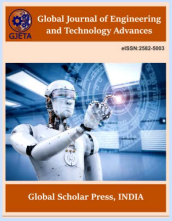
Global Journal of Engineering and Technology Advances
eISSN: 2582-5003
Cross Ref DOI: 10.30574/gjeta
Journal homepage: https://gjeta.com/



(RESEARCH ARTICLE)

Check for updates

A deep learning framework for SMS spam detection: Comparative performance of LSTM and BiLSTM Models

Gokada Mahesh * and Suneel Kumar Duvvuri

Department of Computer Science, Government College (Autonomous), Rajahmundry, Andhra Pradesh, India.

Global Journal of Engineering and Technology Advances, 2026, 27(01), 050-063

Publication history: Received on 05 March 2026; revised on 11 April 2026; accepted on 13 April 2026

Article DOI: https://doi.org/10.30574/gjeta.2026.27.1.0086

Abstract

Short Message Service (SMS) continues to be an important communication medium for personal, commercial, and institutional use because of its simplicity, accessibility, low cost, and immediate delivery. Despite the rapid growth of internet-based messaging platforms, SMS remains widely used for banking alerts, authentication codes, service notifications, promotional communication, and direct personal interaction. However, the increasing spread of unsolicited and deceptive SMS messages has made spam detection a critical challenge in modern mobile communication. Spam messages often include fraudulent offers, phishing links, fake alerts, and misleading promotional content, which may reduce user trust and create privacy and security risks. Conventional filtering approaches based on rules, blacklists, and manually selected keywords often struggle to adapt to the changing linguistic patterns and structural variations of spam messages. In this context, deep learning provides a more adaptive and effective solution for short text classification.

This study proposes a deep learning-based SMS spam detection framework using Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM) models. For experimental evaluation, a synthetically prepared and labelled SMS dataset was used. Before model training, the dataset was pre-processed through lowercase conversion, URL removal, number removal, punctuation removal, tokenization, and sequence padding. These preprocessing steps transformed raw SMS text into a structured numerical format suitable for sequence-based learning. Both models were implemented and evaluated under the same experimental settings to ensure a fair comparison. The framework used a vocabulary size of 20,000 words, a sequence length of 100, and an 80:20 train-test split. Model performance was evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score.

The experimental results showed highly effective classification performance on the prepared dataset, with both the LSTM and BiLSTM models achieving 100.00% test accuracy in the reported experiment. These findings indicate that sequence-based deep learning architectures can be highly effective for SMS spam detection within the experimental setting of this study. However, the reported performance should be interpreted carefully, as it may also reflect the characteristics and separability of the prepared dataset. Overall, this study provides a comparative evaluation of LSTM and BiLSTM models and offers a useful foundation for future research on intelligent message filtering, secure mobile communication, and more robust spam detection systems.

Keywords: SMS Spam Detection; Deep Learning; LSTM; BiLSTM; Text Classification; Natural Language Processing.

1. Introduction

Short Message Service (SMS) continues to be an important mode of communication in both personal and institutional contexts despite the rapid expansion of internet-based messaging platforms. Its simplicity, fast transmission, low cost,

* Corresponding author: Gokada Mahesh.

and wide accessibility have ensured its continued relevance in modern communication systems. SMS is still extensively used for banking alerts, one-time passwords, appointment reminders, delivery notifications, and customer service interactions, making it a trusted and practical medium for exchanging important information [1]. However, this widespread use has also made SMS a common target for misuse through unsolicited and deceptive messages known as SMS spam [2].

SMS spam has become a major concern because it extends beyond simple promotional disturbance and often includes misleading offers, fake rewards, phishing links, fraudulent notifications, and impersonation-based content. Such messages are deliberately designed to attract user attention and manipulate recipients into taking unsafe actions. As mobile communication continues to grow, the presence of spam messages creates not only inconvenience but also serious risks related to privacy, security, and financial safety [3]. Since SMS is often treated as a trusted source of information, deceptive messages delivered through this medium can have a significant harmful impact and weaken user confidence in mobile communication systems [4].

A major challenge in SMS spam detection lies in the nature of SMS text itself. Unlike long-form documents, SMS messages are short, brief, and often unstructured. They may contain abbreviations, shortened words, irregular spelling patterns, symbols, and informal expressions, which make classification difficult. Traditional spam detection approaches based on keyword filtering, blacklists, and manually defined rules are often unable to manage such variability effectively [5]. Although these methods may identify certain known spam patterns, they usually fail when message wording changes or when spammers intentionally disguise malicious intent. To address these limitations, intelligent data-driven methods such as machine learning and deep learning have gained increasing importance in spam detection research. These methods can learn meaningful patterns directly from labelled message data and provide better adaptability than fixed-rule systems [6]. As a result, intelligent classification techniques have become increasingly important for developing reliable and robust SMS filtering systems. [7].

