Sentiment Analysis Project Report

Esposito Giorgio Angelo - 664640

May 28, 2025

1 Introduction

Analyzing this textual data to extract meaningful insights is a crucial task across various domains such as marketing, customer service, and research. One of the significant applications of text analysis is sentiment analysis, which involves determining the sentiment expressed in a piece of text.

Sentiment analysis, also known as opinion mining, is a Natural Language Processing (NLP) technique used to determine whether data is positive, negative, or neutral. It is widely used to analyze customer reviews, social media conversations, and other forms of textual data to understand public opinion and sentiment. This project focuses on performing sentiment analysis on movie reviews from the IMDB dataset.

The IMDB dataset comprises 50,000 movie reviews labeled as either positive or negative. The goal of this project is to build a machine learning model that can accurately classify these reviews based on their sentiment. To achieve this, various text processing techniques and machine learning models will be employed, including neural networks.

2 Related work

The paper ”BiERU: Bidirectional Emotional Recurrent Unit for Conversational Sentiment Analysis” [1] introduces the Bidirectional Emotional Recurrent Unit (BiERU), a neural network model designed to improve sentiment analysis in conversations by effectively capturing emotional context. Unlike traditional models that rely on explicit speaker information, BiERU processes dialogue without distinguishing between different speakers, making it more adaptable to various conversational settings. The model leverages a bidirectional recurrent structure to capture sentiment dependencies from both past and future contexts, ensuring a more comprehensive understanding of the emotional flow within a conversation. By focusing on emotional context rather than speaker-specific features, BiERU generalizes well across different datasets and conversational styles. The experimental results demonstrate that BiERU outperforms standard LSTMs, BiLSTMs, and Transformer-based models in sentiment classification accuracy and robustness. This makes it particularly valuable for applications such as chatbots, customer service analysis, and social media sentiment tracking, where understanding emotions in multi-turn conversations is essential.

The paper ”Sentiment Analysis Using Simplified Long Short-term Memory Recurrent Neural Networks” [2] explores ways to reduce the complexity of LSTM-based models for sentiment analysis while maintaining high accuracy. The authors propose six simplified versions of LSTM by modifying or removing certain components, aiming to lower computational costs without significantly affecting performance. The study evaluates these models on a Twitter sentiment analysis dataset, comparing their effectiveness to standard LSTMs. The results indicate that some simplified architectures achieve comparable accuracy to full LSTMs while being more efficient, making them suitable for real-time sentiment analysis tasks where computational resources are limited. The paper also investigates the impact of bidirectional LSTM layers, showing that while they can improve performance, their necessity depends on the dataset and the level of contextual understanding required. These findings provide valuable insights for optimizing deep learning models in sentiment classification, particularly in resource-constrained environments.

The paper ”Sentiment Analysis Using Gated Recurrent Neural Networks” [3] examines the application of Gated Recurrent Neural Networks (GRNNs) for sentiment classification, highlighting their ability to capture long-term dependencies in text. The study provides an overview of deep learning techniques used in sentiment analysis, emphasizing the advantages of GRNNs over traditional recurrent architectures like vanilla RNNs and LSTMs. By leveraging gating mechanisms, GRNNs effectively filter and retain relevant information, reducing the risk of vanishing gradient issues and improving sentiment prediction accuracy. The authors evaluate GRNN-based models on multiple sentiment analysis datasets, demonstrating their superior performance in extracting emotional context from text compared to standard recurrent models. The findings suggest that GRNNs are well-suited for sentiment classification tasks, particularly in scenarios where capturing nuanced emotional expressions is essential, such as customer feedback analysis and opinion mining.

3 Dataset

The dataset used for this project is the IMDB movie reviews dataset. It contains 50,000 reviews, labeled as either positive or negative, with an equal distribution of both classes. The dataset is divided into a training set, validation set and a test set, with the training set being the 80% (40,000 reviews) of the whole dataset, the remaining 20% is further divided into the validation set (16%, 8,000 reviews) and the test set (4%, 2,000 reviews). The dataset is available here [4].

