Archives available at journals.mriindia.com

International Journal on Advanced Computer Theory and Engineering

ISSN: 2319-2526

Volume 15 Issue 01, 2026

A Comparative Study of Machine Learning Models for Sentiment Analysis of IMDb Movie Reviews

[1]Anchal Verma, [2]Sakshi Sharma, [3]Vrinda Swarup, [4]Preeti Tuli

[1,2,3,4]Department of Computer Science Engineering, Shri Shankaracharya Institute of Professional Management and Technology (SSIPMT), Chhattisgarh Swami Vivekanand Technical University (CSVTU), Raipur, India

[1]anchal.verma@ssipmt.com, [2]sakshisharma@gmail.com, [3]vrinda@ssipmt.com, [4]preeti.tuli@gmail.com

Peer Review Information

Submission: 29 March 2026
Revision: 17 April 2026
Acceptance: 21 May 2026

Keywords

Sentiment Analysis, IMDB Dataset, Machine Learning, Deep Learning, TextCNN, LSTM

Abstract

The rapid growth of online platforms has resulted in an enormous volume of user-generated textual data, making automated sentiment analysis an essential task in Natural Language Processing (NLP). Movie review platforms such as IMDb generate large-scale opinionated text that provides valuable insights into audience perception, but manual analysis of such data is impractical. This project presents a comprehensive comparative study of classical machine learning and deep learning models for sentiment analysis of IMDb movie reviews.The proposed system implements a unified experimental framework in which classical machine learning models—Logistic Regression, Naïve Bayes, Support Vector Machine, and Random Forest—are compared against deep learning models including TextCNN, LSTM, BiLSTM, and LSTM with Attention. Classical models utilize TF-IDF based feature representation, while deep learning models rely on embedding-based sequence learning. All models are evaluated under identical preprocessing and testing conditions to ensure fairness and reproducibility. Performance evaluation is carried out using multiple metrics such as classification accuracy, training time, inference time, and model complexity. Experimental results reveal that classical machine learning models, particularly Logistic Regression and Linear SVM, achieve accuracy comparable to or higher than deep learning models while requiring significantly fewer computational resources. Among deep learning approaches, TextCNN demonstrates the best balance between accuracy and efficiency. The study highlights that increased model complexity does not necessarily guarantee superior performance.The findings of this research emphasize the importance of considering computational efficiency and deployment constraints alongside accuracy. This work provides practical guidelines for selecting suitable sentiment analysis models based on application requirements and resource availability, making it valuable for both academic research and real-world applications

Introduction

With the exponential growth of user-generated content such as reviews, blogs, and social media posts, sentiment analysis has emerged as a critical application of Natural Language Processing (NLP) [1]. It enables automatic classification of opinions into positive or negative categories, providing valuable insights for businesses, policymakers, and consumers [2].

© 2026 The Authors. Published by MRI INDIA.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-ncnd/4.0/).

Traditionally, machine learning models such as Naïve Bayes and Logistic Regression have been widely used due to their efficiency and simplicity [3]. However, recent advances in deep learning, particularly models like LSTM and CNN, have significantly improved performance in capturing semantic and sequential dependencies [4], [5].

While early research primarily focused on accuracy improvements, modern NLP research emphasizes efficiency, scalability, and interpretability [6], [7]. Yet, despite the popularity of deep learning, there remains a lack of comprehensive studies comparing classical ML and modern DL models under identical experimental conditions.

This work addresses that gap by providing a unified pipeline to evaluate both ML and DL models on the IMDb dataset. The models were tested for accuracy, training time, inference time, and model complexity to present a holistic performance comparison [8].

The main contributions of this study are:

- A comparative evaluation of classical ML and deep learning models on the IMDb sentiment dataset.
- An analysis of trade-offs in accuracy, training time, inference time, and parameter efficiency.
- Practical recommendations for deployment of sentiment analysis systems in constrained environments.