The challenge is further intensified by the fact that many spam messages are intentionally crafted to resemble authentic communication from banks, telecom providers, online services, and other trusted organizations. In such cases, users may find it difficult to distinguish genuine messages from fraudulent ones [8]. Moreover, the short and unstructured format of SMS reduces contextual richness, which further limits the effectiveness of conventional filtering methods based on static patterns [9]. These issues highlight the need for automated and adaptive systems capable of identifying spam messages with greater accuracy and consistency. Manual identification is not always practical because users receive numerous messages every day, and a misleading message may easily be overlooked as legitimate. An automated SMS spam detection system can therefore reduce user burden by providing fast, consistent, and reliable classification of incoming messages [10]. Since spam content continues to evolve, intelligent models are better suited than conventional keyword-based or rule-based techniques [11]. Therefore, a deep learning-based approach is needed to improve the accuracy, adaptability, and reliability of SMS spam detection [12].

In this context, the present study proposes a deep learning framework for classifying SMS messages into spam and ham categories [13]. The research focuses on two sequence-based deep learning models, namely Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM). These models are well suited for textual sequence analysis because they can capture contextual dependencies and learn from the order of words within a message [14]. To evaluate the effectiveness of the proposed models, the study employs standard classification measures such as accuracy, precision, recall, and F1-score [15]. The study is limited to binary classification of text-based SMS content in a dataset-based experimental setting and does not extend to e-mail, social media, multimedia messages, or real-time commercial deployment environments [16][17].

The significance of this study is both practical and academic. From a practical perspective, an effective SMS spam detection system can help protect users from misleading and harmful content in a communication channel that remains important for alerts, reminders, and authentication services [18]. From an academic perspective, the study contributes to research in text classification, natural language processing, and deep learning by examining the suitability of LSTM and BiLSTM models for short-text spam detection [19]. Furthermore, the comparative analysis presented in this work may assist future researchers in selecting appropriate sequence-based models for spam detection and related communication security applications [20]. Overall, this study presents a focused deep learning-based approach to SMS spam detection and aims to support the development of safer, more reliable, and more trustworthy mobile communication systems.

2. Literature Review

SMS spam detection has become an important research area in mobile communication security because Short Message Service (SMS) is still widely used for banking alerts, one-time passwords, reminders, delivery notifications, and other service-related communication. Although internet-based messaging applications are now very common, SMS remains relevant because of its simplicity and practical value. However, the growing use of SMS has also increased the spread of spam, phishing, and deceptive mobile messages. Since SMS messages are usually short and unstructured, they are more difficult to classify than longer text documents. As a result, research in this area has moved from simple rule-based filtering methods to advanced machine learning and deep learning approaches.

In the early stage, spam filtering mainly relied on keyword matching, blacklist rules, and manually defined patterns. These methods were easy to use, but they were not flexible when spammers changed wording or message structure. As spam messages became more deceptive, researchers started using machine learning methods that could learn directly from labelled SMS data. Over time, the literature expanded into practical mobile deployment studies, comparative machine learning frameworks, statistical learning models, language-specific methods, multilingual systems, hybrid deep learning architectures, and contextual transformer-based approaches. This development shows that SMS spam detection has become a mature research area, although performance still depends on the dataset, language, preprocessing strategy, and model design.

Narayan and Saxena [21] studied SMS spam detection on Android platforms and reported about 56% accuracy. Their work is important because it focused on real mobile deployment and showed that practical spam detection is more difficult than controlled experimental classification. A stronger machine learning foundation was later provided by Shirani-Mehr [22], who reported about 98.5% accuracy and established SMS spam filtering as an important short-text classification problem. Suleiman and Al-Naymat [23] further strengthened the field by proposing an H2O-based comparative framework in which Random Forest achieved 97.7% accuracy. These studies together helped build strong practical and machine learning baselines for later research.

Researchers also explored alternative and language-specific methods. Pudasaini et al. [24] used a Relevance Vector Machine and reported an F1-score of 0.975175, showing that alternative statistical learning methods can also perform well. Al Saidat et al. [25] proposed an Arabic SMS spam detection model and reported 96.99% accuracy, highlighting the importance of language-specific modelling. More recent studies moved toward advanced contextual and deep learning methods. Shen et al. [26] proposed a BERT-based model with multi-graph convolution and reported 99.28% and 93.78% accuracy on two benchmark datasets. This work demonstrated the value of contextual language modelling in detecting more complex spam patterns.

Another important trend in the literature is multilingual and hybrid deep learning. Sachan et al. [27] reported 99.0% accuracy for a multilingual spam detection model, showing that high performance can also be achieved in linguistically diverse settings. Similarly, Ghourabi et al. [28] proposed a hybrid CNN-LSTM model for Arabic and English SMS spam detection and reported 98.74% overall accuracy. Their work showed that combining local pattern learning with sequential modelling can be highly effective for short-text classification.