3.1 Data preprocessing

Steps:

1. Loading the Dataset: The dataset is loaded from a CSV file.
2. Data Inspection: The first few rows of the dataset are displayed to understand its structure.
3. Lowercasing: All text in the reviews is converted to lowercase to ensure uniformity.
4. Text Cleaning: URLs, special characters, numbers, and punctuation are removed from the text.
5. Tokenization: The text is tokenized into words using NLTK.
6. Label Encoding: Sentiment labels are converted from text to numerical values (positive to 1, negative to 0).
7. Stopword Removal: Common stopwords are removed from the text using NLTK's stopwords list.
8. Text to Sequence Conversion: The text is converted to sequences of integers using a tokenizer, and the sequences are padded to ensure they all have the same length.
9. Loading GloVe Embeddings: Pre-trained GloVe embeddings are loaded and an embedding matrix is created for use in the neural network.

4 Models used

During the project, various models were evaluated to determine the best approach for sentiment analysis. The models evaluated include:

- Plain Neural Network (NN): A basic neural network with dense layers. It includes:
  - An embedding layer to convert words into dense vectors of fixed size.
  - A flatten layer to convert the 2D matrix into a vector.
  - Dense layers with ReLU activation for learning.
  - Dropout layers for regularization.
  - A final dense layer with sigmoid activation for binary classification.
- Recurrent Neural Network (RNN): A neural network with LSTM layers to handle sequence data. It includes:
  - An embedding layer to convert words into dense vectors of fixed size.

- An LSTM layer to capture sequential dependencies in the data.
- Dense layers with ReLU activation for learning.
- Dropout layers for regularization.
- A final dense layer with sigmoid activation for binary classification.

- Bi-directional RNN (Bi-RNN): A neural network with bidirectional LSTM layers to capture dependencies in both directions. It included:
  - An embedding layer to convert words into dense vectors of fixed size.
  - A bidirectional LSTM layer to capture dependencies in both forward and backward directions.
  - Dense layers with ReLU activation for learning.
  - Dropout layers for regularization.
  - A final dense layer with sigmoid activation for binary classification.

5 Hyperparameters tuning

A grid search was performed to find the best combination of hyperparameters. This involves training the model with different combinations of hyperparameters and evaluating their performance on the validation set. The hyperparameters tuned include:

- Batch Size: 32, 64
- Epochs: 10, 20, 30
- Neurons: 16, 32, 64, 128
- Learning Rate: 0.01, 0.001, 0.0001
- Dropout Rate: 0.0, 0.3, 0.6, 0.9

6 Results

The grid search was performed three times, in order to find the best hyperparameters for three kinds of models: plain Neural Network, RNN with LSTM units and bi-directional RNN with LSTM units. In the following subsections the results of these grid searches and the results obtained are reported.

6.1 Plain Neural Network (NN)

Best hyperparameters:

- Neurons: 128
- Learning Rate: 0.001
- Dropout Rate: 0.6
- Batch Size: 64
- Epochs: 20

The best validation accuracy achieved for the Plain NN model was approximately 72.0%. The learning curves for the best model are visible in figure 1.
The neural network model achieved a strong overall performance in the sentiment analysis task, demonstrating balanced precision, recall, and accuracy across both classes. The classification report (figure 2) indicates an accuracy of 84%, with both the negative and positive classes having an F1-score of 0.84. The model's precision and recall are nearly identical for both classes, suggesting a well-calibrated model that does not exhibit significant bias toward either positive or negative sentiments.