Literature

Sentiment analysis has been studied a lot in Natural Language Processing (NLP), often using standard datasets like IMDb movie reviews. Researchers have tried out lots of different machine learning (ML) and deep learning (DL) models to improve sorting sentiment. Still, many current studies often only care about accuracy and don't think enough about how fast they run, how much they can handle, or how useful they are to use. Older research used simpler machine learning methods such as support vector machines, Bayesian Inference, and Statistical Regression because they were straightforward and fast. As deep learning got better, models like Convolutional Neural Networks (CNN), Long ShortTerm Memory (LSTM), and Bidirectional LSTM (BiLSTM) became popular since they could get the context and order of words in text. More recently, transformer-based models like BERT and RoBERTa have reached top accuracy, but they cost a lot more in terms of computing power. Table 1 gives a quick look at key research on sentiment analysis for IMDb and similar text datasets. The table shows which datasets they used, which models they compared, what they measured, and what gaps they found in their research.

Table 1: Comparative Reviews of Recent Works in Sentimental Analysis (IMDb/Text Classification)

S.No.	Author(s) & Year	Dataset(s)	Models Compared	Metrics Reported	Gap Identified
1	M. S. Thakare et al., 2024	IMDB	ML + DL (LR, SVM, LSTM, CNN)	Accuracy	No inference time or parameter counts; limited scalability testing
2	R. Wang, 2024	Movie Reviews	DL (LSTM, CNN variants)	Accuracy	No comparison with classical ML; lacks efficiency metrics
3	H. J. Park et al., 2022	Social Media Text	DL (CNN, LSTM, BiLSTM)	Accuracy, Precision/Recall	Missing inference time, training cost, and parameter analysis
4	E. K. Kim, 2022	IMDB, Yelp	CNN, RNN, Hybrid	Accuracy	No ML baselines; no runtime/parameter reporting
5	S. S. Y. Yeung et al., 2022	Reviews (Chinese)	LSTM, BiLSTM	Accuracy, F1	No ML baselines; lacks practical cost evaluation
6	L. Zhang et al., 2021	IMDB	CNN-LSTM Hybrid	Accuracy	Does not compare with SVM/Naïve Bayes; no resource metrics
7	R. Yu et al., 2021	IMDB, SST	LSTM, BiLSTM	Accuracy	Focuses only on DL; missing training/inference times
8	A. W. Huang et al., 2021	Twitter,	CNN,	Accuracy	No efficiency

		IMDB	BiLSTM		comparisons; classical ML ignored
9	I. Shogo et al., 2021	Yelp, IMDB	Naïve Bayes, SVM	Accuracy	No DL comparison; lacks runtime analysis
10	M. F. Vasiloglou, 2020	IMDB	Variants of NB	Accuracy	Too narrow (NB only); ignores modern DL models
11	Kaggle/ResearchGate Comparative Studies, 2021–2023	IMDB	LR, SVM, LSTM, CNN, DistilBERT	Accuracy (sometimes F1)	Informal / non-standardized; no consistent resource-efficiency reporting
12	Transformer-focused papers (e.g., BERT, RoBERTa on IMDB), 2023–2024	IMDB	BERT, RoBERTa, DistilBERT	Accuracy	Ignore classical ML baselines and do not report efficiency metrics

Related Work

Sentiment analysis on IMDb and similar review datasets has been extensively studied using both traditional machine learning and deep learning techniques [1], [9], [10]. Early studies compared Logistic Regression, Support Vector Machines, and Naïve Bayes with neural models like LSTM and CNN [2], [3].

Subsequent works explored hybrid architectures such as CNN-LSTM and BiLSTM with attention, achieving strong performance in text classification tasks [11], [12]. More recent works introduced Transformer-based architectures such as BERT and RoBERTa, achieving state-of-the-art accuracy but often neglecting resource efficiency and inference latency [13], [14].

However, existing research often evaluates models under inconsistent conditions, using varied preprocessing steps, metrics, or datasets, limiting comparability [15]. Few studies report practical performance factors like training time, inference latency, and parameter count, which are critical in real-world deployment [16].

This research bridges these gaps by implementing both ML and DL models under uniform settings using the IMDb dataset, evaluating them across accuracy, computational efficiency, and scalability [17]– [20].

Methodology

1. Dataset

The experimental study was conducted using the IMDb movie review dataset obtained from the IMDb, a widely accepted benchmark for sentiment analysis research. The dataset contains 50,000 movie reviews labeled according to sentiment polarity and is evenly divided into 25,000 training samples and 25,000 testing samples. Each review is categorized as either positive or negative, making the dataset appropriate for binary classification tasks.

