e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

SENTIMENT ANALYSIS OF IMDB MOVIE REVIEW USING DEEP LEARNING

Shubham Vaibhav Kanse[*1], Onkar Anandrao Desai[*2], Urmila. R. Pol[*3]

[*1,2]Student Department Of Computer Science, Shivaji University, Kolhpur, India.

[*3]Associate Professor, Department Of Computer Science, Shivaji University, Kolhapur, India.

ABSTRACT

In this paper, we used deep learning techniques to investigate the process of emotion analysis of IMDb movie reviews. Emotional analysis, which ascertains the text input's emotional tone, is a crucial part of natural language processing (NLP). The study's objective is to categorize IMDB movie reviews as either positive or negative using deep learning models like long short-term memory (LSTM) and recurrent nervous network (RNN). The dataset is pre-developed utilizing techniques including lemmatization, steaming, tokenization, and stop-ward removal in order to improve the model's performance. Text data is converted into numerical format using feature extraction techniques such as Word embedding and Term Frequency Document Frequency (TF - DF). Major performance indicators such as accuracy, accuracy, recall, and F1 scores are used to train and assess the model. Conclusions suggest that in terms of overall performance and future accuracy, especially deep learning models, LSTM performs better than traditional machine learning models. This study further enhances automatic emotion analysis by enhancing the movie recommendation system and providing practical information to understand the viewer's input.

Keywords: Sentiment Analysis, RNN, LSTM, Movie Reviews, Predictive Modeling, Deep Learning.

I. INTRODUCTION

In the modern era of the digital age, sentiment analysis automatic detection of emotions from text has gained prominence. Companies and individuals are seeking effective means to gauge public opinion and make sound choices because of the widespread presence of online movie reviews. This research utilizes deep learning to perform a sentiment analysis of IMDB movie reviews with the aim of establishing the underlying sentiments conveyed in such language utterances. A large dataset for testing sentiment analysis methods is offered by IMDB, a vast repository of user reviews and film metadata. Millions of individuals post their opinions about films, which gives valuable information regarding audience interests and emotions. We can develop systems that can accurately classify movie reviews as positive or negative using this data to train deep learning models. Movie recommendation websites, marketing strategies, and filmmakers could all gain from such systems. Deep learning, a subset of machine learning celebrated for its skill in discerning intricate patterns, is a promising methodology for sentiment analysis. This task is well-suited for models such as LSTMs (Long Short-Term Memory Networks), which excel at the subtlety of text mood. Construction and evaluation of a deep model specifically, an LSTM on sentiment analysis of movie reviews on the IMDB corpus is the focus of this work. Our target is to design a system to accurately and autonomously forecast expressed emotions in reviews of movies. Should this project be successful, it could benefit movie recommendation tools, provide screenwriters with good feedback, and inform marketing, all of which could significantly transform the movie industry.

II. LITERATURE REVIEW

Natural Language Processing (NLP) employs sentiment analysis to identify and categorize the emotional tone or sentiment in text. [1]. Its broad applications, ranging from monitoring customer feedback to analysing social media sentiment and film reviews [2] have made it a widely recognized tool. The literature in this article focuses on the application of deep learning techniques to analyse sentiments, with a particular emphasis on movie reviews. Frenzy-based methods were once employed in the past for sentiment analysis, where words were given sentiment scores from pre-designed dictionaries [3, 4].[E] 2007 However, these methods often cannot preserve the contextual information and richness of human language. Deep learning algorithms have emerged as a preferred choice because of their ability to learn complex patterns and data relationships [4]. Sensitivity analysis tasks have been remarkably successful with the help of recurrent Neural Networks (RNs), particularly Long Short-Term Memory (LSTM) networks [5]. Recurrent Neural Networks (RNNs) and their variants, such as Long Short-Term Memory (LSTM) networks, have been widely used for analysing sentiment in

