Final Project: Movie Review Text Classification

IST 664: Natural Language Processing

Professor: Michael Larche

Team Members:

Teera Yong

Smit Parikh

Vinay Kumar Chandra

Due Date: 12/09/2024

Table of Contents

1. Introduction

Sentiment analysis, a pivotal task in natural language processing (NLP), has garnered significant attention in recent years due to its ability to extract subjective opinions and emotions from textual data. This technology underpins a wide array of applications, such as customer feedback analysis, social media monitoring, and market research, where understanding public sentiment is critical. By classifying text data into predefined sentiment categories, sentiment analysis enables businesses, researchers, and policymakers to make informed decisions based on public opinions.

The Kaggle competition “Sentiment Analysis on Movie Reviews” provides an intriguing and challenging dataset to explore the complexities of sentiment classification. The dataset consists of phrases extracted from movie reviews and is annotated with sentiment labels. Each phrase in the dataset is manually labeled using crowd-sourcing, where annotators assign one of five sentiment categories: negative (0), somewhat negative (1), neutral (2), somewhat positive (3), and positive (4). This granular sentiment labeling offers a unique opportunity to analyze nuanced sentiment expressions, such as differentiating between strongly negative and mildly negative opinions. For instance, a phrase like "barely watchable" might fall into the "somewhat negative" category, whereas "a complete disaster" would be labeled as "negative."

The dataset is comprehensive, containing 156,060 phrases in the training file (train.tsv) and an additional test file (test.tsv) for evaluation. Its size and diversity present an ideal challenge for researchers to develop and benchmark machine learning models. By tackling this dataset, we aim to contribute to the broader field of sentiment analysis, leveraging the richness of movie reviews to advance our understanding of text classification methodologies.

Additionally, the problem of sentiment classification is far from trivial. Movie reviews often contain complex sentence structures, idiomatic expressions, and mixed sentiments, which make accurate classification a challenging task. For instance, a phrase such as "visually stunning but narratively weak" conveys both positive and negative sentiments, requiring sophisticated models to discern the overall tone. The manual labeling process ensures high-quality annotations, but it also highlights the subjective nature of sentiment analysis, as interpretations can vary between individuals.

2. Objective of the Project

The primary goal of this project is to build robust classification models for predicting sentiment labels using the Kaggle movie review dataset. This involves processing data by tokenizing phrases into words, removing stop words and punctuations, and engineering feature sets such as unigram bag-of-words, bigrams, and sentiment lexicons like Subjectivity, LIWC, and AFINN. The project leverages these features to explore and compare various classification techniques, including traditional machine learning algorithms such as Naïve Bayes and Random Forest, as well as modern deep learning approaches like Multi-layer Perceptron and Recurrent Neural Networks. By using the Kaggle dataset as a testbed, the findings from this project not only benchmark the performance of these models but also provide valuable insights into the practical applications of sentiment analysis across similar domains.

3. Methodology

3.1. Data Preprocessing

The training dataset was preprocessed through two routes. The first route, each phrase was tokenized into individual token while preserving the sentiment labels (“negative”, “somewhat negative”, “neutral”, “somewhat positive”, “positive”) in the form of numbers (0,1,2,3,4). The second route is almost similar to the first route except unnecessary stop words and punctuations were removed. It should be pointed out that not all stop words were removed especially those stop words associated with negations as they may contain significant negation features that may guide classifiers for better prediction. However, some extra stop words that were not in NLTK library were also included. These extra stop words are ("'s", "n't", "'re", "'ve", "'d", "also", "thing", "maybe", "would", "could", "should", "might", "must", "lot", "etc", "ok", "okay", "oh", "uh", "um").

3.2. Feature Engineering

- *Unigram Bag-of-Words*: Creating a sparse representation of text using the frequency of individual words. We only selected the most frequent 1500 words.
- *Negations*: creating negation word features such as words that follow these negations ('no', 'not', 'never', 'none', 'nowhere', 'nothing', 'noone', 'rather', 'hardly', 'scarcely', 'rarely', 'seldom', 'neither', 'nor').
- *Bigram Features*: Capturing contextual information by analyzing consecutive word pairs.
- *Sentiment Lexicons*: Incorporating external resources like:
    - Subjectivity Lexicon (SL) to detect subjective versus objective content.
    - LIWC (Linguistic Inquiry and Word Count) to analyze psychological attributes.
    - AFINN to assign sentiment scores based on word intensity.
- *Combined features:* combination of unigrams, negations, bigrams, sentiment lexicons, etc.
- *Size of the vocabulary:* not part of feature engineering. We changed the size of the unigrams (1500, 3000, and 5000) words to see the effects on the performance of the model.

