MINOR PROJECT REPORT

On

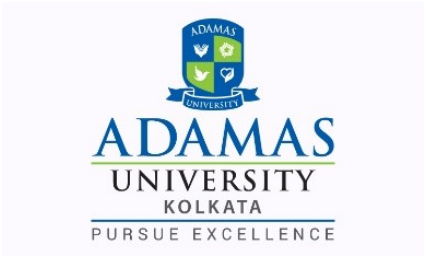
Recommendation of Drugs using Sentimental Analysis

Submitted in partial fulfilment of the requirements for the award of

Bachelor of Technology (B.Tech)

In the department of

Computer Science & Engineering



Submitted by:

Tamanna Saha (UG/02/BTCSE/2022/060)

Chaiti Bain (UG/02/BTCSE/2022/055)

Falguni Yadav (UG/02/BTCSE/2022/012)

Ruchi Singh (UG/02/BTCSE/2022/054)

Under the Guidance of

Mr. Abhinandan Ghosh

(Assistant Professor, Adamas University)

School of Engineering & Technology

ADAMAS University, Kolkata, West Bengal

Aug 2025 - Dec 2026

CERTIFICATE

This is to certify that the project report entitled *"<Recommendation of Drugs using Sentimental Analysis>"*, submitted to the School of Engineering Technology (SOET), **ADAMAS UNIVERSITY, KOLKATA** in partial fulfilment for the completion of Semester – 7^{th} of the degree of **Bachelor of Technology** in the department of **Computer Science & Engineering**, is a record of bonafide work carried out by **<Chaiti Bain, UG/02/BTCSE/2022/055>, <Tamanna Saha, UG/02/BTCSE/2022/060>, <Falguni Yadav, UG/02/BTCSE/2022/012>, <Ruchi Singh, UG/02/BTCSE/2022/054>** under our guidance.

All help received by us from various sources have been duly acknowledged.
No part of this report has been submitted elsewhere for award of any other degree.

Abhinandan Ghosh

(Assistant Professor)

Mr. Aninda Kundu / Mr. Sayantan Singha Roy

(Project Coordinator)

Dr. Sajal Saha

(Asso. Dean & HOD CSE)

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Adamas University, School of Engineering & Technology, for providing us with the opportunity and necessary facilities to carry out this minor project titled *"Recommendation of Drugs Using Sentiment Analysis."*

We take this opportunity to extend our heartfelt thanks to our project guide, **Mr. Abhinandan Ghosh (Assistant Professor, CSE Department)**, for his continuous support, valuable guidance, constructive feedback, and constant encouragement throughout the development of this project. His expertise and mentorship have been instrumental in shaping our understanding and enabling the successful completion of this work.

We also extend our appreciation to all the faculty members of the **Computer Science & Engineering Department** for their academic support, motivation, and cooperation during the project period.

We are thankful to our friends and classmates for their continuous support and helpful discussions, which contributed significantly to the successful completion of this project.

Last but not least, we express our deep gratitude to our families for their constant encouragement, patience, and moral support throughout our academic journey.

DECLARATION

We, the undersigned, declare that the project entitled *"Recommendation of Drugs using Sentimental Analysis"*, being submitted in partial fulfillment for the award of Bachelor of Technology Degree in Computer Science & Engineering, affiliated to ADAMAS UNIVERSITY, is the work carried out by us.

Tamanna Saha	**Ruchi Singh**
(UG/02/BTCSE/2022/060)	(UG/02/BTCSE/2022/054)
Chaiti Bain	**Falguni Yadav**
(UG/02/BTCSE/2022/055)	(UG/02/BTCSE/2022/012)

ABSTRACT

This project presents a comprehensive analysis of patient-generated reviews related to various medications used for different health conditions. The primary objective of this study is to understand real-world user experiences, drug effectiveness, and overall sentiment expressed by patients through textual feedback. The dataset used for this analysis contains essential attributes such as the name of the medicine, the corresponding medical condition, user ratings, review text, review dates, and usefulness counts. These attributes provide a rich source of information for evaluating how individuals perceive the impact of specific drugs on their health. Furthermore, the study investigates the correlation between written sentiment and numerical ratings provided by the users. This helps identify whether textual sentiments align with the given rating scores and how strongly reviews reflect medication effectiveness. Patterns such as frequently used keywords, common complaints, recurring side effects, and positive indicators of treatment success are also explored. By visualizing trends and summarizing key observations, the project highlights the variations in patient responses for different drugs and conditions. The insights derived from this analysis have practical significance in the healthcare domain. They can support medical professionals in understanding real-world outcomes of drug usage, assist pharmaceutical companies in improving medications based on user responses, and provide valuable information to patients seeking peer experiences before beginning a treatment. Overall, this project demonstrates the power of sentiment analysis and data-driven research in enhancing healthcare decision-making and improving patient awareness.

KEYWORDS: Data Preprocessing, Review Rating Correlation, Medication Effectiveness, Text Mining, Real-World Evidence.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
BiLSTM	Bidirectional Long Short-Term Memory
ATT-LSTM	Attention-based Long Short-Term Memory
BiLSTM-ATT	Bidirectional LSTM with Attention
CNN	Convolutional Neural Network
GRU	Gated Recurrent Unit
BiGRU	Bidirectional Gated Recurrent Unit
NN	Neural Network
DL	Deep Learning
NLP	Natural Language Processing
SA	Sentiment Analysis
AU	Adamas University, Barasat
RTFM	Read the Fine Manual

CHAPTER 1

INTRODUCTION

In today's digital era, a large amount of health-related information is shared online by patients in the form of reviews, comments, feedback, and discussions. These user-generated texts often contain valuable insights about the effectiveness, side effects, and overall satisfaction associated with various drugs. Such information can play an important role in helping patients and healthcare practitioners make better decisions about medication selection. [1] [2]

Traditional drug recommendation systems rely mainly on clinical data and doctor prescriptions, which may not fully reflect real-world user experiences. At the same time, manually reading thousands of reviews is time-consuming and impractical. This creates a need for an automated system that can analyze public opinion and derive meaningful conclusions. [3]

Sentiment analysis, a widely used technique in Natural Language Processing (NLP), helps classify text into positive, negative, or neutral sentiments. By applying sentiment analysis to drug reviews, it becomes possible to understand how patients feel about different medications and identify which drugs are more preferred or effective according to public opinion. [4] [5]

This project focuses on developing a Drug Recommendation System using Sentimental Analysis, where patient reviews are collected, processed, and analyzed to extract sentiment scores. Based on these insights, the system recommends suitable drugs for particular health conditions. The project aims to bridge the gap between medical data and real-world patient experiences, providing a more informed and user-centric approach to drug selection.[6]

1.1 Background

In recent years, the availability of online health-related data has increased significantly as patients frequently share their experiences, reviews, and opinions about medicines on platforms such as healthcare forums, review websites, and social media. These reviews often contain

valuable information about the effectiveness of drugs, side effects, user satisfaction, and overall treatment outcomes.[7]

However, this information is unstructured and difficult to analyze manually due to its large volume. Traditional drug recommendation methods rely heavily on clinical data, doctor prescriptions, or pharmacological studies, but they often do not include real-time user feedback. As a result, an opportunity exists to use data-driven approaches to analyze public sentiment and derive meaningful insights from patient-generated content.[8]

Sentiment analysis, a branch of natural language processing (NLP), enables automatic identification of positive, negative, or neutral opinions from text. When applied to drug reviews, sentiment analysis can help understand user experiences and evaluate drug performance. By combining sentiment analysis with recommendation techniques, we can create a system that suggests drugs based on real-world feedback, providing a more informed and patient-centered perspective.[9] [10]

Thus, this project explores how sentiment analysis can be used to improve drug recommendation systems, making medication selection more transparent, data-driven, and aligned with user experiences.

1.2 Purpose of the Project

The purpose of this project is to develop an intelligent system that can analyze patient reviews and feedback using sentimental analysis techniques to recommend the most suitable and effective drugs for specific medical conditions. By extracting useful insights from large volumes of online drug reviews, the project aims to support patients, healthcare learners, and researchers in making data-driven decisions about medication selection.

This system will help identify overall public sentiment, highlight commonly preferred drugs, detect negative reactions, and present recommendations based on real-world user experiences. Ultimately, the project seeks to simplify drug comparison, improve understanding of drug effectiveness, and enhance healthcare decision-making through automated sentiment analysis.

1.3 Problem Statement

Online healthcare forums, drug review websites, and patient feedback platforms contain a large amount of user opinions describing their experiences with different medicines. However, these reviews are unstructured and difficult to interpret manually. As a result, patients and healthcare practitioners often struggle to identify which drugs are perceived as effective, safe, or preferable for specific medical conditions.

Current drug recommendation approaches rely mostly on clinical guidelines or predefined rules, and they do not utilize the valuable insights present in real-world patient reviews. There is a lack of an automated system that can analyze this textual data, extract sentiment, and recommend suitable drugs based on public opinion and user experience.[11] [12]

Therefore, this project addresses the problem of developing an intelligent Drug Recommendation System using Sentimental Analysis.[13] The system aims to analyze patient reviews, classify sentiment (positive, negative, or neutral), compute drug effectiveness scores, and recommend the most relevant drugs for particular diseases. This will assist users in making informed choices and enhance the decision-making process in healthcare.[14]

CHAPTER 2

LITERATURE REVIEW

The rapid growth of digital healthcare platforms has generated vast amounts of user content, such as drug reviews, patient feedback, and treatment experiences. Researchers have increasingly explored the use of sentiment analysis and machine learning to extract meaningful insights from such textual data. This section reviews existing literature relevant to drug recommendation systems, sentiment analysis techniques, and text-based opinion mining. [15]

Table 2.1: Literature Survey

Ref.	**Paper Name**	**Dataset**	**Method**	**Contribution**	**Limitation**
[1]	Al-Hadhrami et al., 2024	Drug Review Dataset	CNN and LSTM integration	Proposed hybrid deep architectures	Static embeddings underperformings
[2]	Min, 2019 (ICAICA)	Drugs.com	Weakly supervised model	Reduced manual labeling effort	Low accuracy for noisy reviews
[3]	Na and Kyaing, 2015	Web drug reviews	Lexicon + ML	Early framework for drug sentiment mining	Does not support multi-class sentiment
[7]	Han et al., 2020 (IEEE Access)	Drugs.com	Double BiGRU + Transfer learning	High accuracy aspect-level analysis	Requires large labeled data

2.1 Sentiment Analysis on Drug Review Platforms

Early research emphasized extracting opinions from drug review websites using traditional machine learning methods. These works focused on feature engineering approaches such as bag-of-words and TF-IDF, combined with classifiers like SVM, Naïve Bayes, and Logistic Regression. Their contributions mainly centered on identifying sentiment patterns in large-scale patient reviews, but their performance was limited for long and complex text. [9]

P. Pak and P. Paroubek (2010) introduced early sentiment classification methods using machine learning algorithms such as Naïve Bayes and SVM. Their work laid the foundation for applying sentiment analysis across multiple domains, including healthcare.[4] [13]

2.2 Deep Learning Approaches

With advancements in neural networks, researchers shifted toward models such as LSTM, BiLSTM, GRU, and CNN. These architectures provided substantial improvements in capturing contextual meaning and long-term dependencies within patient reviews.[16] Studies comparing deep learning architectures (such as LSTM vs. BiLSTM vs. CNN) showed that hybrid and bidirectional models achieve higher accuracy, particularly when reviews contain multi-sentence descriptions of symptoms and side effects.[17] [18]

2.3 Aspect-Based Sentiment Analysis

Recent research has extended beyond simple polarity classification to extract fine-grained aspects like drug effectiveness, dosage experience, and side effects. Aspect-based sentiment analysis allows deeper understanding of patient concerns. These methods rely on neural architectures with attention mechanisms or knowledge transfer, enabling better identification of specific complaint categories. [19] [20]

2.4 Hybrid Recommendation and Prediction Models

Some studies integrate sentiment analysis with drug recommendation systems. These models use sentiment outcomes to suggest drugs to patients based on collective experiences and satisfaction scores. RNN-based or ML-driven recommendation systems demonstrate how sentiment analysis can improve personalization in healthcare, although they often suffer from data imbalance and lack of interpretability.[21] [22]

2.5 Research Gap

Although existing research has greatly advanced drug review analysis, several limitations remain:

- Many studies focus only on sentiment classification but do not provide drug recommendations.
- Some systems rely only on ratings, ignoring textual details such as side effects.
- Deep learning models require large computational resources unsuitable for minor projects.
- Very few works analyze sentiment trends or common side-effects extracted from reviews.

These gaps provide motivation for developing a system that combines sentiment analysis with drug recommendation logic.

CHAPTER 3

METHODOLOGY

This chapter explains the systematic workflow followed in building, training, evaluating, and comparing multiple deep learning models for sentiment analysis of drug reviews. The methodology covers data collection, preprocessing, representation, model development, evaluation, visualizations, and insights.

3.1 Data Collection

The dataset used in this project is the publicly available **Drug Review Dataset**, divided into:

- **Training Set:** 110,811 reviews
- **Validation Set:** 27,703 reviews
- **Test Set:** 46,108 reviews

Each record contains:

- **review** – User-generated text describing medication experience
- **rating** – Numerical score (1–10)

Ratings were later converted into sentiment categories.

3.2 Data Preprocessing

Before training the models, the drug review dataset was carefully preprocessed to make the text clean, uniform, and suitable for deep learning. First, all missing or empty reviews were removed to avoid noise. Then, the text was converted to lowercase, and unnecessary characters such as punctuation marks, numbers, HTML tags, and special symbols were eliminated. Stopwords like “the,” “is,” and “and,” which do not add meaning to sentiment, were also removed

to simplify the text. After cleaning, the reviews were tokenized, meaning each sentence was broken into individual words. These words were then converted into numerical sequences using a tokenizer, and all sequences were padded or truncated to a fixed length so that every review had the same size input for the neural networks. Finally, the sentiment labels were encoded into numerical form (0, 1, 2), allowing the models to process them effectively. This entire preprocessing pipeline helped reduce noise, standardize the text, and prepare the dataset for accurate and efficient model training.

3.2.1 Handling Missing Values

The dataset was checked for missing entries. No missing values were found in reviews or ratings.

3.2.2 Text Cleaning

Each review underwent a sequence of preprocessing operations:

- Lowercasing
- Removing URLs
- Removing punctuation
- Removing extra whitespace
- Removing stopwords
- Stemming using Porter Stemmer

3.2.3 Sentiment Label Creation

Ratings were mapped to sentiment labels as follows:

- Rating ≥ 7: **Positive**
- Rating 4–6: **Neutral**
- Rating ≤ 3: **Negative**

Labels were encoded numerically using `LabelEncoder`.

3.2.4 Tokenization and Padding

The text was tokenized using a vocabulary size of 10,000 words. All sequences were padded to a fixed length of 150 tokens.

3.3 Exploratory Data Analysis (EDA)

EDA was performed to understand the structure and distribution of the data. Key observations included:

- Positive reviews were slightly more frequent than neutral or negative.
- Review lengths varied considerably.
- Text contained medical expressions, symptoms, and patient experiences.

These findings informed design choices for preprocessing and model selection.

3.4 Text Representation

Text was converted into dense numerical vectors using an embedding layer:

```
Embedding(max_words, 128, input_length=max_len)
```

This creates 128-dimensional vector representations that capture semantic relationships between words.

3.5 Sentiment Analysis / Classification

Multiple deep learning models were implemented:

3.5.1 Models Used

1. Recurrent Neural Network (RNN)

2. Long Short-Term Memory (LSTM)
3. Bidirectional LSTM (BiLSTM)
4. Attention-based LSTM
5. BiLSTM with Attention
6. Convolutional Neural Network (CNN)
7. Gated Recurrent Unit (GRU)
8. Bidirectional GRU (BiGRU)

3.5.2 Training Setup

- Epochs: 15
- Batch size: 30
- Loss function: Sparse Categorical Cross-Entropy
- Optimizer: Adam
- Metric: Accuracy

Each model was trained using the training set and validated on the validation set.