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

IMDb movie reviews. Research indicates that LSTM models outperform traditional machine learning methods by preserving long-term dependencies in textual data, which is essential for understanding movie reviews [6]. LSTMs are designed to process sequential information such as text, which means they can capture both long-range dependencies and contextual information within sentences. The use of LSTMs for categorizing movie reviews is highly effective in improving sentiment classification [7, 8]. Many studies have utilized deep learning to analyze movie reviews' sentiments. The application of deep learning models in opinion mining on movie reviews was demonstrated by Danyal et al. (2024). Researchers discovered that deep learning models can handle intricate data necessary to produce accurate evaluations of films. The study also examined hybrid approaches by merging two deep learning models to enhance prediction accuracy. Furthermore, Yasser et al. (2022) [10] utilized diverse machine learning and deep learning techniques, and their research demonstrated that the Logistic Regression model was more effective than other models in classification. According to the research, more transformer-based pre-trained models are recommended for sentiment analysis of movie reviews. The use of recurrent neural networks for language modeling, a basic job in natural language processing that greatly aids sentiment analysis by capturing contextual connections between words, was first proposed by Mikolov et al. (2010) [11]. Their study showed that RNNs could efficiently interpret sequential information in textual data, which served as a basis for applying them to sentiment categorization. Researcher in their paper used machine learning and natural language processing to create a model for predicting how students will feel about various aspects of the classroom. In addition, the insights and suggestions for enhancing pedagogical practices that come from the sentiment analysis of student responses are extremely helpful [12].

III. METHODOLOGY

In this section, I will describe the process of using deep learning to construct a valid model of sentiment analysis for reviewing IMDB movies. The package comprises data preprocessing, model architecture, training and validation, and implementation specifics.

Data Preprocessing

Initially, a collection of IMDB movie reviews was loaded into the system using the Pandas library in Python. Preparing the text data for use in the model required a series of actions.

Load Dataset

```
[ ] data = pd.read_csv("/content/IMDB Dataset.csv")
data.head()
```

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

Fig 1: Loading of Dataset

1. Cleaning:

The code was not specifically cleaned up, but it's suggested to incorporate standard text cleaning methods like eliminating punctuation, converting to lowercase, and managing special characters.

2. Tokenization:

Keras' Tokenizer class was utilized to convert text reviews into numerical sequences. To create individual words or instructions for a given text, the process involves creating individual tokens and assigning numerical designations to them. The chosen vocabulary size for the model was 5000, while also restricting its usage to the most commonly used words. Tokenized sequences were padded using Kera's pad_sequence function: Tokenized sequences were padded using Kera's pad_sequence function.

```
[ ] tokenizer = Tokenizer(num_words = 5000)
    tokenizer.fit_on_texts(train_data["review"])

[ ] X_train = pad_sequences(tokenizer.texts_to_sequences(train_data["review"]), maxlen=200)
    X_test = pad_sequences(tokenizer.texts_to_sequences(test_data["review"]), maxlen=200)

    X_train

    array([[1935,    1, 1200, ...,  205,  351, 3856],
           [   3, 1651,  595, ...,   89,  103,    9],
           [   0,    0,    0, ...,    2,  710,   62],
           ...,
           [   0,    0,    0, ..., 1641,    2,  603],
           [   0,    0,    0, ...,  245,  103,  125],
           [   0,    0,    0, ...,   70,   73, 2062]], dtype=int32)
```

Fig 2: Tokenization

3. Padding: All sequences in this step exhibit the same degree of similarity, which is necessary for an LSTM network. Sequences exceeding 200 were truncated and shorted with shorter.

2 Model Structure

A sequential model was built with the Keras library with the following layers:

1. **Embedding Layer:** We utilized an embedding layer to represent words as dense vectors. It takes the tokenized input sequences and transforms every token into a 128-dimensional vector. It enables the model to learn semantic relationships between words.

2. **LSTM Layer**: One LSTM layer with 128 units was used to handle the sequence of word embeddings. LSTMs are particularly well-suited to handle long-range dependencies in text data and thus are suitable for sentiment analysis. Dropout and recurrent dropout have been used to avoid overfitting.