3.3. Model Selection

Implementing and evaluating diverse machine learning models:

- Naïve Bayes: A probabilistic approach that assumes feature independence. This model will be used to test each features mentioned above on 5000 phrases in each experiment.
- Random Forest: An ensemble method for robust predictions.
- Multi-layer Perceptron (MLP): A feedforward neural network capable of learning complex relationships.
- Recurrent Neural Network (RNN): A sequential model using LSTM layers to capture long-term dependencies in text.

Each model classifier was conducted on the entire training set 156060 phrases along.

3.4.Model Evaluation:

- Using precision, recall, and F1-scores as metrics to compare the performance of various models.
- Employing cross-validation to ensure model robustness and prevent overfitting.

4. Result and Analysis

4.1 Experiment on Developing Features

<u>Experiment 1:</u> Unigram features

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7321
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0             0.224      0.210      0.214
1             0.239      0.370      0.289
2             0.830      0.642      0.724
3             0.229      0.369      0.282
4             0.150      0.217      0.177

Macro Average Precision Recall          F1      Over All Labels
              0.335      0.362      0.337

Label Counts {0: 218, 1: 827, 2: 2642, 3: 1016, 4: 297}
Micro Average Precision Recall          F1      Over All Labels
              0.543      0.497      0.507
Show 10 most informative features:

Most Informative Features
                 V_dull = True               0 : 2      =     55.7 : 1.0
                 V_cold = True               0 : 2      =     38.5 : 1.0
              V_failure = True               0 : 2      =     30.0 : 1.0
            V_pointless = True               0 : 2      =     30.0 : 1.0
            V_cinematic = True               4 : 2      =     27.4 : 1.0
              V_already = True               0 : 2      =     21.4 : 1.0
                V_avoid = True               0 : 2      =     21.4 : 1.0
                 V_dumb = True               0 : 2      =     21.4 : 1.0
               V_leaves = True               0 : 2      =     21.4 : 1.0
          V_manipulative = True               0 : 2      =     21.4 : 1.0
```

Figure 1: Result of Naïve Bayes Classifier with unigram features before removing stop words and punctuations.

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7163
Use Classifier:  Naive Bayes
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0             0.120      0.221      0.155
1             0.215      0.378      0.271
2             0.861      0.624      0.723
3             0.250      0.404      0.308
4             0.135      0.298      0.184

Macro Average Precision Recall          F1      Over All Labels
              0.316      0.385      0.328
Micro Average Precision Recall          F1      Over All Labels
              0.554      0.502      0.507
Show 10 most informative features:

Most Informative Features
                 V_dull = True               0 : 2      =     55.7 : 1.0
                 V_cold = True               0 : 2      =     38.5 : 1.0
              V_failure = True               0 : 2      =     30.0 : 1.0
            V_pointless = True               0 : 2      =     30.0 : 1.0
            V_cinematic = True               4 : 2      =     27.4 : 1.0
              V_already = True               0 : 2      =     21.4 : 1.0
                V_avoid = True               0 : 2      =     21.4 : 1.0
       V_disappointingly = True              0 : 2      =     21.4 : 1.0
                 V_dumb = True               0 : 2      =     21.4 : 1.0
               V_leaves = True               0 : 2      =     21.4 : 1.0
```

Figure 2: Result of Naïve Bayes Classifier with unigram features after removing stop words and punctuations.

Removing stop words and punctuation reduced the number of unique words but had negligible effects on overall model performance, as the Micro Average F1-Score remained consistent at 0.507. While certain class-specific F1 scores changed slightly, the most influential features and overall trends remained stable, with only two unigram features, *V_manipulative* and *V_disappointingly*, swapping places in the rankings. This demonstrates that stop word and punctuation removal may not be critical for this dataset, as key sentiment-bearing words are preserved in both cases. Moreover, omitting stop words and punctuation results in fewer tokens to process, reducing computational complexity while retaining meaningful tokens for unigram features.

