DOI: 10.22034/tjee.2025.67687.5036

Attention-based CNN-BiLSTM Deep Model for Sentiment Analysis of User Opinions in Social Media

Reza Latifi Ardahaei[1], Mohammad Asadpour[1*], Hadi Seyedarabi[1]

[1] Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran
[*] Corresponding Author: m_asadpour@tabrizu.ac.ir

Received: 06/06/2025, Revised 24/10/2025, Accepted: 24/12/2025.

Abstract

Extracting sentiment from textual data is crucial for understanding public opinion and guiding strategic decisions. We introduce a hybrid deep-learning pipeline that combines convolutional feature detectors, bidirectional recurrent units, and a custom attention mechanism. First, convolutional layers with pooling condense local n-gram patterns into compact feature maps. These maps are fed into a bidirectional LSTM network that captures sequence information in both forward and reverse directions. A specialized attention module then assigns relevance scores to individual tokens, sharpening the final classification. Evaluations on widely used sentiment benchmarks show that our method outperforms leading models in terms of accuracy as well as requires fewer computational resources, making it a practical solution for scalable emotion detection in text.

Keywords

Sentiment analysis, Deep learning, CNN, Bi-LSTM, Attention mechanism, Social media.

1. Introduction

Sentiment analysis involves examining a piece of text to judge whether its tone is positive, negative, or neutral and it has become essential for interpreting customer reviews, survey responses, and open-ended feedback streams ([1]). Utilizing advanced language processing algorithms in conjunction with data driven learning models often termed “emotion detection” or “affective computing”researchers analyze sentiment at three granularity levels:

Document-level, which assigns an overall polarity to an entire text;Sentence-level, which evaluates each sentence independently;Aspect-level, which isolates opinions about specific attributes (e.g., product quality or service efficiency) ([2], [3]). Organizations across sectors leverage these insights to monitor brand health, engage customers more effectively, and inform strategic decisions. In e-commerce, for instance, aggregating large-scale feedback enables dynamic pricing adjustments and guides iterative improvements in product design. In healthcare, mining patient comments and social discussions aids in early detection of public health trends, adverse drug reactions, and emotional well-being. Government and policy makers also employ sentiment analysis to gauge public reaction to legislation and social initiatives ([4]).

However, real-world text is often noisy: social media posts abound with misspellings, slang, emojis, and hyperlinks, all of which complicate automatic classification. Robust pre-processing such as whitespace normalization, stop-word removal, stemming or lemmatization, and part-of-speech tagging is therefore critical before model training ([5]). Converting text into numerical form traditionally relies on methods like TF-IDF or one-hot encoding, but these fail to capture semantic relationships. Modern approaches instead use embedding layers (e.g., Word2Vec or GloVe) to produce dense vector representations that preserve contextual meaning [6]. Deep learning architectures have further advanced the field: CNNs excel at extracting local n-gram features, while RNNs—particularly LSTM and GRU are designed to model sequential dependencies and mitigate vanishing-gradient issues through gating mechanisms ([7], [8]). Bidirectional LSTMs extend this capability by processing sequences in both forward and backward directions, thereby enriching contextual understanding ([9]). Introducing attention layers enables models to assign varying importance to critical tokens on the fly, sharpening their focus on words that carry sentiment ([10]). More recently, models employing the transformer architecture, notably BERT and XLNet, which rely solely on attention layers, have set new performance benchmarks without the gradient-stability problems of earlier RNN architectures ([11]).

