Leveraging Word2Vec-Enhanced CNN-LSTM Hybrid Architecture for Sentiment Analysis in E-Commerce Product Reviews

Original Scientific Paper

Kosala Natarajan*
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India
kosala.nataraj@gmail.com

Nirmalrani V
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India
nirmalrani.it@sathyabama.ac.in

Gowri S
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India.
gowri.it@gmail.com

Ramya G Franklin
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India.
mikella.prabu@gmail.com

Poornima D
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India.
poorniramesh2011@gmail.com

Jabez J
Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology,
Jeppiar Nagar, Chennai, Tamil Nadu - 600119, India.
jabezme@gmail.com

*Corresponding author

Abstract *– The amalgamation of machine learning (ML) techniques and natural language processing (NLP) is leveraged to evaluate the sentiment of textual input. With the increasing popularity of e-commerce platforms like Amazon, product reviews have emerged as an essential source of information for potential purchasers, providing insights into product quality and performance from the consumers' viewpoints. This study aims to systematically organize and analyze customer opinions to effectively capture consumer sentiment based on product reviews. In this study, we propose a deep learning framework that combines a stacked 1D convolutional layer (CNN) with a Long Short-Term Memory (LSTM) network, using pre-trained Word2Vec embedding as fixed input representations. Evaluated on a large Amazon product review dataset, our model — StackedCNN-LSTM-W2V — achieves a classification accuracy of* **99 %**, *outperforming traditional CNN, LSTM, and logistic regression baselines.*

Keywords*: Sentiment analysis, Amazon product reviews, StackedCNN-LSTM, Text classification, Deep learning, Word embedding*

Received: April 18, 2025; Received in revised form: August 13, 2025; Accepted: August 20, 2025

1. INTRODUCTION

Sentiment analysis (SA) is a branch of Natural Language Processing (NLP) that utilizes machine learning methods to evaluate textual information. It has garnered considerable interest from researchers and developers owing to its efficacy in assessing the polarity of textual content—positive, negative, or neutral. SA has been extensively utilised across many text data types, including product reviews on e-commerce platforms such as Amazon.

Amazon has become a central hub for product feedback, where consumers share their experiences with various items, from electronics to household goods. These reviews serve as valuable insights for both potential buyers and businesses. Customers rely on product reviews to make informed purchasing decisions, while companies analyze sentiment trends to improve product quality, enhance customer satisfaction, and refine marketing strategies [1].

With the transition from traditional to digital marketing, consumer behavior has changed significantly, but word of mouth remains crucial. Platforms like Amazon, Meesho, and Ajio enable customers to share their insights on products and services, influencing other buyers and business strategies. Likewise, major social platforms contribute significantly to shaping consumer perception and corporate reputations.

Customers can acquire significant information regarding a product's quality by perusing the reviews, hence facilitating more informed purchasing judgments. Corporations can utilise this feedback to gauge client happiness and enhance their products or services [2]. Large volumes of customer reviews make manual analysis impractical and inefficient.

The vast amount of unstructured textual data requires transformation into a computable format for efficient processing. Sentiment analysis provides a feasible solution by utilizing text mining techniques to extract subjective information and classify sentiment polarity [3]. Traditional approaches struggle to identify intricate patterns and contextual nuances in reviews, making deep learning(DL) techniques a more viable approach.

Numerous approaches exist for processing product reviews, with DL being one of the most used methods, employing neural networks with numerous layers. DL has been widely utilised in numerous academic domains, including NLP and SA.

Deep learning is chosen for analyzing Amazon reviews due to its ability to capture complex text patterns, improving sentiment classification accuracy. In contrast to conventional machine learning methods, deep learning models exhibit enhanced efficacy in comprehending context, managing long-range dependencies, and deriving significant insights from user feedback.

Deep learning approaches, particularly LSTM and CNN, have shown remarkable improvements in sentiment classification [4]. Traditional machine learning models, though effective, are limited in capturing long-term dependencies and complex linguistic structures. Deep learning models, particularly those combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, have demonstrated strong performance in extracting both spatial and temporal features. This study proposes an advanced model integrating deep stacked CNN layers, frozen Word2Vec embedding, and a robust regularization strategy, specifically optimized for Amazon product reviews, a domain known for its verbose, informal, and sentimentally complex text.

The key contributions of this research are as follows:

- **Utilization of Word2Vec for Feature Extraction**: This study uses Word2Vec to transform text into numerical vectors, capturing rich semantic meanings from review data
- **Development of a Hybrid CNN-LSTM Model**: The proposed StackedCNN-LSTM -WordtoVec model integrates CNN and LSTM layers to extract spatial and sequential features from review text.
- **Feature Extraction using CNN**: CNN is utilized to identify important textual patterns, such as word combinations and sentiment cues, contributing to enhanced feature learning.
- **Context Understanding with LSTM**: By integrating LSTM, the model can learn and retain extended contextual and sequential patterns.

2. RELATED WORKS

Sharma *et. al.* [5] demonstrated that Word2Vec embedding, particularly when kept frozen, outperforms Glove and FastText in deep learning models such as CNN-LSTM when applied to product review datasets. This was attributed to Word2Vec's ability to retain meaningful local word relationships, which is critical for sentiment classification.