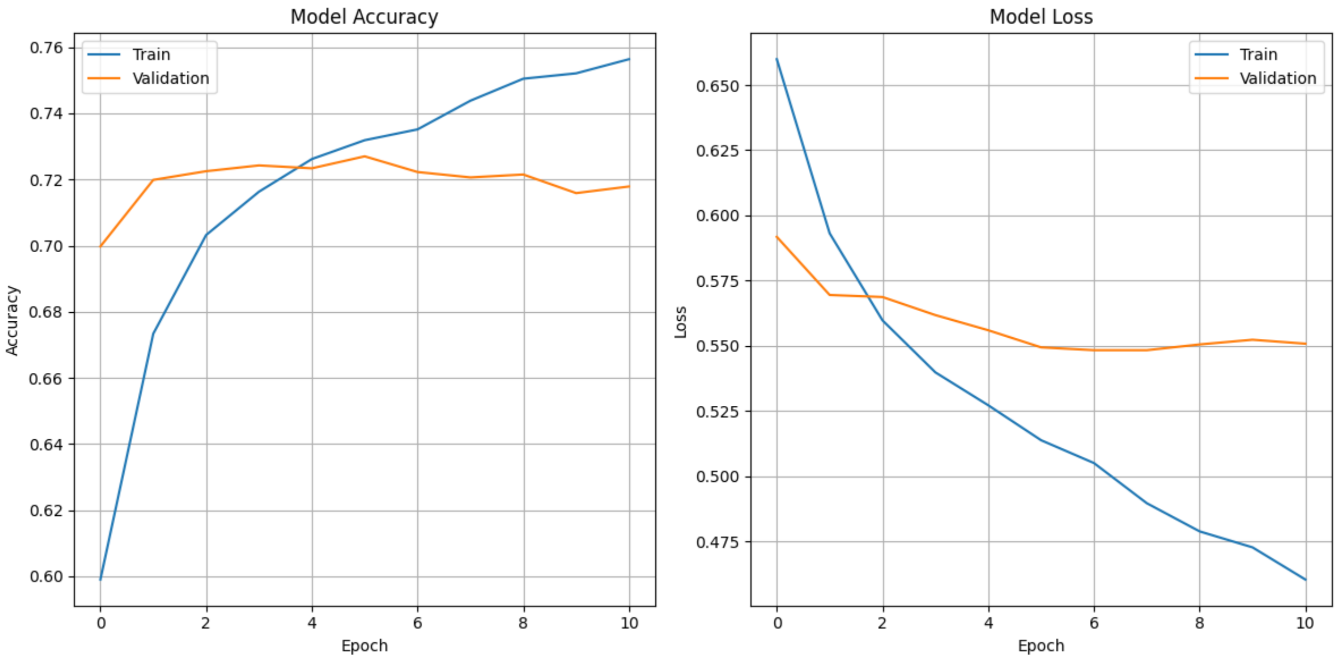
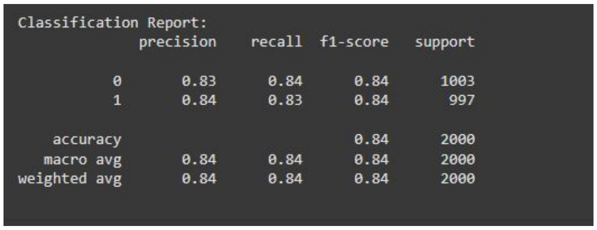


Figure 1: Training and validation accuracy and loss for the best NN model



```
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.84      0.84      1003
           1       0.84      0.83      0.84       997

    accuracy                           0.84      2000
   macro avg       0.84      0.84      0.84      2000
weighted avg       0.84      0.84      0.84      2000
```

Figure 2: Classification report for the best NN model

The confusion matrix (figure 3) provides further insights into the model's classification behavior. 845 negative samples were correctly classified, while 158 were misclassified as positive. Similarly, 829 positive samples were correctly identified, with 168 being incorrectly labeled as negative. The distribution of false positives and false negatives is relatively balanced, implying that the model does not disproportionately favor one class over the other. However, the presence of 326 misclassified instances (sum of false positives and false negatives) suggests that there is still room for improvement, possibly through hyperparameter tuning or the incorporation of more sophisticated architectures.
The ROC curve (figure 4) further supports the effectiveness of the model. The AUC (Area Under the Curve) score of 0.90 indicates strong discriminative ability, as the model effectively distinguishes between positive and negative sentiments. The steep rise in the ROC curve suggests that a high proportion of true positives are captured while maintaining a relatively low false positive rate. This confirms that the neural network exhibits reliable classification performance, with minimal signs of overfitting or underfitting.