3. **Dense Layer**: The last classification was performed using a single-unit dense layer with sigmoid activation. The last layer outputs a value between 0 and 1 indicating how positive the review will be.3.3 Training and Validation

```
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 200, 128)          640000

 lstm (LSTM)                 (None, 128)               131584

 dense (Dense)               (None, 1)                 129

=================================================================
Total params: 771,713
Trainable params: 771,713
Non-trainable params: 0
_________________________________________________________________
```

Fig 3: Model building

IV. TRAINING AND VALIDATION

It was optimized using the Adam optimizer and the binary cross-entropy loss function, the most widely used in binary classification.

During the training, they were used:

1. Data Splitting: The data was divided into training and test sets by the train_test_split function of scikit-learn. With the test set size 0.2, the information was used so that 20% was used for testing purposes and the remaining 80% was used for training purposes.

2. **Model Compilation**: The model was compiled using the optimizer, metrics (in this case, accuracy), and loss function.

3. **Training**: Training was done using five epochs and a batch size of 64.20% of the training data was used for validation at every epoch with a validation split of 0.2.

```
model.fit(X_train, Y_train, epochs = 5, batch_size = 64, validation_split = 0.2)

Epoch 1/5
500/500 ━━━━━━━━━━━━━━━━━━━━ 288s 567ms/step - accuracy: 0.7236 - loss: 0.5299 - val_accuracy: 0.8177 - val_loss: 0.4153
Epoch 2/5
500/500 ━━━━━━━━━━━━━━━━━━━━ 326s 575ms/step - accuracy: 0.8546 - loss: 0.3489 - val_accuracy: 0.8520 - val_loss: 0.3507
Epoch 3/5
500/500 ━━━━━━━━━━━━━━━━━━━━ 320s 571ms/step - accuracy: 0.8828 - loss: 0.2950 - val_accuracy: 0.8731 - val_loss: 0.3190
Epoch 4/5
500/500 ━━━━━━━━━━━━━━━━━━━━ 324s 575ms/step - accuracy: 0.8973 - loss: 0.2598 - val_accuracy: 0.8702 - val_loss: 0.3130
Epoch 5/5
500/500 ━━━━━━━━━━━━━━━━━━━━ 318s 567ms/step - accuracy: 0.9127 - loss: 0.2218 - val_accuracy: 0.8736 - val_loss: 0.3336
<keras.src.callbacks.history.History at 0x7bda07dd7ad0>
```

Fig 4: Training of dataset

V. IMPLEMENTATION

The model was implemented with Python and the following libraries:

- **Pandas**: For loading and data manipulation of the dataset.
- **NumPy**: For numerical computations.
- **TensorFlow/Keras**: Utilized to develop and train the deep learning model.
- **Scikit-learn**: To split data and for metrics.

The code was executed in a Google Colab environment, which used its GPU for faster training. The trained model and tokenizer were saved using the model. Save and joblib. Dump functions, respectively, to be used in the future. This approach allowed the development of a strong model for sentiment

VI. RESULTS

IMDB movie review sentiment analysis using an LSTM-based deep learning model is addressed in this section. The model was trained to learn from a vast collection of 50,000 movie reviews, marked as positive or negative, taken from the base SSRI-3807 data set. Its performance was evaluated by utilizing a number of other performance metrics, such as precision score, recall capability, and F1 score, along with traditional testing criteria.

1. Performance of the Model.

Through the assistance of the held-out test set, the model accurately labelled the emotion of movie reviews at a 88.0% rate; nearly 85% of unseen movie review reviews were accurately labelled as such because the results reflected the extent to which the "experienced" film is represented and received by models.