Zarei et al. [6] showed that Word2Vec outperforms other embeddings in tasks involving longer English texts, such as e-commerce or social media reviews. Their study highlighted that while glove captures global word co-occurrence, Word2Vec is more effective at encoding local semantic structure, which helps in capturing subtle sentiment nuances.

Hashmi et al. [7] share a similar goal with our research, enhancing sentiment classification performance on Amazon product reviews using various embedding strategies and classification algorithms. While their work adopts a hybrid modelling framework integrating multiple machine learning and deep learning methods, our study emphasizes performance analysis using carefully selected embedding techniques and overfitting control strategies. Both approaches aim to improve classification accuracy and interpretability on real-world e-commerce review data. BERT model reached an accuracy of 89%. The comparative results help validate the effectiveness of different modeling strategies, offering valuable insights into the advantages and drawbacks of different sentiment analysis techniques.

Shamal et al. [8] employed the LSTM model in their research, yielding enhanced outcomes for SA tasks. Furthermore, Guner et al. [9] established that the LSTM outshone competing models in terms of accuracy for binary SA. Atikur employed a solitary convolutional layer on two distinct datasets to demonstrate aspect extraction in Bangla reviews using a CNN. Although the SVM demonstrated remarkable precision, the proposed CNN model attained the maximum recall and F1-score across both datasets [10]. S. M. Qaisar [11] applied an LSTM-based model for sentiment analysis using the IMDB movie review dataset. The study emphasized the importance of preprocessing to improve classifier compatibility and demonstrated that LSTM effectively captured contextual dependencies in textual data. The model achieved a classification accuracy of 89.9%.

Mathieu Cliché, et al. [12] employed a CNN and LSTM-based methodology trained on the SemEval-2017 Twitter dataset, utilizing an extensive collection of unlabeled data and pre-trained word embedding. This hybrid methodology exhibited substantial enhancements in classification precision. The outlined approach consisted of five essential stages: reading the

CSV file containing Twitter data, preprocessing, feature extraction, and classification. The investigation employed two methodologies: the initial method implemented a conventional ML technique on the dataset, whereas the subsequent method leveraged deep neural network-based approaches.

Priya Darshini [24] proposed a hybrid architecture named HAF-wBiLSTM for customer satisfaction prediction using Amazon product reviews. The model integrates Bag-of-Words features with a weighted bidirectional LSTM, allowing for the extraction of contextual and dependent information. Their model was evaluated against CNN, LSTM, Tree-LSTM, and SVM, and showed superior performance and dependability, with notable improvements in accuracy of 94% and interpretability across benchmark datasets.

Anbumani [25], a novel sentiment analysis framework combining BERT, BiGRU, and Graph Neural Networks was proposed for classifying customer feedback in e-commerce settings. This deep learning-based method effectively extracted sentiment-related features and achieved a high classification accuracy of 93.35%. The study emphasizes its utility in monitoring customer and employee sentiments to support strategic decisions. The proposed system demonstrated superior performance over existing models on multiple datasets.

The polarity of tweets was predicted using an LSTM model in [13], while text sentiment classification was accomplished using a hybrid technique that combined CNN and LSTM in [14]. Cliche [15] proposed an ensemble model combining CNN and LSTM architectures, which ranked first in all five English sub-tasks of SemEval-2017. With a focus on classifying sentiment polarity in social media data, Meena et al. [16] presented the use of CNN for SA. A remarkable accuracy rate of 95.4% was achieved by methodically classifying user comments from a variety of ethnic groups into sentiment classifications.

Basiri et al. [17] introduced a bidirectional CNN-RNN model that integrates BiLSTM and BiGRU layers with an attention mechanism for the sentiment analysis of product evaluations on Twitter. The model efficiently caught both historical and prospective context, emphasizing significant sentences through attention mechanisms. It attained an accuracy of 92.44% and concentrated on document-level sentiment, proposing enhancements for forthcoming recommender systems.

In order to classify customer reviews into four different sentiment classifications, et al. [18] presented a sophisticated method that uses LSTM and fuzzy logic. Three benchmark datasets were used to evaluate this model: customer reviews of Amazon products, reviews of Amazon video games, and reviews of Amazon mobile phones. The corresponding accuracy rates were 96.03%, 83.82%, and 90.92%.

Singh *et. al.* [19] used Logistic Regression as a baseline model. Despite being straightforward, their method produced findings that were reasonably accurate, with an F1-score of 83.5% an accuracy of 85%. This model serves as a basic standard for sentiment classification tasks and emphasizes the need for more sophisticated models to adequately capture subtle sentiments, despite its limitations in handling complicated linguistic patterns.

In order to achieve significant gains in sentiment analysis performance, Anbumani and Selvaraj [20] presented a CNN model with a new SigTan-Beta activation function. The CNN-SigTan-Beta model achieved 94.5% accuracy. This model is appropriate for capturing local dependencies in Amazon product evaluations since it can successfully identify sentiment-indicative keywords and phrases in text. Nevertheless, despite its strength, the CNN design might not be able to properly capture lengthy text relationships, which could hinder its ability to process complicated sentence patterns.

The GRU model was used by Chen et al. [21] to examine Amazon reviews. In this investigation, GRUs, which are renowned for their effectiveness when processing sequential data, achieved an accuracy of 92%. The model was successful in correctly classifying both positive and negative attitudes, as evidenced by its high precision and balanced recall. Because of its effectiveness, the GRU can be used for sentiment analysis tasks that call for capturing contextual dependencies without incurring the computational costs of more intricate models.