Overall, the literature shows that SMS spam detection has evolved from rule-based filtering to machine learning baselines, comparative frameworks, language-specific models, contextual transformer-based methods, multilingual systems, and hybrid deep learning architectures. Although many studies have reported strong performance, they differ in dataset, language, and model complexity. Therefore, there is still a need for a focused comparison of closely related sequence-based models under a controlled binary SMS spam detection setting. This is the gap addressed by the present study, which compares LSTM and BiLSTM models within a consistent dataset-based framework in Table 1.

Table 1 Comparative Analysis of Previous Studies on SMS Spam Detection and the Present Study

Author(s) with Ref. No.	Year	Method / Model Used	Reported Accuracy / Performance	Key Observation
Narayan and Saxena [20]	2013	Android-based SMS spam detection evaluation	56% accuracy	Showed the practical difficulty of spam detection in real Android environments
Shirani-Mehr [21]	2014*	Machine learning approach	98.5% accuracy	Established a strong machine learning baseline for SMS spam filtering
Suleiman and Al-Naymat [22]	2017	H2O framework with Random Forest	97.7% accuracy	Demonstrated the value of comparative machine learning evaluation
Pudasaini et al. [23]	2023	Relevance Vector Machine	F1-score = 0.975175	Showed that alternative statistical learning methods can be effective
Al Saidat et al. [24]	2024	Arabic SMS spam detection model	96.99% accuracy	Highlighted the importance of language-specific spam detection
Shen et al. [25]	2025	BERT with multi-graph architecture	99.28% and 93.78% accuracy	Demonstrated strong contextual performance using transformer-based methods
Sachan et al. [26]	2025	Multilingual spam detection model	99.0% accuracy	Showed high effectiveness in multilingual spam detection
Ghourabi et al. [27]	2020	Hybrid CNN-LSTM model	98.74% accuracy	Proved that hybrid deep learning models can handle SMS spam effectively
Present Study	2026	LSTM and BiLSTM models for SMS spam detection	100% accuracy	Provides a focused comparison of sequence-based deep learning models under a controlled binary classification setting

3. Methodology

3.1. System Overview

The proposed SMS spam detection system is developed as a deep learning-based text classification framework. It classifies SMS messages into two categories, namely spam and ham.

3.2. Research Design

The study follows an experimental research design with a step-by-step workflow. It includes data preparation, preprocessing, model building, training, and evaluation in Fig1.

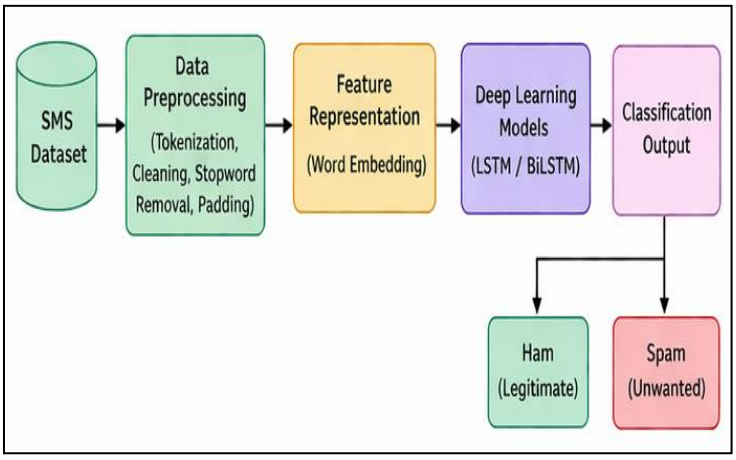


Figure 1 Overview of the Proposed SMS Spam Detection System

3.3. Development Environment and Tools Used

The project was implemented using Python and supporting libraries for data processing, visualization, and deep learning. NumPy and Pandas were used for data handling, Matplotlib, Seaborn, and WordCloud were used for visualization, Regular Expressions and the string module were used for text cleaning, Scikit-learn was used for data splitting and label encoding, and TensorFlow/Keras[29] was used for tokenization, padding, model building, training, and evaluation.

3.4. Dataset Description

The dataset used in this study is *sms_spam_ham_80000_clean.csv*, which contains 80,000 SMS messages. It has two columns, namely *label* and *message*, and is treated as a binary classification dataset. The dataset is a synthetic SMS dataset developed for experimental spam detection analysis, where the messages are organized into spam and ham classes for model training and evaluation in Table2.

Table 2 Dataset Description

Attribute	Description
Dataset Name	sms_spam_ham_80000_clean.csv
Total Messages	80,000
Number of Columns	2
Column Names	label, message
Classification Type	Binary classification
Target Classes	ham, spam
Dataset Nature	Synthetic SMS dataset

3.5. Data Inspection and Quality Checking

After loading the dataset, its structure, data types, missing values, and duplicate rows were checked. Duplicate records were removed to improve data quality and model reliability.

3.6. Class Distribution Analysis

The distribution of spam and ham messages was examined before model training. This helps in understanding dataset balance and its effect on classification performance.