2. Related Works

Deep learning models, including LSTM, GRU, Bi-LSTM, and CNN, are widely employed for sentiment classification tasks ([12]). ([13]) They enhanced a conventional LSTM based sentiment classifier by embedding a sentiment lexicon, thereby augmenting word representations with specialized emotional information. An additional attention layer was incorporated to aggregate insights from the entire text sequence, rather than restricting focus to individual tokens. ([14]) The proposed ABCDM framework merges convolutional and recurrent components into a bidirectional, attention-enhanced architecture, using separate bidirectional LSTM and GRU modules to extract contextual information from both earlier and later portions of the text. In another effort, composite models integrating LSTM units, convolutional networks, and support vector machines were evaluated on eight diverse tweet and review corpora to assess performance across multiple domains. ([15]) In recent sentiment analysis research, distributed word representations are commonly employed; Yet, these approaches usually concentrate on encoding word meanings and fail to account for their emotional connotations. To overcome this limitation, an enhanced word representation technique is proposed, It augments the classic TF-IDF method with sentiment cues to create weighted word embeddings. These sentiment-aware vectors are then passed through a Bi-LSTM network, enhancing its ability to model contextual dependencies and yielding richer comment representations ([16]). For advanced text preprocessing, pre-trained Word2Vec embeddings are used to transform words into dense vector representations. These vectors are then input into a deep convolutional neural network, which efficiently extracts relevant features aligned with user-specific patterns or preferences ([17]). Recurrent neural networks excel at modeling temporal relationships within sequence data but are not well suited for parallel extraction of localized patterns. Additionally, Training often encounters problems where gradients either diminish to near zero or escalate without bound , which undermines their capacity to learn long-range dependencies effectively ([16]). Lightweight models for resource-constrained environments have been proposed to address computational challenges, improving efficiency [18]. Recurrent networks often fail to learn long-term patterns in sequences due to vanishing or exploding gradients during training. To address this, bidirectional LSTM models process data in both forward and reverse directions, allowing the system to draw on context from earlier and later time steps and thereby improve its understanding of sequential information ([19]). GloVe-DCNN combines pre-trained GloVe embeddings with a deep convolutional network to perform sentiment detection on Twitter. Tweets are labeled as positive or negative, and benchmark tests confirm that this approach delivers both high accuracy and faster inference times ([20]). This work integrates a comparison ‐ driven module into a BiLSTM network enhanced with multi-head attention, streamlining the computation of feature interactions. Across three benchmark sentiment corpora, this architecture consistently achieves higher accuracy than a wide range of earlier models ([21]). They present the ACL-SA framework, which integrates convolutional feature extraction and recurrent sequence modeling with an attention layer. Initially, data undergoes cleaning and transformation using TF-IDF and pre-trained GloVe vectors to form robust feature inputs. Convolutional layers followed by max-pooling distill local context and reduce dimensionality, while a bidirectional LSTM captures extended dependencies. An attention module then highlights salient tokens from the convolutional outputs, and the inclusion of Gaussian noise along with dropout regularizes the network to mitigate overfitting ([23]). A hybrid convolutional–recurrent framework was developed to jointly capture local feature patterns and long-range dependencies, with its complementary structure helping to alleviate overfitting. The authors also stress that the fidelity of embedding representations is pivotal: off-the-shelf Word2Vec or GloVe vectors, if used without further adaptation, can sometimes undermine sentiment classification accuracy ([24]). ([25]) They introduced the ARC framework, integrating recurrent and convolutional layers with an attention mechanism to process tweets and reviews. A single bidirectional GRU layer first encodes sequence information, and its output is then passed through convolutional filters to extract both local n-gram cues and broader textual patterns for sentiment categorization. ([26] , [6]) They devised a sequential framework that blends bidirectional gated recurrent units with convolutional layers to perform sentiment analysis. By uniting GRU-driven sequence modeling and CNN-based feature extraction, the architecture captures both fine-grained textual patterns and overarching contextual relationships. ([27]) They introduced a hybrid architecture combining recurrent layers with a convolutional attention module. First, CNN layers distill high-level representations from input sentences. An attention layer then computes relevance scores over these features, emphasizing those most critical for downstream prediction. ([28]) They developed the SAMF-BiLSTM framework, which enriches BiLSTM networks incorporating a self-attention mechanism and multiple input channels. Initially, the system learns sentence-level embeddings, then applies Bi-LSTM to aggregate these into document-wide sentiment vectors. Evaluation results indicate that SAMF-BiLSTM surpasses state-of-the-art techniques in accuracy. Additionally, the SR-LSTM architecture featuring two stacked hidden layers was introduced to better handle long-form text and inter-sentence semantics, offering improved performance on document-level sentiment tasks compared to conventional models that falter over extended sequences ([29]). ([30]) They proposed the CRAN framework, which fuses recurrent processing with convolutional attention modules arranged in a multi-level hierarchy for sentiment categorization. In parallel, the field has seen a surge of machine learning–driven strategies that consistently boost text sentiment accuracy, as a wide array of innovative architectures continues to emerge and enhance classification performance. ([31]) They introduced a composite architecture that unites a deep convolutional network with long short-term memory units. The CNN segment employs multiple layers of convolutions and max-pooling to distill hierarchical feature representations, while the LSTM component models extended sequential dependencies. By fusing these capabilities, the hybrid framework captures both fine-grained patterns and long-range context, leading to enhanced sentiment classification accuracy. ([32]) Researchers have developed RU-BiLSTM, a recurrent network that combines BiLSTM layers, word embedding, and an attention module to perform sentiment analysis on Roman Urdu. ([33]) They developed a convolutional LSTM pipeline for sentiment detection, initially validated on the 50,000-review IMDb corpus. To counteract label imbalance, the approach uses data augmentation driven by a pretrained RoBERTa model. Additionally, the system integrates Facebook AI Research's fastText classifier and applies the combined framework to analyze US airline–related tweets, demonstrating its capacity to extract actionable opinions from social media ([34]). The Wilson Cowan framework, a neural mass network model tailored for metapopulation dynamics, represents subcortical brain areas as interacting nodes within a network. The links between these nodes reflect different types of neural connectivity including structural, functional, and effective pathways capturing the communication patterns among distinct brain regions. ([35]).

3. Proposed model

The proposed model employs a deep learning approach to classify reviews into positive and negative categories. It integrates data preprocessing, word representation, and a hybrid CNN-Bi-LSTM network. The preprocessing step enhances data quality by correcting errors and eliminating noise. A CNN augmented with max-pooling layers is employed to distill key features and compress their dimensionality. The Bi-LSTM layer captures long-term dependencies, while an attention mechanism highlights the importance of individual words in the output. Experimental evaluation using movie and airline review datasets demonstrates that This approach demonstrates superior performance compared to other deep learning methods, offering advantages in both accuracy and computational efficiency. An overview of the proposed sentiment analysis framework is illustrated in Figure. 1.