A Bidirectional LSTM network was used by Kumar et al. [22] to identify both forward and backward dependencies in Amazon product reviews. With an accuracy of 93.5%, this strategy demonstrated remarkable efficacy. Because of its bidirectional construction, the Bi-LSTM is very useful for SA, where word order has a big influence on sentiment classification.

The literature review indicates that machine learning techniques have demonstrated effectiveness in sentiment classification tasks. Recently, advanced techniques in deep learning have been utilized to enhance accuracy and predictive performance. This served as motivation for us to solve our issue set using a hybrid CNN-LSTM architecture. This methodology involves the initial application of a CNN, succeeded by the integration of an LSTM layer and attention mechanism. This architecture improves performance and achieves a superior F1-score when compared to conventional methods.

3. PROPOSED METHODOLOGY

Our research aims to classify product reviews into Positive and Negative sentiments using a hybrid CNN-LSTM model referred to as Stacked CNN-LSTM- Word-2vec. Fig. 1 shows the proposed model.

This study leverages large-scale review data from e-commerce platforms to analyze customer feedback, offering valuable insights into product performance and consumer satisfaction. The proposed model integrates convolutional layers to extract local textual features

and LSTM units to capture sequential dependencies. This architecture significantly improves sentiment classification accuracy by effectively modeling both spatial and temporal aspects of the review texts.

While the CNN-LSTM hybrid architecture has been explored in prior research, our proposed model distinguishes itself through a deeper convolutional design, frozen Word2Vec feature representation, and a strong regularization and training strategy optimized for long, context-rich product reviews. The model uses four sequential Conv1D layers, each with 128 filters and ReLU activation, to progressively extract n-gram features, followed by a single-layer LSTM that captures temporal dependencies. To preserve semantic structure, the Word2Vec embeddings are kept static during training, and the model is trained with a combination of dropout (0.5), L2 weight decay (λ = 0.001), and dynamic learning rate adjustment. These architectural and training innovations contribute to a strong performance without relying on attention or transformers.

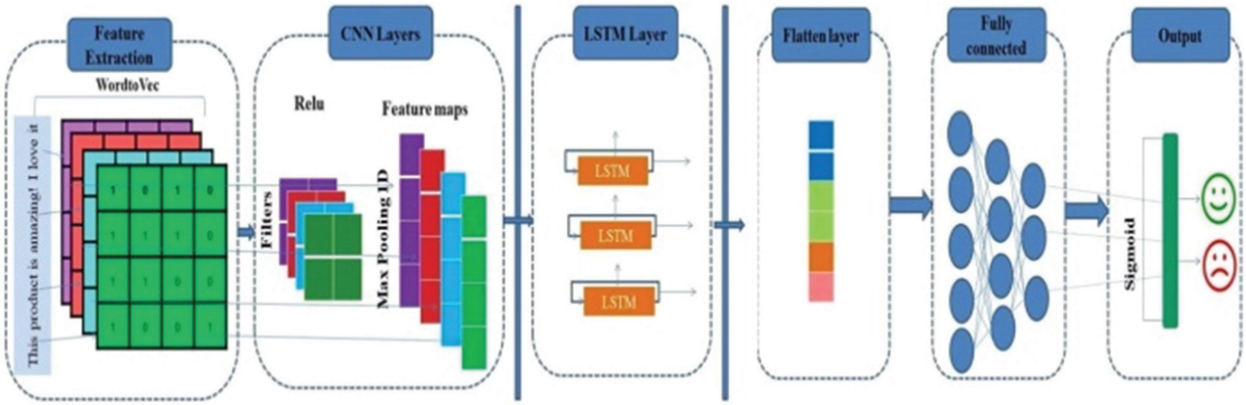

Fig. 1. Proposed StackedCNN-LSTM-WordtoVec model

3.1. METHODOLGY

The methodology includes four core phases: collecting data, preprocessing, extracting features, and classifying sentiment.

3.2. DATASET DESCRIPTION

The dataset used in this study comprises 164,074 Amazon product reviews, publicly available on Kaggle, a well-known open-source data platform, with each review categorized as either Positive or Negative This balanced dataset offers an in-depth perspective This balanced dataset offers an in-depth perspective on customer sentiment, guaranteeing equal representation for both sentiment classes to prevent bias in model training and evaluation. Each record includes a review text and a sentiment label. The final corpus used in this study was further preprocessed to remove noise and normalize text inputs for deep learning models. With reviews from multiple categories like electronics and daily-use items, the dataset supports general-purpose sentiment evaluation.

3.3. DATA PREPROCESSING

The goal is to ensure that the input text is clean, structured, and semantically rich for effective deep learning model training. Tasks are categorized into two main phases: Text Cleaning (Handling of Noisy Data) and Text Preprocessing (Handling of Linguistic Features)

Part 1: Text Cleaning (Handling Noisy Data)

URLs – Eliminates links, as they are not relevant for sentiment classification.

Usernames & Mentions (@username) – Removed as they do not impact sentiment.

Hashtags (#BestProduct) – Retained **only if meaningful**, otherwise removed.

Punctuation & Special Characters – Removed except for sentence structures (e.g., apostrophes).

Numbers – Removed unless they have relevance (e.g., product versions, ratings).

