International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0255
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1738 - 1742

Sentiment Analysis with Cnn-Lstm Model for Opinion Mining

Shalini S[1], Sreemathi G[2], Suwathini Sree G[3], Sudha Devi K[4]
[1, 2, 3] Student-Department of computer science and engineering, Paavai Engineering College, Namakkal, Tamilnadu, India.
[4]Associate professor- Department of Computer Science and Engineering, Paavai Engineering College, Namakkal, Tamilnadu, India.
EmailID:shalinisivasankar21@gmail.com[1],sreemathigovindasamy046@gmail.com[2], gsuwathinisree82@gmail.com[3], sudhajay03@gmail.com[4]

Abstract
Sentiment analysis is an important research area in natural language processing that focuses on identifying opinions, emotions, and attitudes expressed in textual data. With the increasing popularity of social media platforms and online review systems, large volumes of user-generated content are created every day. Extracting meaningful insights from this textual data requires advanced analytical techniques. This research proposes a hybrid deep learning model that combines convolutional neural networks (CNN) and long short-term memory (LSTM) networks for sentiment classification. The component captures long-term contextual dependencies. Experimental results demonstrate that the hybrid CNN-LSTM architecture significantly improves sentiment classification accuracy compared to traditional machine learning approaches.
Keywords: *Sentiment Analysis, Opinion Mining, CNN, LSTM, Deep Learning, Natural Language Processing*

1. Introduction
The rapid advancement of digital communication technologies has led to a significant increase in the volume of textual data generated daily. This includes product reviews, social media posts, blogs, and customer feedback. Analyzing this data to extract meaningful insights is crucial for organisations aiming to improve their services and understand customer preferences[1]. Sentiment analysis, also known as opinion mining, is the computational study of opinions, sentiments, and emotions expressed in text. It plays a vital role in applications such as brand monitoring, customer satisfaction analysis, and market research. Conventional approaches to sentiment analysis include rule-based systems and machine learning models such as Naïve Bayes and Support Vector Machines. Traditional approaches rely on machine learning models such as Naïve Bayes and Support Vector Machines. These methods require extensive feature engineering and often fail to capture the contextual nuances of language. Deep learning techniques have revolutionized NLP by enabling automatic feature extraction and improved contextual understanding. In this work, a hybrid CNN-LSTM architecture is proposed to leverage the strengths of both models, thereby improving sentiments classification performance.

2. Related Works
Numerous studies have been conducted in the field of sentiment analysis using different approaches. Lexicon-based approaches rely on predefined sentiment dictionaries where each word is assigned a polarity score. While these methods are simple and interpretable, they lack the ability to handle context, sarcasm, and domain-specific language. Machine learning approaches such as Naïve Bayes, Decision Trees, and Support Vector Machine use labelled dataset to train classifier. Although these models perform better than lexicon-based method, they depend heavily on manual feature extraction techniques such as n-grams and TF-IDF. Deep learning models, particularly CNN-LSTM, have demonstrated superior performance in sentiment analysis tasks. CNNs are effective in capturing local features and identifying key patterns in text, whereas LSTMs are capable of modelling long-term dependencies in sequential data. Recent research has focused on hybrid architectures combining CNN and LSTM to achieve improved performance leveraging both local and global contextual information[2].

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0255
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1738 - 1742

3. Problem Statement

Despite significant advancements in sentiment analysis several challenges continue to hinder its overall effectiveness and reliability. One of the primary concerns is the difficulty in capturing contextual relationship within text. Language is inherently complex, and the meaning of words often depends on their context within a sentence or paragraph Table 1. Traditional models frequently fail to account for these nuances, leading to incorrect sentiment predictions, especially when dealing with polysemous words or context-dependent phrases. Another major limitation lies in the inability of conventional models to efficiently process long textual sequences. As the length of the input increases, these models tend to lose important information from earlier parts of the text, resulting in incomplete understanding Figure 1. This issue becoming particularly critical in applications such as reviews, articles, and social media discussions, where sentiments are expressed over extended passages rather than short, isolated sentences. Furthermore, many existing approaches rely heavily on manual feature engineering, which involves selecting features from the text data, This process is not only time-consuming but also requires domain expertise, making it less scalable and adaptable to diverse datasets. Manually engineered features may fail to capture hidden patterns and complex linguistic structures, thereby limiting the model's ability to generalize across different contexts and domains. In addition to these challenges, sentiment analysis systems often perform poorly when handling ambiguous and sarcastic expressions. Sarcasm, irony, and implicit meanings are common in human communications, yet they are difficult for machines to interpret accurately[2]. A sentence may appear positive on the surface but convey a negative sentiment in reality, leading to misclassification and reduced model performance. These limitations collectively result in decreased accuracy, robustness, and reliability of sentiment classification systems. Consequently, there is a growing need for a more advanced and robust model that can automatically learn and extract meaningful feature from raw text data while effectively capturing long-range dependencies and contextual relationships. Such a model would significantly improve the performance of sentiment analysis systems, making them more suitable for real-world applications.