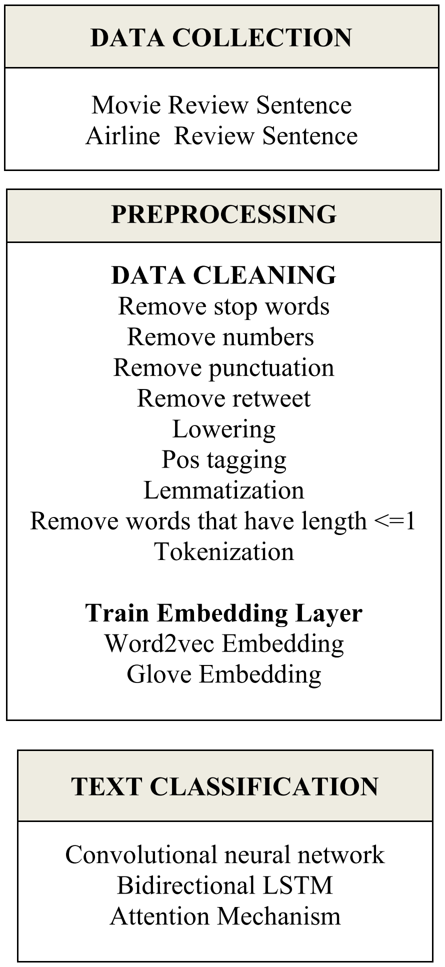

Fig 1. Proposed Model

3.1. Data Preprocessing

Data preprocessing involves converting raw, unorganized data into a consistent, dependable format ready for analysis. Social media datasets, being large and varied, often include noise, inconsistencies, and incomplete or missing data due to the diverse sources and contributors. These challenges are tackled by employing a suite of preprocessing steps lemmatizing words, stripping out punctuation, filtering stop words, segmenting text into tokens, and assigning part-of-speech labels . These steps ensure the text is clean and ready for model training ([1]) , ([5]).

3.2. Train Embedding Layer

A variety of machine learning and deep learning strategies encode vast volumes of textual data into numerical formats. Word embedding methods assign words to vectors based on a dictionary. Classical approaches, such as Bag of Words (BOW) and TF-IDF, emphasize word frequency but don't capture the syntactic or semantic connections between words. In contrast, methods like word2vec and GloVe create word embeddings, which more accurately reflect the semantic relationships and subtleties of word meanings. These models are trained on vast text datasets in an unsupervised manner and excel in learning the meaning of words.

Word2Vec crafts continuous, distributed word embeddings through two primary frameworks : Continuous Bag of Words and Skip-Gram each implemented as a straightforward neural network with a single hidden layer. Word2Vec learns word vectors by processing a text corpus, constructing a vocabulary, and refining the vectors through backpropagation and stochastic gradient descent. Skip-Gram takes a single focus word and uses it to predict its neighboring words, whereas CBOW does the reverse using the surrounding words to infer the central word. Both architectures use a three-layer feedforward neural network to generate word vectors ([1]), ([7]) . The architectural diagram for word embedding model (CBOW and skip-gram) is presented in Figure. 2.

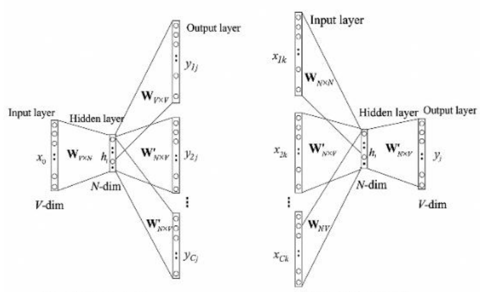

Fig. 2. CBOW & Skip-gram Model

3.3. Convolutional neural network

CNN are layered, feed-forward models originally created for image recognition that demand only basic data preparation. When applied to NLP, they treat word or token embeddings as a 1D signal, using sliding convolutional filters to automatically extract local n-gram patterns and key features from text ([1]).

3.3.1. Convolution Layer

The "convolution" in CNN refers to its core operation of applying learnable kernels over the input to reveal salient patterns. In image tasks, these kernels slide across small patches to capture edges and textures while maintaining spatial structure. Analogously in text classification, one-dimensional convolutional filters move over embedding sequences to build a feature map that summarizes meaningful word or phrase combinations, enabling efficient downstream analysis.

3.3.2. Pooling Layer

When processing high-dimensional inputs, reducing feature size while retaining important patterns is essential to limit the number of learnable parameters. This is typically done by incorporating pooling layers between convolutional stages. Pooling also known as downsampling shrinks the spatial resolution of feature maps to create a more compact and informative representation. Common techniques include max pooling, which selects the highest activation in a given window; average pooling, which computes the mean; and sum pooling, which totals the values. The method adopted in this study applies max pooling to emphasize the most significant features in each region.

3.3.3. Fully Connected Layer

The fully connected layer often implemented as a multi-layer perceptron serves as the final stage in the network, where a softmax function is applied to produce probability distributions over target classes. Here, each neuron links to every unit in the next layer, allowing for a full integration of the extracted features. The preceding convolutional and pooling operations encode high-level abstractions, which the dense layer leverages to perform classification based on learned representations from training.

3.4. Recurrent neural network

RNNs are built to model sequences by incorporating memory of previous states, in contrast to feed-forward networks that process inputs independently. While standard RNNs can learn short-range patterns, they have difficulty preserving information across long sequences, a challenge referred to as the long-term dependency issue. To address this, Long LSTM units were introduced, offering specialized gating mechanisms that enable the network to maintain and update long-term contextual information more effectively ([32]).

3.5. Long short-term memory (LSTM)

LSTM architectures are a refined RNN variant, purpose-built to counteract the vanishing and exploding gradient issues endemic to traditional recurrent models. At every timestep, the LSTM fuses the current input with its previous hidden state to produce an output, which it then forwards through the sequence just like a standard RNN. An LSTM unit comprises a cell state c_t, that preserves information over arbitrary durations, plus three nonlinear gates input i_t, a forget gate f_t, an output gate o_t. which control the cell's information intake and release (see Fig. 3(a)) ([18]).