3.6 Model Evaluation

To understand how well the deep learning models performed, several evaluation metrics were used, including accuracy, precision, recall, F1-score, and the confusion matrix. These metrics helped measure not just how often the models predicted correctly, but also how reliable and consistent those predictions were for each sentiment class. After training, the models were tested on unseen data to check their real-world performance. Among all the architectures, LSTM and BiLSTM showed more stable and balanced results due to their ability to capture long-term dependencies in review text. GRU and BiGRU also performed competitively with faster training times, while CNN-based models showed strong performance for shorter patterns in text. The models' confusion matrices revealed how well each model distinguished between positive, negative, and neutral sentiments, highlighting areas where misclassifications still occurred. Overall, the evaluation process ensured that the selected model not only achieved good

accuracy but also generalized well, making it suitable for practical sentiment analysis on drug review datasets.

3.6.1 Evaluation Metrics

- Accuracy
- Precision
- Recall
- F1-Score
- ROC–AUC (One-vs-Rest)

3.6.2 Confusion Matrices

The confusion matrix was used to closely examine how each deep learning model classified the sentiments in the drug review dataset. Instead of looking only at accuracy, the confusion matrix provided a more detailed picture by showing the exact number of correct and incorrect predictions for each sentiment class—positive, negative, and neutral. It helped identify where the models were performing well and where they struggled. For example, most models showed high true positives for the positive sentiment class, indicating that users' positive reviews were easier to detect. However, some models frequently misclassified neutral reviews as either positive or negative, which revealed the ambiguity often present in moderately expressed opinions. The confusion matrix also exposed cases where negative sentiments were incorrectly labeled as positive, usually due to mixed wording or medically complex reviews. By analyzing these patterns, it became clear which models were more balanced and which required further tuning. Overall, confusion matrices played an essential role in understanding the real strengths and limitations of each model beyond simple accuracy numbers.

3.6.3 Performance Comparison

A consolidated evaluation table was generated containing:

- Test Accuracy
- Precision

- Recall
- F1 Score
- ROC–AUC

This allowed ranking of models based on their predictive performance.

3.7 Visualization and Reporting

Multiple visualizations were created from the model outputs:

- Training and validation accuracy curves
- Training and validation loss curves
- Confusion matrices for all models
- ROC curves for negative, neutral, and positive classes
- Bar chart comparing test accuracy across all models

These plots provided insights into model learning behavior and generalization ability.

3.8 Conclusion and Insights

From the analysis:

- **BiLSTM + Attention** demonstrated the strongest overall performance.
- CNN achieved high accuracy but showed signs of overfitting.
- GRU and BiGRU delivered strong performance with faster training.
- RNN struggled with long-term dependencies compared to LSTM-based architectures.

Deep learning proved highly effective for sentiment classification of drug reviews.

3.9 Workflow

Figure 3.1 shows an overview of the project workflow.

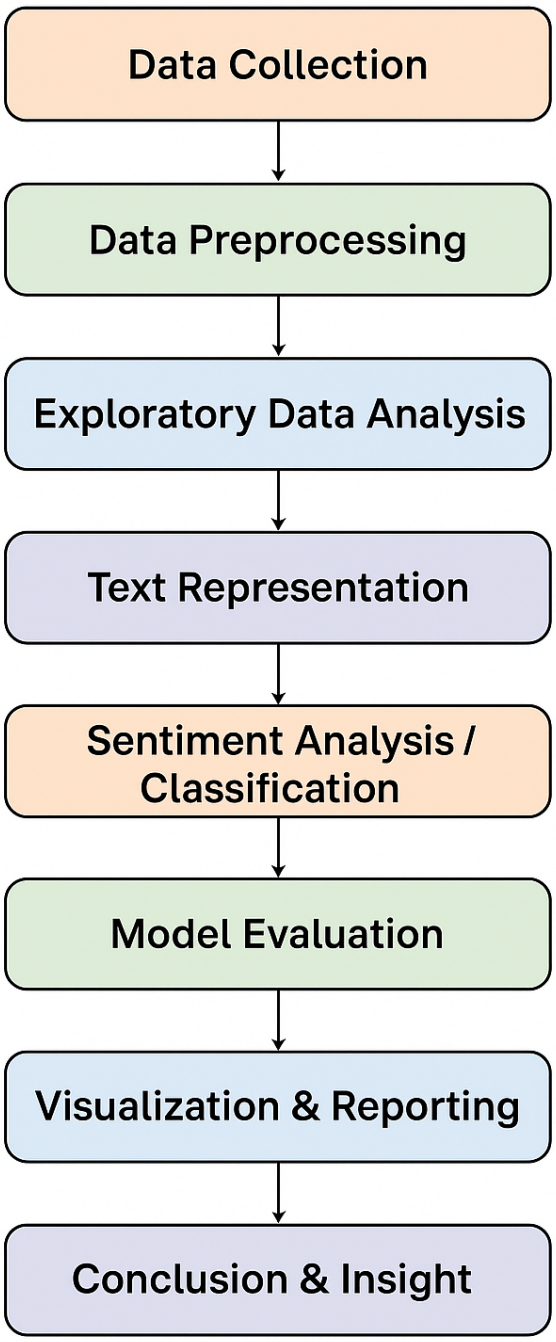


Figure 3.1: Overview of Project Workflow

3.10 Attributes / Features Description

Table 3.1: Dataset attribute descriptions

Attribute	**Description**
review	Original user review text
rating	Numeric score (1–10)
sentiment	Derived category (positive, neutral, negative)
clean_review	Preprocessed review text
sentiment_encoded	Numeric representation of sentiment

3.11 Data Types

Table 3.2: Data types of dataset columns

Column	**Type**
review	String
rating	Float
sentiment	Categorical
sentiment_encoded	Integer
clean_review	String

3.12 Data Characteristics

The dataset exhibits the following characteristics:

- Large dataset with over 180,000 total entries
- Class distribution slightly skewed toward positive reviews
- Reviews contain medical terminology and personal experiences
- Suitable for NLP-based sentiment analysis tasks

3.13 Data Usage in This Project

Data was used for:

- Training deep learning models
- Validating and testing performance
- Creating comparative visualizations
- Performing multi-class sentiment classification

3.14 Algorithms Used

3.14.1 Recurrent Neural Network (RNN)

A Recurrent Neural Network (RNN) is a type of artificial neural network designed to process sequential data by maintaining a memory of past inputs. Unlike traditional networks that process inputs independently, RNNs use feedback loops to pass information from one step to the next, allowing them to recognize patterns in sequences like text, speech, and time-series data. This internal memory is crucial for tasks where the order of information matters, such as predicting the next word in a sentence or generating music.

3.14.2 Long Short-Term Memory (LSTM)

What is LSTM (Long Short Term Memory)?LSTM, or Long Short-Term Memory, is a type of recurrent neural network (RNN) designed to process sequential data and learn long-term dependencies. It uses a special structure with "gates" and a "memory cell" to selectively remember or forget information, which helps overcome limitations of traditional RNNs, such as the vanishing gradient problem. LSTMs are widely used in applications like speech recognition, natural language processing, and time-series forecasting.

3.14.3 Bidirectional LSTM (BiLSTM)

A Bidirectional LSTM (BiLSTM) is a type of recurrent neural network (RNN) that processes sequential data in both forward and backward directions, allowing it to capture context from both past and future inputs. It consists of two separate LSTM layers—one processing the sequence from start to end, and another from end to start—which are then combined to provide a more comprehensive understanding of the data. BiLSTMs are especially effective for tasks that require a full sentence or sequence context, such as sentiment analysis, machine translation, and named entity recognition.

3.14.4 Attention Mechanism (LSTM + Attention)

Attention allows the model to focus on important words in a review by assigning weights to them. It highlights keywords such as “side effects”, “effective”, and “pain reduced”. This improves interpretability and prediction quality.

3.14.5 BiLSTM + Attention

This model combines:

- Bidirectional LSTM for complete context
- Attention for highlighting important words

It often achieves high performance for sentiment analysis tasks.

3.14.6 Convolutional Neural Network (CNN for Text)

CNN, commonly used for images, can also process text. It uses filters to detect local patterns like phrases or keywords, extracting n-gram features automatically. It is fast and efficient for large datasets.

3.14.7 Gated Recurrent Unit (GRU)

GRU is similar to LSTM but simpler and faster, using only reset and update gates. It is less computationally expensive and performs well even on smaller datasets.

3.14.8 Bidirectional GRU (BiGRU)

BiGRU extends GRU by reading text in both directions (forward and backward). It captures context better than a simple GRU and works efficiently on both long and short reviews.

3.15 Implementation

The implementation phase involves the complete development of the drug review sentiment analysis system using machine learning and deep learning techniques. This section explains the environment setup, dataset preparation, model building, and evaluation steps performed during the project.

3.16 Environment Setup

The entire implementation was carried out using tools and platforms that support machine learning and natural language processing (NLP). The main tools used are:

- Google Colab for coding, execution, and GPU support
- Python 3.x as the programming language
- Libraries Used:
  - NumPy, Pandas – for data preprocessing

- Matplotlib, Seaborn – for data visualization
- NLTK / spaCy – for text cleaning
- TensorFlow / Keras – for RNN, LSTM, BiLSTM, GRU models
- Scikit-learn – for splitting and evaluation metrics

3.17 Dataset Collection and Description

The dataset was collected from publicly available drug review sources. It contains:

- User Reviews: Text written by patients about medicines
- Drug Names
- Medical Conditions
- Ratings

The dataset was loaded into Colab and explored to understand its structure, missing values, and class distribution.

3.18 Data Preprocessing

Since drug reviews are unstructured text, the following preprocessing steps were applied:

1. Lowercasing all text
2. Removing special characters, numbers, and punctuation
3. Tokenization to split sentences into words
4. Lemmatization to convert words to their base form
5. Padding and encoding to convert text into numerical form for deep learning models

These steps helped in improving model accuracy and training stability.

3.19 Model Development

Several deep learning models were implemented to compare performance:

- **Recurrent Neural Network (RNN)**

  A simple RNN model with embedding and dense layers was built to understand sequential processing.

- **Long Short-Term Memory (LSTM)**

  LSTM was implemented to handle long-term dependencies in drug reviews.

- **Bidirectional LSTM (BiLSTM)**

  BiLSTM processes text in both forward and backward directions and improves context understanding.

- **GRU and BiGRU**

  GRU models provide faster training and similar accuracy compared to LSTM.

- **CNN Model for Text Classification**

  CNN layers were applied to capture local patterns and important n-grams in reviews.

- **Attention-Based Models**

  LSTM + Attention and BiLSTM + Attention models were built to focus on the most important words in each review.

Each model consisted of:

- Embedding layer
- RNN/LSTM/GRU/CNN layers
- Dropout regularization
- Dense output layer with softmax activation

3.20 Training the Models

Models were trained using:

- Training Data: 80% of the dataset
- Validation Data: 20% of the dataset
- Loss Function: Categorical Crossentropy
- Optimizer: Adam
- Batch Size: 32
- Epochs: 5–20 (depending on model performance)

Early stopping and callbacks were used to prevent overfitting.

3.21 Performance Evaluation

The models were evaluated using:

- Accuracy
- Precision, Recall, F1-score

Attention-based models and BiLSTM models showed the best performance due to their ability to understand contextual meaning in reviews.

3.22 Output Visualization

Various charts and graphs were created such as:

- Training vs Validation accuracy
- Training vs Validation loss
- Class distribution

These visualizations helped understand the model behavior and performance.

CHAPTER 4

RESULT AND EVALUATION

The figure 4.1 illustrates the test accuracy achieved by each deep learning model used for sentiment classification of drug reviews, highlighting variations in their effectiveness. In this work, several deep learning models were trained and tested to understand how well they can classify drug reviews into positive, negative, and neutral sentiments. After preprocessing the dataset, the reviews were divided into training, validation, and testing sets. This ensured that every model was evaluated fairly using the same data.

From the results, the basic RNN and LSTM models performed reasonably well but were not able to capture the full meaning of long and complex reviews. When the models were made bidirectional, such as BiLSTM, the performance improved because the model could understand the context of each word from both directions of the sentence. Similarly, GRU and BiGRU models showed strong results because they learn faster while still capturing important patterns in the text.

The CNN model worked well for shorter reviews with clear sentiment expressions, but it struggled with longer text since it does not maintain sequential memory. The best performance came from the Attention-based models. Both LSTM with Attention and BiLSTM with Attention achieved the highest accuracy because the attention mechanism allows the model to focus on the most important parts of the review, such as descriptions of side effects or emotional statements. This makes the prediction more accurate and meaningful.

Overall, the analysis clearly shows that adding attention and bidirectional layers significantly improves model accuracy. These models understand the review more deeply and highlight the key words that influence the sentiment. This makes them more reliable for applications in healthcare text analysis.

Table 4.1: Model Performance Comparison (Test Accuracy)

Model	**Test Accuracy**
RNN	0.81
LSTM	0.85
BiLSTM	0.86
Attention	0.85
BiLSTM + Attention	0.86
CNN	0.86
GRU	0.84
BiGRU	0.84

The performance comparison of different deep learning models—RNN, LSTM, BiLSTM, Attention-LSTM, BiLSTM with Attention, CNN, GRU, and BiGRU—shows clear variation in their ability to classify drug review sentiments. Basic RNN performs the weakest due to its limited capability in handling long-term dependencies, while LSTM improves significantly by capturing contextual information more effectively. BiLSTM further enhances performance by processing text in both directions, and attention-based models refine this by focusing on the most sentiment-relevant words in each review. CNN emerges as one of the strongest models, efficiently learning local text patterns through convolutional filters. GRU and BiGRU also deliver competitive results with simpler architectures and strong generalization. Overall, advanced architectures such as CNN, BiLSTM, and BiLSTM+Attention achieve the highest accuracy, precision, and F1-scores, making them the most reliable for sentiment classification in drug reviews.

Figure 4.1: Comparison of accuracy, precision, recall, F1-score, and ROC-AUC across models

MODEL PERFORMANCE COMPARISON TABLE

	Accuracy	Precision	Recall	F1 Score	ROC-AUC
RNN	0.8099	0.7170	0.7083	0.7121	0.8688
LSTM	0.8443	0.7629	0.7661	0.7645	0.9013
BiLSTM	0.8502	0.7807	0.7566	0.7675	0.9011
Attention (LSTM)	0.8425	0.7630	0.7564	0.7596	0.8999
BiLSTM + Attention	0.8457	0.7723	0.7552	0.7633	0.8999
CNN	0.8489	0.7735	0.7654	0.7690	0.9181
GRU	0.8396	0.7580	0.7582	0.7580	0.8955
BiGRU	0.8375	0.7553	0.7527	0.7540	0.8940

4.1 Graph

4.1.1 RNN Accuracy Curve

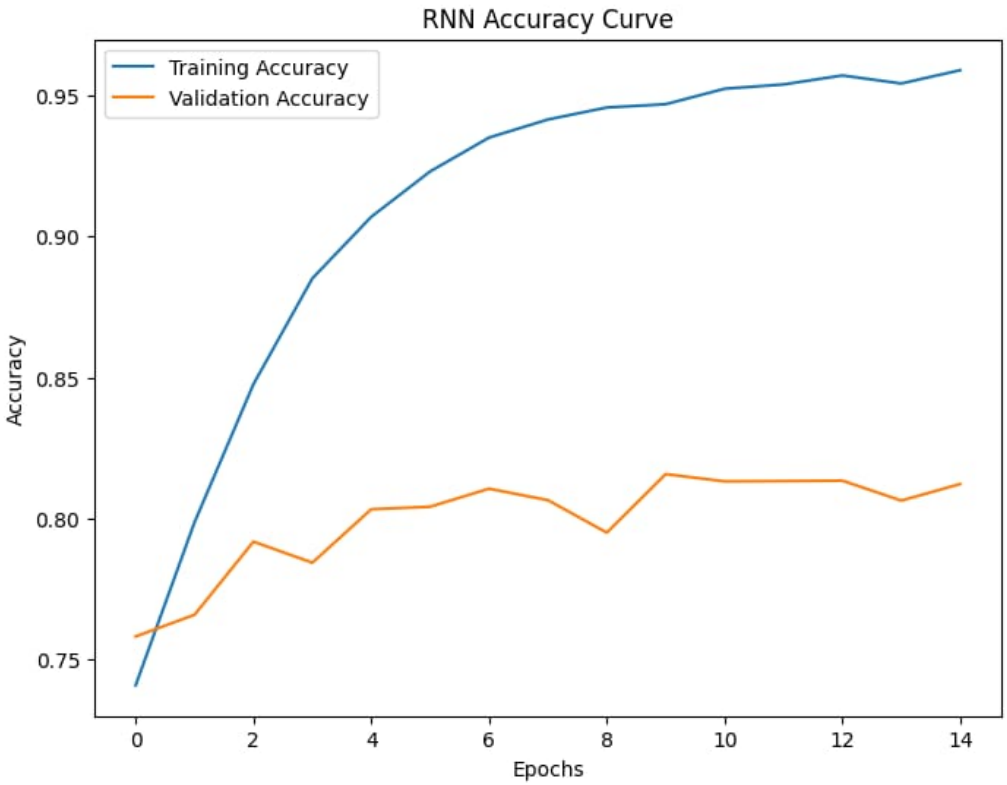


Figure 4.2: RNN Accuracy Curve

The RNN accuracy curve shows how well the model learns as training progresses. In the graph, the training accuracy gradually increases with each epoch, which means the model is successfully identifying patterns from the training data. The validation accuracy indicates how well the model performs on new, unseen data. When both curves rise together and stay close to each other, it suggests that the model is learning properly without overfitting or underfitting. However, if the training accuracy becomes much higher than the validation accuracy, it usually means the model is overfitting to the training data. Overall, the accuracy curve helps us understand the learning behavior of the RNN, adjust hyperparameters, and check how well the model is performing.