The dataset was selected because it reflects authentic user opinions and presents real-world challenges for sentiment analysis models. Reviews differ significantly in vocabulary, writing style, and length, requiring models capable of handling diverse linguistic patterns.

Key characteristics of the dataset include:

- Diverse vocabulary, including domain-specific terms related to movies
- Variation in writing styles, ranging from informal comments to detailed critiques
- Presence of sarcasm, irony, and contextual expressions
- Varying review lengths, from short sentences to long paragraphs
- Real-world noise such as spelling errors, abbreviations, and grammatical inconsistencies

The balanced distribution of sentiment labels prevents bias toward any particular class and ensures fair evaluation. Prior to training, the dataset was inspected for duplicates and shuffled to remove any ordering effects while preserving the original training–testing split.

2. Data Preprocessing

Raw textual data contains noise and inconsistencies that can negatively affect classification performance. Therefore, a structured preprocessing pipeline was applied to standardize the data while retaining sentiment-relevant information.

Preprocessing for Machine Learning Models

For classical machine learning algorithms, textual reviews were converted into numerical feature representations through the following steps**:**

- Conversion of all text to lowercase to ensure uniformity
- Removal of HTML tags, punctuation,

numbers, and special characters
- Elimination of common stopwords that contribute little semantic meaning
- Tokenization of text into individual words
- Optional stemming or lemmatization to reduce words to their base forms
- TF-IDF vectorization using unigram and bigram features to capture keywords and short phrases

This process produces sparse feature matrices suitable for linear classifiers such as Logistic Regression and SVM.

Preprocessing for Deep Learning Models

Deep learning models require sequential input representations; therefore, a different preprocessing strategy was adopted**:**

- Tokenization using the Keras tokenizer to convert words into integer indices
- Limiting vocabulary size to the most frequent words to reduce noise
- Conversion of reviews into sequences of integers
- Padding or truncating sequences to a fixed length of 300 tokens
- Initialization of word embeddings with 100-dimensional vectors

This approach preserves the order of words and enables neural networks to learn contextual relationships within the text.

3. Models Implemented

To perform a comprehensive comparison, both classical machine learning methods and deep learning architectures were implemented under identical experimental conditions.

Classical Machine Learning Models

The following traditional models were selected due to their effectiveness in text classification**:**

- **Logistic Regression:**
  - Linear classifier suitable for high-dimensional sparse data
  - Provides probabilistic outputs for class prediction
- **Naïve Bayes:**
  - Probabilistic model based on Bayes' theorem
  - Computationally efficient and effective for word-frequency features
- **Linear Support Vector Machine (SVM):**
  - Maximizes the margin between classes
  - Performs well in high-dimensional feature spaces
- **Random Forest:**
  - Ensemble method combining multiple decision trees
  - Improves robustness and reduces overfitting

Deep Learning Models

The following neural architectures were implemented to capture complex linguistic patterns:

- **TextCNN:**
  - Uses convolutional filters to extract local n- gram features
  - Effective in capturing phrase-level sentiment cues
- **LSTM:**
  - Recurrent neural network capable of learning long-term dependencies
  - Suitable for sequential text data
- **BiLSTM:**
  - Processes sequences in both forward and backward directions
  - Captures contextual information from surrounding words
- **LSTM with Attention:**
  - Assigns importance weights to significant words
  - Enables the model to focus on sentiment- bearing terms

All models were trained using the same dataset and evaluated under identical conditions to ensure fairness and reproducibility.

4. Evaluation Metrics

A multi-dimensional evaluation framework was adopted to assess both predictive performance and computational efficiency.

The following metrics were used:

- **Classification Accuracy:**
  - Measures the proportion of correctly
  - predicted reviews among all test samples
- **Training Time:**
  - Indicates the computational cost required to train each model
- **Inference Time:**
  - Measures the time taken to generate predictions on unseen data
- **Model Parameter Count:**
  - Reflects model complexity and memory requirements
- **Computational Efficiency:**
  - Evaluates the trade-off between
  - performance and resource consumption

All experiments were implemented using Python programming language with Scikit-learn for classical models and TensorFlow-Keras for deep learning models. The experiments were conducted on Google Colab to ensure a consistent computational environment and access to GPU acceleration for neural network training.