6.2 Recurrent Neural Network (RNN)

Best hyperparameters:

- Neurons: 32
- Learning Rate: 0.001
- Dropout Rate: 0.0

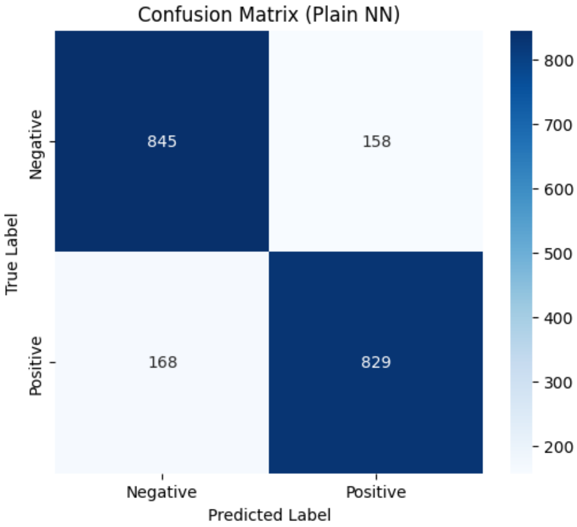


Figure 3: Classification matrix of the best NN model

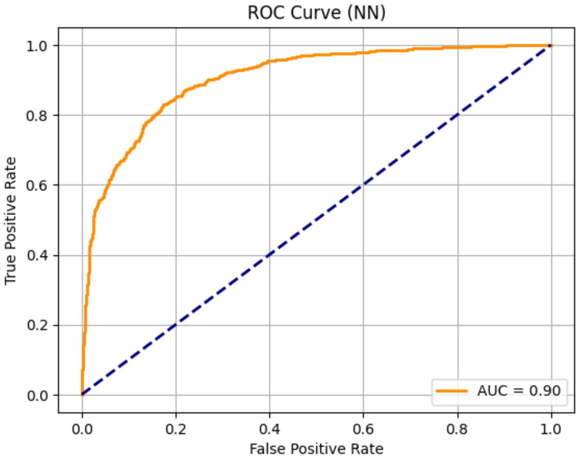


Figure 4: ROC curve for the best NN model

- Batch Size: 64
- Epochs: 50
- L2 rate: 0.001

The best validation accuracy achieved for the RNN model was approximately 83.0%. The learning curves for the best model are visible in figure 5.
demonstrates a classification performance comparable to the plain neural network (NN), achieving an overall accuracy of 83%. The classification report (figure 6) indicates that both the negative and positive classes have an F1-score of 0.83, with precision and recall values remaining balanced. The model correctly classifies a high proportion of samples from both classes, suggesting effective sentiment differentiation. However, compared to the plain NN, there is a slight decrease in accuracy, which may indicate that the RNN architecture has not fully leveraged sequential dependencies in this particular dataset.
The confusion matrix (figure 7) provides a deeper look into misclassification patterns. The model correctly classified 825 negative samples, while 178 were misclassified as positive. Conversely, 837 were correctly labeled as positive, while 160 were misclassified as negative. The false positive rate is slightly higher than in the plain NN, which may suggest that the RNN model is more prone to mistaking negative sentiments for positive ones. Despite this, the false negative rate is slightly lower, implying that the model might be slightly better at recognizing positive sentiment. The total number of misclassified instances (338) is slightly higher than that of the plain NN model (326), further confirming that this model does not necessarily outperform the previous one in overall accuracy.
The ROC curve (figure 8) offers an additional perspective, with an AUC score of 0.91, slightly outperforming the plain NN model's 0.90. This suggests that while the RNN model may have a similar overall classification accuracy, it has a marginally stronger ability to distinguish between classes across different classification thresholds. The curve maintains a steep rise at the beginning, indicating that a high proportion of true positives are captured while keeping false positives relatively low. This improved AUC score may suggest that, while the model struggles slightly more with exact classification, it retains strong predictive capabilities when adjusting decision thresholds.
So the RNN model demonstrates comparable accuracy to the plain NN, with slight trade-offs in false positive and false negative rates. While the AUC score suggests a stronger discriminative ability, the overall classification metrics do not show a clear improvement over the simpler NN model.