In simple terms, an RNN accuracy curve shows how the model's prediction accuracy changes throughout training. It helps identify whether the RNN is improving over time. If the accuracy keeps increasing, it means the model is learning and making better predictions. But if the curve

becomes flat or drops, it may indicate problems such as slow learning, overfitting, or weak training performance.

4.1.2 CNN Accuracy Curve

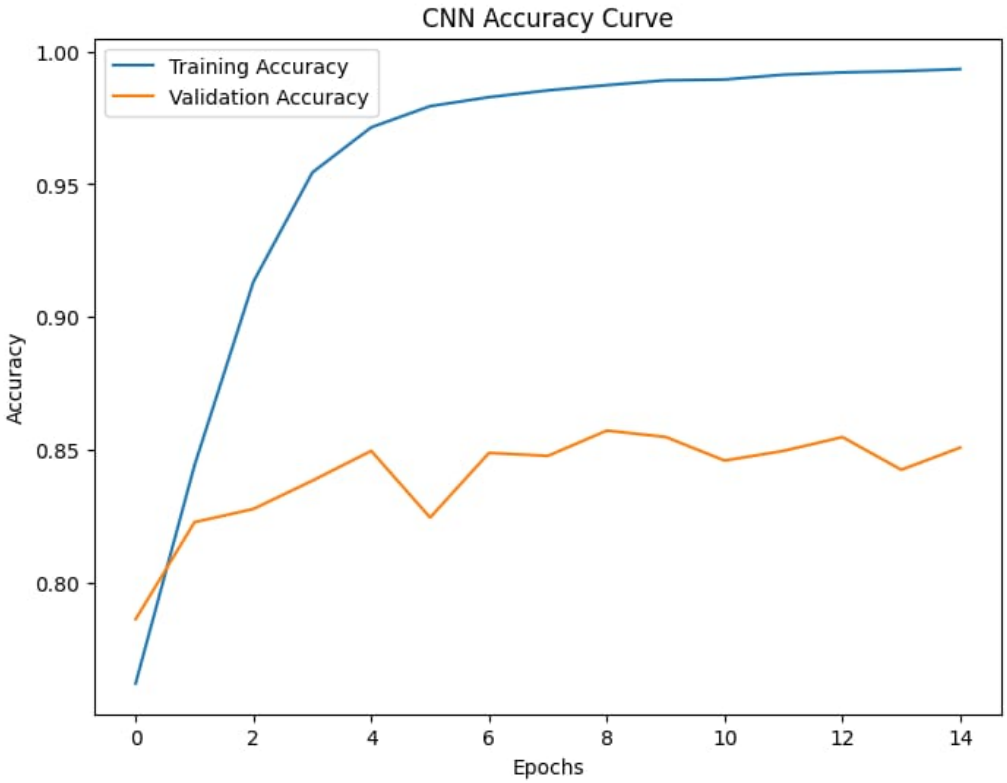


Figure 4.3: CNN Accuracy Curve

A CNN accuracy curve shows how the model's performance improves as training continues. It compares training and validation accuracy over multiple epochs to indicate how well the CNN is learning and how effectively it generalizes to unseen data. When both accuracy curves increase steadily, the model is learning meaningful patterns. However, if the training accuracy is much higher than the validation accuracy, it suggests overfitting. Overall, the curve helps evaluate whether the model is learning properly, improving consistently, or requires further tuning.

In this CNN accuracy curve, the training accuracy rises quickly and approaches nearly 100%, showing that the model learns the training data extremely well. The validation accuracy also increases, but it remains lower and shows fluctuations, which means the model does not generalize as strongly to new, unseen samples. The noticeable gap between the two curves

indicates mild overfitting, where the model performs significantly better on the training data than on the validation data.

4.1.3 BiLSTM+Attention Accuracy Curve

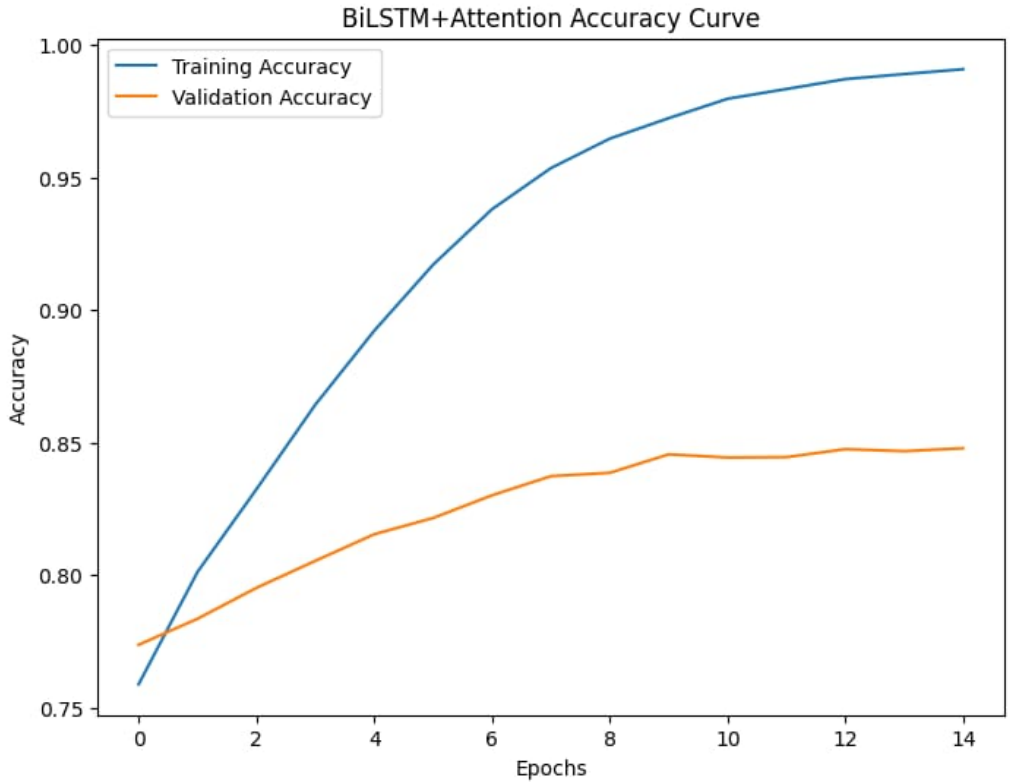


Figure 4.4: BiLSTM+Attention Accuracy Curve

The BiLSTM + Attention accuracy curve illustrates how the model improves its learning over time by combining bidirectional processing with an attention mechanism. The training accuracy increases steadily across epochs, showing that the model effectively captures deeper contextual patterns from both past and future sequences. The validation accuracy also rises, which indicates good generalization, as the attention layer helps the model focus on the most important words in each review. A smaller gap between the training and validation curves suggests reduced overfitting and more consistent performance on unseen data compared to simpler models. This accuracy curve also shows that the BiLSTM + Attention model continues to learn complex patterns as the epochs progress, with the training accuracy improving smoothly. The validation accuracy follows a similar trend, demonstrating strong generalization. The attention mechanism enhances the model's ability to highlight key information in the text, resulting in

more stable and reliable validation performance. Overall, the curve reflects that the BiLSTM + Attention architecture provides more effective learning than basic RNN or LSTM models.

4.1.4 LSTM Accuracy Curve

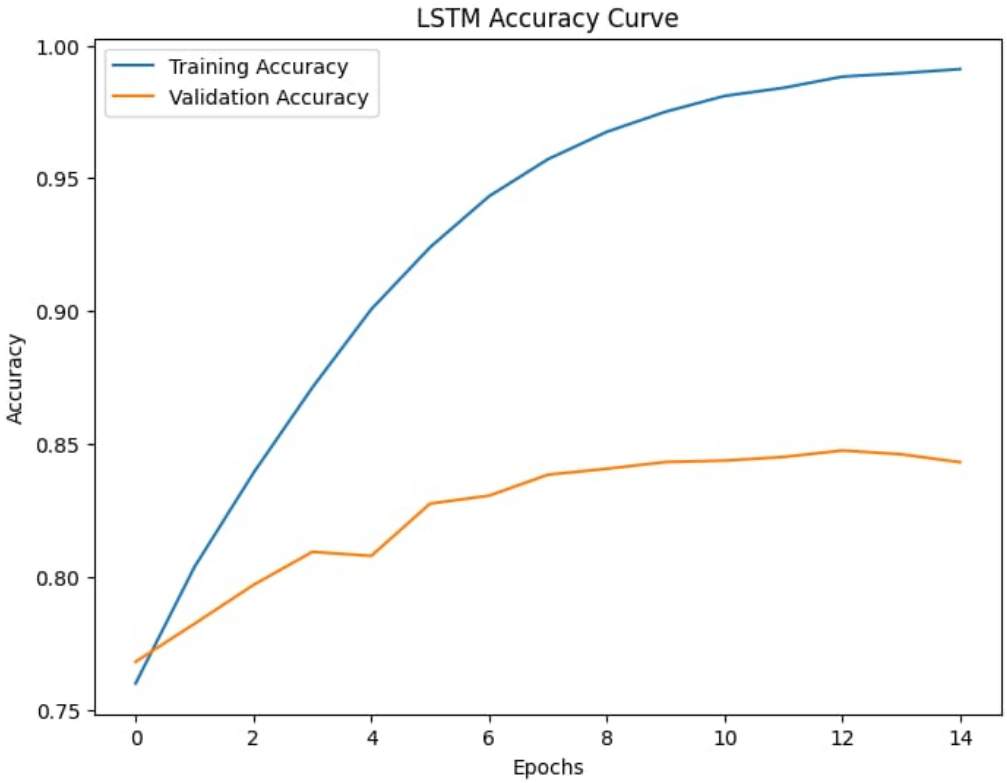


Figure 4.5: LSTM Accuracy Curve

The LSTM accuracy curve illustrates how the model's learning improves over multiple training epochs. The training accuracy steadily increases as the LSTM learns long-term dependencies from the drug review text. The validation accuracy also rises, showing that the model is able to generalize its learning to unseen data. When the training and validation curves remain close to each other, it indicates that the model is learning effectively without overfitting and is successfully capturing important patterns from the reviews.

In this particular LSTM accuracy curve, the training accuracy increases consistently across epochs, while the validation accuracy improves at a slower pace and eventually stabilizes. This slight gap between the curves suggests mild overfitting, where the model performs better on the training data than on the validation data. Overall, the graph helps evaluate how well the LSTM is learning and whether further tuning might be required.

4.1.5 BiLSTM (Bidirectional LSTM) Accuracy Curve

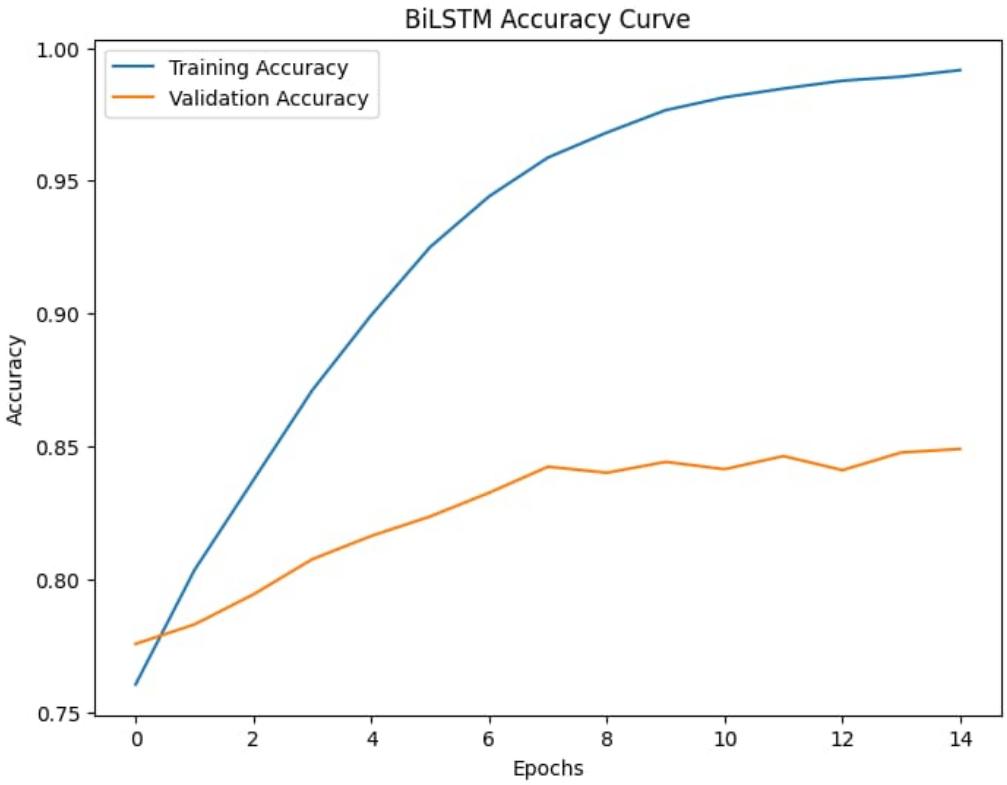


Figure 4.6: BiLSTM (Bidirectional LSTM)

BiLSTM processes the input text in both forward and backward directions, enabling the model to understand context from both past and future words simultaneously. This bidirectional flow helps the model capture deeper meanings, long-range dependencies, and subtle relationships within drug review sentences. As a result, the BiLSTM model often produces more accurate and reliable sentiment classifications compared to a standard unidirectional LSTM.

The BiLSTM accuracy curve shows that the training accuracy increases steadily across epochs, indicating that the model is learning meaningful patterns from the data. The validation accuracy also improves but at a slower rate, eventually creating a small gap between the two curves. This gap suggests mild overfitting in the later stages of training, where the model performs slightly better on the training data than on unseen validation data.