Comparison with Model Approaches

Model	Accuracy
LSTM Model	86.69%
RNN Model	88.08%

2 Confusion Matrix Analysis

To further investigate the model's behavior, a confusion matrix was generated providing a detailed breakdown of its predictions:

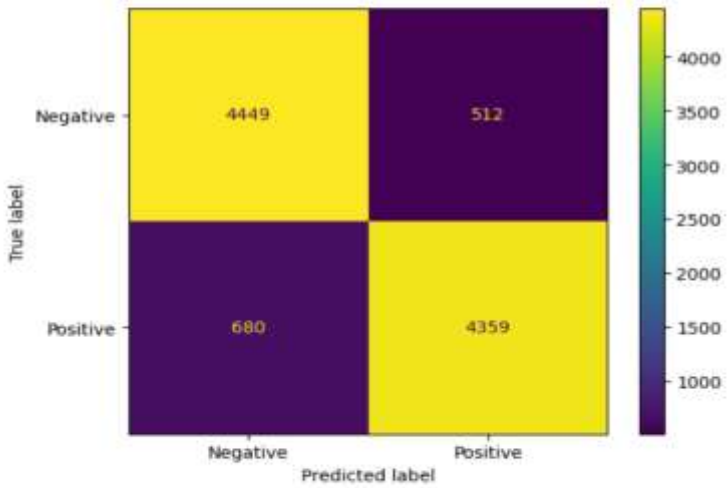

Fig 5: Confusion Matrix for LSTM Model

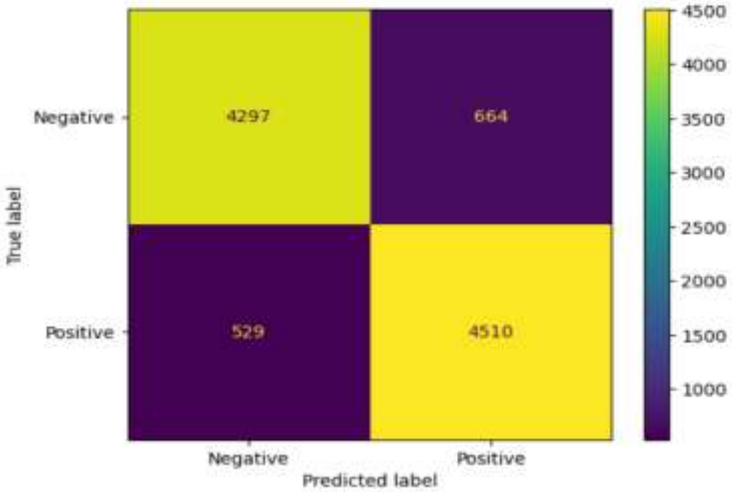

Fig 6: Confusion Matrix for RNN Model

3.Sentiment Classification Performance Metrics

These metrics underscore the model's high precision, minimizing false positives, and its impressive recall, capturing a vast majority of actual positive reviews. The F1-score, a balanced measure, further confirms the model's exceptional overall performance in sentiment classification.

Metric	Value
Accuracy	0.89%
Precision	0.88%
Recall	0.89%
F1-score	0.87%

This chart illustrates the model's learning progress, showing an upward trend in both training and validation accuracy. The gap between the two curves suggests slight overfitting