3.7. Label Encoding

The original labels were converted into binary numerical form for model training. In this study, ham was encoded as 0 and spam was encoded as 1.

$$y = \begin{cases} 0, & \text{if the message is ham} \\ 1, & \text{if the message is spam} \end{cases}$$

This transformation enables the problem to be handled as a binary classification task and supports the use of binary cross-entropy loss during training.

3.8. Text Cleaning and Preprocessing

Raw SMS text was cleaned by converting it to lowercase and removing URLs, numbers, punctuation, and extra spaces. The cleaned text was then used as the final input for model training.

$$T = \{w_1, w_2, w_3, \ldots, w_n\} \quad (1)$$

$$T' = f(T) \quad (2)$$

where f(T) represents the preprocessing function applied to the raw text. After preprocessing, the cleaned SMS message was used as the final input for tokenization and modelling.

3.9. Feature Representation

The cleaned SMS messages were converted into numerical form using tokenization and sequence padding. This step makes the text suitable for deep learning model input [30].

3.9.1. Tokenization

Each word in the SMS message was converted into a unique integer index. This transforms the message into a numerical sequence.

$$X = [w_1, w_2, \dots, w_n] \quad (3)$$

where each element represents the tokenized form of the input text sequence.

3.9.2. Sequence Padding

All tokenized sequences were padded to the same length for model training. In this study, the maximum sequence length was fixed at 100.

$$X_{padded} = [w_1, w_2, \dots, w_n, 0, 0, \dots, 0] \quad (4)$$

where the total sequence length becomes 100. This ensures fixed-size input for the neural network models.

3.10. Train-Test Split

The dataset was divided into training and testing sets using an 80:20 ratio. This split helps in evaluating the model on unseen data.

3.11. Proposed LSTM Model

The first model used in this study is LSTM, which is suitable for learning sequential patterns from text data [31]. It consists of an Embedding layer, an LSTM layer, a Dropout layer, and a Dense output layer.

3.11.1. Embedding Layer

The embedding layer transforms tokenized SMS sequences into dense vector representations. It helps the model learn meaningful numerical features from the text.

$$E \in \mathbb{R}^{V \times D} \quad (5)$$

where V is the vocabulary size and D is the embedding dimension.

3.11.2. LSTM Cell Equations

The LSTM unit consists of three gates and a memory cell [32]

Forget Gate

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (6)$$

The forget gate decides how much information from the previous cell state should be retained.

Input Gate

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (7)$$

The input gate controls how much new information should be added to the memory cell.

Candidate State

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (8)$$

This represents the new candidate information generated from the current input and previous hidden state.

Cell State Update

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (9)$$

This equation updates the memory cell by combining retained past information and new input.

Output Gate

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (10)$$

The output gate controls how much information from the cell state contributes to the hidden state.

Hidden State

$$h_t = o_t * \tanh(C_t) \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (11)$$

The hidden state acts as the output of the LSTM cell and is passed to the next time step. In this way, the LSTM model captures sequential patterns in SMS text and supports effective spam classification.

3.12. Proposed BiLSTM Model

The second model used in this study is the BiLSTM model, which processes the input sequence in both forward and backward directions [33]. This helps in capturing better contextual information than the standard LSTM model in Fig2.

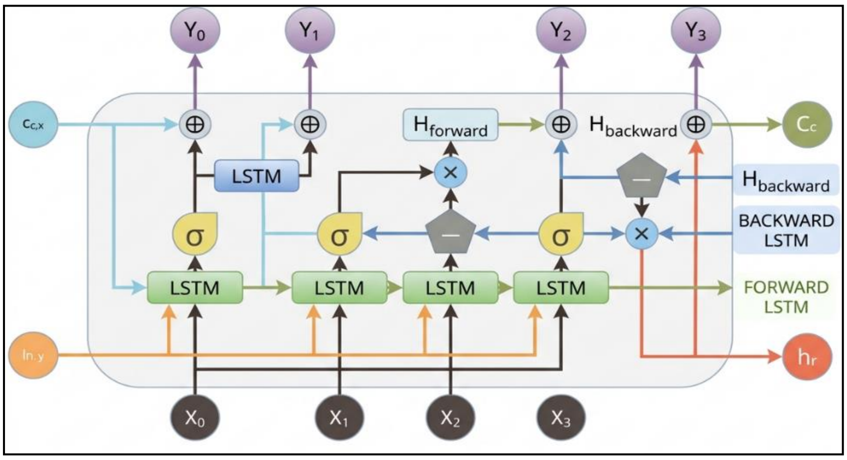


Figure 2 Architecture of the Proposed BiLSTM Model for SMS Spam Detection

The figure illustrates the bidirectional processing mechanism of the BiLSTM model, where the input sequence is learned in both forward and backward directions to improve contextual understanding for SMS spam classification.

Final output:

$$y_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t} \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \dots (12)$$

where $\overrightarrow{h_t}$ represents the forward hidden state and $\overleftarrow{h_t}$ represents the backward hidden state. Because of bidirectional learning, the BiLSTM model is expected to capture contextual dependencies more effectively than the standard LSTM model.