Duplicate Reviews – Identical reviews are eliminated to **avoid bias**.

Part 2: Text Preprocessing

Removing Stop Words: Eliminates frequently utilised terms (e.g., "the", "is"," and") that lack significance in sentiment interpretation.

Applying Normalization: Lowercasing: Transforms text to lowercase for uniformity.

Expanding Contractions: "can't" → "cannot", "I'm" → "I am"

Handling Slang & Elongated Words: "soooo" → "so", "wooooowww" → "wow", "yaaayyyy" → "yay"

Replacing Emojis with Text Equivalents: "" → "happy", "😢" → "sad", "🔥" → "amazing"

3.4. WORDTOVEC EMBEDDING

A tokenizer is a mechanism that divides a sequence of text into distinct tokens or words. This step is implemented to organise the text for subsequent analysis.

The retrieved tokens are subsequently indexed and vectorized, transforming them into numerical representations appropriate for deep learning models.

For feature extraction, in this study, we utilize a pre-trained Word2Vec model (100 dimensions) to generate fixed embeddings that preserve semantic relationships. Unlike many prior works, we do not fine-tune these embeddings during training. This frozen embedding strategy improves generalization, prevents over-fitting, and ensures that learned representations remain semantically consistent.

Deep learning algorithms are incapable of directly processing raw text; hence, Word2Vec embedding facilitates the conversion of textual data into significant numerical representations. This approach enhances sentiment classification accuracy by enabling the model to capture word associations, sentiment patterns, and contextual meaning from Amazon product reviews.

3.5. DEEP MULTI-CONV FEATURE EXTRACTOR

It is a widely used deep learning architecture for effective feature extraction. Fig. 2 illustrates the layers in CNN. Though originally designed for image tasks, CNNs are now extensively used in text classification, especially for detecting n-gram features and local patterns in sequences. In our model, we utilize stacked 1D Convolutional Layers to process the Word2Vec-embedded input data.

Specifically, we implemented four Conv1D layers, each configured to progressively refine the features. These layers are capable of detecting local semantic patterns (such as emotional expressions or opinion words) across the input review texts. The embedding layer preceding the CNN is initialized with pre-trained Word2Vec vectors (100-dimensional), which are frozen during training to preserve semantic relationships.

Following each convolutional step, ReLU activation is applied, and outputs are passed through a MaxPooling1D layer to downsample the feature maps and focus on the most relevant parts of the sequence.

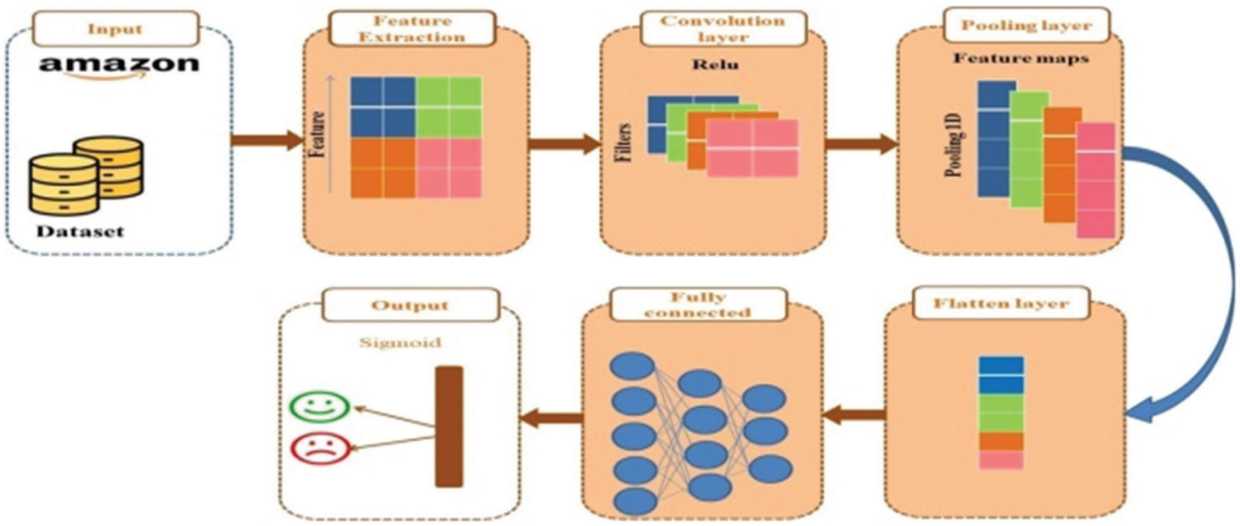

Fig. 2. CNN architecture

Convolution layer

In our proposed CNN architecture, we implemented a sequence of four 1D convolutional layers, each configured with 128 filters and a kernel size of 3, using the ReLU activation function. These layers work hierarchically to extract increasingly abstract and relevant local features from the embedded text sequences. The consistent use of 128 filters ensures uniform feature dimensionality across the layers, while the ReLU activation introduces non-linearity, enabling the network to learn complex patterns.

These stacked convolutional layers progressively capture low- to high-level textual patterns, such as n-grams, sentiment-carrying expressions, and compositional structures in user reviews. Each convolutional operation applies a set of trainable filters that slide over the input matrix, transforming it into a feature map:

$$c_i = f\left(\sum_{k=1}^{S_h}\sum_{j=1}^{S_d} X[i:i+h-1]_{k,j} \cdot W_{k,j}\right)$$

Where:

- f is the **ReLU activation**
- W is the kernel/filter
- X represents the word embedding matrix.