Therefore, the model without stop words and punctuation, as implemented above, was selected as the baseline for subsequent experiments, **which also omit stop words and punctuation for consistency.**

Experiment 2: *Use unigrams and negations*

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7163
Use Classifier:  Naive Bayes
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0               0.204      0.188      0.195
1               0.226      0.353      0.274
2               0.801      0.641      0.712
3               0.295      0.410      0.342
4               0.162      0.232      0.190

Macro Average Precision Recall          F1      Over All Labels
                0.338      0.365      0.343
Micro Average Precision Recall          F1      Over All Labels
                0.539      0.502      0.511
```

Figure 3: Result of Naïve Bayes Classifier using unigrams and negations

Negation handling slightly improved the Micro Average F1-Score (from 0.507 to 0.511) compared to results of only use unigram features in experiment 1. Improvements were observed in Label 3 (Somewhat Positive), with a noticeable increase in F1-score (from 0.308 to 0.342), suggesting that negation handling helps in nuanced sentiment categories.

Experiment 3: *Use different vocabulary sizes (3000 and 5000)*

For 3000 most common words:

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7163
Use Classifier:  Naive Bayes
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0             0.083      0.309      0.126
1             0.190      0.352      0.246
2             0.864      0.615      0.718
3             0.257      0.405      0.314
4             0.106      0.246      0.148

Macro Average Precision Recall          F1      Over All Labels
              0.300      0.386      0.310
Micro Average Precision Recall          F1      Over All Labels
              0.550      0.493      0.498
```

Figure 4: Result of Naïve Bayes Classifier using unigrams with 3000 common words

Increasing the feature set from 1500 to 3000 most common words resulted in a marginal decline in Micro Average F1-Score (from 0.507 to 0.498). Neutral Sentiment (Label 2) maintained robust performance, with high precision (0.864) and recall (0.615), indicating that it benefits from additional vocabulary. Negative Sentiment (Label 0) and Positive Sentiment (Label 4) struggled, showing a decline in F1-scores compared to the smaller vocabulary size. While a larger vocabulary provides more features, it also increases sparsity and noise in the dataset, which can degrade classifier performance.

For 5000 most common words:

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7163
Use Classifier:  Naive Bayes
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0             0.053      0.219      0.079
1             0.181      0.399      0.247
2             0.882      0.604      0.716
3             0.233      0.394      0.291
4             0.091      0.240      0.122

Macro Average Precision Recall          F1      Over All Labels
              0.288      0.371      0.291
Micro Average Precision Recall          F1      Over All Labels
              0.551      0.489      0.489
```

Figure 4: Result of Naïve Bayes Classifier using unigrams with 5000 common words

Expanding the feature set to 5000 most common words resulted in no improvement in Micro Average F1-Score (0.489), similar to results from the 3000-word feature set. Macro Average F1-Score (0.291) slightly decreased, indicating challenges in classifying minority and extreme sentiment classes (Labels 0 and 4). Neutral Sentiment (Label 2) maintained its position as the best-performing class with high precision and recall. Negative Sentiment (Label 0) experienced significant performance degradation, with very low precision (0.053) and F1-score (0.079). Positive Sentiment (Label 4) also showed weak results, suggesting that the inclusion of less significant words dilutes the classifier's ability to identify positive sentiment.

There is no improvement in performance even after increasing feature sets from 1500 to 3000 and 5000 most common words. In fact, the performance seems to get worse with higher number of common words due to introducing additional noise, leading to sparsity in the feature matrix. Therefore, 1500 most common words as unigram features are preferred.

Experiment 4: Use Subjectivity Lexicons

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7159
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0               0.000       0.000       0.000
1               0.064       0.301       0.104
2               0.902       0.598       0.719
3               0.278       0.349       0.309
4               0.017       0.177       0.030

Macro Average Precision Recall          F1      Over All Labels
              0.252       0.285       0.232

Label Counts {0: 218, 1: 827, 2: 2642, 3: 1016, 4: 297}
Micro Average Precision Recall          F1      Over All Labels
              0.545       0.447       0.462
```

Figure 5: Result of Naïve Bayes Classifier using only subjectivity lexicons

Using subjectivity lexicons as features led to moderate overall performance, with a Micro Average F1-score of 0.462, slightly lower than the previous baseline experiment. While effective for identifying neutral and somewhat positive sentiments, this approach struggled with minority classes (Labels 0 and 4), with extremely low precision and recall. The subjectivity lexicon features may not adequately represent the linguistic patterns associated with extreme sentiments. The results suggest that subjectivity lexicons alone are insufficient for robust sentiment classification

and should be complemented with additional features such as unigrams, bigrams, or contextual embeddings.