6.3 Bi-directional RNN

Best hyperparameters:

- Neurons: 128
- Learning Rate: 0.001
- Dropout Rate: 0.6
- Batch Size: 64
- Epochs: 50
- L2 rate: 0.001

The best validation accuracy achieved for the bi-directional RNN model was approximately 82.20%. The learning curves for the best model are visible in figure 9.
From the classification report (figure 10), the model achieves an overall accuracy of 82%, with a precision of 0.82 for class 0 and 0.83 for class 1. The recall values are 0.83 for class 0 and 0.82 for class 1, leading to an F1-score of 0.82 for both classes. These results indicate a balanced performance across both classes, suggesting that the model does not exhibit a strong bias toward either class.
The confusion matrix (figure 11) provides a more detailed view of the misclassification patterns. The bi-directional RNN correctly classifies 833 negative samples and 813 positive samples, while 170 negative instances are misclassified as positive, and 184 positive instances are misclassified as negative. This shows a relatively even distribution of errors, with a slight tendency to misclassify positive samples

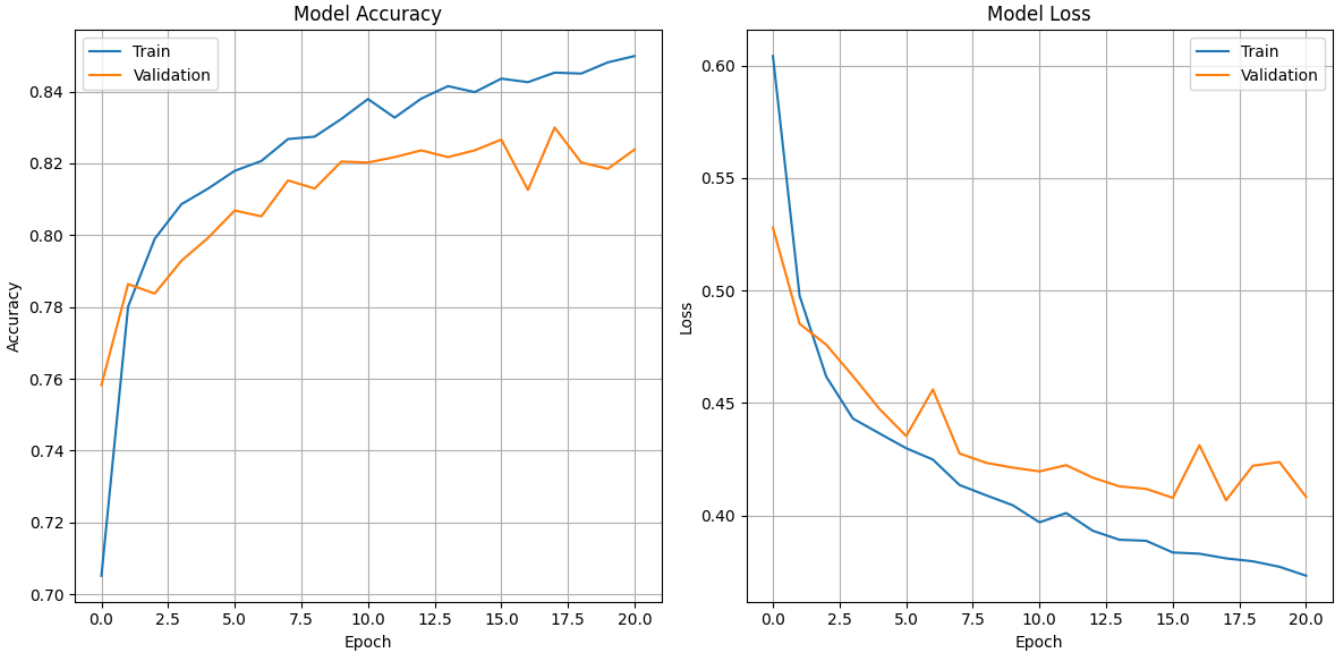
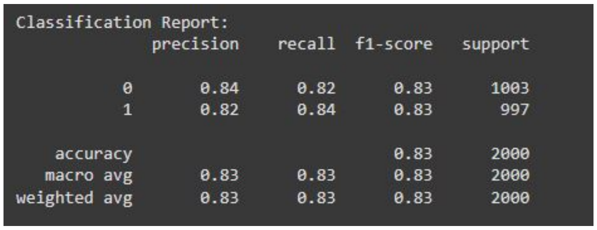