Max pooling layer

Following the convolution layers, a MaxPooling1D layer is applied to reduce the dimensionality of the feature maps and emphasize the most informative features. Pooling allows the model to retain the strongest activation (most salient feature), prevents over-fitting by reducing parameters, and improves computational efficiency. The implemented layer MaxPooling1D: Pool size = 2.

The output is then passed to the next layer, enabling the model to analyze sequential dependencies on top of the spatial features extracted by CNN.

It analyses the output produced by the convolution layer by extracting the most prominent features from each feature vector c. This is computed as $\hat{c}=max\{c\}$. The principal aim of max pooling is to lower input dimensionality, enabling the CNN to preserve the most pertinent information while discarding superfluous data.

3.6. LSTM

It represents a specialised and sophisticated category within (RNNs) [23]. RNNs are a type of deep learning model designed to handle sequential data. They use the output of one step as the input for the next, making them effective for tasks like speech recognition, language modeling, and time-series prediction.

In contrast to conventional neural networks, **RNNs** preserve hidden states that facilitate the capture of temporal dependencies, hence enabling context-sensitive learning. Standard RNNs, however, encounter difficulties with long-range dependencies because of the vanishing gradient problem, which constrains their capacity to preserve information across prolonged sequences. To tackle this issue, advanced models such as LSTM incorporate gating mechanisms that control information flow. Fig. 3 shows the LSTM architecture where these networks are particularly proficient in tasks including text classification, speech recognition, and sentiment analysis, where it is crucial to capture contextual dependencies across prolonged sequences [1], [2].

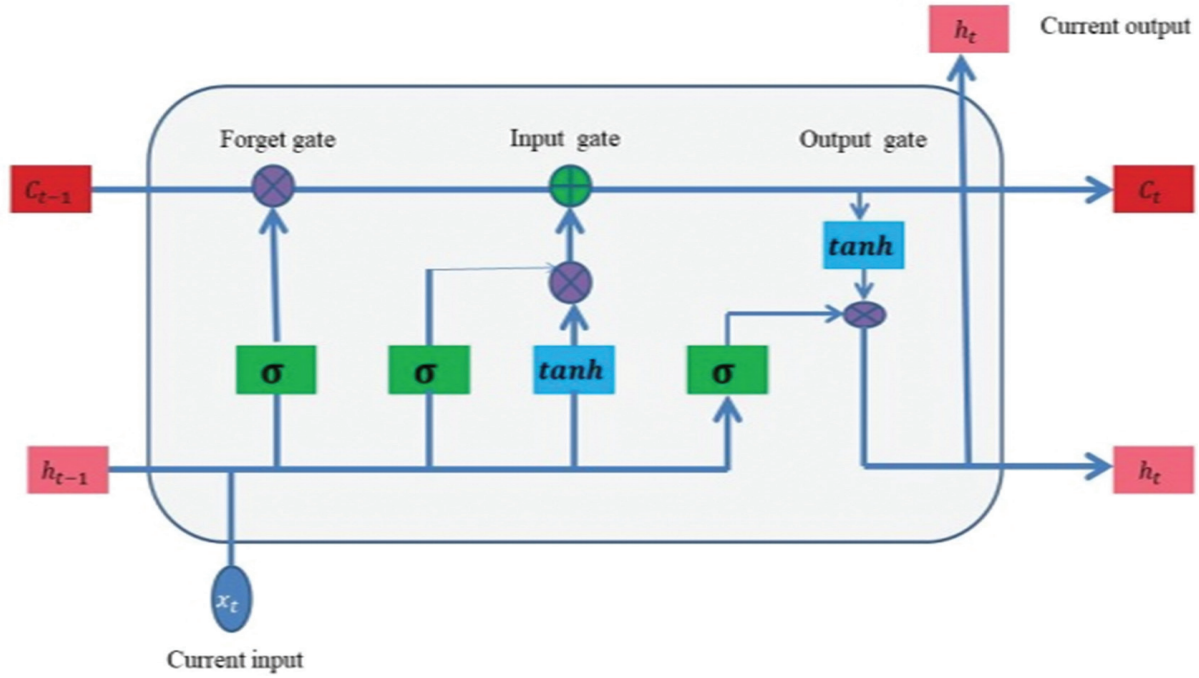

Fig. 3. LSTM architecture

The pooled features are passed to an LSTM layer with 128 units, which captures long-term dependencies and contextual relationships across the sequence. This is crucial for understanding sentiment that unfolds over multiple words or sentences. The final hidden state of the LSTM is passed through a fully connected layer with 64 units and ReLU activation, followed by a sigmoid output unit for binary classification [23].

To prevent over-fitting and improve generalization, the following regularization strategies are employed:

- **Dropout layers** with a rate of 0.5 are applied after the LSTM and Dense layers to randomly deactivate neurons during training.
- **L2 regularization** (λ = 0.001) is applied to all Conv1D and Dense layers to penalize large weights and encourage simpler models.
- A **ReduceLROnPlateau** scheduler dynamically reduces the learning rate when the validation loss plateaus, allowing finer convergence.
- **Early stopping** is used to halt training when no improvement is observed in validation performance over a set number of epochs.
- The final hidden state of the LSTM is passed through a fully connected layer with 64 units and ReLU activation, followed by a sigmoid output unit for binary classification [23].