3.13. Model Configuration

Both models were trained using the same configuration to ensure fair comparison. The main settings include vocabulary size 20,000, sequence length 100, embedding dimension 128, LSTM units 64, dropout rate 0.5, epochs 10, batch size 64, and validation split 0.2. in Table3.

Table 3 Hyperparameter Configuration of the Proposed Models

Hyperparameter	Value
Vocabulary Size	20,000
Sequence Length	100
Embedding Dimension	128
LSTM Units	64
Dropout Rate	0.5
Epochs	10
Batch Size	64
Validation Split	0.2

3.14. Training Strategy

Both models were trained for 10 epochs with batch size 64 and validation split 0.2. Early Stopping with patience value 3 was used to reduce overfitting.

3.15. Loss Function and Optimizer

The models were compiled using the Adam optimizer [34] and binary cross-entropy loss function. This setup is suitable for binary classification tasks such as spam and ham detection.

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] \quad \ldots\ldots (13)$$

where $N$is the total number of samples, $y_i$is the actual class label, and $\hat{y}_i$is the predicted probability. This configuration provides an effective training setup for binary SMS classification.

3.16. Evaluation Metrics

The model performance was evaluated using accuracy, precision, recall, and F1-score. These metrics provide a balanced understanding of spam classification results.

Accuracy: Accuracy measures the overall proportion of correctly classified messages out of the total number of messages.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad \ldots\ldots (14)$$

Precision: Precision measures the proportion of correctly predicted spam messages among all messages predicted as spam.

$$\text{Precision} = \frac{TP}{TP+FP} \quad \ldots\ldots (15)$$

Recall: Recall measures the proportion of actual spam messages that are correctly identified by the model.

$$\text{Recall} = \frac{TP}{TP+FN} \quad \ldots\ldots (16)$$

F1-score: F1-score is the harmonic mean of precision and recall and provides a balanced measure of classification performance.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (17)$$

Pseudo algorithm

The complete workflow of the proposed SMS spam detection framework is summarized in Algorithm.

Algorithm: SMS Spam Detection using LSTM and BiLSTM

Begin

Input:
SMS dataset containing labels and messages

Output:

Performance comparison of LSTM and BiLSTM models

1. Load the SMS dataset.
2. Inspect the dataset and remove duplicate records.
3. Analyze the class distribution of spam and ham messages.
4. Encode labels into binary form.
5. Preprocess the text by converting to lowercase and removing URLs, numbers, punctuation, and extra spaces.
6. Tokenize the cleaned text and convert it into numerical sequences.
7. Pad all sequences to a fixed length.
8. Split the dataset into training and testing sets.
9. Configure model hyperparameters.
10. Build, train, and evaluate the LSTM model.
11. Build, train, and evaluate the BiLSTM model.
12. Compare the performance of both models using accuracy, precision, recall, and F1-score.
13. Report the final results.

The algorithm outlines the major steps involved in dataset preparation, preprocessing, sequence generation, model training, and performance evaluation.

3.17. Summary of Methodology

The proposed methodology combines preprocessing, sequence generation, and deep learning-based classification for SMS spam detection. It provides a clear framework for comparing the performance of LSTM and BiLSTM models.

4. Results and Discussion

4.1. Experimental Setup

The proposed SMS spam detection models were implemented using Python with deep learning libraries. The dataset was divided into training and testing sets using an 80:20 ratio, and both LSTM and BiLSTM models were trained under the same experimental settings to ensure a fair comparison. The main training parameters included 10 epochs, batch size of 64, validation split of 0.2, Adam optimizer, binary cross-entropy loss function, and early stopping with patience value 3. This setup helped the models learn efficiently while also reducing the possibility of overfitting during training.

4.2. Preliminary Data Analysis

Before model training, the dataset was carefully inspected and pre-processed to ensure its suitability for classification. After data inspection and cleaning, the dataset contained no missing values and no duplicate records, which made it suitable for reliable model training. After preprocessing, the SMS messages were converted into numerical sequences using tokenization with a vocabulary size of 20,000 and padding to a fixed sequence length of 100. These steps transformed the raw text into a structured format suitable for deep learning-based classification.

4.3. Training Performance

Both LSTM and BiLSTM models showed strong learning behavior during training. The LSTM model achieved a training accuracy of 0.9582 in the first epoch, while the BiLSTM model achieved 0.9597 in the first epoch. In both cases, validation accuracy quickly reached 1.0000, and the loss values reduced sharply across epochs. These observations indicate that both models learned the message patterns effectively and converged in a stable manner without major training instability. The training behavior of both models is further illustrated through the training and validation accuracy and loss plots shown in Fig3.