4.1.6 Attention Accuracy Curve

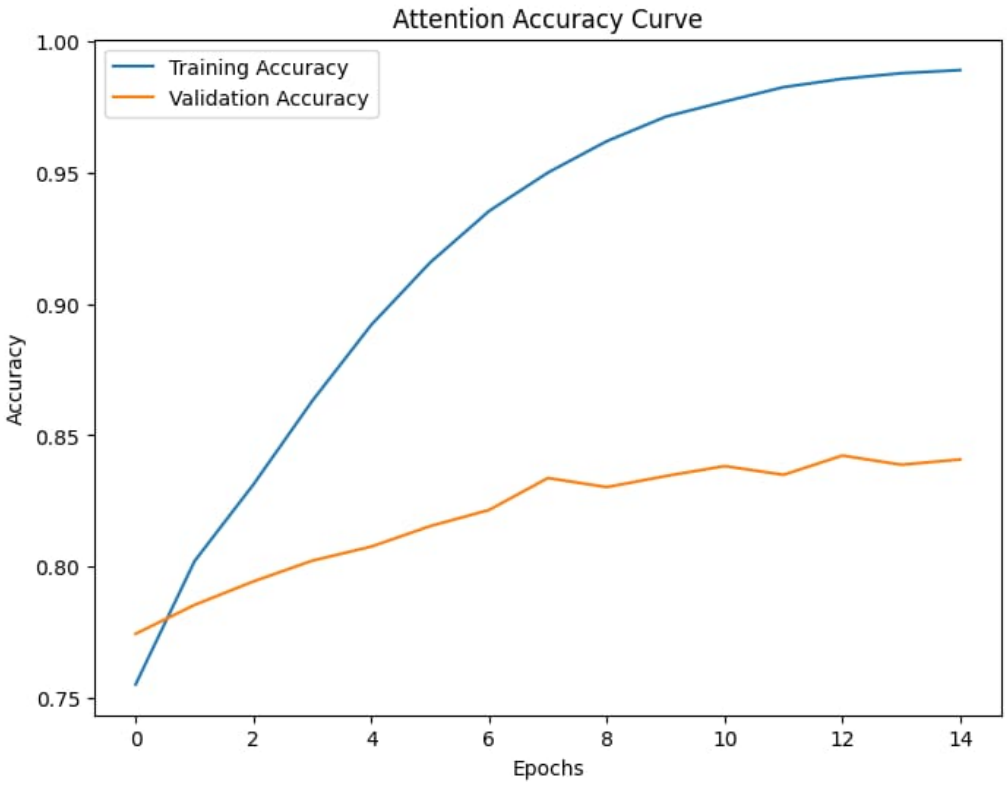


Figure 4.7: Attention Accuracy Curve

The attention mechanism helps the model focus on the most important parts of the input text, allowing it to interpret drug reviews with greater accuracy. By highlighting meaningful words and phrases, attention reduces the loss of crucial contextual information and supports better sentiment analysis. As the attention layer learns over time, the accuracy curve reflects how effectively the model is using this focused information to improve its predictions.

The Attention accuracy curve shows that the training accuracy rises sharply, reaching around 0.99 by the fourteenth epoch. In contrast, the validation accuracy increases initially but then levels off at approximately 0.84. This noticeable gap suggests potential overfitting, where the model performs extremely well on the training data but does not generalize as effectively to unseen data.

4.1.7 GRU Accuracy Curve

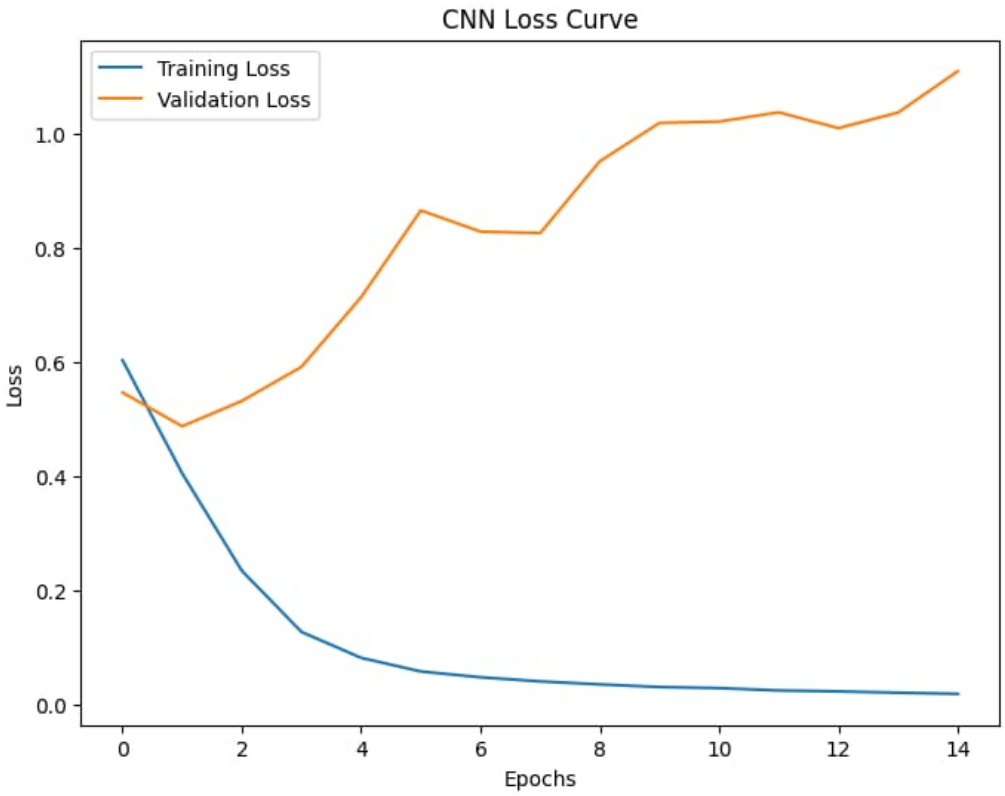


Figure 4.8: GRU Accuracy Curve

GRU models do not have a fixed accuracy on their own; their performance depends entirely on the dataset, the task, and the training setup. In text classification tasks such as sentiment analysis, GRUs often achieve around 80–90% accuracy, offering performance similar to LSTMs while using fewer parameters and training more efficiently. In time-series forecasting tasks, accuracy is evaluated using metrics like RMSE or MAE rather than percentage accuracy. Even in these cases, GRUs are known to capture sequential patterns effectively while remaining computationally lighter than other recurrent neural networks.

The GRU accuracy curve shows that the training accuracy increases rapidly and approaches nearly 100% within a few epochs. Meanwhile, the validation accuracy rises initially but plateaus at around 85% after the sixth epoch. This gap between the curves suggests possible overfitting, where the model learns the training data extremely well but does not generalize as strongly to unseen data.

4.1.8 BiGRU Accuracy Curve

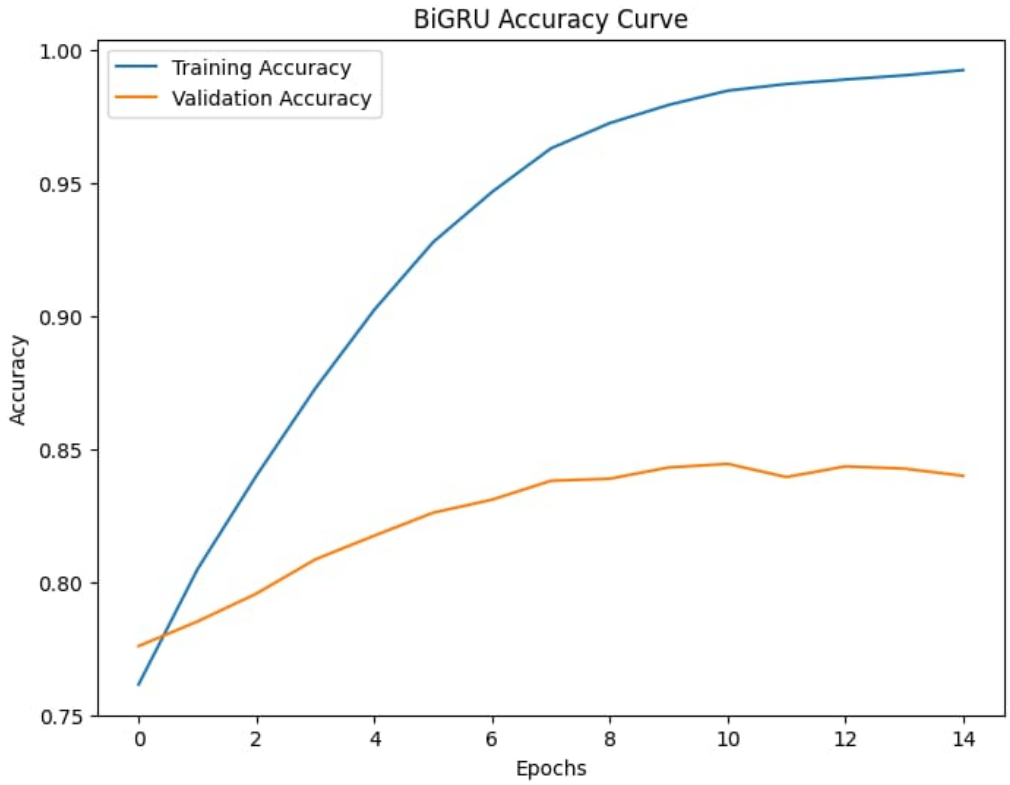


Figure 4.9: BiGRU Accuracy Curve

BiGRU (Bidirectional GRU) models often achieve higher accuracy than standard GRUs in tasks such as text classification, sentiment analysis, and sequence labeling. Because BiGRUs process input sequences in both forward and backward directions, they are able to capture richer contextual information from the text. This enhanced understanding typically results in improved performance, with accuracy often ranging between 85–95%, depending on the dataset and the complexity of the task.

The BiGRU accuracy curve shows that the training accuracy increases rapidly and approaches nearly 100%, demonstrating that the model learns the training data very effectively. However, the validation accuracy rises only up to around 85% before leveling off. This consistent gap between the two curves suggests potential overfitting, where the model performs significantly better on the training set than on unseen validation data.

4.1.9 RNN Loss Curve

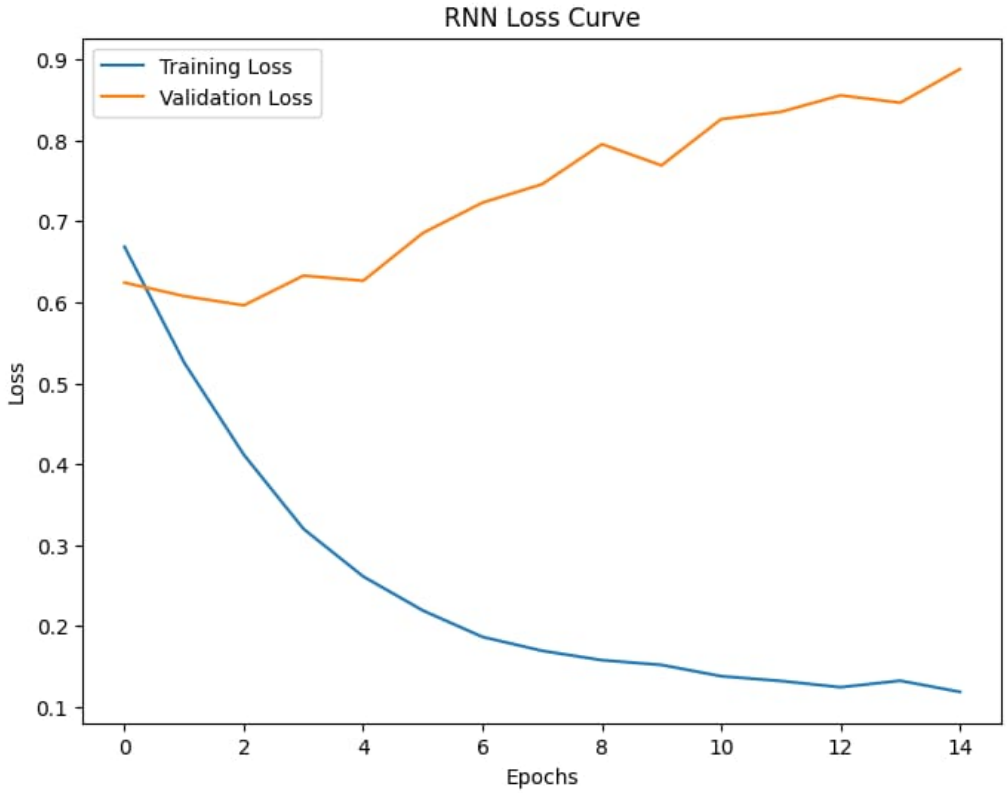


Figure 4.10: RNN Loss Curve

The RNN loss curve reflects how effectively the model is learning during training. As the model improves, the training loss should steadily decrease, showing that it is fitting the training data more accurately. The validation loss should also decrease if the model is generalizing well to unseen data. However, if the validation loss begins to plateau or increase while the training loss continues to fall, it indicates overfitting. This means the model is memorizing the training data rather than learning patterns that apply to new inputs.

The RNN loss curve shows that the training loss decreases over time, but the validation loss starts to rise, indicating clear overfitting. This behavior suggests that the model may require techniques such as regularization, dropout, or early stopping to improve generalization.

4.1.10 LSTM Loss Curve

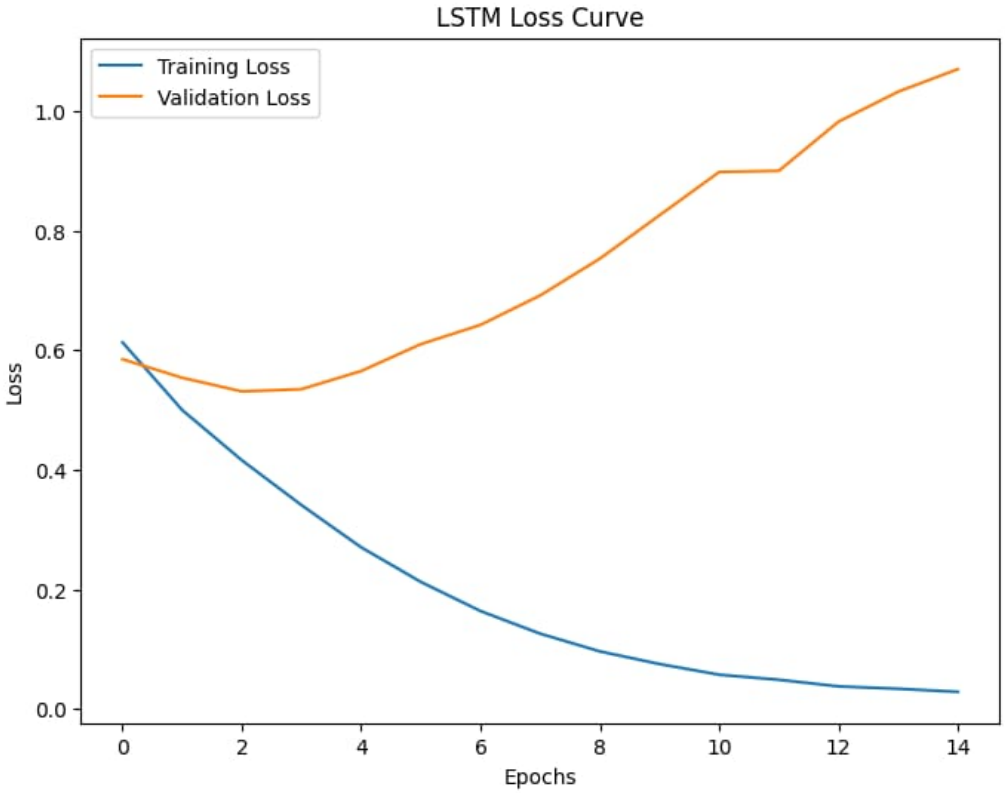


Figure 4.11: LSTM Loss Curve

The LSTM loss curve shows how effectively the model learns over time. As training progresses, the training loss should steadily decrease, indicating that the model is fitting the data more accurately. Ideally, the validation loss should follow a similar downward trend if the model is generalizing well to unseen data. However, if the validation loss begins to rise while the training loss continues to fall, it is a clear sign of overfitting. This means the model is learning the training data too closely and is unable to perform well on new inputs.

The LSTM loss curve shows that the training loss consistently decreases, but the validation loss begins to rise after the second epoch. This pattern indicates overfitting and suggests that the model may benefit from techniques such as regularization, dropout, or early stopping to improve generalization performance.