3.6 Gated recurrent unit (GRU)

A GRU employs two gates an update gate (z_t) and the reset gates (r_t) to jointly regulate state updates. The reset gate r_t determines how much of the previous hidden state h_{t-1} contributes to the candidate activation $\tilde{h}_t$; lower values of r_t lead to greater omission of past information. (see Fig. 3(b)) , ([8]).

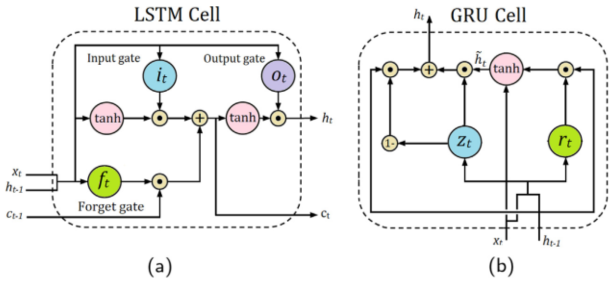

Fig. 3. LSTM vs. GRU: (a) LSTM architecture, (b) GRU architecture

3.7. Bidirectional LSTM

Bi-LSTM extends the capabilities of standard LSTM networks by processing input sequences in both forward and backward directions. By processing sequences in both directions, this method strengthens the model's contextual understanding of preceding and subsequent tokens, leading to superior sequential text analysis versus traditional LSTMs. Fig.4 shows the Bi-LSTM architecture [9].

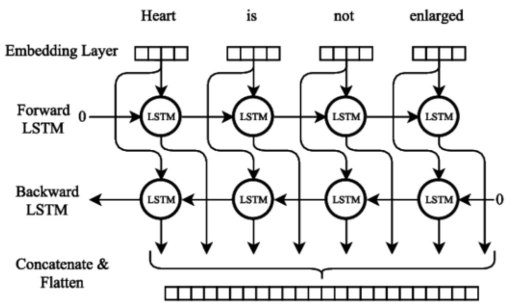

Fig. 4. Architecture of a Bi-LSTM

3.8. Attention-Based Bidirectional Network

Conventional LSTM networks are constrained by their unidirectional nature, relying solely on prior time steps for context. Conversely, our Bi-LSTM framework integrates an attention layer to harness context from both earlier and later tokens. This is accomplished by employing three independent bidirectional hidden layers, whose outputs are merged and subsequently processed through an attention layer to refine the final representation. Our proposed The bidirectional architecture is built from two hidden LSTM layers: a forward layer L_f and L_b, respectively. The BiLSTM module processes word embeddings bidirectionally, extracting full-spectrum contextual information. An attention mechanism is applied to emphasize tokens that carry stronger sentiment relevance, selectively weighting significant features while minimizing the influence of less informative ones. The enhanced feature representations are routed into a dense softmax layer to deliver the final classification results.

Attention Mechanism

Each hidden state h_t from the LSTM layers is fed into the attention module, which allocates unique importance scores to different tokens. The context vector is then computed as a weighted sum of all these hidden states (see Fig. 5) .

$$u_t = \tanh(W \times h_t + \mathrm{b}) \tag{1}$$

$$a_t = \frac{\exp(u_t^T u_w)}{\sum t \exp(u_t^T u_w)} \tag{2}$$

$$c = \sum_t^T a_t h_t \tag{3}$$

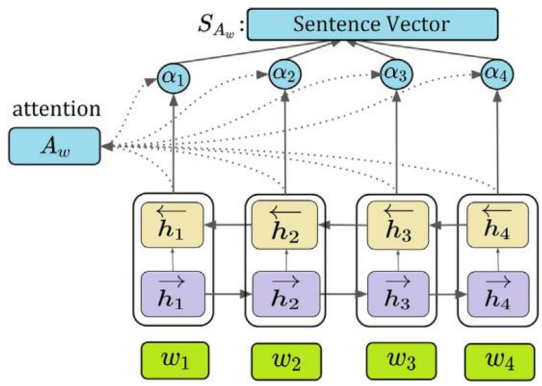

Fig. 5. The bidirectional model's attention module

Let u_t denote the hidden representation derived from state h_t and let u_w be a context vector that is randomly initialized and then optimized during training. To gauge the relevance of each word, we compute the similarity between u_t and u_w. For an input sequence of length T, this yields a raw score at each time step t, which is then normalized into an attention weight a_t. Finally, we form the context vector c by taking the weighted sum of the hidden states using these attention weights, so that *c* succinctly captures the aggregated information from all words in the review.

3.9. Running-Time Scalability Study

Time complexity quantifies how long a procedure takes to run on a given input. A model with low time complexity will require less time to train and make predictions. A running time record of proposed method and ACL-SA ([22]) model on US-Airline Twitter dataset under the same circumstance in colab with GPU runtime are listed in Table 1. Empirical results demonstrate that our model enhances sentiment analysis effectiveness without extending training duration.

Table 1. Epoch-wise training duration on the US-Airline Twitter dataset

Dataset	Models	Accuracy	Training-time(sec)
US-Airline Twitter dataset	Attention-CNN-BiLSTM	**97.65%**	**71.27**
ACL-SA ([22])	CNN-Attention -BiLSTM	94.01%	82.98

4. Experiments and Analysis

This section outlines the experimental setup and presents an in-depth evaluation of the obtained results. The proposed model is tested on two datasets: US-Airline for short text analysis and IMDB for long-form content. The primary objective is to rigorously measure the model's effectiveness. In the final stage, the performance of our approach is benchmarked against existing methods to validate its predictive strength.