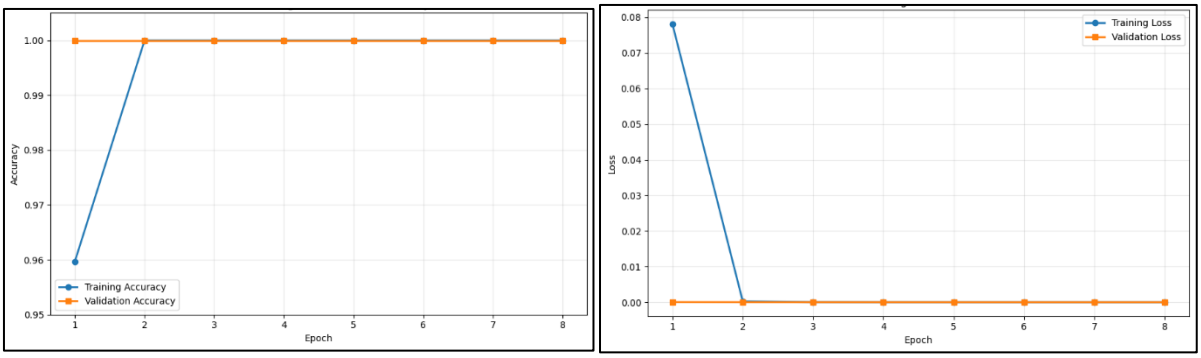


Figure 3 BiLSTM Model Performance Curves Showing Accuracy and Loss During Training

4.4. Test Performance

After training, both models were evaluated on the test dataset. The LSTM model achieved a test accuracy of 1.0000 (100%), and the BiLSTM model also achieved a test accuracy of 1.0000 (100%). This shows that both models correctly classified all test messages in the reported experiment. Such a result highlights the strong ability of the proposed deep learning framework to distinguish between spam and ham messages on the selected dataset.

4.5. Confusion Matrix Analysis

The confusion matrix of the BiLSTM model demonstrates excellent classification performance on the test dataset. As shown in Figure 4, all predictions are concentrated along the main diagonal of the matrix, indicating that the model correctly classified both ham and spam messages without misclassification in Fig4.

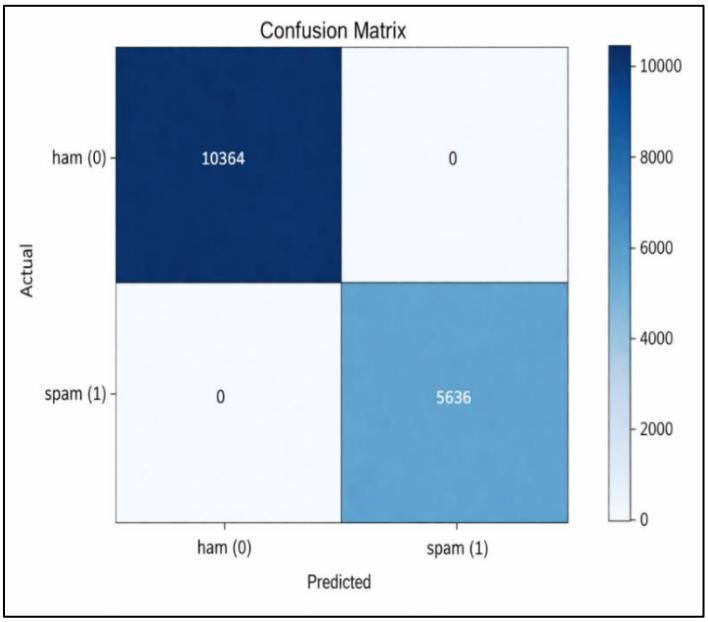


Figure 4 Confusion Matrix of the BiLSTM Model on the Test Dataset

The matrix shows that 10,364 ham messages were correctly classified as ham, while 5,636 spam messages were correctly classified as spam. At the same time, the off-diagonal values are zero, which means that no legitimate messages were incorrectly predicted as spam and no spam messages were incorrectly predicted as ham. This result indicates the absence of both false positive and false negative classifications in the reported experiment.

A strong diagonal pattern in the confusion matrix is important because it reflects the model's ability to clearly separate the two classes. In practical terms, this means that the BiLSTM model was able to preserve legitimate messages while also detecting spam messages accurately. Therefore, the confusion matrix further confirms the effectiveness of the proposed BiLSTM-based SMS spam detection model within the experimental setting of this study.

4.6. Evaluation Metrics

In addition to the confusion matrix analysis, the performance of the BiLSTM model was evaluated using four standard classification metrics, namely accuracy, precision, recall, and F1-score. These metrics provide a balanced and detailed assessment of classification quality in binary spam detection.

In the reported experiment, the BiLSTM model achieved 1.0000 (100%) for accuracy, precision, recall, and F1-score on the selected test dataset. The accuracy value indicates that all test messages were classified correctly. The precision score confirms that all messages predicted as spam were actually spam, while the recall score shows that all real spam messages were successfully detected. The F1-score also reached 1.0000, indicating a perfect balance between precision and recall.