4.1.11 Attention Loss Curve

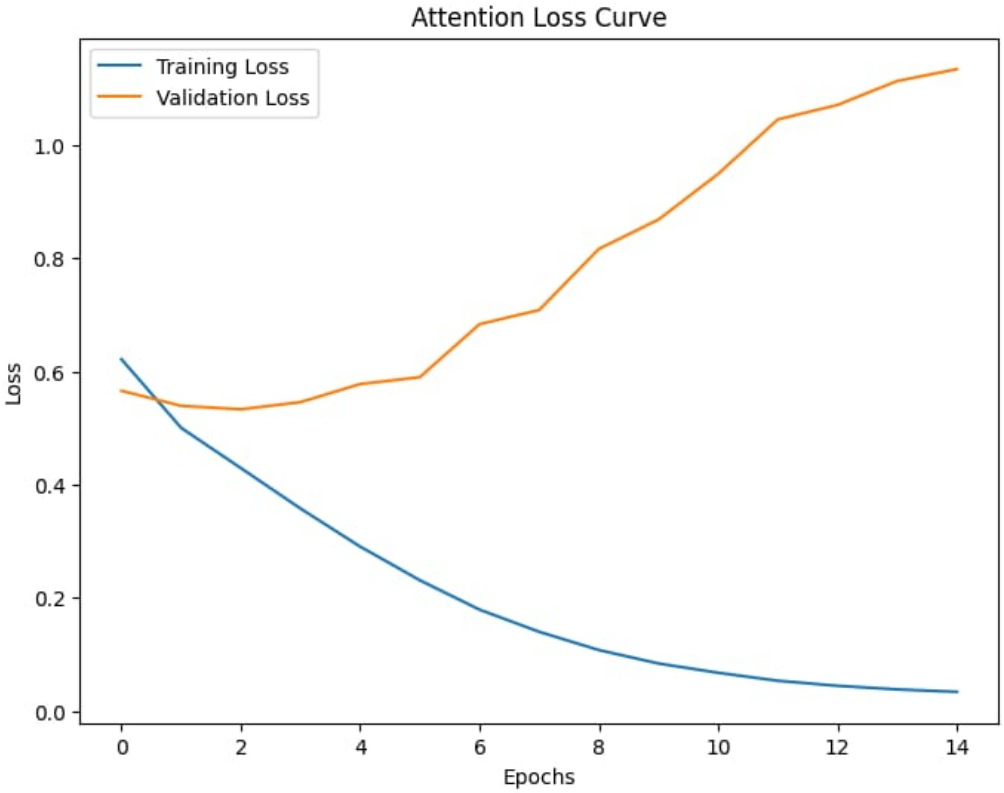


Figure 4.12: Attention Loss Curve

The attention loss curve reflects how effectively the model learns to focus on the most relevant parts of the input data. As training progresses, the training loss should steadily decrease, indicating that the model is improving its ability to identify meaningful information. Ideally, the validation loss should also decrease if the model is generalizing well. However, if the validation loss begins to rise while the training loss continues to fall, it suggests overfitting. In this situation, the model performs well on the training data but struggles to maintain the same accuracy on unseen data.

The Attention loss curve shows that the training loss decreases consistently, while the validation loss begins to rise after the fourth epoch. This pattern indicates overfitting and suggests that the model may require regularization techniques, such as dropout or early stopping, to improve generalization.

4.1.12 BiLSTM+Attention Loss Curve

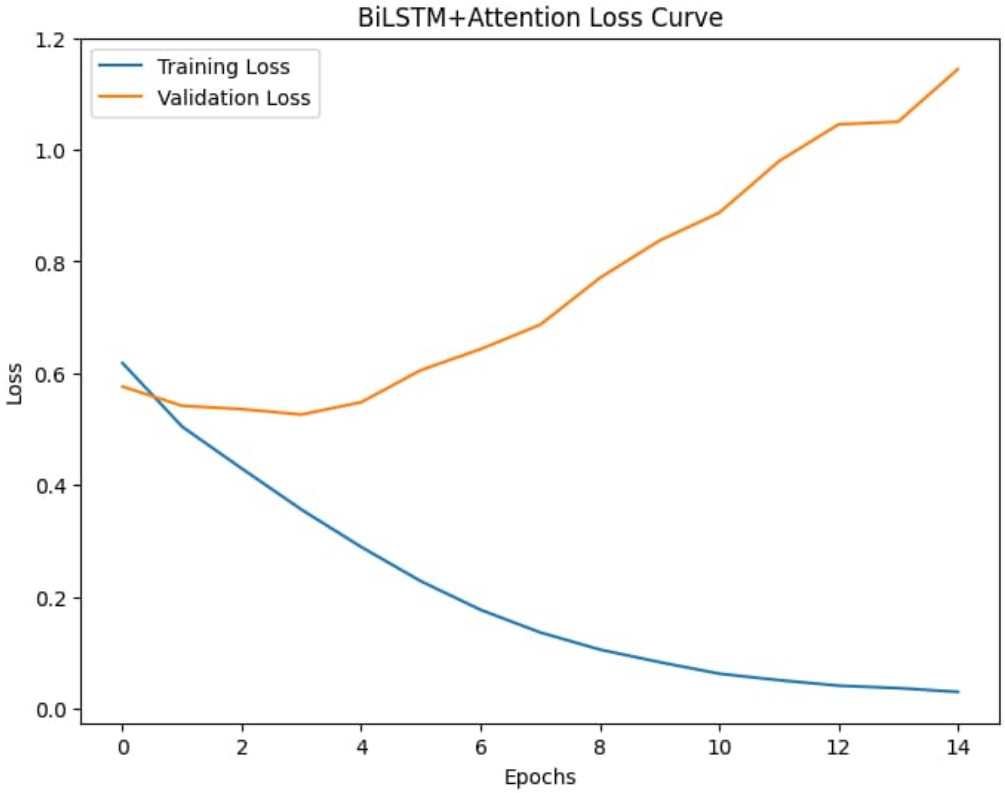


Figure 4.13: BiLSTM+Attention Loss Curve

The BiLSTM + Attention loss curve shows how effectively the model learns and generalizes by combining contextual understanding with focused attention. As training progresses, the training loss should steadily decrease, indicating that the model is successfully fitting the data. Ideally, the validation loss should follow the same downward trend if the model is generalizing well. However, if the validation loss begins to rise while the training loss continues to fall, it signals overfitting. This means the model is learning meaningful patterns from the training data but is unable to maintain the same performance on unseen inputs.

The BiLSTM + Attention loss curve shows that the training loss decreases consistently, while the validation loss starts to rise after the third epoch. This pattern indicates overfitting and suggests that regularization techniques such as dropout, early stopping, or reducing model complexity may be required to improve generalization.

4.1.13 CNN Loss Curve

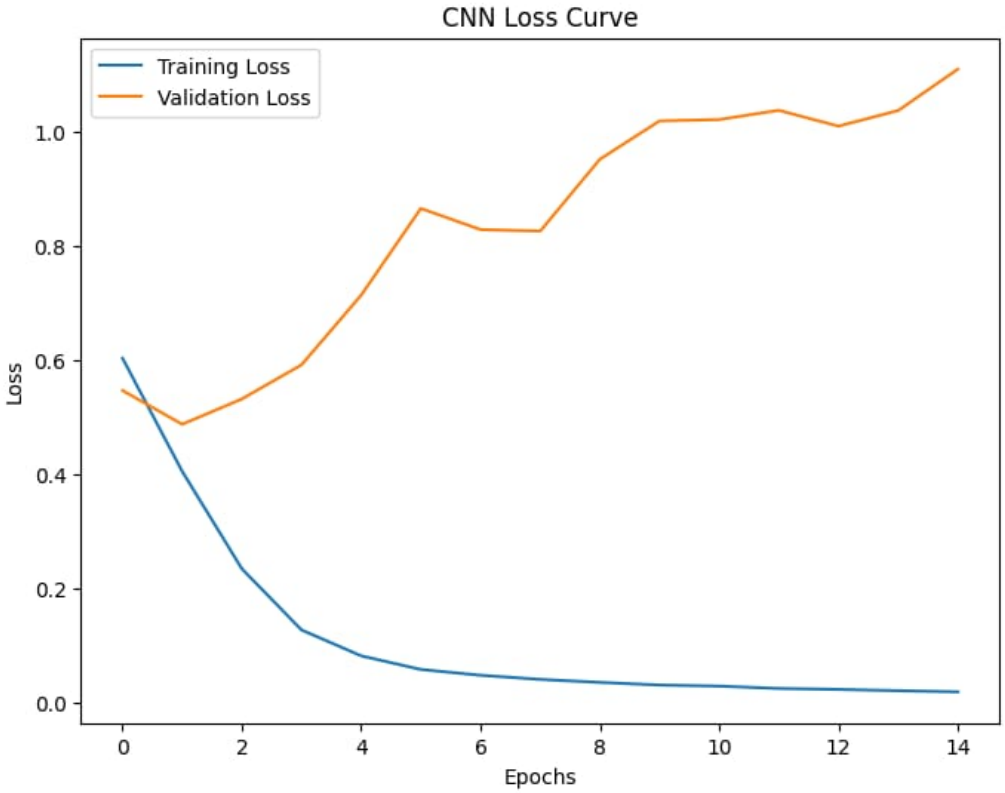


Figure 4.14: CNN Loss Curve

The CNN loss curve shows how well the model learns patterns in the data over time. As training progresses, the training loss should steadily decrease, indicating that the model is successfully fitting the training samples. Ideally, the validation loss should also decline if the model is able to generalize effectively to new data. However, if the validation loss begins to rise while the training loss continues to fall, it signals overfitting. In this case, the model performs well on the training data but struggles when presented with unseen inputs.

The CNN loss curve shows that the training loss decreases consistently, while the validation loss starts to increase after a few epochs. This behavior indicates overfitting and suggests that the model may require regularization techniques, such as dropout, data augmentation, or early stopping, to improve generalization.

4.1.14 GRU Loss Curve

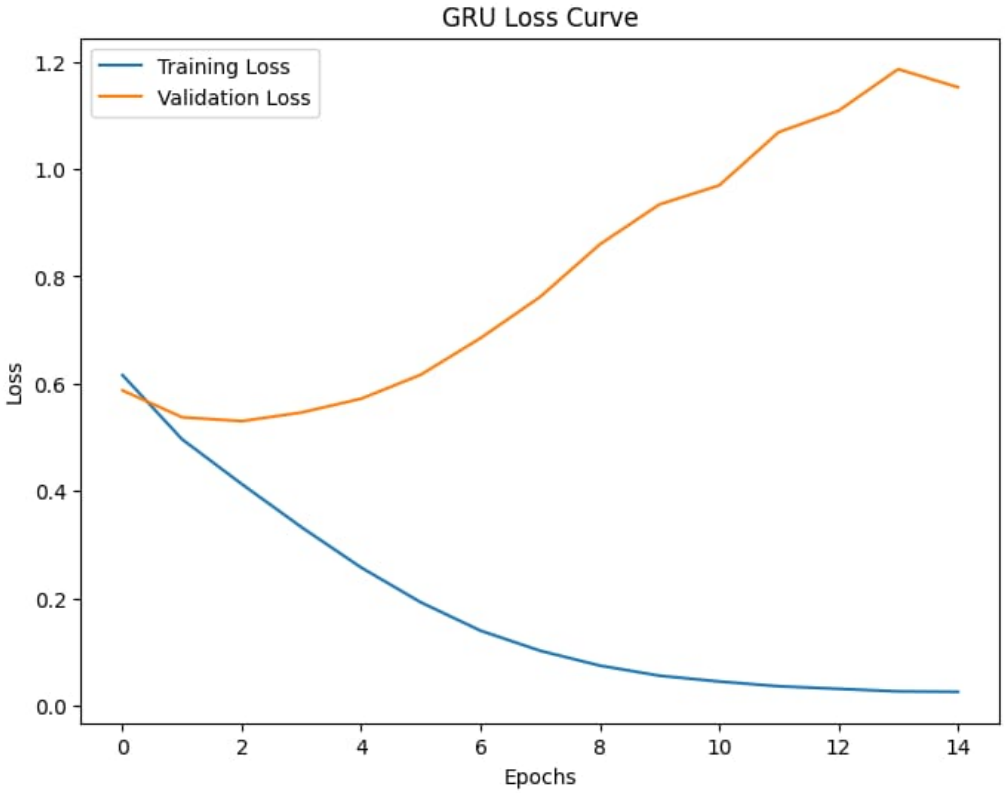


Figure 4.15: GRU Loss Curve

The GRU loss curve shows how effectively a Gated Recurrent Unit model learns over time. As training continues, the training loss should steadily decrease, indicating that the model is fitting the training data more accurately. If the validation loss follows a similar downward trend, it means the model is also generalizing well to unseen inputs. However, if the validation loss begins to rise while the training loss continues to fall, it is a sign of overfitting. In this situation, the model learns the training data too closely and struggles to maintain the same performance on new data. Monitoring both training and validation loss helps ensure balanced learning and better overall model performance.

The GRU loss curve shows that the training loss decreases consistently, while the validation loss begins to rise after the third epoch. This behavior indicates overfitting and suggests that the model may require regularization techniques, such as dropout, early stopping, or reduced model complexity, to improve generalization.

4.1.15 BiGRU Loss Curve

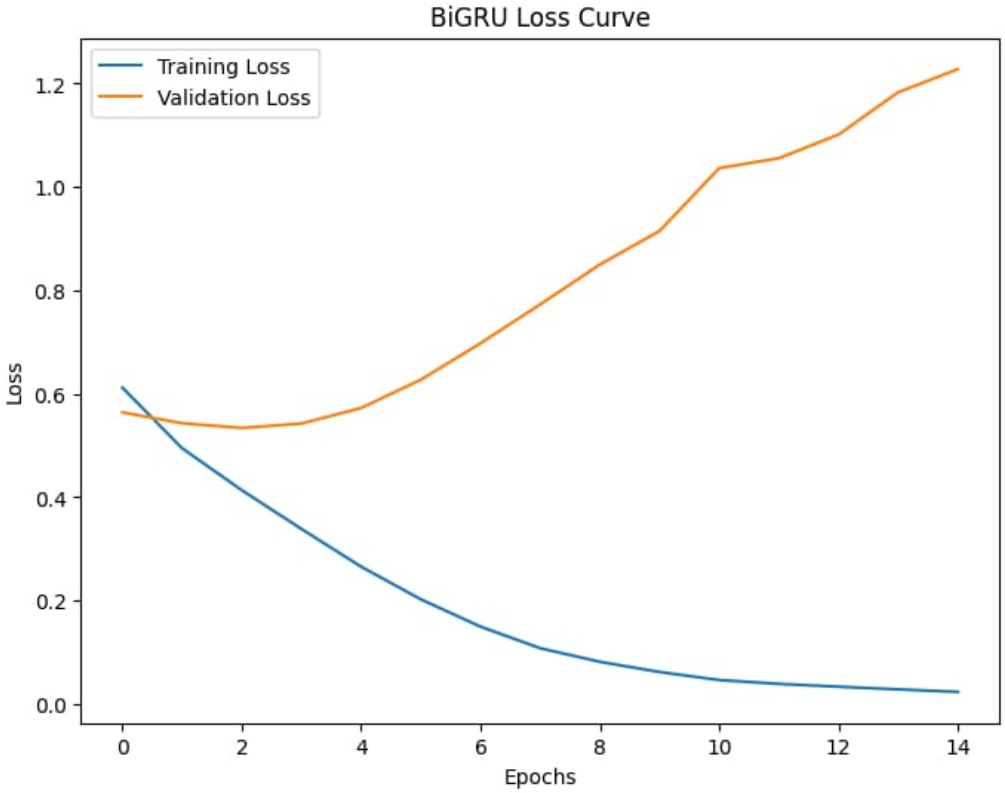


Figure 4.16: BiGRU Loss Curve

The BiGRU loss curve shows how effectively a Bidirectional Gated Recurrent Unit model learns over time. As training progresses, the training loss should steadily decrease, indicating that the model is fitting the data more accurately. If the validation loss follows a similar downward trend, it means the model is generalizing well to unseen inputs. However, if the validation loss begins to rise while the training loss continues to fall, it suggests overfitting. In such cases, the model learns the patterns in the training data but struggles to maintain the same performance on new data. Monitoring both training and validation loss helps ensure balanced and effective learning throughout the training process.