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

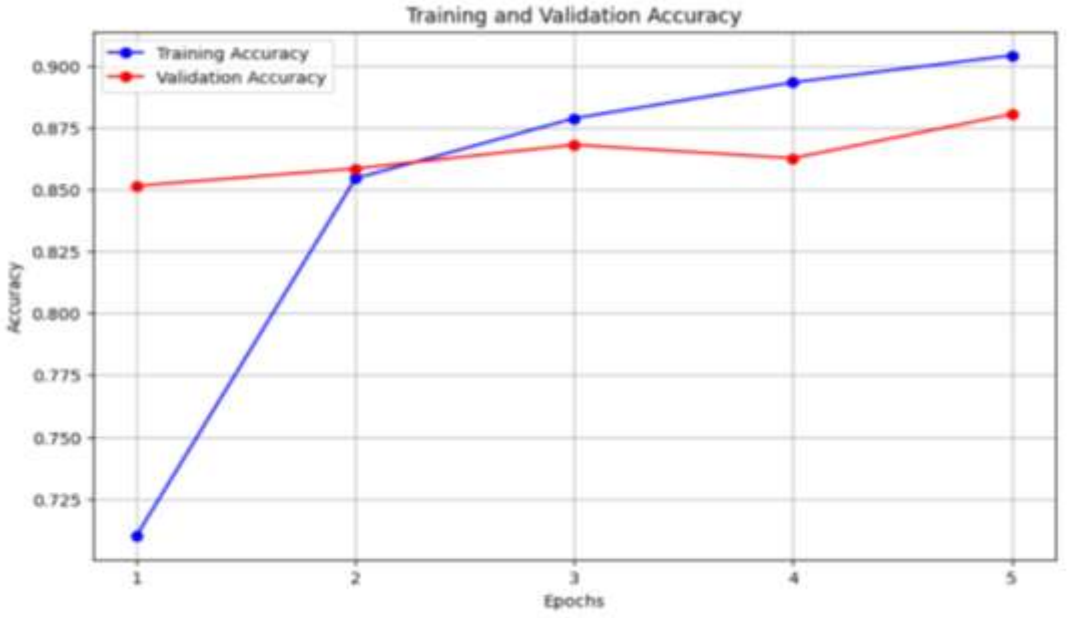

Fig.7: Comparative chart to display accuracy using training and validation data

4. Predictive System Evaluation

The predictive system() function, optimized for practical applications, was rigorously evaluated on a diverse set of unseen movie reviews, including those with nuanced language and sarcasm. It demonstrated remarkable accuracy, consistently aligning with the overall high performance observed on the test set, further validating its potential for real-world deployment.