Together, these evaluation metrics confirm the strong effectiveness of the proposed BiLSTM model for SMS spam detection within the experimental setting of this study. However, these perfect results should be interpreted with caution, as they may also reflect the clean, structured, and synthetic nature of the prepared dataset.

4.7. Comparative Discussion

The table below presents a comparative summary of the LSTM and BiLSTM models used in this study.

Table 4 Comparison Between LSTM and BiLSTM Models

Aspect	**LSTM**	**BiLSTM**
Processing Direction	Forward only	Forward and backward
Contextual Learning	Learns past context	Learns past and future context
Test Accuracy	100.00%	100.00%
Precision	100.00%	100.00%
Recall	100.00%	100.00%
F1-Score	100.00%	100.00%
Computational Requirement	Lower	Slightly higher
Overall Observation	Strong performance	Strong performance with richer contextual understanding

A comparison between LSTM and BiLSTM shows that both models performed exceptionally well on the selected dataset. Although both achieved 100.00% across all evaluation metrics, the BiLSTM model offers an additional advantage because it learns contextual information in both forward and backward directions. This enables it to capture richer sequence information than the standard LSTM model. However, bidirectional learning generally requires slightly higher computational effort. Therefore, the choice between the two models may depend on the balance between contextual learning capability and computational cost.

5. Discussion

The overall results indicate that deep learning models are highly effective for SMS spam detection in the current experimental setting. The strong performance of both LSTM and BiLSTM suggests that sequence-based learning is well suited for short-text classification problems such as SMS spam filtering. At the same time, the perfect results should be

interpreted carefully, because they may also be influenced by the clean, structured, and synthetic nature of the dataset used in the study. Even so, the results clearly demonstrate that the proposed framework provides a strong foundation for accurate and reliable SMS spam classification.

6. Conclusion

This study presented a deep learning-based SMS spam detection framework using LSTM and BiLSTM models for binary classification of SMS messages into spam and ham categories. The experimental results showed that both models achieved excellent performance on the selected dataset, while the BiLSTM model provided stronger contextual learning through bidirectional sequence processing. The evaluation metrics, confusion matrix analysis, and comparative discussion confirmed the effectiveness of the proposed approach for accurate SMS spam detection. Although the perfect performance should be interpreted carefully because of the clean and synthetic nature of the dataset, the study clearly demonstrates that sequence-based deep learning models provide a reliable and effective solution for SMS spam classification.

Future Work

In future work, the proposed system can be extended by using larger and more diverse real-world SMS datasets to improve generalization. Additional deep learning approaches such as GRU, attention-based models, and transformer-based architectures can also be explored for comparative analysis. The system may further be improved by supporting multilingual spam detection, real-time deployment, and more complex spam patterns that occur in practical mobile communication environments.

Compliance with ethical standards

Acknowledgement

The Author is thankful to the Department of Computer Science at Government College (Autonomous), Rajahmundry, for their support and encouragement throughout the course of this work.

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] K. A. Kannisto, M. H. Koivunen, and M. A. Välimäki, "Use of mobile phone text message reminders in health care services: A narrative literature review," *J. Med. Internet Res.*, vol. 16, no. 10, p. e222, Oct. 2014, doi: 10.2196/jmir.3442.

[2] T. A. Almeida, J. María Gómez Hidalgo, and T. P. Silva, "Towards SMS Spam Filtering: Results under a New Dataset." [Online]. Available: http://www.cloudmark.com/en/article/

[3] D. A. Oyeyemi and A. K. Ojo, "SMS Spam Detection and Classification to Combat Abuse in Telephone Networks Using Natural Language Processing," *Journal of Advances in Mathematics and Computer Science*, vol. 38, no. 10, pp. 144–156, Oct. 2023, doi: 10.9734/jamcs/2023/v38i101832.

[4] M. Salman, M. Ikram, and M. A. Kaafar, "An Empirical Analysis of SMS Scam Detection Systems," Oct. 2022, [Online]. Available: http://arxiv.org/abs/2210.10451

[5] G. José De Sousa, D. Carlos, G. Pedronette, P. Papa, and I. Rizzo Guilherme, "SMS Spam Detection Through Skip-gram Embeddings and Shallow Networks."

[6] M. A. Uddin, M. N. Islam, L. Maglaras, H. Janicke, and I. H. Sarker, "ExplainableDetector: Exploring Transformer-based Language Modeling Approach for SMS Spam Detection with Explainability Analysis," May 2024, [Online]. Available: http://arxiv.org/abs/2405.08026

[7] S. K. DUVVURI, *Applications of Artificial Intelligence Across Domains* . Commissionerate of Collegiate Education, Government of Andhra Pradesh , 2026. doi: 10.5281/zenodo.18623057.