4. Existing System

Existing sentiment analysis systems primarily rely on traditional machine learning techniques and rule-based approaches Table 2. These methods have been widely used due to their simplicity and effectiveness on smaller datasets. Commonly used techniques include Naïve Bayes classifier, Support Vector Machines, and Logistic Regression. In addition, lexicon-based sentiment analysis approaches are also employed, where predefined dictionaries of positive and negative words are used to determine sentiment polarity[3]. Manual feature extraction required, Limited contextual understanding, Lower classification accuracy.

5. Proposed System

The proposed system uses a CNN-LSTM hybrid architecture to improve sentiment classification[4].

Text preprocessing Module: The text preprocessing module is responsible for cleaning and preparing raw textual data before it is fed into the model. It involves steps such as converting text to lowercase, removing punctuation and special characters, tokenizing sentences into words, and optionally removing stop words or applying stemming and lemmatization. This process helps reduce noise and ensures that the input data is consistent and structured, which ultimately improves the performance of the model[5].

Word Embedding Layer: The word embedding layer converts the pre-processed text into numerical vector representations the capture the semantic meaning of words. Each word is mapped to a dense vector, allowing the model to understand relationships between words, such as similarity and context. These embeddings can be either pre-trained or learned during training, and they serve as the input for the subsequent layers in the model.

Convolutional Neural Network: The CNN layer is used to extract local feature from the embedded text. It applies convolutional filters to identify important patterns such as phrases, keywords, and sentiment-bearing expressions within the text. By using pooling operations, the CNN reduces dimensionality while

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0255
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1738 - 1742

retaining the most relevant features, making it effective for capturing meaningful information from short text segments.

Long Short-Term Memory (LSTM) Network: The LSTM network is designed to capture sequential dependencies and contextual relationships in the text. Unlike traditional neural networks, LSTM can remember important information over long sequences, enabling it to understand how words influence each other in a sentence. This is particularly useful for sentiment analysis, where context and word order play a crucial role.

Advantages

- Improved contextual understanding
- Enhanced classification accuracy
- Scalability for large datasets

-Reduced dependency on manual feature engineering

6. Opinion Mining

Opinion mining focuses on extracting opinions from text data.

The process includes:

- Identifying sentiment polarity
- Detecting subjectivity
- Extracting opinion targets

Applications include product review analysis, political opinion monitoring, and social media analytics.

7. Dataset Description

The dataset used in this research consists of customer reviews collected from online platforms.

Each review is labelled according to sentiment polarity.

Table 1 Classes Positive / Negative

Dataset	Summary
Total Reviews	50,000
Training Data	40,000
Testing Data	10,000

8. Dataset Analysis

Dataset analysis helps understand the characteristics of the textual data.

Class Distribution: Balanced datasets ensure fair classification performance.

Text Length Analysis: Sentence length varies across reviews and influences neural network input size[6].

9. Methodology

Data Acquisition: The dataset used in this study consists of labelled textual data obtained from online platforms such as product review websites and social media. Each data instance contains a text review along with its corresponding sentiment label (positive or negative)[7]. The datasets is divided into training and testing sets to evaluate model performance effectively.

Data Preprocessing: Data preprocessing is a crucial step in preparing raw text for model training. It involves cleaning and transforming the text into a suitable format. Key preprocessing steps include:

- Tokenization: Splitting text into individual words or tokens.
- Stop word removal: Eliminating common words that do not contribute to sentiment.
- Lemmatization: Converting words to their base form.
- Text normalization: Removing special characters, punctuation, and converting text to lowercase.

These steps help improve the quality of input data and enhance model performance.

Text Representation

Text data must be converted into numerical form before being processed by machine learning models.

Common techniques include:

Bag of Words (BOW)

- Term Frequency-Inverse Document Frequency (TF-IDF)
- Word Embeddings

Among these, word embeddings are preferred as they capture semantic relationships between words and provide dense vector representations.

Word Embedding: Word embeddings map words into continuous vector spaces where semantically similar words are positioned close together.

Popular embedding techniques include:

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0255
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1738 - 1742

- Word2Vec
- GloVe
- FastText

These embeddings improve the model's ability to understand context and relationships between words.

10. Model Description

CNN for Text Classification: CNN is used to extract local feature from text data. It applies convolutional filters over word embeddings to detect important patterns such as phrases and sentiment indicators. The output of the convolutional layer is passed through pooling layers to reduce dimensionality and retain the most significant features.

LSTM Networks: LSTM is a type of Recurrent Neural Network (RNN) designed to handle sequential data and capture long-term dependencies. It uses memory cells and gating mechanisms to retain important over time, making it highly effective for understanding context in sentences[8].

