous/neutral reviews**

Bidirectional LSTM and lower LR (Adam 0.0005) — no accuracy gain (still 88%).

★ ★ ★ ★ ★ ★ ★ ★ ★ ★

CONCLUSION

- Compared **Logistic Regression**, **Naive Bayes**, and **Neural Network** on IMDB reviews
- **Naive Bayes**: 85% on clear sentiment, weak on mixed
- **Logistic Regression**: improved from 62% → 88% after tuning
- **Neural Network**: highest accuracy (88%), best on ambiguous reviews
- **SAS**: verified sentiment and topic labeling on full dataset

THANKS!

Do you have any questions?