[8] D. Goel, H. Ahmad, A. K. Jain, and N. K. Goel, "Machine Learning Driven Smishing Detection Framework for Mobile Security," Dec. 2024, [Online]. Available: http://arxiv.org/abs/2412.09641

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," Sep. 2013, [Online]. Available: http://arxiv.org/abs/1301.3781

[10] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification." [Online]. Available: https://github.com/facebookresearch/

[11] Y. Kim, "Convolutional Neural Networks for Sentence Classification." [Online]. Available: http://nlp.stanford.edu/sentiment/

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2019, [Online]. Available: https://arxiv.org/abs/1810.04805

[13] P. Zhou, Z. Qi, S. Zheng, J. Xu, H. Bao, and B. Xu, "Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling."

[14] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," Aug. 2015, [Online]. Available: http://arxiv.org/abs/1508.01991

[15] J. Howard and S. Ruder, "Universal Language Model Fine-tuning for Text Classification." [Online]. Available: http://nlp.fast.ai/ulmfit.

[16] A. Vaswani *et al.*, "Attention Is All You Need."

[17] D. P. Patinavalasa and D. Suneel Kumar, "Scalable Email Spam Detection Using BiLSTM with Large-Scale Hybrid Datasets," *International Journal Of Recent Trends In Multidisciplinary Research*, p. 96, Mar. 2026, doi: 10.59256/ijrtmr.20260602016.

[18] P. Zhou *et al.*, "Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification."

[19] F. J. Schwebel and M. E. Larimer, "Using text message reminders in health care services: A narrative literature review," Sep. 01, 2018, *Elsevier B.V.* doi: 10.1016/j.invent.2018.06.002.

[20] I. Gurol-Urganci, T. de Jongh, V. Vodopivec-Jamsek, R. Atun, and J. Car, "Mobile phone messaging reminders for attendance at healthcare appointments," Dec. 05, 2013, *John Wiley and Sons Ltd*. doi: 10.1002/14651858.CD007458.pub3.

[21] A. Narayan and P. Saxena, "The curse of 140 characters: Evaluating the efficacy of SMS spam detection on Android," in *Proceedings of the ACM Conference on Computer and Communications Security*, 2013, pp. 33–41. doi: 10.1145/2516760.2516772.

[22] H. Shirani-Mehr, "SMS Spam Detection using Machine Learning Approach."

[23] D. Suleiman and G. Al-Naymat, "SMS Spam Detection using H2O Framework," in *Procedia Computer Science*, Elsevier B.V., 2017, pp. 154–161. doi: 10.1016/j.procs.2017.08.335.

[24] S. Pudasaini, A. Shakya, S. P. Pandey, P. Paudel, S. Ghimire, and P. Ale, "SMS Spam Detection using Relevance Vector Machine," in *Procedia Computer Science*, Elsevier B.V., 2023, pp. 337–346. doi: 10.1016/j.procs.2023.12.089.

[25] M. R. Al Saidat, S. Y. Yerima, and K. Shaalan, "A Novel Approach for Arabic SMS Spam Detection Using Hybrid Deep Learning Techniques," in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 260–267. doi: 10.1016/j.procs.2024.10.199.

[26] L. Shen, Y. Wang, Z. Li, and W. Ma, "SMS spam detection using BERT and multi-graph convolutional networks," *International Journal of Intelligent Networks*, vol. 6, pp. 79–88, Jan. 2025, doi: 10.1016/j.ijin.2025.06.002.

[27] R. K. Sachan, V. Tiwari, N. Raghav, and P. Rajpoot, "Spam Text Messages Detection Using Multilingual Embedding," in *Procedia Computer Science*, Elsevier B.V., 2025, pp. 390–398. doi: 10.1016/j.procs.2025.03.215.

[28] A. Ghourabi, M. A. Mahmood, and Q. M. Alzubi, "A hybrid CNN-LSTM model for SMS spam detection in arabic and english messages," *Future Internet*, vol. 12, no. 9, Sep. 2020, doi: 10.3390/FI12090156.

[29] M. Suja, P. Lavanya, P. D. Prasad, and S. K. Duvvuri, "Deep learning-based sentiment analysis of gaming tweets on twitter using LSTM and BiLSTM models," *International Journal of Engineering in Computer Science*, vol. 8, no. 1, pp. 215–222, Jan. 2026, doi: 10.33545/26633582.2026.v8.i1b.269.

[30] K. Cho *et al.*, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," Sep. 2014, [Online]. Available: http://arxiv.org/abs/1406.1078

[31] J. Y. Lee and F. Dernoncourt, "Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks," Mar. 2016, [Online]. Available: http://arxiv.org/abs/1603.03827

[32] Pandiri Lavanya, Patinavalasa Durga Prasad, and Suneel Kumar Duvvuri, "Context-Aware Sentiment Classification of Movie Reviews Using Bidirectional LSTM Networks," *Int. J. Sci. Res. Sci. Eng. Technol.*, vol. 13, no. 2, pp. 159–171, Mar. 2026, doi: 10.32628/IJSRSET261371.

[33] X. Ma and E. Hovy, "End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF," May 2016, [Online]. Available: http://arxiv.org/abs/1603.01354

[34] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Jan. 2017, [Online]. Available: http://arxiv.org/abs/1412.6980