Experiment 5: Use LIWC Lexicon

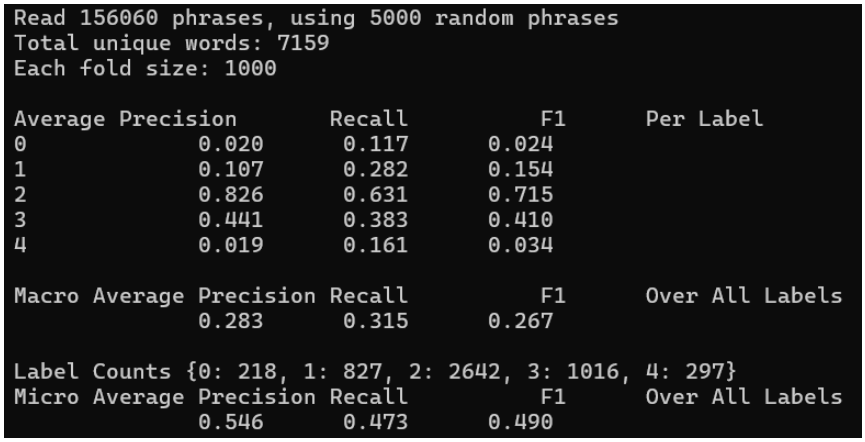

Figure 6: Result of Naïve Bayes Classifier using only LIWC lexicons

Using LIWC features improved overall performance compared to subjectivity lexicons, achieving a Micro Average F1-Score of 0.490. The results suggest LIWC features are better suited for capturing neutral and somewhat positive sentiments but remain insufficient for accurately identifying extreme sentiments (Labels 0 and 4), with F1-scores of 0.024 and 0.034, respectively. To address these limitations, LIWC features could be combined with unigrams, bigrams, or contextual embeddings to enhance classification performance for underrepresented classes.

Experiment 6: Use unigrams, negations, SL features, LIWC features

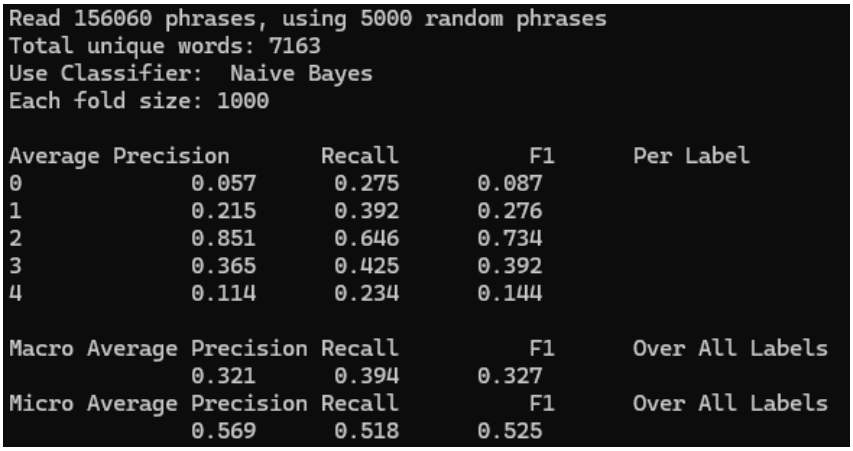

Figure 7: Result of Naïve Bayes Classifier using unigrams, negations, SL, and LIWC features

The use of a combined feature set (unigrams, negations, SL features, and LIWC features) yielded the best overall performance so far, with a Micro Average F1-Score of 0.525 and a Macro Average F1-Score of 0.327. This approach effectively leveraged the strengths of each feature type, improving classification for nuanced sentiment categories (Labels 1 and 3) while maintaining strong performance for neutral sentiment (Label 2). However, challenges remain for extreme sentiment categories (Labels 0 and 4), suggesting the need for additional features or model adaptations to handle these classes more effectively.

Experiment 7: Use combined features (unigrams, negations, bigrams, SL features, LIWC)

```
Read 156060 phrases, using 5000 random phrases
Total unique words: 7163
Use Classifier:  Naive Bayes
Each fold size: 1000

Average Precision       Recall          F1      Per Label
0             0.057      0.275      0.087
1             0.215      0.392      0.276
2             0.851      0.646      0.734
3             0.365      0.425      0.392
4             0.114      0.234      0.144

Macro Average Precision Recall          F1      Over All Labels
              0.321      0.394      0.327
Micro Average Precision Recall          F1      Over All Labels
              0.569      0.518      0.525
```

Figure 8: Result of Naïve Bayes Classifier using unigrams, bigrams, negations, SL, and LIWC.