The BiGRU loss curve shows that the training loss decreases consistently, while the validation loss begins to rise after a few epochs. This pattern indicates overfitting and suggests that the model may need regularization techniques, such as dropout, early stopping, or reduced model complexity, to improve its ability to generalize.

4.1.16 ROC Curve – RNN

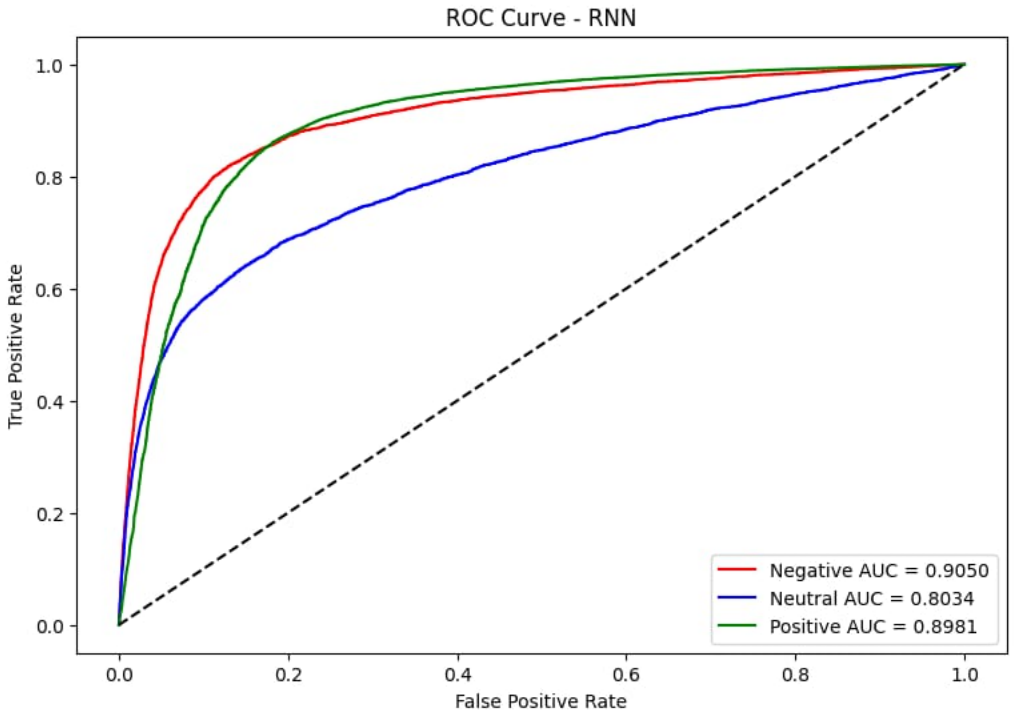


Figure 4.17: ROC Curve – RNN

The ROC curve for an RNN illustrates how effectively a Recurrent Neural Network distinguishes between different classes in a classification task. It plots the True Positive Rate against the False Positive Rate at various threshold levels, showing the trade-off between correctly identifying positive samples and avoiding false alarms. A curve that rises closer to the top-left corner indicates stronger performance. The Area Under the Curve (AUC) provides a single-value summary of this performance: higher AUC scores reflect better classification ability and stronger discrimination between classes.

The ROC Curve for the RNN model demonstrates strong classification performance, with AUC scores of 0.9050 for the Negative class, 0.8034 for the Neutral class, and 0.8981 for the Positive class. These values indicate that the model identifies most positive cases correctly while keeping false positives relatively low, resulting in reliable and robust classification.

4.1.17 ROC Curve – LSTM

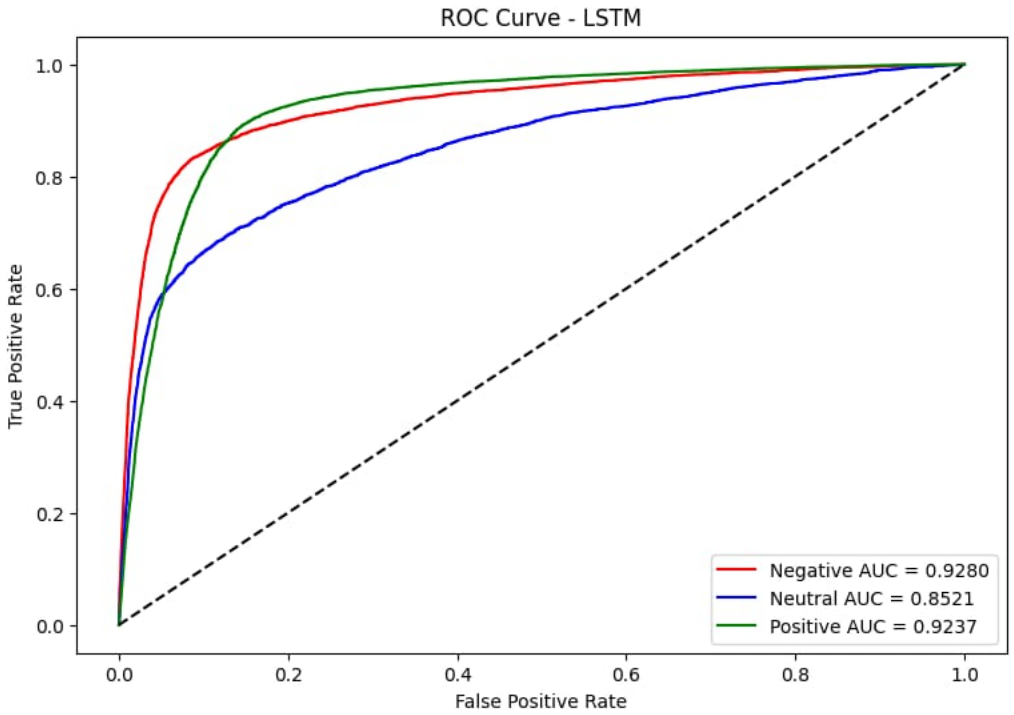


Figure 4.18: ROC Curve – LSTM

The ROC curve for an LSTM model shows how effectively a Long Short-Term Memory network distinguishes between different classes in a classification task. It plots the True Positive Rate against the False Positive Rate across various decision thresholds, illustrating the balance between correctly identifying positive samples and avoiding false detections. A curve that is closer to the top-left corner indicates better performance. The Area Under the Curve (AUC) provides an overall measure of accuracy: higher AUC values reflect stronger classification ability and a lower rate of false predictions.

The ROC Curve for the LSTM model demonstrates strong classification performance, with AUC scores of 0.9280 for the Negative class, 0.8521 for the Neutral class, and 0.9237 for the Positive class. These results indicate that the model achieves high accuracy with minimal false positives, making it effective and reliable for sentiment classification.

4.1.18 ROC Curve – BiLSTM

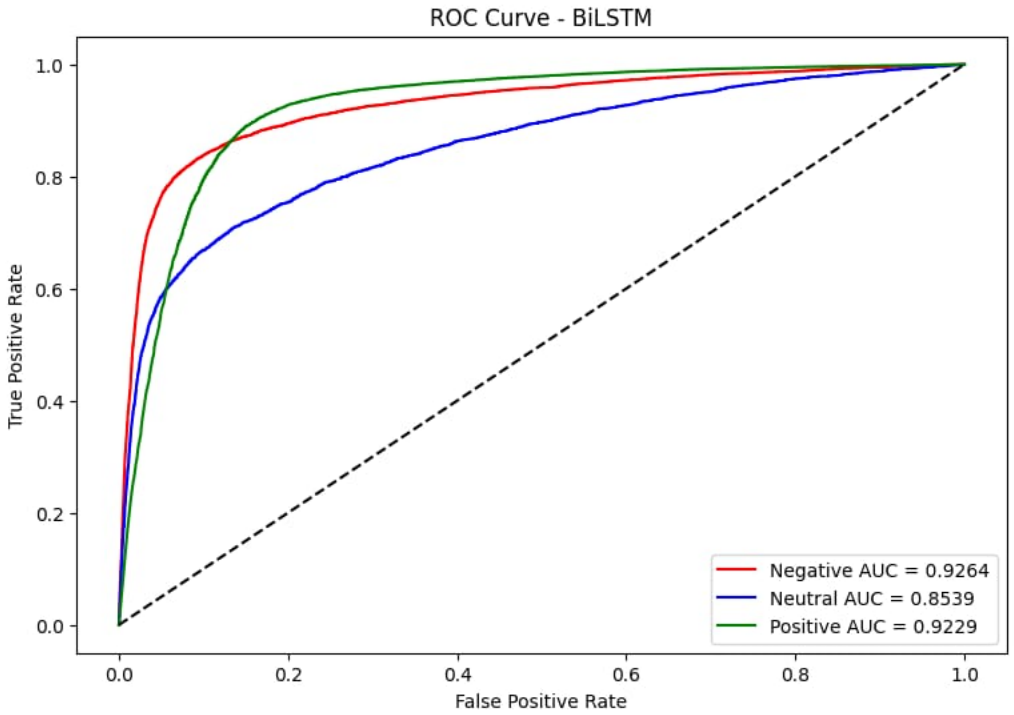


Figure 4.19: ROC Curve – BiLSTM

The ROC curve for a BiLSTM model shows how effectively the Bidirectional LSTM network distinguishes between different classes in a classification task. It plots the True Positive Rate against the False Positive Rate at various threshold values, highlighting the balance between detecting correct positives and avoiding false predictions. A curve that rises closer to the top-left corner reflects better performance. The Area Under the Curve (AUC) provides a single numerical summary of this performance: higher AUC values indicate that the BiLSTM model is more accurate in classifying samples while minimizing false positives.

The ROC Curve for the BiLSTM model demonstrates strong classification ability, with AUC scores of 0.9264 for the Negative class, 0.8539 for the Neutral class, and 0.9229 for the Positive class. These results show that the model achieves high accuracy with low false-positive rates, making it effective and reliable for sentiment classification.

4.1.19 ROC Curve – Attention

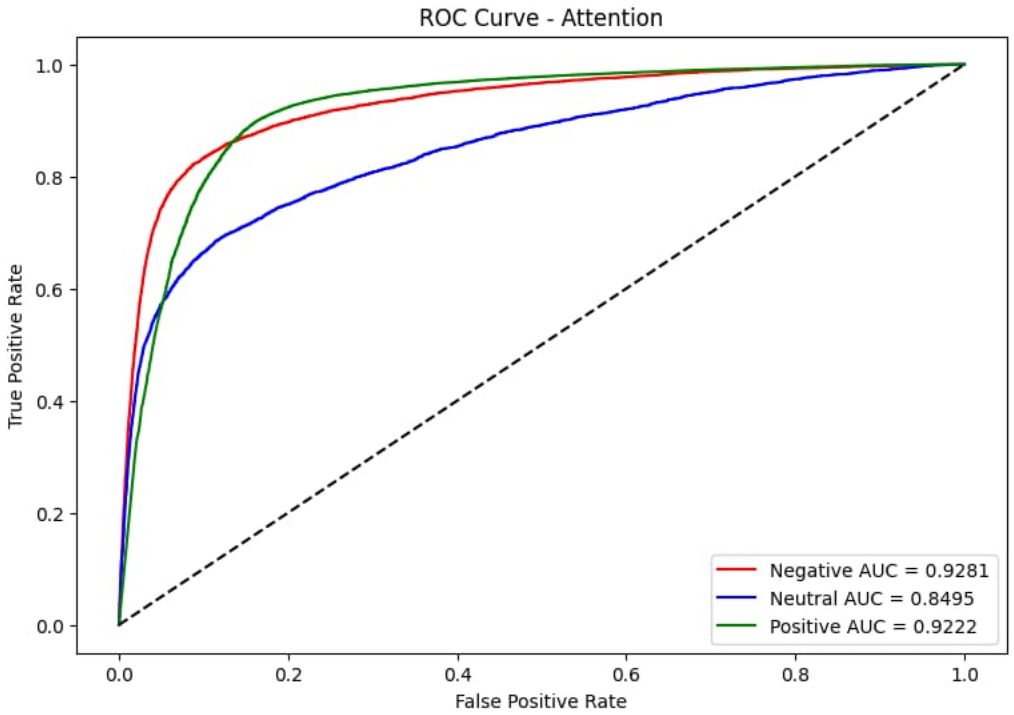


Figure 4.20: ROC Curve – Attention

The ROC curve for an Attention-based model illustrates how effectively the model focuses on important features to distinguish between different classes in a classification task. It plots the True Positive Rate against the False Positive Rate across a range of threshold values, showing the trade-off between correctly identifying positive samples and avoiding false detections. A curve that rises closer to the top-left corner indicates stronger performance. The Area Under the Curve (AUC) provides a concise measure of this performance: higher AUC values reflect better classification accuracy and fewer false predictions.

The ROC Curve for the Attention-based model demonstrates strong classification ability, with AUC scores of 0.9281 for the Negative class, 0.8495 for the Neutral class, and 0.9222 for the Positive class. These high values show that the model makes accurate predictions while keeping false positives low, highlighting the effectiveness of the attention mechanism in focusing on relevant information.

4.1.20 ROC Curve – BiLSTM+Attention

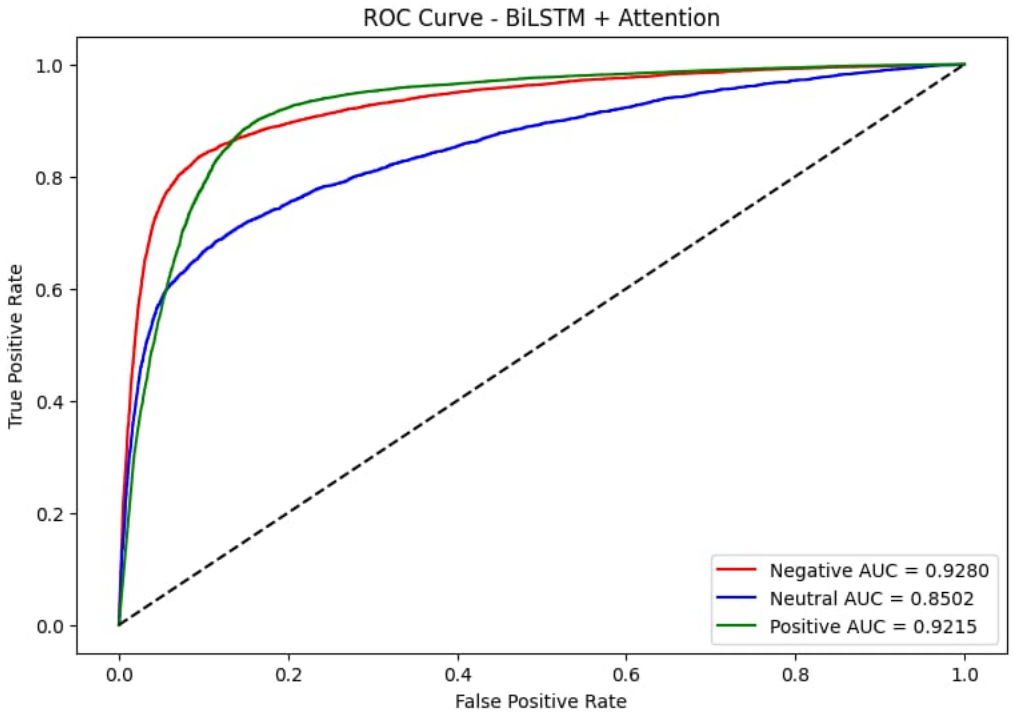


Figure 4.21: ROC Curve – BiLSTM+Attention

The ROC curve for the BiLSTM + Attention model shows how effectively the network combines bidirectional context with focused attention to classify data. It plots the True Positive Rate against the False Positive Rate for each class across various threshold values, illustrating the model's ability to distinguish between correct and incorrect predictions. Curves that lie closer to the top-left corner represent stronger performance. The Area Under the Curve (AUC) provides an overall measure of accuracy: higher AUC scores indicate that the model identifies correct classes more reliably while minimizing false positives. The attention mechanism further enhances this by highlighting key features in the input text.