Hybrid CNN-LSTM Architecture: The hybrid model combines CNN and LSTM to leverage their respective strengths. Workflow: Input Text -> Word Embeddings -> CNN Layer -> LSTM Layer -> Fully Connected Layer -> Output. This architecture enables the model to capture both local and global features, resulting in improved classification performance.

11. System Architecture

The system architecture consists of:

- Data Layer
- Preprocessing Layer
- Feature Extraction Layer
- CNN Layer
- LSTM Layer
- Classification Layer

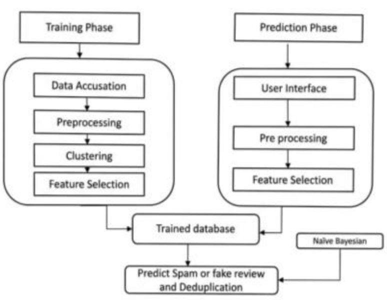

Figure 1 System Architecture

12. Model Training

The model is trained using supervised learning with labelled datasets. Key components include:

- Loss Function: Binary Cross Entropy
- Optimizer: Adam
- Batch Size and Epochs: Configurable parameters

During training, the model adjusts its weights to minimize the loss function and improve prediction accuracy.

13. Hyperparameter Tuning

Learning rate, Batch size, Epochs, Convolution filters, LSTM hidden units.

14. Experimental Setup

The proposed model is implemented using:

- Programming Language: Python
- Libraries: TensorFlow, Keras, NumPy, Pandas
- Hardware: GPU-enabled system for faster computation

The dataset is split into training sets, and experiments are conducted to evaluate model performance.

15. Evaluation Metrics

The performance of the model is evaluated using standard metrices:

- Accuracy: Overall correctness of predictions
- Precision: Proportion of correctly predicted positive instances
- Recall: Ability to identify all positive instances
- F1 Score: Harmonic mean of precision and recall

These metrices provide a comprehensive evaluation of the model's effectiveness.

16. Performance Comparison

Table 2 Model accuracy comparison

Model	Accuracy
Naive Bayes	78%
SVM	84%
CNN	89%
LSTM	90%
CNN-LSTM	93%

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0255
e ISSN: 2584-2854
Volume: 04 Issue: 05
May 2026
Page No: 1738 - 1742

17. Applications

Applications of sentiment analysis include:

- Social media monitoring
- Customer feedback analysis
- Brand reputation tracking
- Market research.

Conclusion

The paper presents a hybrid CNN-LSTM model for sentiment analysis, combining convolutional Neural Network (CNNs) and Long Short-Term Memory (LSTM) networks to capture both local and sequential features from textual data. The model leverages CNNs to extract local features, such as n-gram patterns, and LSTMs to capture sequential dependencies and long-range relationships. The architecture consists of an embedding layer, CNN layer, LSTM layer, and dense layer, producing improved sentiment classification accuracy compared to traditional machine learning methods. The hybrid model demonstrates strong potential for large-scale opinion mining applications, such as social media monitoring and customer feedback analysis. Experimental results on benchmark datasets, including IMDS and Twitter US Airline Sentiment, show significant improvements in accuracy and F1-score. Future directions include incorporating pre-trained word embedding, integrating attention mechanisms, and extending model to support multiple languages and dialects.

References

[1]. S. Poria, E. Cambria, and A. Hussain, "Multimodal Sentimental Analysis: Addressing Key Issues and setting up the Baselines,'' IEEE Intelligent Systems, 2017.

[2]. A. Zadeh, M. Chen, and L. Morency, "Tensor Fusion Network for Multimodal Sentiment Analysis'', IEEE Conference on Empirical Methods in Natural Language Processing (EMNLP), 2017.

[3]. Z. LIU, M. Sun, and T. Zhao, "Multimodal Sentiment Analysis with deep neural networks", IEEE Transactions on Neural Network and Learning Systems, 2021.

[4]. M. He, "A Review of data Fusion and Deep Learning Models for multi model Sentiment Analysis,'' IEEE Access, 2022.

[5]. K. Zhao, "Multi model Sentiment Analysis: A Comprehensive Survey from a Fusion methods Perspective," IEEE Transactions on Affective Computing, 2022.

[6]. Y. Wang, L. Huang, and X. Zhao, "Attention-Based Multi model Fusion for sentiment classification", IEEE Access,2022.

[7]. H. Fan and J. Chen, "Position-Perspective Multi-Hop Fusion Network for Multi model Aspect-Based Sentiment Analysis," IEEE Access, 2023.

[8]. R. Chen, "MSAF-CF: A Multi model Sentiment Analysis Framework Based on Feature Enhancement and Cross-Fusion," IEEE Transaction on Multimedia,2023.