Using a fully combined feature set (unigrams, negations, bigrams, SL features, and LIWC features) maintained the highest **Micro Average F1-Score of 0.525**, consistent with Experiment 6. However, the addition of bigrams did not yield measurable improvements, indicating their limited contribution to this dataset and model. The results highlight that while combining diverse features improves overall performance and balances across labels, challenges remain for minority and extreme sentiment classes (Labels 0 and 4).

Result Summary:

Table 1 summarizes the overall micro-average performance scores across experiments. Experiments 6 and 7, which utilized combined features including unigrams, negations, SL features, and LIWC features, achieved the highest performance with an F1-score of 0.525, representing a modest improvement over the baseline. Interestingly, the inclusion of bigram features in

Experiment 7 did not contribute to further performance gains, suggesting redundancy in their addition. It is important to note that these results were derived from a sample of only 5,000 phrases, used as a preliminary test to select feature sets. Running these experiments on the full dataset of 156,060 phrases would likely yield more significant differences in the results shown in Table 1. The combined features are particularly promising, as they effectively capture relationships between tokens and provide richer representations of the data's underlying structure.

Table 1: Summary results of micro average score of Naïve Bayes classifier using various features

Performance	Baseline	Negation Features	Vocabulary Size		SL Features	LIWC Features	Combined Features w/o bigrams	Combined Features with Bigrams
			3000	5000				
Precision	0.554	0.539	0.55	0.551	0.545	0.546	0.569	0.569
Recall	0.502	0.502	0.493	0.489	0.447	0.473	0.518	0.518
F-1 Score	0.507	0.511	0.498	0.489	0.462	0.49	0.525	0.525

4.2. Experiment on Advanced Classifiers and New Features

a. Naïve Bayes

We developed a new combined feature set that incorporates additional AFINN lexicon features while removing bigram and negation word features. These features were excluded because they constituted a large portion of the combined feature set without significantly enhancing model performance. Moreover, including these features would have made classification over the entire dataset computationally prohibitive. To compare performance, we employed four classifiers: Naïve Bayes, Random Forest, Multi-layer Perceptron, and Recurrent Neural Network. Each classifier was trained on the full training dataset of 156,060 phrases, and the best-performing model was subsequently evaluated on the test set.

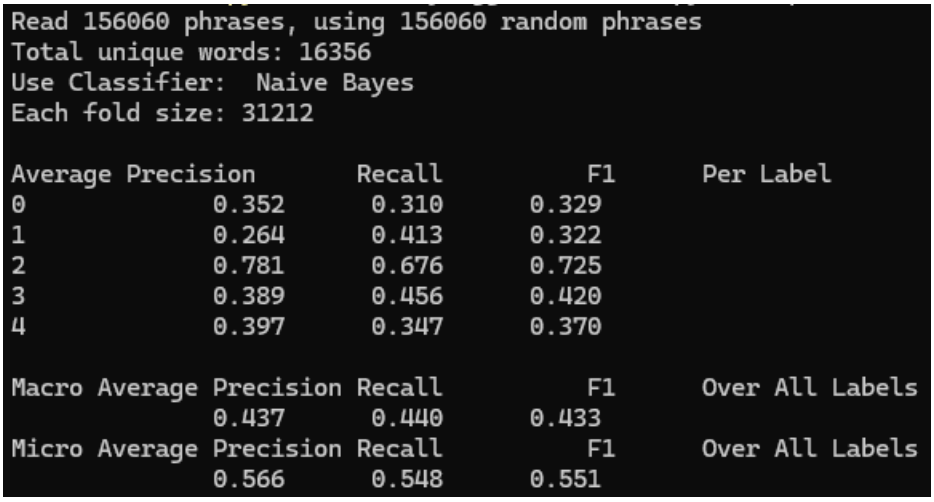

Figure 9: Result of using Naïve Bayes along with the new combined feature set (unigram, SL, LIWC, AFINN lexicons) on full training set.