Figure 5: Training and validation accuracy and loss for the best RNN model



```
Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.82      0.83      1003
           1       0.82      0.84      0.83       997

    accuracy                           0.83      2000
   macro avg       0.83      0.83      0.83      2000
weighted avg       0.83      0.83      0.83      2000
```

Figure 6: Classification report for the best RNN model

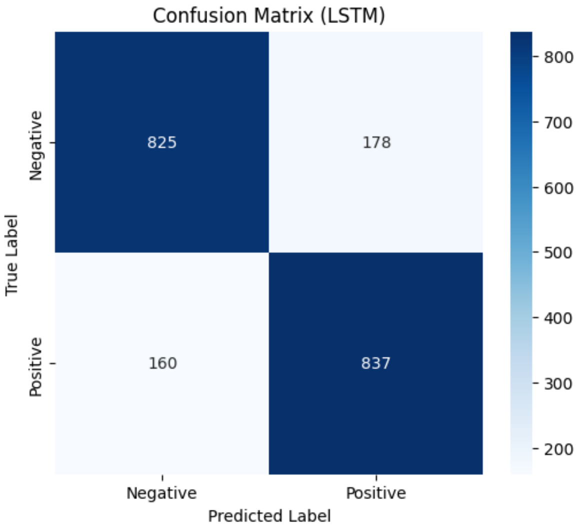


Figure 7: Classification matrix of the best RNN model

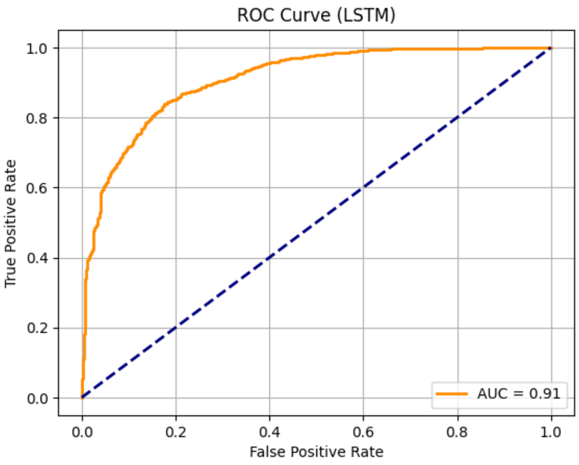


Figure 8: ROC curve for the best RNN model

more frequently.
The ROC curve (figure 12) further supports the model's effectiveness in distinguishing between classes. The AUC value of 0.90 indicates strong discriminatory power, suggesting that the model performs well in ranking predictions and distinguishing between positive and negative cases.
Overall, while the bi-directional RNN captures sequential dependencies in both forward and backward directions, its performance does not significantly exceed that of standard RNN models. The classification metrics and AUC suggest that the model is effective, but its improvements over simpler architectures might be marginal.

7 Strenghts & Weaknesses

- Strenghts:
  1. Use of Pre-trained Embeddings: The use of pre-trained GloVe embeddings helps in capturing semantic information from the text, which improves the model's performance.
  2. Thorough Data Preprocessing: Extensive data preprocessing steps, including text cleaning, tokenization, and stopword removal, ensure that the data fed into the model is clean and relevant.
  3. Variety of Models Evaluated: Evaluating different types of models (Plain NN, RNN, bi-directional RNN) allows for a comprehensive understanding of which model architecture works best for this specific problem.
- Weaknesses:
  1. Limited Dataset: The dataset, although balanced, is limited to movie reviews from IMDB. The model may not generalize well to other types of text data or reviews from different domains.
  2. Computationally Intensive: Training models, especially RNNs and bi-directional RNNs, is computationally intensive and time-consuming, which may not be feasible for all users.
  3. Potential Overfitting: Despite using dropout layers, there is still a risk of overfitting, especially with complex models like RNNs and bi-directional RNNs. This might require additional techniques such as regularization or data augmentation to mitigate.