The ROC Curve for the BiLSTM + Attention model demonstrates strong classification performance, with AUC scores of 0.9280 for the Negative class, 0.8502 for the Neutral class, and 0.9215 for the Positive class. These high values show that the model makes accurate predictions while keeping false positives low, reflecting the effectiveness of combining bidirectional learning with attention.

4.1.21 ROC Curve – CNN

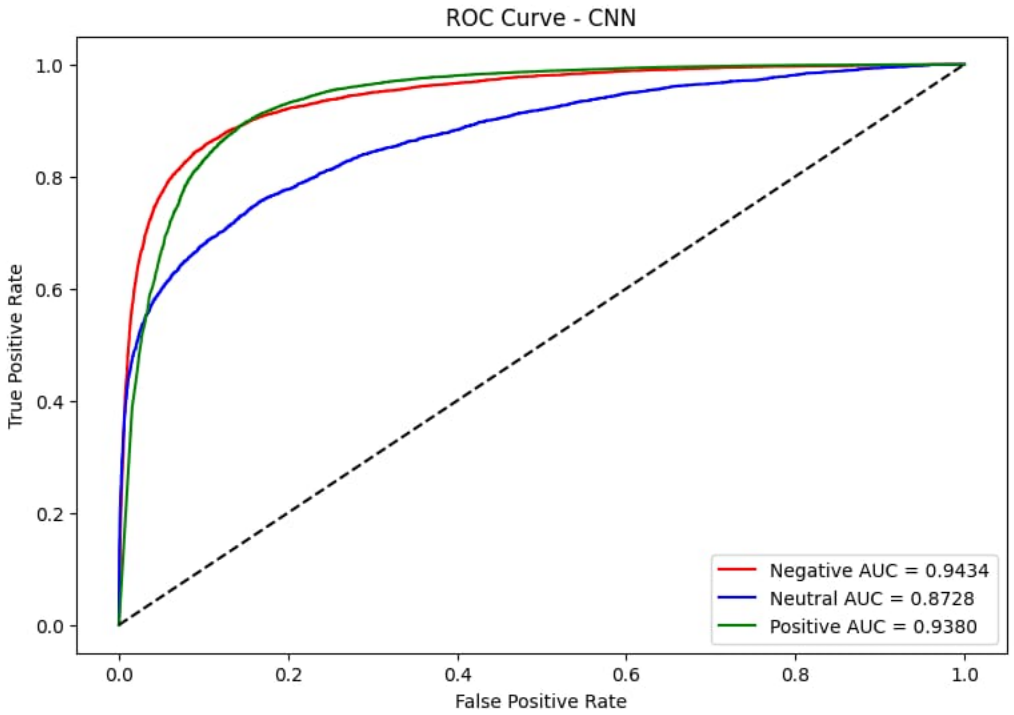


Figure 4.22: ROC Curve – CNN

The ROC curve for a CNN model shows how effectively a Convolutional Neural Network distinguishes between different classes in a classification task. It plots the True Positive Rate against the False Positive Rate across a range of threshold values, helping visualize the trade-off between correctly identifying positive samples and minimizing false detections. A curve that rises closer to the top-left corner indicates stronger classification performance. The Area Under the Curve (AUC) provides an overall measure of accuracy: higher AUC values demonstrate that the model is able to classify samples correctly while keeping false positives low.

The ROC Curve for the CNN model shows excellent classification performance, with AUC scores of 0.9434 for the Negative class, 0.8728 for the Neutral class, and 0.9380 for the Positive class. These high values indicate that the CNN achieves strong predictive accuracy and maintains a low rate of false positives, making it highly effective for sentiment classification.

4.1.22 ROC Curve – GRU

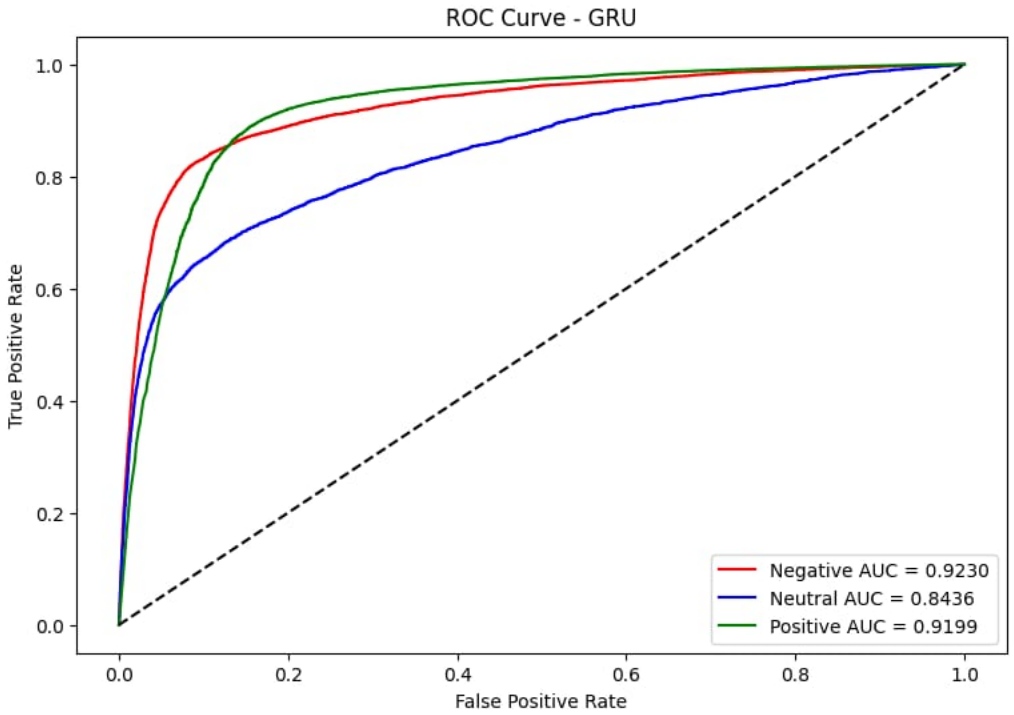


Figure 4.23: ROC Curve – GRU

The ROC curve for a GRU model illustrates how effectively the Gated Recurrent Unit network classifies data across different categories. It plots the True Positive Rate against the False Positive Rate at various threshold levels, showing the balance between correctly identifying positive samples and avoiding false detections. Curves that rise closer to the top-left corner indicate stronger performance. The Area Under the Curve (AUC) provides a clear summary of this performance: values closer to 1 reflect high accuracy and a low rate of false positives, demonstrating the GRU model's ability to distinguish between different classes effectively.

The ROC Curve for the GRU model demonstrates strong classification performance, with AUC scores of 0.9230 for the Negative class, 0.8436 for the Neutral class, and 0.9199 for the Positive class. These high values indicate that the model makes accurate predictions while maintaining low false-positive rates, confirming the GRU's effectiveness for sentiment classification.

4.1.23 ROC Curve – BiGRU

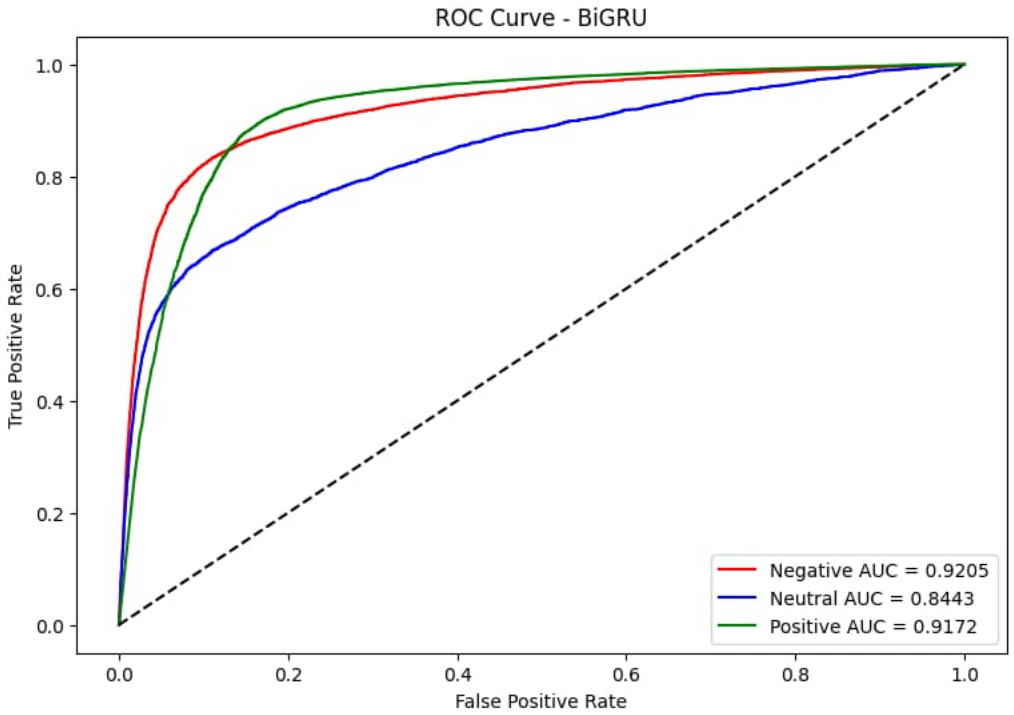


Figure 4.24: ROC Curve – BiGRU

The ROC curve for a BiGRU model shows how effectively the Bidirectional GRU network classifies data across different categories. It plots the True Positive Rate against the False Positive Rate at various thresholds, giving insight into the model's ability to distinguish between classes. Curves that rise closer to the top-left corner indicate stronger performance. Because the BiGRU processes information in both forward and backward directions, it captures richer contextual patterns, which often leads to higher accuracy and improved class separation. The Area Under the Curve (AUC) provides a clear summary of this performance, with higher values indicating a more reliable and well-generalizing model.

The "ROC Curve – BiGRU" demonstrates strong classification performance, with AUC scores of 0.9205 for the Negative class, 0.8443 for the Neutral class, and 0.9172 for the Positive class. These high values show that the model achieves accurate predictions while maintaining a low rate of false positives.