The Naïve Bayes classifier achieved a Micro Average F1-score of 0.551 and a Macro Average F1-score of 0.433 on the full dataset. While it performed well on the majority class (Neutral), it struggled with minority classes (Negative and Somewhat Negative), highlighting its limitations in handling imbalanced datasets. Labels 0 (Negative) and 1 (Somewhat Negative) showed low precision and recall, likely due to class imbalance and overlapping language patterns with other sentiments. These results confirm that Naïve Bayes is effective as a baseline model but may require additional feature engineering or alternative techniques to improve performance on underrepresented sentiments.

b. Random Forest

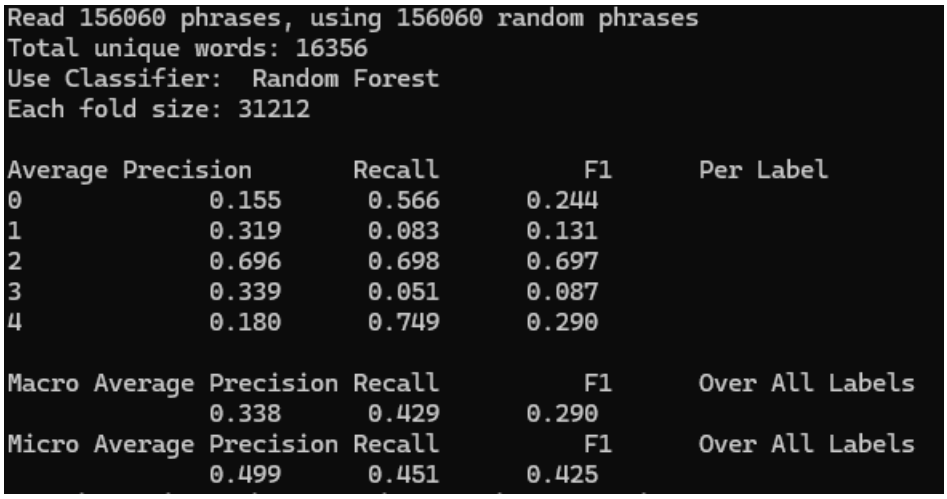

Figure 10: Result of using Random Forest along with the new combined feature set (unigram, SL, LIWC, AFINN lexicons) on full training set.

The Random Forest classifier achieved a Micro Average F1-score of 0.425 and a Macro Average F1-score of 0.290, reflecting moderate overall performance with significant challenges in handling imbalanced sentiment classes. While it performed well for the majority class (Neutral) and showed high recall for Positive sentiment, it struggled with precision and underrepresented labels, limiting its effectiveness as a standalone model. Further improvements could involve rebalancing techniques or feature engineering to enhance its ability to classify minority classes.

c. Multi-layer Perceptron

```
Read 156060 phrases, using 156060 random phrases
Total unique words: 16356
Use Classifier:  Multi-layer Perceptron
Each fold size: 31212

Average Precision       Recall          F1      Per Label
0               0.507      0.268      0.349
1               0.523      0.369      0.432
2               0.678      0.844      0.752
3               0.554      0.484      0.516
4               0.555      0.307      0.393

Macro Average Precision Recall          F1      Over All Labels
                0.563      0.454      0.488
Micro Average Precision Recall          F1      Over All Labels
                0.610      0.627      0.607
```

Figure 11: Result of using Multi-layer Perceptron along with the new combined feature set (unigram, SL, LIWC, AFINN lexicons) on full training set.

The Multi-layer Perceptron (MLP) classifier achieved a **Micro Average F1-score of 0.607** and a **Macro Average F1-score of 0.488**, representing the best overall performance among the tested models so far. Its ability to handle nuanced and frequent sentiment categories, coupled with its capacity for learning complex feature relationships, makes it a strong candidate for sentiment classification. However, addressing challenges with minority classes remains crucial for further improvement.