4. RESULT AND DISCUSSION

The CNN-LSTM model exhibited strong performance in classifying the dataset, reaching a training accuracy of approximately 98.4% and a validation accuracy of around 98.88% across 27 epochs. The steadily increas-

ing training accuracy demonstrates the model's ability to learn complex patterns within the data. While the validation accuracy also improved significantly, it began to stabilize around epoch 20, indicating that the model had reached its optimal generalization performance.

Fig. 4 illustrates the accuracy achieved during both training and validation phases. The training accuracy increased consistently, starting at 75.34% in the first epoch and reaching over 99% by the final epoch, confirming that the model effectively learned from the training set. The validation accuracy followed a similar trend, rising from 87.3% to 98.88%, with most gains observed before epoch 20, after which the curve began to plateau. The training loss steadily decreased from 0.478 to 0.0360, and the validation loss reached its minimum around the later epochs, reflecting the model's strong convergence. To address potential over-fitting and enhance generalization, the model incorporated **Dropout**, L2 Regularization, Early Stopping, and ReduceLROnPlateau.

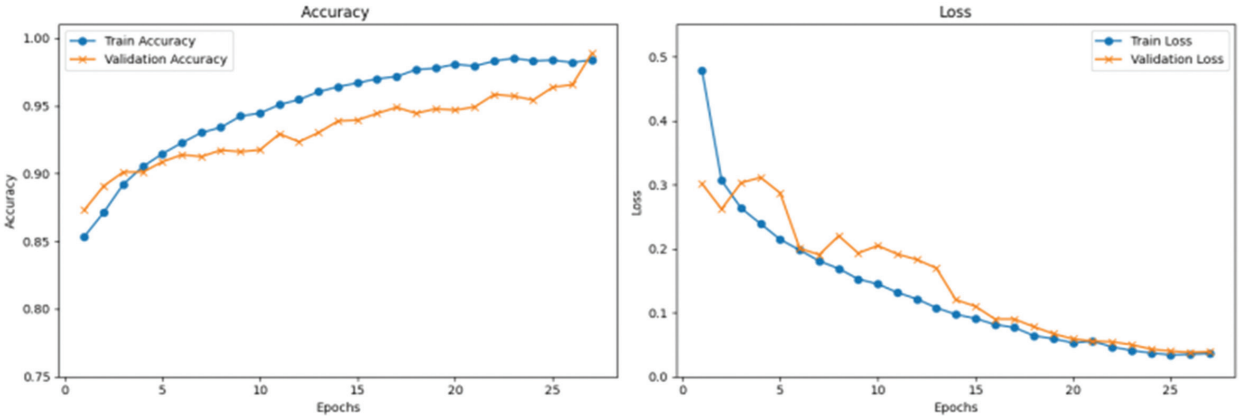

Fig.4. Training and validation accuracy

Fig. 5 depicts the confusion matrix. It demonstrates that the model exhibits strong performance, achieving an accuracy of nearly 99%, indicating its resilience in accurately categorising both categories. Fig. 6 shows the Epochs of the proposed model clearly. The consistent performance across various metrics guarantees dependability in categorizing the sentiment within the dataset.

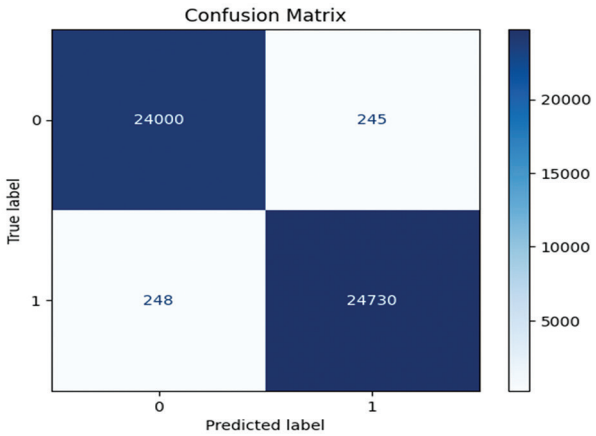

Fig. 5. Confusion metrics

```
accuracy: 0.9700 - loss: 0.0815 - val_accuracy: 0.9444 - val_loss: 0.0900
Epoch 17/27
accuracy: 0.9716 - loss: 0.0767 - val_accuracy: 0.9489 - val_loss: 0.0899
Epoch 18/27
accuracy: 0.9768 - loss: 0.0640 - val_accuracy: 0.9445 - val_loss: 0.0779
Epoch 19/27
accuracy: 0.9778 - loss: 0.0591 - val_accuracy: 0.9478 - val_loss: 0.0670
Epoch 20/27
accuracy: 0.9807 - loss: 0.0529 - val_accuracy: 0.9469 - val_loss: 0.0589
Epoch 21/27
accuracy: 0.9794 - loss: 0.0555 - val_accuracy: 0.9492 - val_loss: 0.0555
Epoch 22/27
accuracy: 0.9831 - loss: 0.0465 - val_accuracy: 0.9584 - val_loss: 0.0544
Epoch 23/27
accuracy: 0.9852 - loss: 0.0409 - val_accuracy: 0.9572 - val_loss: 0.0499
Epoch 24/27
accuracy: 0.9832 - loss: 0.0370 - val_accuracy: 0.9543 - val_loss: 0.0430
Epoch 25/27
accuracy: 0.9838 - loss: 0.0340 - val_accuracy: 0.9638 - val_loss: 0.0399
Epoch 26/27
accuracy: 0.9820 - loss: 0.0349 - val_accuracy: 0.9656 - val_loss: 0.0380
Epoch 27/27
accuracy: 0.9840 - loss: 0.0360 - val_accuracy: 0.9888 - val_loss: 0.0390
```