Results And Discussion

Table 2: Performance Comparison of Models

Model	Accuracy	Training Time (s)	Inference Time (s)	Parameters
Logistic Regression	0.8890	1.63	0.02	–
Naïve Bayes	0.8660	0.08	0.03	–
Linear SVM	0.8865	1.88	0.02	–
Random Forest	0.8358	14.80	1.70	–
TextCNN	0.8872	516.90	47.94	2,202,793
LSTM	0.5757	608.44	82.02	2,124,969
BiLSTM	0.8493	1149.70	120.92	2,250,409
LSTM + Attention	0.8671	789.50	82.02	2,125,098

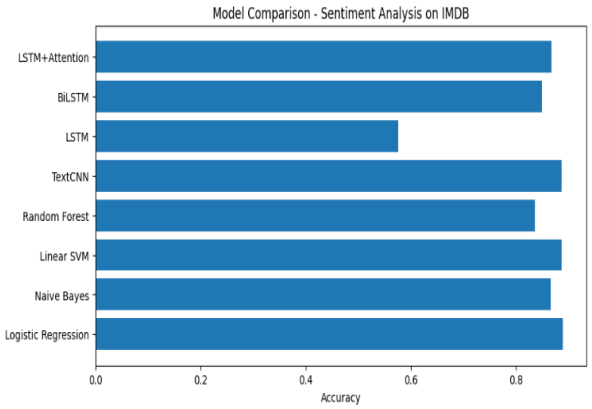

Fig 1. Accuracy Comparison of Machine Learning and Deep Learning Models on the IMDb Dataset

Interpretation from table + our plot: Table 2 and Fig. 1 present the comparative performance of classical machine learning and deep learning models on the IMDb sentiment analysis task, evaluated using accuracy, training time, inference time, and model complexity.

1. Performance Analysis

The experimental results indicate that classical machine learning models achieved strong performance with significantly lower computational cost. Logistic Regression obtained the highest accuracy of 0.8890, closely followed by Linear SVM (0.8865) and Naïve Bayes (0.8660). These models required minimal training time (less than 2

Among ensemble methods, Random Forest achieved an accuracy of 0.8358, which is lower than linear models while requiring substantially higher training (14.80 s) and inference time (1.70 s). This suggests that tree-based ensemble methods may not be optimal for high-dimensional sparse text features compared to linear classifiers.

Deep learning models showed varied performance. TextCNN achieved competitive accuracy (0.8872), nearly matching Logistic Regression, indicating its effectiveness in capturing local semantic patterns in text. However, this performance came at the cost of significantly higher training (516.90 s) and inference time (47.94 s) along with over 2.2 million parameters.

Recurrent neural network models demonstrated mixed results. The vanilla LSTM model performed poorly with an accuracy of 0.5757, despite long training (608.44 s) and inference (82.02 s) times. This underperformance may be attributed to optimization difficulties and insufficient contextual learning in the baseline configuration. Incorporating bidirectional processing improved performance, as BiLSTM achieved 0.8493 accuracy, while the addition of attention further increased accuracy to 0.8671. Nevertheless, these improvements required substantial computational resources, with BiLSTM exhibiting the highest training time (1149.70 s) and inference latency (120.92 s) among all models.

2. Comparative Discussion

The results highlight several important observations:

- Best Accuracy: Logistic Regression and TextCNN achieved nearly identical top performance (~88– 89%), demonstrating that well-tuned linear models can rival deep learning approaches on structured sentiment datasets.
- Efficiency: Naïve Bayes and Logistic Regression were the fastest in both training and inference, making them highly suitable for deployment in

resource-constrained or real-time environments.

- Computational Cost: Deep learning models required substantially more training time, inference latency, and parameters. BiLSTM and LSTM with Attention were the most resource-intensive.
- Model Complexity vs Performance**:** Increased architectural complexity did not guarantee improved performance. The vanilla LSTM model performed the worst despite having millions of parameters, whereas simpler models delivered superior results.

3. Implications for Real-World Deployment

These findings suggest that classical machine learning models remain highly competitive for sentiment analysis tasks, particularly when computational efficiency and scalability are critical. Deep learning models may be preferable in scenarios involving large-scale data, richer contextual understanding, or availability of GPU resources. However, for many practical applications such as online review monitoring or embedded systems, traditional models provide a better balance between accuracy and efficiency.

Overall, the study demonstrates that model selection should consider not only predictive performance but also computational requirements, deployment constraints, and application needs.