d. Recurrent Neural Network

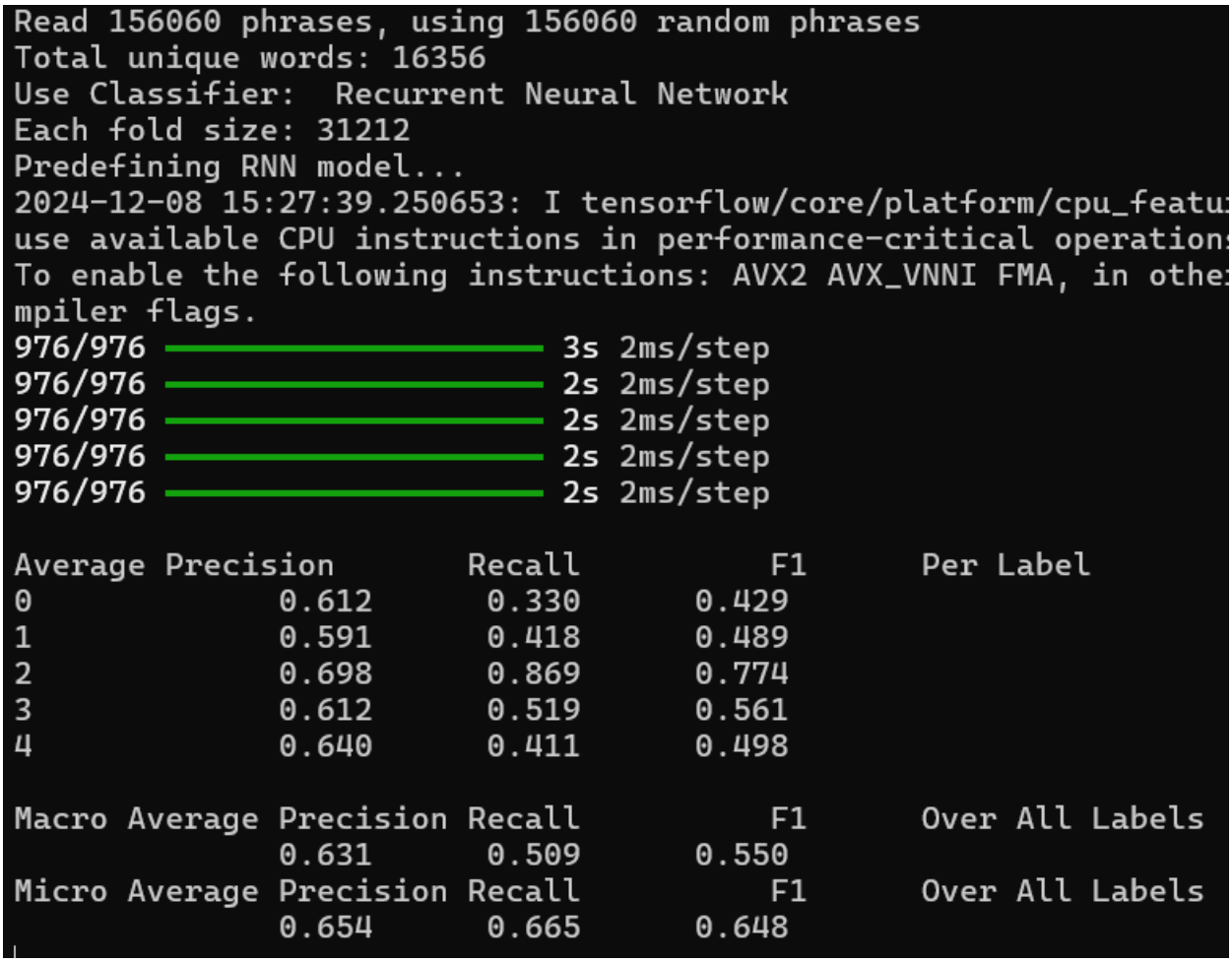

Figure 12: Result of using Recurrent Neural Network along with the new combined feature set (unigram, SL, LIWC, AFINN lexicons) on full training set.

The Recurrent Neural Network (RNN) with Bidirectional Transformer achieved the best overall performance, with a Micro Average F1-score of 0.648 and a Macro Average F1-score of 0.550. Its ability to leverage bidirectional context improved classification for nuanced and frequent sentiment classes while maintaining balanced predictions across all labels. Despite its strengths, further efforts are needed to enhance classification for minority and extreme sentiments (Labels 0 and 4). This makes the RNN with Bidirectional Transformer the most promising model for sentiment analysis in this study

Result Summary:

Table 2 shows that Recurrent Neural Network (RNN) achieved the best performance across all metrics, with the highest Precision (0.654), Recall (0.665), and F1-score (0.648). Its ability to utilize bidirectional context and capture sequential dependencies contributed to its strong performance. Multi-layer Perceptron (MLP) ranked second in performance, with a Precision of 0.610, Recall of 0.627, and F1-score of 0.607. The MLP's deep learning architecture enabled it to handle complex relationships in the data, improving results over traditional models. Naïve Bayes performed moderately well, achieving a Precision of 0.566, Recall of 0.548, and F1-score of 0.551. While simpler, it demonstrated strong baseline performance, particularly for frequent sentiment classes. Random Forest showed the weakest performance, with the lowest Precision (0.499), Recall (0.451), and F1-score (0.425). The model struggled with imbalanced classes and subtle sentiment distinctions. These results suggest that deep learning approaches, particularly RNNs, are the most effective for sentiment classification in this context.