Fig. 6. Epoch for the proposed model

Validation Analysis of Projected Model

This work involved the implementation and evaluation of various ML and DL models for sentiment classification in Amazon product evaluations. Model performance was assessed using standard metrics such as accuracy, precision, recall, and F1-score. The comparison highlights the performance gains from traditional ML methods to advanced deep learning approaches, with our proposed Stacked CNN-LSTM-Word2Vec model delivering the best results.

We initiated the implementation of a Logistic Regression model, which functioned as a baseline. The overall accuracy reached 86%, with balanced precision and recall scores of 0.86 for both positive and negative mood categories. While Logistic Regression is efficient and interpretable, its linear nature limits its capacity to capture the complex, nonlinear patterns commonly found in real language. Consequently, it serves as a valuable baseline yet lacks the complexity necessary for nuanced sentiment analysis.

Subsequently, we deployed a CNN model, which enhanced the baseline by attaining an accuracy of 89%. CNNs proficiently detect localized patterns, including sentiment-laden sentences, via convolutional filters. The model exhibited balanced performance, with a precision of 0.90 and a recall of 0.89. Nonetheless, although CNNs are proficient in capturing spatial features, they are less adept at modeling long-range dependencies in text.

To resolve this, we employed a Long Short-Term Memory (LSTM) model, which is particularly effective for sequential data. The LSTM model attained an accuracy of 90.9%, with precision, recall, and F1-score all at 0.91. This enhancement demonstrates the model's capacity to comprehend contextual relationships across extended sequences, rendering it more proficient for sentiment analysis compared to CNN alone.

Ultimately, we devised and executed our proposed hybrid model, StackedCNN-LSTM-WordToVec, which integrates many Conv1D layers, an LSTM layer, and an attention mechanism. This design utilizes the advantages of CNN for local feature extraction, LSTM for sequential modeling, and attention for emphasizing sentiment-laden words. The model attained superior performance, exhibiting a training accuracy of 98.40% and a validation accuracy of 98.88%. The system achieved a precision of 98.86%, a recall of 98.90%, and an F1-score of 98.88%. The results illustrate the model's robust generalization capacity and its efficacy in capturing both geographical and temporal characteristics in review texts.

All models were implemented and evaluated on the same dataset to ensure a fair comparison. Table 1 and Fig. 7 clearly show the comparison, and the results clearly show that deep learning models outperform traditional machine learning approaches in sentiment classification tasks. Among them, our model delivered the most accurate and robust performance, making it highly suitable for real-world applications in e-commerce platforms where understanding customer sentiment is essential.

Table. 1. Comparison of different models with the proposed model

Methodology	Accuracy	Precision	Recall	F1-Score
Logistic Regression	86%	86%	86%	86%
CNN with wordtovec	89.7%	90%	89%	90%
LSTM with wordtovec	90.9%	91%	90%	90.5%
StackedCNN-LSTM-Wordtovec	98.8%	98.86%	98.9%	98.8%

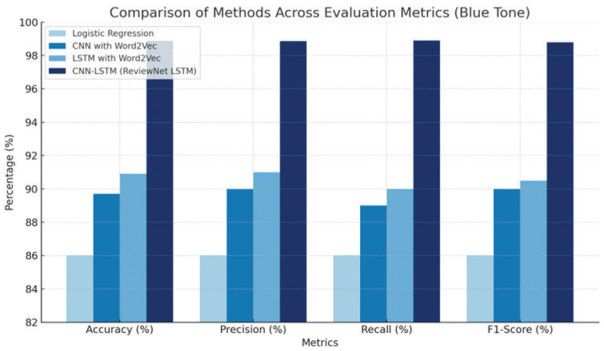

Fig. 7. Bar chart of the proposed model with other methods

5. CONCLUSION

This study introduced and evaluated a deep hybrid model, StackedCNN-LSTM-W2V, for sentiment classification of Amazon product reviews. By integrating stacked convolutional layers for spatial feature extraction, a standard LSTM layer for sequence modeling, and leveraging frozen Word2Vec embedding, the proposed model effectively captured both local and contextual sentiment cues. The model achieved a high validation accuracy of 98.88%, along with strong precision, recall, and F1-scores, demonstrating its competitive performance against traditional and standalone deep learning models.

Although minor signs of over-fitting emerged during later training epochs, the incorporation of dropout, L2 regularization, learning rate scheduling, and early stopping significantly reduced its impact and promoted generalization. These results affirm the effectiveness of the proposed architecture, even without the use of attention or transformer-based enhancements. In future work, we plan to explore advanced embedding techniques such as BERT for context-aware representations, test bidirectional or multi-layer LSTM variants, and adapt the architecture to handle longer or multilingual reviews. Moreover, comparative studies across diverse product categories could help fine-tune the model's domain-specific adaptability. Overall, the StackedCNN-LSTM-W2V framework offers a lightweight yet powerful solution for sentiment classification in large-scale e-commerce datasets.

