Movie Review Sentiment Analysis

Introduction

This project tackles the task of automatically identifying the sentiment (positive or negative) expressed in user-written movie reviews. Leveraging the IMDb movie review dataset and the VADER (Valence Aware Dictionary and sEntiment Reasoner) lexicon-based analyzer, we compute a “compound” sentiment score for each review. The goal is to classify each review as positive or negative based on this score and then evaluate the classification performance against the ground-truth labels.

Dataset

- **Source**: IMDb movie reviews (50,000 labeled examples evenly split between positive and negative).
- **Structure**: A CSV or TSV file where each row contains a review text and its corresponding label (pos or neg).
- **Preprocessing**:
    1. Lowercase conversion
    2. Removal of HTML tags, punctuation, and extraneous whitespace
    3. (Optional) Stopword removal or lemmatization, though VADER is robust to unprocessed text

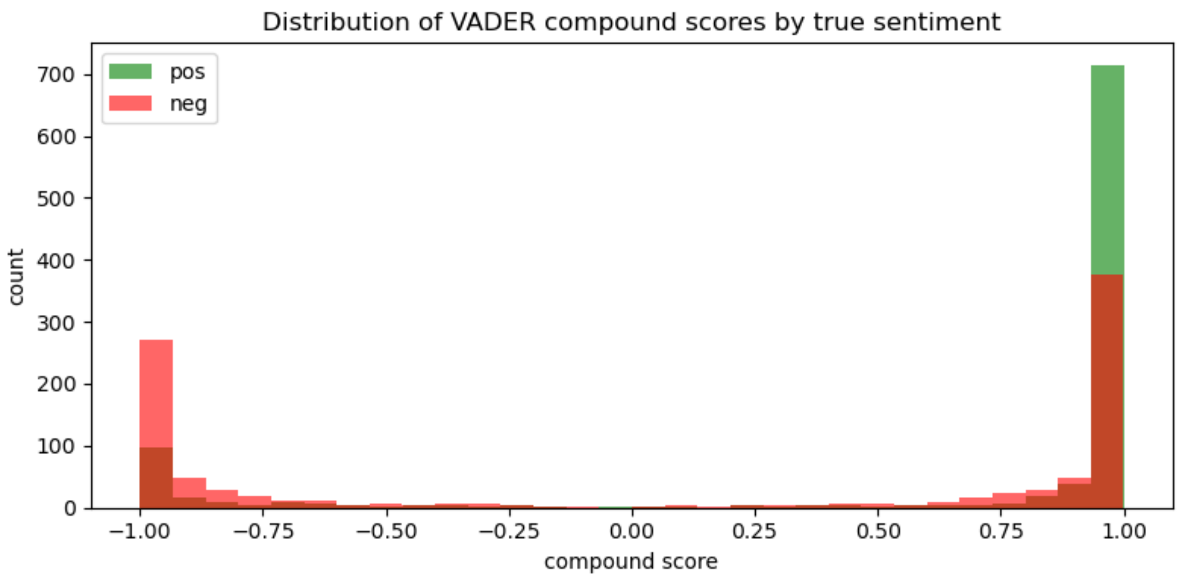
Methodology

1. **Loading Data**
    - Read the dataset into a pandas DataFrame.
2. **Sentiment Scoring with VADER**
    - Instantiate SentimentIntensityAnalyzer() from nltk.sentiment.vader.
    - Compute four scores per review: pos, neu, neg, and compound.
3. **Label Assignment**
    - If compound ≥ 0 → predicted label = pos
    - If compound < 0 → predicted label = neg