Table 2: Summary results of micro average score of different classifiers using a combined feature set of unigrams, SL sentiment, LIWC sentiment, and AFINN sentiment

Performance	Naïve Bayes	Random Forest	Multi-layer Perceptron	Recurrent Neural Network
Precision	0.566	0.499	0.61	0.654
Recall	0.548	0.451	0.627	0.665
F-1 Score	0.551	0.425	0.607	0.648

Finally, the Recurrent Neural Network model was used to predict the sentiment of the test.tsv dataset, and the result was saved in test_prediction.csv.

5. Discussion

5.1 Key Observations

Feature Engineering:

- Unigram features with 1500 most common words served as a strong baseline but increasing the vocabulary size (3000 and 5000 words) introduced sparsity and noise, leading to reduced performance.
- Negation handling provided incremental improvements, especially for nuanced sentiment categories like "Somewhat Positive."
- Combining diverse features (unigrams, negations, SL features, LIWC features) yielded the best results, with a Micro Average F1-score of 0.525 using Naïve Bayes, demonstrating the value of feature diversity.

Classifier Performance:

- The Recurrent Neural Network (RNN) with Bidirectional Transformer achieved the best overall performance (F1-score: 0.648), leveraging its ability to model bidirectional context and sequential dependencies effectively.
- The Multi-layer Perceptron (MLP) followed closely, showcasing its capacity to capture complex relationships in the data (F1-score: 0.607).
- Naïve Bayes provided a reliable baseline, performing well for frequent classes but struggling with minority and extreme sentiments.
- Random Forest exhibited the weakest performance (F1-score: 0.425), struggling with imbalanced classes and subtle distinctions between sentiment categories.

Impact of Feature Combinations:

- Combining unigrams, SL features, LIWC features, and AFINN lexicons improved classification performance, especially in the case of Naïve Bayes classifier.
- Adding bigrams did not yield measurable improvements, indicating redundancy in this feature type for the dataset.

5.2 Lessons Learned

Model-Feature Alignment:

Different classifiers benefit from different feature sets. Simpler models like Naïve Bayes perform well with basic unigram features, while deep learning models benefit from rich, diverse feature representations.

Importance of Balance:

While addressing class imbalance is critical, techniques like **class weighting** during model training provided effective results without introducing noise or computational overhead. This approach proved preferable to oversampling techniques like SMOTE, which can distort sparse unigram representations.

Deep Learning Advantages:

RNNs and MLPs demonstrated superior performance due to their ability to learn complex patterns and contextual relationships, making them more effective for nuanced sentiment classification.

Feature Engineering Efficiency:

Removing stop words and unnecessary punctuation reduced computational complexity without sacrificing performance, emphasizing the need for efficiency in preprocessing.

5.3 Limitations

- Minority classes (e.g., Negative and Positive) consistently underperformed due to insufficient representation in the dataset, leading to biased predictions.
- Unigram-based sparse matrices presented challenges for classifiers, particularly Random Forest, which struggled to handle sparsity effectively.
- While combined features improved performance, the inclusion of bigrams did not provide significant benefits, suggesting limitations in their utility for this dataset.
- The computational cost of deep learning models like RNNs on the full dataset was significant, requiring substantial resources and time for training and evaluation.

6. Future Work

- Utilize class weighting in loss functions for deep learning models or during model training for traditional classifiers to ensure balanced performance across all sentiment classes. Explore under-sampling techniques for the majority class to reduce the bias toward frequent labels without introducing synthetic samples that may distort feature space.

- Incorporate contextual embeddings from transformer models like BERT or RoBERTa to capture richer linguistic patterns.
- Explore data augmentation techniques and domain adaptation to improve generalization across datasets with varying language patterns.
- Investigate hybrid models that combine the strengths of traditional machine learning (e.g., Random Forest) and deep learning approaches (e.g., RNNs) to achieve better balance and accuracy.

7. Conclusion

This study demonstrates the effectiveness of combining diverse feature sets and advanced classifiers for sentiment analysis of movie reviews. The Recurrent Neural Network with Bidirectional Transformer emerged as the best-performing model, achieving a Micro Average F1-score of 0.648, showcasing its ability to capture bidirectional context and nuanced sentiment relationships. The Multi-layer Perceptron followed closely, providing robust performance with simpler architecture. While Naïve Bayes served as a reliable baseline, and Random Forest struggled with class imbalance and complexity, the results highlight the importance of feature engineering and model selection in sentiment analysis tasks. Future work should address the limitations of class imbalance and explore advanced embeddings and hybrid models to further enhance sentiment classification performance.