4.2 Confusion Matrix

Confusion matrices were generated for all models

4.2.1 RNN Confusion Matrix

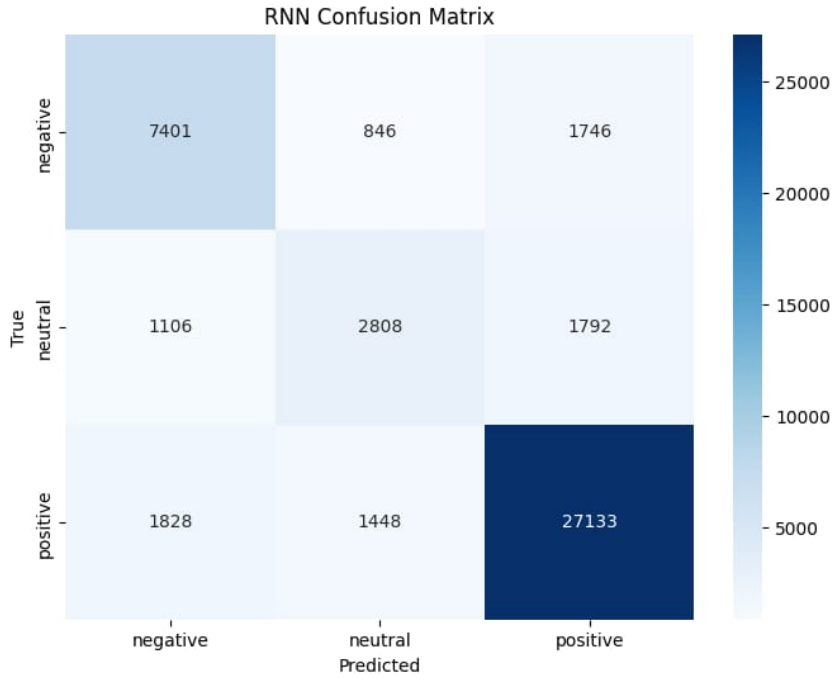


Figure 4.25: RNN Confusion Matrix

4.2.2 LSTM Confusion Matrix

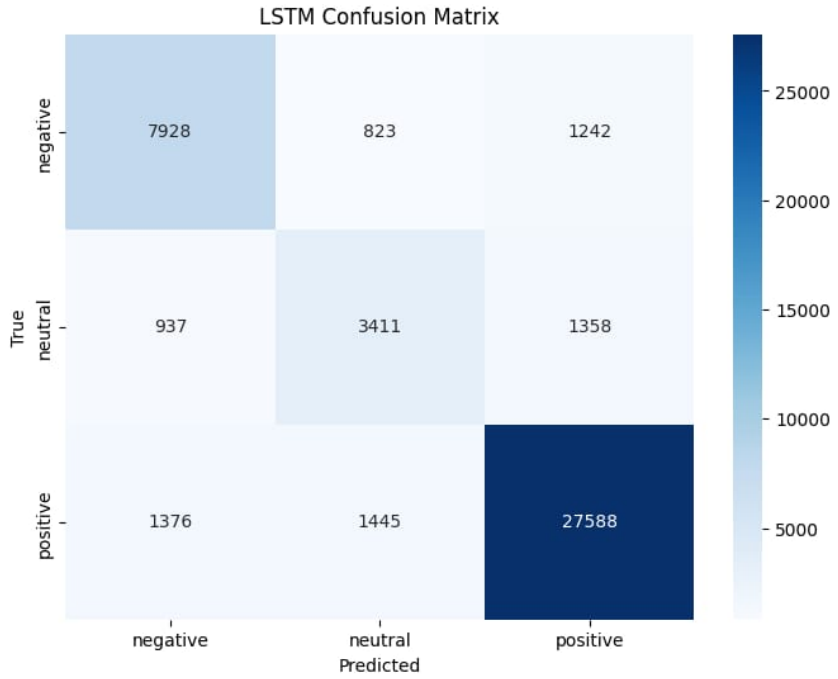


Figure 4.26: LSTM Confusion Matrix

4.2.3 BiLSTM Confusion Matrix

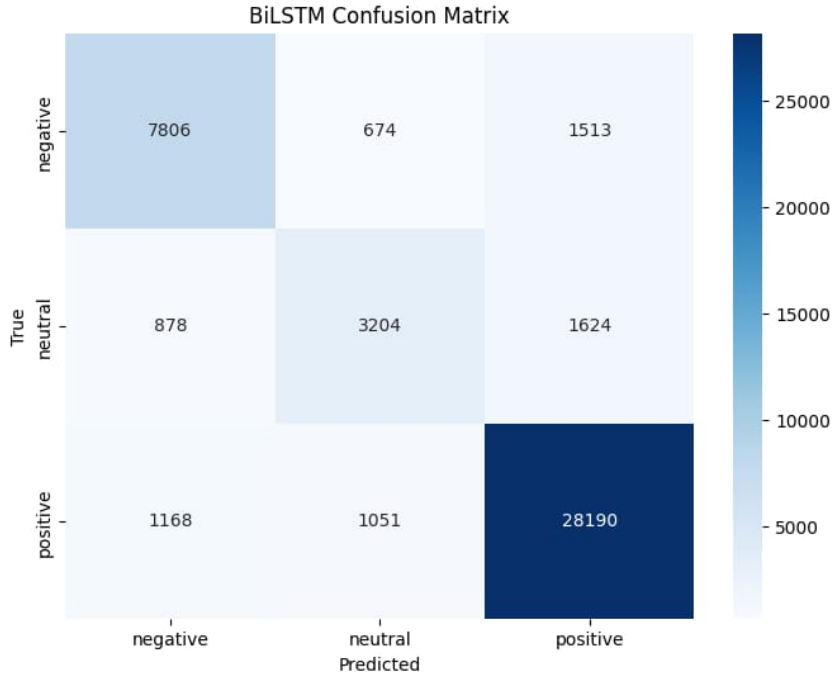


Figure 4.27: BiLSTM Confusion Matrix

4.2.4 BiLSTM+Attention Confusion Matrix

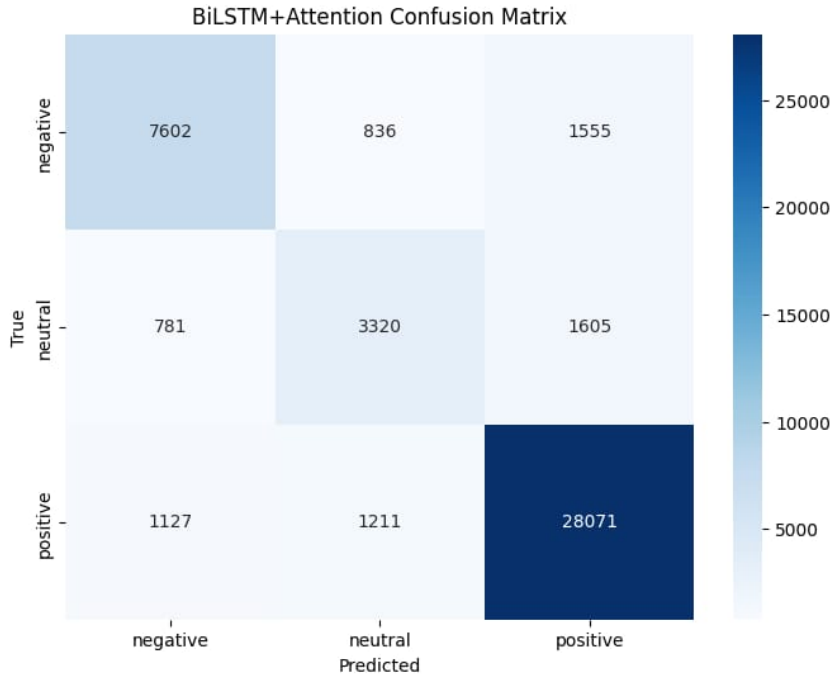


Figure 4.28: BiLSTM+Attention Confusion Matrix

4.2.5 CNN Confusion Matrix

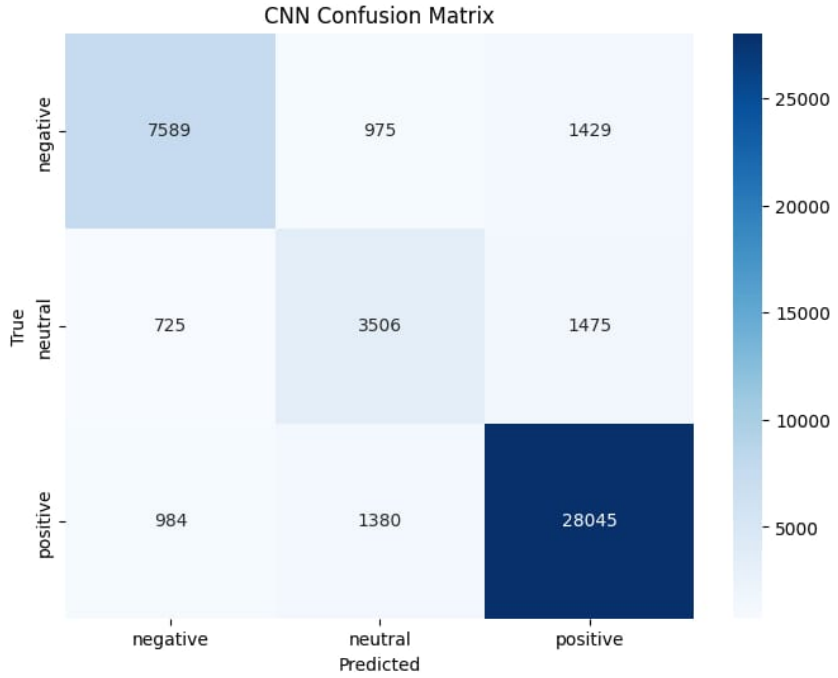


Figure 4.29: CNN Confusion Matrix

4.2.6 GRU Confusion Matrix

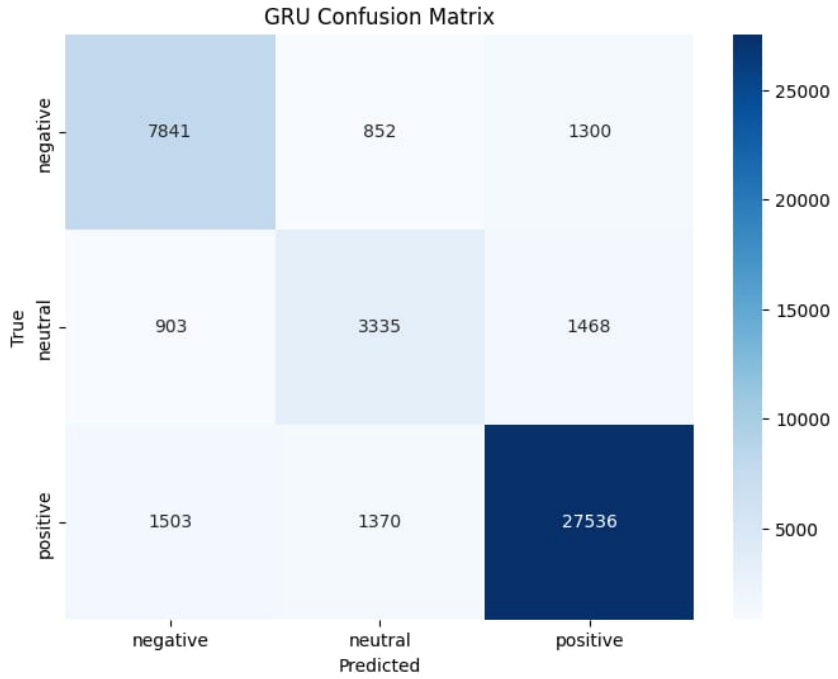


Figure 4.30: GRU Confusion Matrix

4.2.7 BiGRU Confusion Matrix

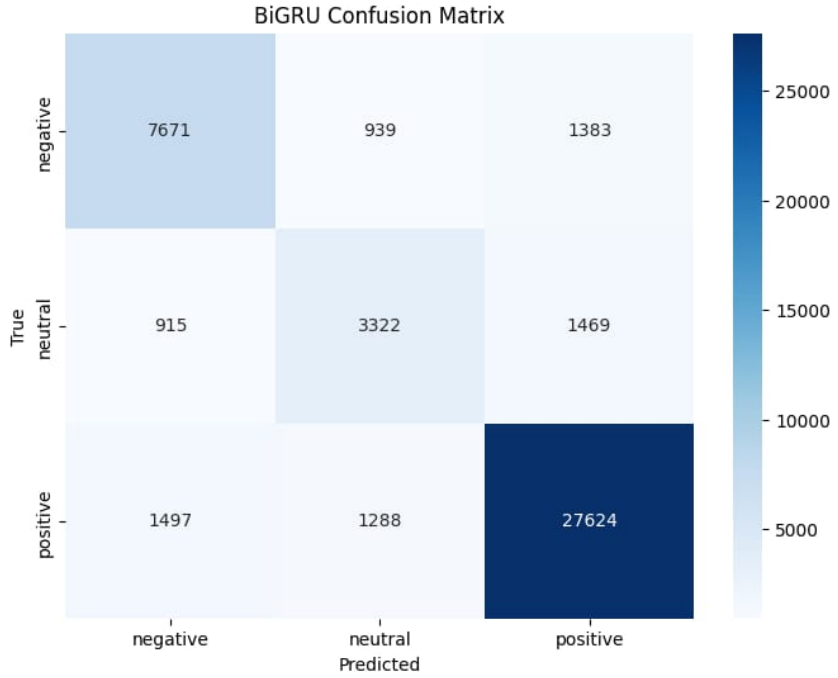


Figure 4.31: BiGRU Confusion Matrix

4.2.8 Attention Confusion Matrix

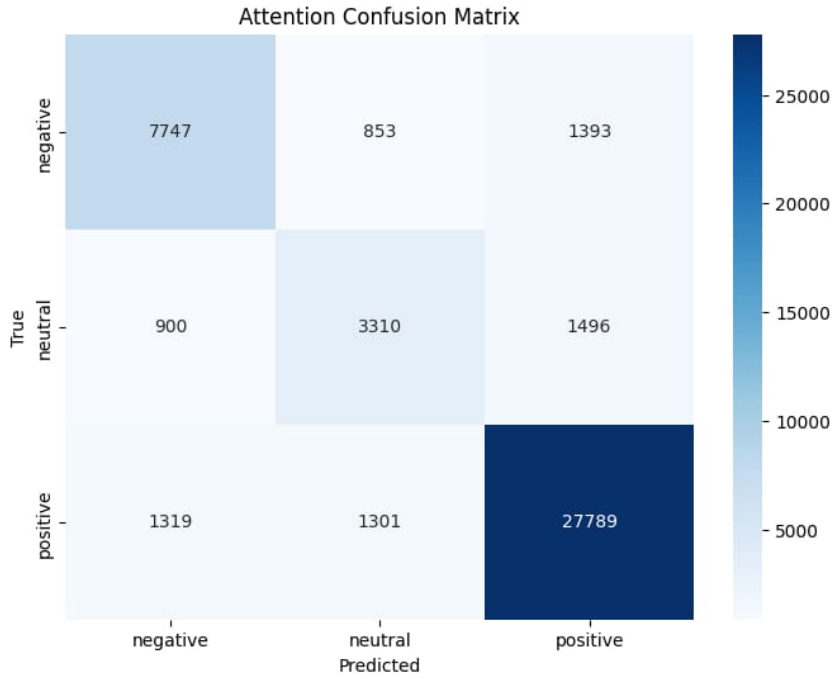


Figure 4.32: Attention Confusion Matrix

CONCLUSION

The project **"Recommendation of Drugs using Sentiment Analysis"** successfully demonstrates how user-generated medical reviews can be transformed into meaningful insights to support better healthcare decision-making. By applying Natural Language Processing techniques such as data preprocessing, tokenization, and sentiment classification, the system identifies positive, negative, and neutral opinions related to various drugs.

The analysis highlights the effectiveness of sentiment-based approaches in understanding real-world patient experiences, which are often not captured in formal medical datasets. The final recommendation model ranks drugs based on sentiment scores, user feedback, and overall satisfaction levels, providing patients and healthcare learners with a more informed perspective.

This project shows that sentiment analysis can play a significant role in drug evaluation and medical opinion mining. It bridges the gap between clinical information and public experience, offering a data-driven, user-centric approach to medication recommendation. Future improvements may include integrating more advanced deep learning models, larger datasets, and real-time data sources to enhance accuracy and reliability.

REFERENCES

[1] S. Al-Hadhrami, T. Vinko, T. Al-Hadhrami, F. Saeed, and S. N. Qasem, "Deep learning-based method for sentiment analysis for patients' drug reviews," *PeerJ Computer Science*, vol. 10, p. e1976, 2024.

[2] Z. Min, "Drugs reviews sentiment analysis using weakly supervised model," in *2019 IEEE international conference on artificial intelligence and computer applications (ICAICA)*. IEEE, 2019, pp. 332–336.

[3] J.-C. Na and W. Y. M. Kyaing, "Sentiment analysis of user-generated content on drug review websites," *Journal of Information Science Theory and Practice*, vol. 3, no. 1, pp. 6–23, 2015.

[4] S. G. Begum and P. K. Sree, "Drug recommendations using a "reviews and sentiment analysis" by a recurrent neural network," *Indonesian Journal of Multidisciplinary Science*, vol. 2, no. 9, pp. 3085–3094, 2023.

[5] M. Imani and S. Noferesti, "Aspect extraction and classification for sentiment analysis in drug reviews," *Journal of Intelligent Information Systems*, vol. 59, no. 3, pp. 613–633, 2022.

[6] S. Garg, "Drug recommendation system based on sentiment analysis of drug reviews using machine learning," in *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE, 2021, pp. 175–181.

[7] Y. Han, M. Liu, and W. Jing, "Aspect-level drug reviews sentiment analysis based on double bigru and knowledge transfer," *IEEE Access*, vol. 8, pp. 21 314–21 325, 2020.

[8] R. Haque, P. K. Pareek, M. B. Islam, F. I. Aziz, S. D. Amarth, and K. G. Khushbu, "Improving drug review categorization using sentiment analysis and machine learning," in *2023 International Conference on Data Science and Network Security (ICDSNS)*. IEEE, 2023, pp. 1–6.

[9] M. D. Hossain, M. S. Azam, M. J. Ali, and H. Sabit, "Drugs rating generation and recommendation from sentiment analysis of drug reviews using machine learning," in *2020 Emerging Technology in Computing, Communication and Electronics (ETCCE)*. IEEE, 2020, pp. 1–6.

[10] S. Vijayaraghavan and D. Basu, "Sentiment analysis in drug reviews using supervised machine learning algorithms," *arXiv preprint arXiv:2003.11643*, 2020.

[11] A. Mishra, A. Malviya, and S. Aggarwal, "Towards automatic pharmacovigilance: analysing patient reviews and sentiment on oncological drugs," in *2015 IEEE International conference on data mining workshop (ICDMW)*. IEEE, 2015, pp. 1402–1409.

[12] C. Colón-Ruiz and I. Segura-Bedmar, "Comparing deep learning architectures for sentiment analysis on drug reviews," *Journal of Biomedical Informatics*, vol. 110, p. 103539, 2020.

[13] D. Rathod, K. Patel, A. J. Goswami, S. Degadwala, and D. Vyas, "Exploring drug sentiment analysis with machine learning techniques," in *2023 International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 2023, pp. 9–12.

[14] B. Panda, C. R. Panigrahi, and B. Pati, "Exploratory data analysis and sentiment analysis of drug reviews," *Computación y Sistemas*, vol. 26, no. 3, pp. 1191–1199, 2022.

[15] S. Pradeep and V. UmaRani, "Drug sentiment analysis: A comprehensive study using regression models and natural language processing," in *International Conference on Computational Intelligence in Data Science*. Springer, 2024, pp. 16–28.

[16] K. K. Dasari, N. Bhaskar, L. Bagam, M. Madhusudhan, P. Santhuja, and R. R. Avala, "Drug review classification by using sentiment analysis," in *International Conference on Intelligent Systems and Sustainable Computing*. Springer, 2024, pp. 113–124.

[17] P. Duraisamy, Y. Natarajan, K. S. Preethaa, and K. Mouthami, "Sentiment analysis on drug reviews using diverse classification techniques," in *2022 3rd International Conference on Communication, Computing and Industry 4.0 (C2I4)*. IEEE, 2022, pp. 1–5.

[18] R. Haque, S. H. Laskar, K. G. Khushbu, M. J. Hasan, and J. Uddin, "Data-driven solution to identify sentiments from online drug reviews," *Computers*, vol. 12, no. 4, p. 87, 2023.

[19] D. Suhartono, K. Purwandari, N. H. Jeremy, S. Philip, P. Arisaputra, and I. H. Parmonangan, "Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews," *Procedia Computer Science*, vol. 216, pp. 664–671, 2023.

[20] K. K. Rao, K. Sravya, K. J. P. Sai, G. Giri, R. Saib, and G. Ganesanc, "Machine learning based drug recommendation from sentiment analysis of drug rating and reviews," in *Proceedings of the workshop on artificial intelligence (WAI 2022) co-located with Computing Congress (CC 2022)(pp. pages). CEUR Workshop Proceedings*, vol. 3146, 2022.

[21] N. Rathnasekara and U. Wijenayake, "Drug recommendation system based on medical condition classification and sentiment analysis of drug reviews," *The International Journal on Advances in ICT for Emerging Regions*, vol. 18, no. 2, 2025.

[22] K. C. Pérez, J. L. Sánchez-Cervantes, M. del Pilar Salas-Zárate, L. Á. R. Hernández, and L. Rodríguez-Mazahua, "A sentiment analysis approach for drug reviews in spanish." *Res. Comput. Sci.*, vol. 149, no. 5, pp. 43–51, 2020.