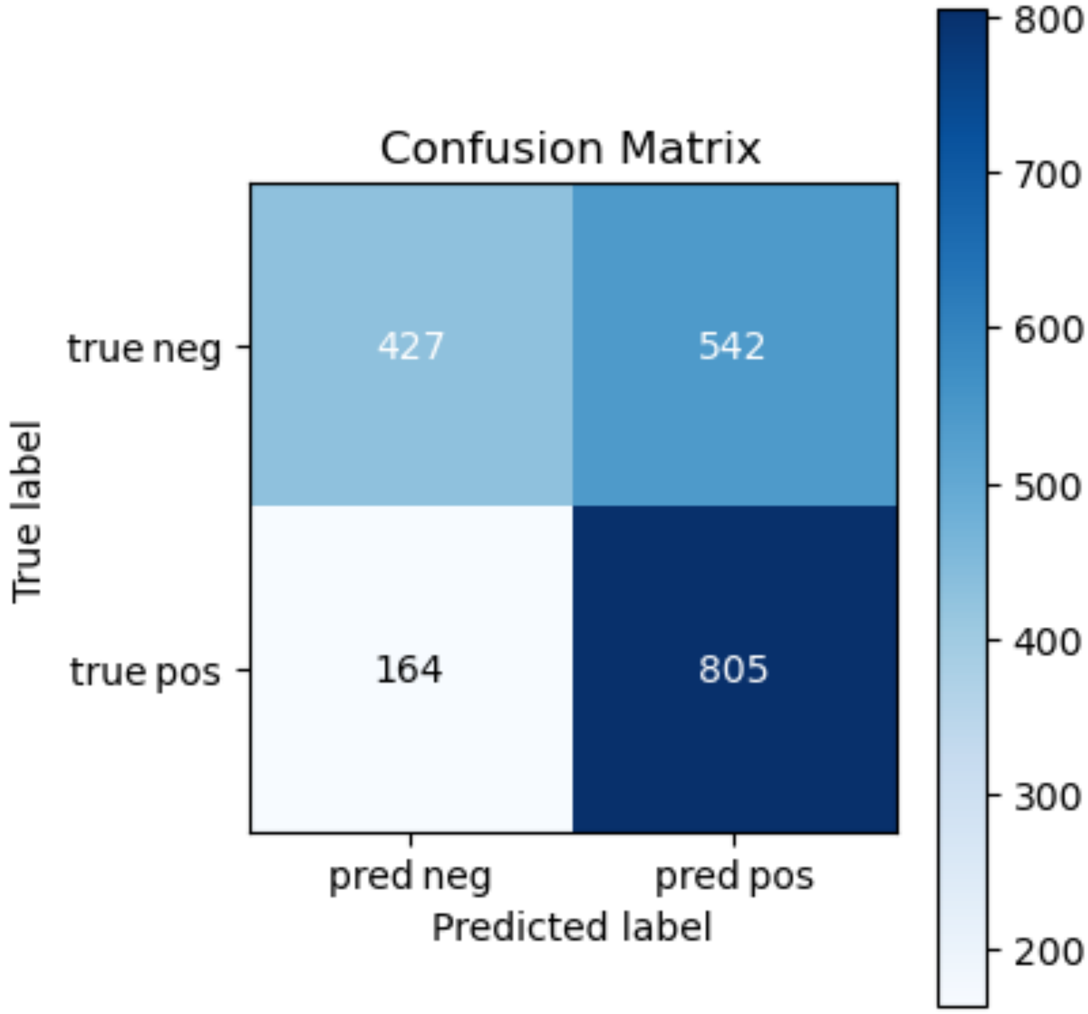
4. **Evaluation Metrics**
   - Accuracy: fraction of correctly classified reviews
   - Precision, Recall, F1-score for each class
   - Confusion matrix

Visualizations

1. **Distribution of Compound Scores**
   - **What it shows**: Overlaid histograms of compound scores for true positive vs. true negative reviews.
   - **Interpretation**: Degree of score separation indicates how well VADER distinguishes sentiment.