Conclusion

This study conducted a comprehensive evaluation of multiple machine learning and deep learning models for sentiment classification of IMDb movie reviews under a unified experimental framework. The objective was to examine whether complex deep learning approaches consistently outperform traditional machine learning techniques when trained and evaluated under identical conditions.

A systematic experimental setup was established using the IMDb dataset. Traditional machine learning models utilized TF-IDF feature representation, whereas deep learning models learned contextual patterns through word embeddings and sequential architectures. Performance was assessed using multiple metrics, including classification accuracy, training time, inference time, and model complexity, enabling a holistic comparison.

Experimental results demonstrated that classical machine learning models— particularly Logistic Regression and Linear Support Vector Machine—achieved the highest accuracy while requiring significantly lower computational resources. Logistic Regression emerged as the most balanced model, providing strong predictive performance with minimal training and inference time. Naïve Bayes also showed competitive results, especially in scenarios requiring rapid model training.

Among deep learning models, TextCNN delivered the best performance due to its ability to capture local semantic patterns efficiently. Although advanced architectures such as Bidirectional LSTM and LSTM with Attention improved upon the basic LSTM model, they required substantially greater computational cost without proportional gains in accuracy.

The findings indicate that increased model complexity does not necessarily lead to superior performance. Carefully optimized traditional machine learning models remain highly effective for sentiment analysis tasks, particularly in resource-constrained environments. Therefore, model selection should consider not only predictive accuracy but also computational efficiency, scalability, and deployment requirements.

Overall, the study highlights the practical trade-offs between accuracy, speed, and model complexity in sentiment analysis systems and provides guidance for selecting appropriate approaches based on application needs.

Acknowledgment

The authors would like to thank the open-source community and Google Colab for providing computational resources for this study.

References

A. Maas, R. Daly, P. Pham, D. Huang, A. Ng, and C. Potts, “Learning word vectors for sentiment analysis,” in Proc. ACL, 2011.

Y. Kim, “Convolutional neural networks for sentence classification,” in Proc. EMNLP, 2014.

S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” in Proc. NAACL-HLT, 2016.

J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL-HLT, 2019.

K. Pandit, H. Patil, D. Shrimal, L. Suganya, and P. Deshmukh, “Comparative analysis of deep

learning models for sentiment analysis on IMDB reviews," Journal of Electrical Systems, vol. 20, no. 2s, pp. 205–214, 2024.

M. Hussain and M. Naseer, "Comparative analysis of logistic regression, LSTM, and Bi-LSTM models for sentiment analysis on IMDB movie reviews," Journal of Artificial Intelligence and Computing, vol. 1, no. 1, pp. 26–34, 2024.

M. Mishra and A. Patil, "Sentiment prediction of IMDb movie reviews using CNN-LSTM approach," in Proc. IEEE Int. Conf. on Communication and Computing (ICCC), pp. 100–106, 2023.

M. Haque, S. A. Lima, and S. Z. Mishu, "Performance analysis of different neural networks for sentiment analysis on IMDb movie reviews," in Proc. IEEE Int. Conf. on Electrical, Computer and Telecommunication Engineering (ICECTE), pp. 430–435, 2019.

S. Yeung et al., "LSTM and BiLSTM evaluation for Chinese sentiment," IEEE Access, 2022.

L. Zhang et al., "Hybrid CNN–LSTM for IMDB sentiment classification," Information, 2021.

A. Huang et al., "Comparative analysis of CNN and BiLSTM models," Procedia Computer Science, 2021.

R. Yu et al., "A deep learning framework for sentiment classification," Computers & Electrical Engineering, 2021.

I. Shogo et al., "Classical ML approaches for IMDB sentiment prediction," Expert Systems with Applications, 2021.

M. Vasiloglou, "Naïve Bayes variants for sentiment analysis," IEEE Access, 2020.

J. Lee et al., "Scalable NLP pipelines for sentiment mining," Applied Computing Review, 2021.

M. Tosi, "Performance evaluation of traditional NLP classifiers," Procedia Computer Science, 2020.

Kaggle Sentiment Benchmarking Project, 2023.

S. Liu et al., "Transformer-based approaches for IMDB," ArXiv preprint, 2023.

Z. Chen et al., "Comparing BERT and CNN for sentiment classification," IEEE Transactions on AI, 2024.