4.1. Datasets

This study employs two benchmark datasets: US-Airline for short-text sentiment and IMDB for long-text analysis. The US-Airline dataset, available on Kaggle, includes 14,641 tweets from 2015 addressing airline-related issues, labeled as positive, negative, or neutral. After filtering out neutral entries, 11,541 tweets were retained for this work. The IMDB dataset features 50,000 movie reviews, evenly divided into 25,000 positive and 25,000 negative samples. Detailed statistics of the dataset are listed in Table 2.

Table 2. Comprehensive dataset metrics

Dataset	Positive	Negative	Total
US Airline	2363	9178	11,541
IMDB	25,000	25,000	50,000

4.2. Experimental Setup

The input layer is primed by merging embeddings from both Word2Vec and GloVe, each with 200-dimensional vectors, alongside other learnable parameters during training. The CNN block consists of three parallel channels with 128, 64, and 32 filters, each employing a kernel size of 3 and ReLU activation. Outputs from these convolutional layers are downsampled via max pooling with a window size of 2. The architecture stacks three Bi-LSTM layers of 128 units each, then transitions to a 128-unit fully connected layer with ReLU activation. Attention mechanisms are applied to the outputs from each Bi-LSTM layer to emphasize relevant features, which are then merged through a concatenation layer. A Sigmoid function is applied in the output layer to facilitate binary classification. Training is performed by minimizing binary cross-entropy loss, with weight updates handled by the Adam optimizer. Experiments were carried out on a Windows 10 workstation powered by a 2.27 GHz Intel Core i5 processor and 4 GB of memory, with hyperparameter details listed in Table 3. The hyperparameters were optimized using both grid search and manual tuning.

Tables 3. Configuration of hyperparameters employed during training of the proposed model

Parameters	Values
Epoch count	[1-10]
Batch size	32
Embedding size	200
Kernel size	3
Filter	32,64,128
Pool size	2
Number of CNN layers	3
Number of Pooling	3
BiLSTM layer output	128
Number of Dropout	3
Dropout rate	0.1
Optimization step size	0.0001
Optimizer	Adam, RMSprop
Activation function type	ReLu
Loss function	Cross-entropy

4.3. Performance evaluation parameters

The proposed model's effectiveness is assessed through conventional classification measures accuracy, precision, recall, and F1-score each calculated from its confusion matrix. Moreover, the classifier's performance is further examined using the Receiver Operating Characteristic curve and its corresponding area under the curve. These evaluation criteria facilitate a comprehensive analysis and comparison of the model's predictive capabilities.

i. **Precision**: It is defined the ratio of true positive prediction to the total number of positive prediction. It measures the exactness of the classifier. It can be expressed as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4}$$

ii. **Recall**: It is defined as the ratio between the number of true positive prediction to the total number of actual positive sample. It is also known as sensitivity.

$$\text{Recall} = \frac{\text{TP}}{\text{TP +FN}} \tag{5}$$

iii. **F-measure**: It is the harmonic mean of Precision and Recall.

$$\text{F1} = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision+Recall}} \tag{6}$$

iv. **Accuracy**: It is defined as the fraction of samples that are predicted correctly.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+}FP\text{+}FN} \tag{7}$$

```
Classification Report:
              precision    recall  f1-score   support

    negative       0.99      0.97      0.98      1839
    positive       0.97      0.99      0.98      1833

    accuracy                           0.98      3672
   macro avg       0.98      0.98      0.98      3672
weighted avg       0.98      0.98      0.98      3672
```

Fig. 6. Precision, Recall, and F1-Score metrics using the CNN-BiLSTM algorithm with an attention mechanism and word embedding based on GloVe on the US-Airline dataset.

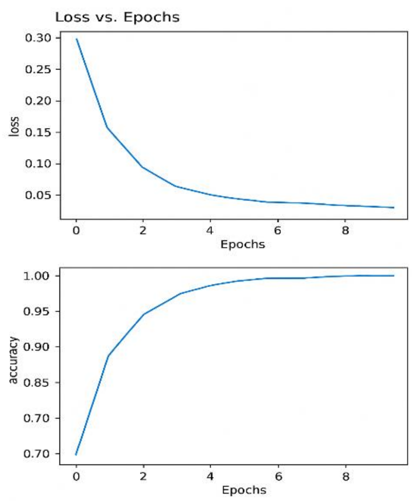

Fig. 7. Loss and accuracy charts using the CNN-BiLSTM algorithm with an attention mechanism and word embedding based on GloVe on the US-Airline dataset.

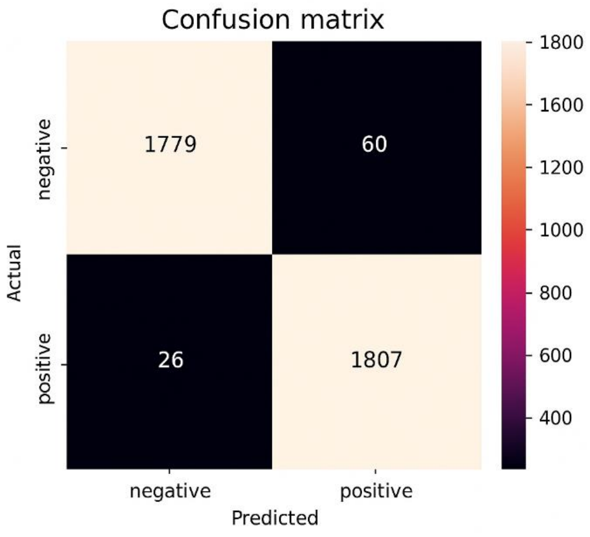

Fig. 8. Confusion matrix using the CNN-BiLSTM algorithm with an attention mechanism and word embedding based on GloVe on the US-Airline dataset**.**

Tables 4. Comparison of the time complexity of the proposed model with the model used in **Kamyab et al. (2021)** under identical parameters on the **US-Airline Twitter** dataset.