8 Conclusion

This project investigated the performance of different neural network architectures for a sentiment analysis task, comparing a standard feedforward neural network (NN), a recurrent neural network

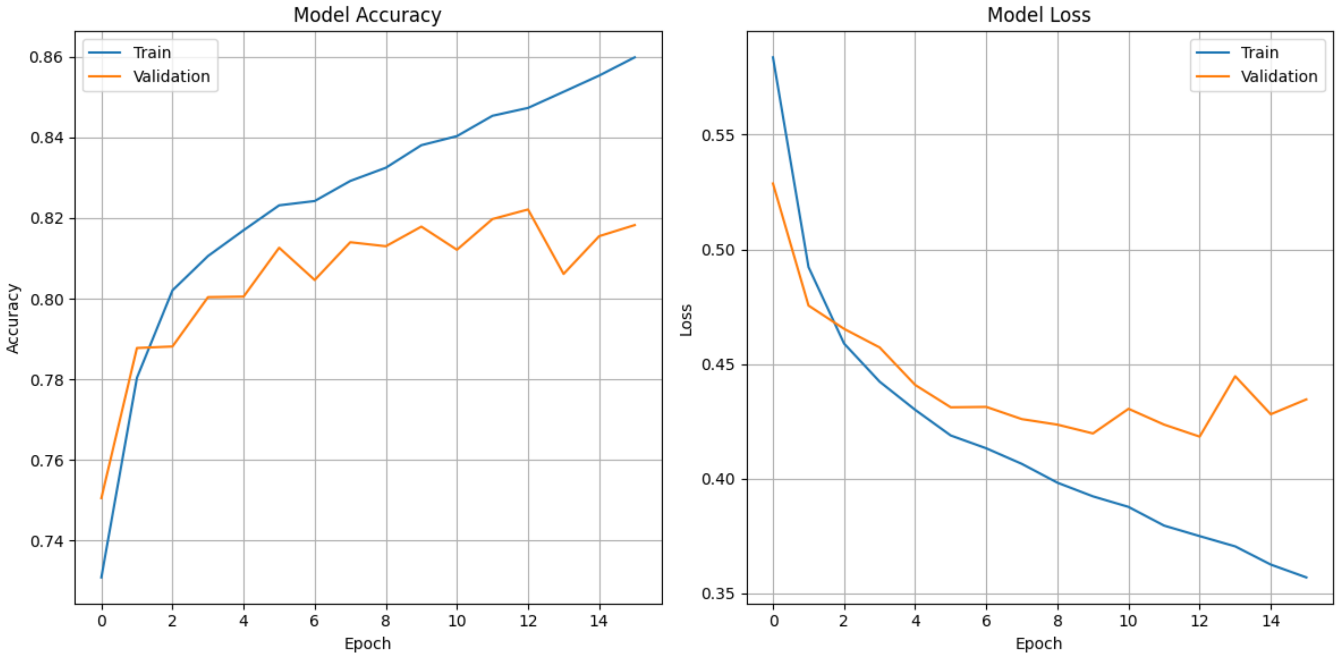
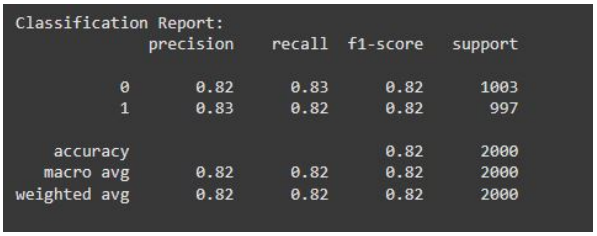


Figure 9: Training and validation accuracy and loss for the best bi-directional RNN model



```
Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.83      0.82      1003
           1       0.83      0.82      0.82       997

    accuracy                           0.82      2000
   macro avg       0.82      0.82      0.82      2000
weighted avg       0.82      0.82      0.82      2000
```