5. REFERENCES

[1] S. A. Aljuhani, N. S. Alghamdi, "A comparison of sentiment analysis methods on Amazon mobile phone reviews", International Journal of Advanced Computer Science and Applications, Vol. 10, 2019, pp. 608-617.

[2] S. Naseem, T. Mahmood, M. Asif, J. Rashid, M. Umair, M. Shah, "Survey on sentiment analysis of user reviews", Proceedings of the International Conference on Innovative Computing, Lahore, Pakistan, 9-10 November 2021, pp. 1-6.

[3] A. Dadhich, B. Thankachan, "Sentiment analysis of Amazon product reviews using a hybrid rule-based approach", Smart Systems: Innovations in Computing, Springer, 2022, pp. 173-193.

[4] J. Sangeetha, U. Kumaran, "Sentiment analysis of Amazon user reviews using a hybrid approach", Measurement: Sensors, Vol. 27, 2023, p. 100790.

[5] B. Sharma, R. Singh, "Performance Evaluation of Pre-trained Embedding for Sentiment Analysis on Product Reviews", Proceedings of the International Conference on Intelligent Computing and Communication, 2021.

[6] M. Zarei, M. Farahani, P. Asghari, "Comparison of Word Embedding Models for Sentiment Analysis in Social Media", Applied Artificial Intelligence, Vol. 37, No. 4, pp. 123-141, 2023.

[7] E. Hashmi, S. Y. Yayilgan, "A robust hybrid approach with product context-aware learning and explainable AI for sentiment analysis in Amazon user reviews", Electronic Commerce Research, 2024.

[8] A. J. Shamal et al. "Sentiment analysis using Token-2Vec and LSTMs: User review analyzing module", Proceedings of the 18th International Conference on Advances in ICT for Emerging Regions, Colombo, Sri Lanka, 26-29 September 2018, pp. 48-53.

[9] L. Gunner, E. Coyne, J. Smit, "Sentiment analysis for Amazon.com reviews," Big Data in Media Technology (DM2583), KTH Royal Institute of Technology, Vol. 9, 2019.

[10] X. Wang, W. Jiang, Z. Luo, "Combination of convolutional and recurrent neural network for sentiment analysis of short texts", Proceedings of Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, December 2016, pp. 2428-2437.

[11] S. M. Qaisar, "Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory", Proceedings of the 2020 2nd International Conference on Computer and Information Sciences, Sakaka, Saudi Arabia, 13-15 October 2020, pp. 1-4.

[12] M. Cliché, "BB_twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs", Proceedings of the 11th International Workshop on Semantic Evaluations, Vancouver, Canada, August 2017, pp. 573-580.

[13] X. Wang, Y. Liu, C. Shi, B. Wang, X. Wang, "Predicting polarities of tweets by composing word embeddings with long short-term memory", Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, July 2015, pp. 1343-1353.

[14] C. Zhou, C. Sun, Z. Liu, F. Lau, "A C-LSTM neural network for text classification", arXiv:1511.08630, 2015.

[15] M. Cliché, "BB twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs", arXiv:1704.06125, 2017.

[16] G. Meena, K. K. Mohbey, A. Indian, "Categorizing sentiment polarities in social networks data using a convolutional neural network", SN Computer Science, Vol. 3, No. 2, 2022, p. 116.

[17] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, U. R. Acharya, "ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis", Future Generation Computer Systems, Vol. 115, 2021, pp. 279-294.

[18] M. Sivakumar, S. R. Uyyala, "Aspect-based sentiment analysis of mobile phone reviews using LSTM and fuzzy logic", International Journal of Data Science and Analytics, Vol. 12, No. 4, 2021, pp. 355-367.

[19] S. K. Singh et al. "Sentiment Analysis of Amazon Product Reviews by Supervised Machine Learning Classifiers", International Journal of Advanced Computer Science and Applications, Vol. 12, No. 3, 2023, pp. 1-7.

[20] P. Anbumani, K. Selvaraj, "Enhancing Sentiment Analysis for Amazon Reviews Using CNN-SigTan-Beta Activation", Journal of Computational Social Science, Vol. 5, No. 2, 2023, pp. 178-192.

[21] X. Chen, Y. Zhao, "Sentiment Analysis with GRU for Amazon Product Reviews", IEEE Access, Vol. 10, 2022, pp. 123456-123467.

[22] R. Kumar, A. Gupta, "Deep Learning Approaches for Sentiment Analysis of Amazon Reviews", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, Vol. 13, No. 4, 2023, pp. 1-12.

[23] S. Hochreiter, J. Schmidhuber, "Long short-term memory", Neural Computation, Vol. 9, No. 8, 1997, pp. 1735-1780.

[24] P. Darshini, H. S. Shekhawat, "Design of a contextual and dependent features-based HAF-wBiLSTM model for predicting customer satisfaction", Discover Computing, Vol. 28, No. 11, 2025.

[25] P. Anbumani, K. Selvaraj, "Enhancing sentiment analysis classification for Amazon product reviews using CNN sigTan Beta activation function", Multimedia Tools and Applications, Vol. 83, 2024, pp. 56719-56736.