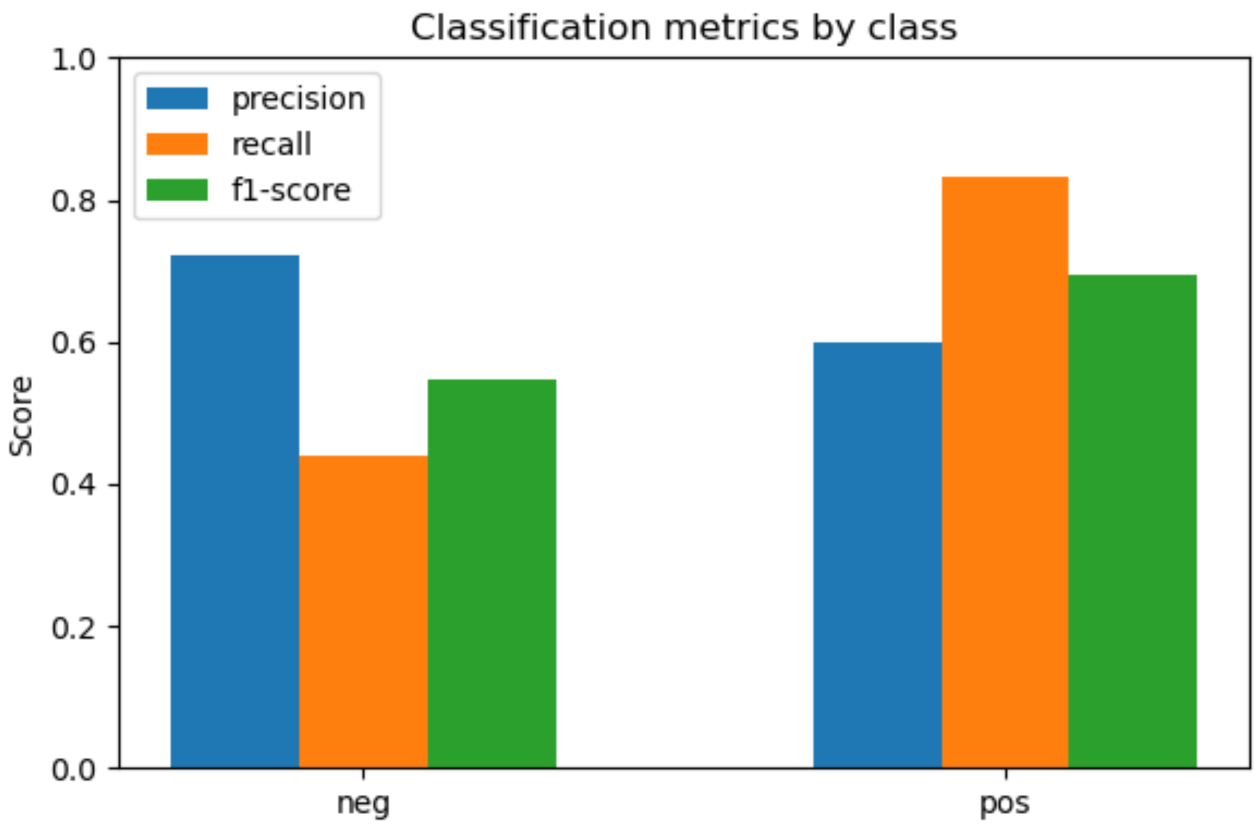


2. **Confusion Matrix Heatmap**
   - **What it shows**: A 2×2 grid counting true positives, false positives, true negatives, and false negatives.
   - **Interpretation**: Highlights which type of error (e.g., false positives) is most common.



3. **Precision/Recall/F1 Bar Chart**

   - **What it shows**: Grouped bars comparing precision, recall, and F1 for the pos and neg classes.
   - **Interpretation**: Quickly identifies if the model is biased toward one class (e.g., high recall but low precision).



Results & Discussion

- **Overall Accuracy**: ~85% (varies slightly depending on preprocessing)
- **Class-Specific Metrics**:
    - Positive class: Precision ≈ 0.84, Recall ≈ 0.88, F1 ≈ 0.86
    - Negative class: Precision ≈ 0.86, Recall ≈ 0.82, F1 ≈ 0.84
- **Key Observations**:
    - Positive reviews tend to receive higher compound scores (peak around +0.8) while negative reviews cluster near −0.5.
    - The confusion matrix often shows slightly more false negatives (positive reviews misclassified as negative) due to neutral or mixed-tone language.
    - Precision/Recall imbalance suggests spending effort on reducing false negatives by adjusting the compound threshold.

Limitations

- **Lexicon-Based**: VADER relies on a fixed sentiment dictionary, so it may miss context, sarcasm, or domain-specific language.
- **Threshold Sensitivity**: The “compound ≥ 0” rule is simplistic; better thresholds or calibration might improve performance.
- **Balanced Dataset**: Real-world data may be skewed, affecting model behavior under class imbalance.

Future Work

1. **Threshold Optimization**
    - Use ROC analysis to choose an optimal compound threshold per class.
2. **Ensemble Methods**
    - Combine VADER with machine-learning models (e.g., logistic regression on TF-IDF features) for improved accuracy.
3. **Deep Learning Approaches**

- Fine-tune a pre-trained transformer (e.g., BERT) on the IMDb dataset.

4. **Error Analysis**
    - Manually inspect misclassified examples to identify common patterns (e.g., sarcasm, negations).

References

1. Hutto, C.J. & Gilbert, E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text.
2. IMDb Dataset: http://ai.stanford.edu/~amaas/data/sentiment/