Figure 10: Classification report for the best bi-directional RNN model

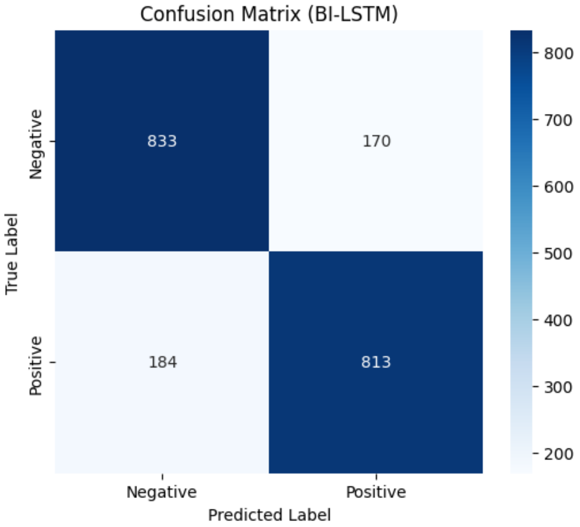


Figure 11: Classification matrix of the best bi-directional RNN model

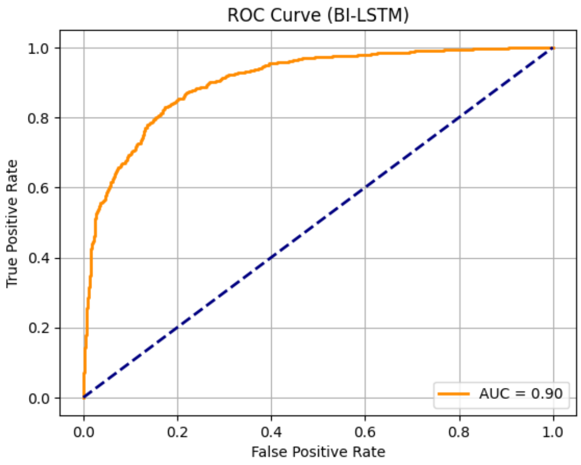


Figure 12: ROC curve for the best bi-directional RNN model

(RNN) with LSTM units, and a bi-directional RNN with LSTM units. The best-performing models for each architecture were obtained through an extensive grid search, ensuring optimal hyperparameter selection. The evaluation was conducted using key performance metrics, including accuracy, precision, recall, F1-score, confusion matrices, and ROC curves.

The results indicate that recurrent architectures significantly outperform the standard NN model. The best validation accuracy achieved was 72% for the NN, 83% for the RNN with LSTM, and 82.20% for the bi-directional RNN. These findings highlight the importance of incorporating sequential dependencies, as RNN-based models demonstrated superior performance in capturing temporal patterns within the data.

The classification reports and confusion matrices further confirm this trend. The RNN achieved the highest overall accuracy and F1-scores, demonstrating a well-balanced trade-off between precision and recall. The bi-directional RNN model, while also effective, did not provide a substantial improvement over the standard LSTM, suggesting that bidirectional processing did not yield significant performance gains in this specific context.

Moreover, the ROC curves reveal that both RNN-based models exhibit strong discriminatory power, with AUC values of 0.91 for RNN and 0.90 for bi-directional RNN. These values confirm the robust classification capabilities of recurrent architectures compared to the traditional NN model.

In conclusion, this study underscores the advantage of RNN-based architectures for the given classification task. While the bi-directional RNN model did not surpass the standard RNN in terms of validation accuracy, both recurrent models outperformed the NN. Future research could explore additional optimizations, such as fine-grained hyperparameter tuning, attention mechanisms, or alternative recurrent architectures, to further enhance performance. A GitHub repository of this project is available here.

References

[1] Wei Li, Wei Shao, Shaoxiong Ji, and Erik Cambria. Bieru: Bidirectional emotional recurrent unit for conversational sentiment analysis. *Neurocomputing*, 467:73–82, 2022.

[2] Karthik Gopalakrishnan and Fathi M Salem. Sentiment analysis using simplified long short-term memory recurrent neural networks. *arXiv preprint arXiv:2005.03993*, 2020.

[3] Sharat Sachin, Abha Tripathi, Navya Mahajan, Shivani Aggarwal, and Preeti Nagrath. Sentiment analysis using gated recurrent neural networks. *SN Computer Science*, 1:1–13, 2020.

[4] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.