Feature	Attention-CNN-BiLSTM (Proposed Model)	ACL-SA (Kamyab et al., 2021)
Dataset	US-Airline Twitter dataset	US-Airline Twitter dataset
Accuracy	**97.65%**	94.01%
Training Time	**71.27 s**	82.98 s
Epoch	10	10
Embedding	GloVe = 200	GloVe = 200
Optimizer	RMSProp	RMSProp
Kernel Regularization	L2 (0.0001)	L2 (0.0001)

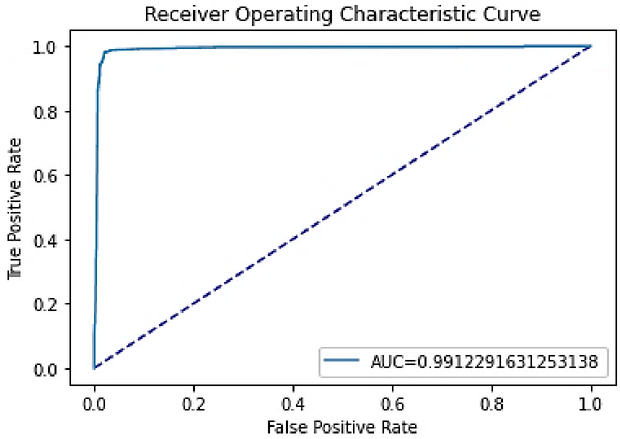

Fig. 9. ROC curve using the CNN-BiLSTM algorithm with an attention mechanism and word embedding based on GloVe on the US-Airline dataset.

4.4. Baseline methods

This study evaluates the proposed model using two benchmark datasets and compares its performance against several recently developed sentiment classification models. Below, we outline a selection of the most relevant and up-to-date baseline methods used for comparative analysis. ABCDM ([15]): The model employs an attention-driven architecture that integrates a bidirectional CNN-RNN framework. It utilizes two separate bidirectional recurrent layers to extract surrounding context by looking at both prior and subsequent tokens. Subsequently, a CNN layer is applied to the combined RNN outputs to compress the feature space and enhance representation efficiency.

ACL-SA ([22]): This architecture leverages CNN to identify local contextual patterns and employs Bi-LSTM to learn long-range dependencies within the text. An attention layer is applied over the CNN output to highlight the most informative features. For representing words, the model utilizes GloVe embeddings that have been pre-trained on large text corpora.

Word embedding-CNN ([3]): CNN-based frameworks were explored and combined with TF-IDF features and word embedding techniques to convert textual input into a suitable format for deep neural networks, enabling effective sentiment prediction.

Co-LSTM ([8]): A hybrid approach combining two deep learning architectures, CNN and LSTM and RNN with memory, is proposed for sentiment classification of reviews across various domains.

([11]) The experiments focused on applying attention mechanisms within recurrent neural architectures, particularly those incorporating long short-term memory units and gated recurrent units. The attention strategies used in this study are adaptable and can be extended beyond sentiment analysis to a wide range of text classification problems and other application domains.

([32]) Introduce RU-BiLSTM, a deep recurrent network that merges bidirectional LSTM layers with word embeddings and an attention module to perform sentiment analysis on Roman Urdu. ([33]) proposed a

CNN-LSTM Approach for sentiment analysis. a data augmentation technique using a pre-trained RoBERTa model to address class imbalance and applied the fast Text classifier of US airlines Twitter dataset ([34]).
The Wilson-Cowan framework, utilized in neural mass network modelling for metapopulations, conceptualizes distinct subcortical brain areas as interacting nodes. The links between these nodes reflect different types of neural connectivity structural, functional, or effective capturing the dynamic relationships among brain regions. ([35])

4.5. Analysis and Interpretation of Results
This section juxtaposes our model's outcomes with those of the baseline methods. Additionally, we applied four different variations of the model using Word2Vec and GloVe word representations, along with various deep learning algorithms. Training proceeded for ten epochs in mini-batches of 32 samples with a learning rate of $1{\times}10^{-4}$ to reach the target accuracy on the datasets.

4.5.1. Analysis of Results on the US-Airline Dataset
Table 4 presents the performance of each model on the US-Airline Twitter dataset. Against the baseline approaches, our proposed model achieves an impressive 97.65% accuracy.

Table 4. Accuracy comparison on the US-Airline Twitter dataset with baseline models

Reference	Models	Accuracy (%)
([13])	ABCDM	92.75
([20])	ACL-SA	94.01
([3])	TF-IDF-CNN	85.45
([3])	TF-IDF-RNN	82.80
([3])	Word embedding-CNN	90.37
([3])	Word embedding-RNN	90.45
([32])	RoBERTa	96.23
Proposed models	**Word2Vec**	**GloVe**
CNN-LSTM	94.55	97.19
CNN-BiLSTM	94.06	96.51
CNN-GRU	93.81	96.35
Attention-CNN-GRU	95.88	96.89
Attention-CNN-BiLSTM	96.26	**97.65**

4.5.2. Analysis of Results on the IMDB Dataset
Table 5 displays the results of evaluating both our model and the chosen baselines on the IMDB dataset, with our approach attaining 90.53% accuracy.