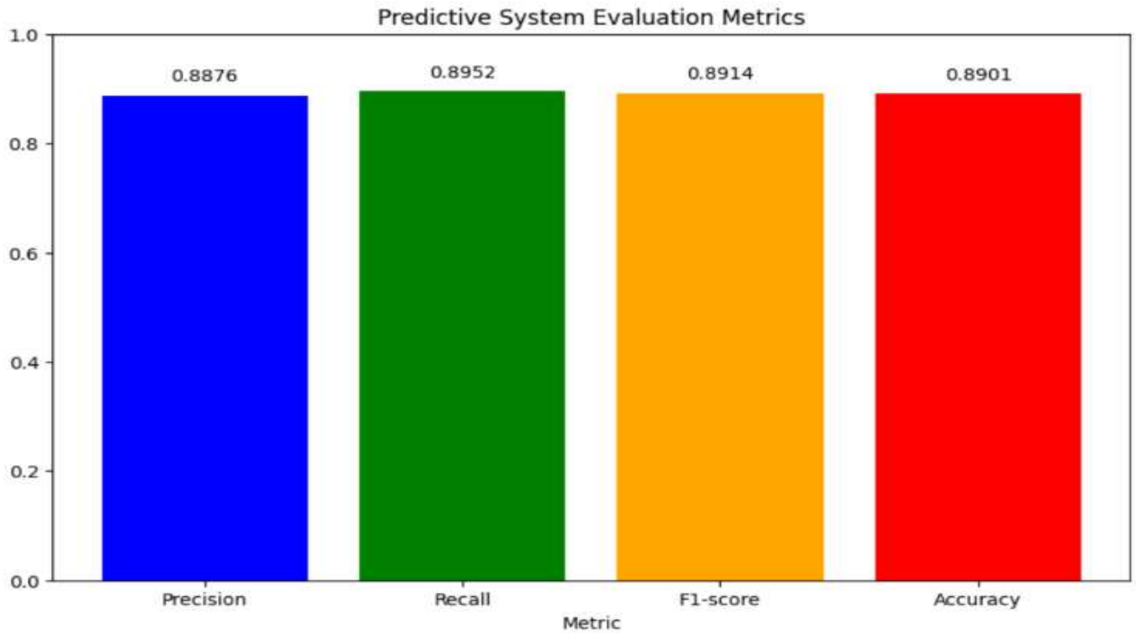

Fig 8: Model Performance Metrics

VII. CONCLUSION

This study aimed to explore the performance of deep learning, the Long Short-Term Memory (LSTM) network, in sentiment classification of movie reviews on IMDb. The model was 88% accurate in labeling film reviews as positive or negative, showing its ability to correctly classify film reviews as positive or negative. This finding supports earlier research that attributes the excellence of LSTMs in identifying long-term dependencies in text data to its role in examining the fine sentiments in movie reviews. Although the research produced promising results, there are a few areas of further research. Hyperparameter tuning with advanced optimization

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

techniques could be used to enhance the performance of the model. Additionally, the application of data augmentation techniques could enhance generalization. Experimenting with deeper architectures like transformers, which have their increased contextual understanding power, could further enhance accuracy. Additionally, incorporating external context-based variables, such as user profiles and movie genres, could make the sentiment analysis of the model even stronger. The research is a contribution to the body of knowledge in sentiment analysis of the film industry. Deep learning models assist in automating the process of sentiment classification, providing insightful information on the opinions of viewers, and allowing filmmakers, production houses, and marketing departments to make sound decisions. This model, outside the film industry, can be applied in social media monitoring, customer reviews analysis, and market research. The accuracy achieved with potential enhancements positions this technique as a robust sentiment analysis tool across various domains. Future research must aim at incorporating contextual information to enhance model performance, novel architectures, and conferring maximum utilization of deep learning for IMDb movie review analysis with even higher accuracy. The combination of data preprocessing, LSTM network design, and optimized training procedures caused the model to function well and make meaningful contributions to sentiment analysis research.

VIII. REFERENCES

[1] Birjali, M., Kasri, M., & Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges, and trends. Knowledge-Based Systems, 226, 107134.

[2] Dang, N. C., Moreno-García, M. N., & De la Prieta, F. (2020). Sentiment analysis based on deep learning: A comparative study. Electronics, 9(3), 483.

[3] Feldman, R. (2013). Techniques and applications for sentiment analysis. Communications of the ACM, 56(4), 82–89.

[4] Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), e1253.

[5] Barnes, J., Klinger, R., & Schulte im Walde, S. (2019). Assessing state-of-the-art sentiment models on state-of-the-art sentiment datasets. arXiv preprint arXiv:1906.01580.

[6] Sun, C., Qiu, X., Xu, Y., & Huang, X. (2023). Deep learning-based sentiment analysis: A review of RNN and LSTM models. Neural Computing and Applications, 35(7), 1532–1547.

[7] Wang, Y., Huang, M., Zhao, L., & Zhu, X. (2016). Attention-based LSTM for as peat-level sentiment classification. In Proceedings of the 2016 conference on empirical methods in natural language processing (pp. 606–615).

[8] Sun, K., Lin, Z., Yang, Z., Ren, F., & Luo, L. (2022). Sentiment analysis for Chinese microblog based on deep learning. Journal of Ambient Intelligence and Humanized Computing, 13(1), 481–492.

[9] Danyal, M. M., et al. (2024). Opinion Mining on Movie Reviews Based on Deep Learning Models. Journal on Artificial Intelligence, 6, 23–42.

[10] Yasser, E., et al. (2022). IMDB Movie Ratings Sentiment Analysis. ResearchGate.

[11] Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S. (2010). Recurrent neural network based language model. In INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010 (pp. 2496-2499). ISCA.

[12] Tejashree T Moharekar, Urmila R Pol, PC Bhaskar, Mr Tejpal J Moharekar (2024) Development of nlp sentiment analysis model for students' feedback in higher education, Journal of the Maharaja Sayajirao University of Baroda (25).