Table 5. Evaluating Accuracy comparison on the IMDB dataset relative to the baseline Models

Reference	Models	Accuracy (%)	
([6])	Co-LSTM	83.13	
([3])	TF-IDF-CNN	82.30	
([3])	TF-IDF-RNN	56.39	
([3])	Word embedding-CNN	86.07	
([3])	Word embedding-RNN	87.05	
([31])	CNN-LSTM	85.0	
([33])	Bert+ Wilson-Cowan model RNN	87.46	
([9])	LSTM	86.21	
	Bi-LSTM	86.69	
	GRU	85.78	
	Self-Att-LSTM+word2vec	89.71	
Proposed models		**Word2Vec**	**GloVe**
	CNN-LSTM	89.19	89.51
	CNN-BiLSTM	89.34	89.49
	CNN-GRU	87.76	88.30
	Attention-CNN-GRU	89.17	89.66
	Attention-CNN-BiLSTM	89.10	**90.53**

4.5.3 Ablation Study of Model Components

study by removing or modifying one module at a time from the proposed configuration (GloVe + CNN + BiLSTM + Attention).

Table X shows that eliminating the attention layer leads to a 1.9% drop in accuracy, while removing the CNN or BiLSTM modules results in decreases of about 3% and 1.8%, respectively.

These results confirm that all three components—CNN for local feature extraction, BiLSTM for long-range context modeling, and the attention mechanism for focusing on sentiment-relevant tokens work together to enhance the model's overall performance.

Model Variant	Description	Accuracy (%)
Full Model (GloVe + CNN + BiLSTM + Attention)	Proposed architecture	**97.65**
– Attention	Without attention layer	95.72
– CNN	Without convolutional layers	94.63
– BiLSTM	Replaced with unidirectional LSTM	93.88
– GloVe	Using Word2Vec only	96.26

5. Conclusion

Utilizing convolutional and recurrent network architectures markedly boosts the accuracy of text categorization. We propose a sentiment-analysis framework that fuses CNN with BiLSTM layers, enhanced by an attention mechanism. The model incorporates multiple stages, including data preprocessing, we integrate pretrained word vectors with neural modules engineered to handle noisy social - media data, then apply convolutional filters to capture local contextual cues and employ max-pooling to distill and compress those features. The Bi-LSTM component captures bidirectional dependencies Via a dual-pass approach on the feature sequence first moving forward, then in reverse. Attention mechanisms are employed to dynamically highlight sentiment-relevant tokens. Experimental evaluations demonstrate that the model achieves strong performance, obtaining 90.53% accuracy on the IMDB dataset and 97.65% on the US-Airline dataset. The architecture maintains a relatively simple design, offers reduced computational overhead, and surpasses several existing models with respect to classification accuracy.

6. Limitations and Future Work

Although the proposed Attention-based CNN-BiLSTM model demonstrates strong results on both short and long textual datasets, several limitations remain. First, since the model relies on monolingual pretrained embeddings (Word2Vec and GloVe), its generalizability to multilingual or code-mixed content may be limited. Moreover, the model's performance could decrease when applied to extremely noisy, informal social-media text that includes emojis, abbreviations, or misspellings. Future work will explore integrating multilingual contextual embeddings such as mBERT or XLM-R and applying domain-adaptive pretraining to enhance robustness. Additionally, extending the framework to multimodal sentiment analysis incorporating visual or emoji features represents a promising direction for improving real-world applicability.

6. References

[1] A. Kumar Sharma, S. Chaurasia and D. Kumar Srivastava, "Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec," *Procedia Computer Science,* vol. 167, pp. 1139-1147, April 2020.

[2] J. S. and . M. M. , "An efficient sentiment analysis methodology based on long short-term memory networks," *Complex & Intelligent Systems,* vol. 7, p. 2485–2501, 18 June 2021.

[3] N. Cach Dang , M. N. Moreno-García and F. De la Prieta, "Sentiment Analysis Based on Deep Learning:A Comparative Study," *Electronics,* 14 March 2020.

[4] M. Birjali, M. Kasri and A. Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches,challenges and trends," *Knowledge-Based Systems,* vol. 226, p. 107134, 17 August 2021.

[5] M. D. and . R. R. , "An efficient sentimental analysis using hybrid deep learning and optimization technique for Twitter using parts of speech (POS) tagging," *International Journal of Speech Technology,* vol. 24, p. 329–339, 1 February 2021.

[6] Zafarani-Moattar, E., Kangavari, M. R., & Rahmani, A. M. (2022). Topic detection on COVID-19 tweets: A comparative study on clustering and transfer learning models. Tabriz Journal of Electrical Engineering, 52(4), 281–291. https://doi.org/10.22034/tjee.2022.15744.]

[7] R. Kumar Behera, M. Jena, S. Kumar Rath and S. Misra, "Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data," *Information Processing and Management,* vol. 58, no. 1, January 2021.

[8] S. Sachin, A. Tripathi, N. Mahajan, S. Aggarwal and P. Nagrath, "Sentiment Analysis Using Gated Recurrent Neural Networks," *SN Computer Science,* March 2020.

[9] S. TAM, R. BEN SAID and Ö. ÖZGÜR TANRIÖVER, "A ConvBiLSTM Deep Learning Model-Based Approach for Twitter Sentiment Classification," *IEEE Access,* vol. 9, pp. 41283 - 41293, 9 March 2021.

[10] S. Kardakis, I. Perikos, F. Grivokostopoulou and I. Hatzilygeroudis, "Examining Attention Mechanisms in Deep Learning Models for Sentiment Analysis," *Applied Sciences 11,* no. 9, 25 April 2021.

[11] Z. Yang, Z. Dai, Y. Yang1, J. Carbonell1, R. Salakhutdinov1 and Q. V. Le, "XLNet: Generalized Autoregressive Pretraining for Language Understanding," January 2020.

[12] M. Umut Salur and I. Aydin, "A Novel Hybrid Deep Learning Model for Sentiment Classification," *IEEE Access,* vol. 8, pp. 58080 - 58093, 23 March 2020.

[13] X. FU, J. YANG, J. LI, M. FANG3 and H. WANG4, "Lexicon-enhanced LSTM with Attention for General Sentiment Analysis," *IEEE Access,* vol. 6, pp. 71884 - 71891, 29 October 2018.

[14] M. E. Basiri, S. Nemati, M. Abdar and E. Cambria, "ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis," *Future Generation Computer Systems,* vol. 115, pp. 279-294, February 2021.

[15] C. N. Dang, M. N. Moreno-Garcı´a and F. De la Prieta, "Hybrid Deep Learning Models for Sentiment Analysis," *Complexity,* 13 August 2021.

[16] G. XU, Y. MENG, X. QIU, Z. YU and X. WU, "Sentiment Analysis of Comment Texts Based on BiLSTM," *IEEE Access,* vol. 7, pp. 51522 - 51532, 9 April 2019.

[17] A. H. Ombabi, O. Lazzez, W. Ouarda and A. M. Alimi, "Deep learning framework based on Word2Vec and CNNfor users interests classification," in *Sudan Conference on Computer Science and Information Technology (SCCSIT)*, 2017.

[18] پورحسینی، محمدرضا؛ عباسی، مهدی و محمدی‌پسند، احسان. (۱۴۰۳). توسعه مدل‌های کوچک یادگیری ماشین برای توزیع بهینه بارهای کاری در شبکه‌های لبه. مجله مهندسی برق دانشگاه تبریز، ۵۴(۴)، ۴۰۴-۴۱۳. https://doi.org/10.22034/tjee.2024.57664.4672

[19] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing,* vol. 337, pp. 325-338, 14 April 2019.

[20] Z. JIANQIANG, G. XIAOLIN and Z. XUEJUN, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," *IEEE Access,* vol. 6, pp. 23253 - 23260, 1 January 2018.

[21] Y. LIN, J. LI, L. YANG, K. XU and H. LIN, "Sentiment Analysis With Comparison Enhanced Deep Neural Network," *IEEE Access,* vol. 8, pp. 78378 - 78384, 22 April 2020.

[22] M. Kamyab, G. Liu and M. Adjeisah, "Attention-Based CNN and Bi-LSTM Model Based on TF-IDF and GloVe Word Embedding for Sentiment Analysis," *Applied Sciences,* no. 23, 27 November 2021.

[23] H. Tien Nguyen and M. Le Nguyen, "An ensemble method with sentiment features and clustering support," *Neurocomputing,* vol. 370, pp. 155-165, 22 December 2019.

[24] M. Kamkarhaghighi and M. Makrehchi, "Content Tree Word Embedding for document representation," *Expert Systems with Applications,* pp. 241-249, 30 December 2017.

[25] S. Wen and J. Li, "Recurrent Convolutional Neural Network with Attention for Twitter and Yelp Sentiment Classification," in *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, 2018.

[26] D. ZHANG, L. TIAN, M. HONG, F. HAN, Y. REN and Y. CHEN, "Combining Convolution Neural Network and Bidirectional Gated Recurrent Unit for Sentence Semantic Classification," *IEEE Access,* vol. 6, pp. 73750 - 73759, 22 November 2018.

[27] U. Mohd, B. Ahmad, E. Song, M. S. Hossain, M. Alrashoud and G. Muhammad, "Attention-based sentiment analysis using convolutional and recurrent neural network," *Future Generation Computer Systems,* pp. 571-578, December 2020.

[28] W. Li, F. Qi, M. Tang and Z. Yu, "Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification," *Neurocomputing,* vol. 387, pp. 63-77, 28 April 2020.

[29] G. Rao, W. Huang, Z. Feng and Q. Cong, "LSTM with sentence representations for document-level sentiment classification," *Neurocomputing,* vol. 308, pp. 49-57, 25 September 2018.

[30] J. DU, L. GUI, Y. HE, R. XU and X. WANG, "Convolution-Based Neural Attention With Applications to Sentiment Classification," *IEEE Access,* vol. 7, pp. 27983 - 27992, 21 February 2019.

[31] A. U. Rehman, A. K. Malik, B. Raza and W. Ali, "A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis," *Multimedia Tools and Applications,* vol. 78, p. 26597–26613, 11 June 2019.

[32] B. Ahmed Chandio, A. Shariq Imran, M. Bakhtyar, S. M. Daudpota and J. Baber, "Attention-Based RU-BiLSTM Sentiment Analysis Model for Roman Urdu," Applied Sciences, no. 7, 4 April 2022.

[33] M. Mishra, A. Patil, " Sentiment Prediction of IMDb Movie Reviews Using CNN-LSTM Approach", International Conference on Control Communication & Computing India (ICCC) , July 2023.

[34] S. A. Sazan, M. Ahmed, T. B. Saad, M. Roy "Advanced Natural Language Processing Techniques for Efficient Sentiment Analysis of US Airline Twitter Data: A High-Performance Framework for Extracting Insights from Tweets", 2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT).

[35] R. Marino, L. Buffoni, L. Chicchi, F. D. Patti, D. Febbe, L. Giambagli, D. Fanelli, " Learning in Wilson-Cowan model for metapopulation", 2024.