International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-14 Issue-6, March 2026

A Comparative Analysis of Deep Learning Models for Smishing Detection in SMS Message

Aqsa Shaikh, Mariya Shaikh, Srivaramangai R

Abstract*: Smishing (SMS phishing) is a cyber threat that is growing rapidly, and it's a tactic in which attackers use SMS messages to deceive users and trick them into revealing their private information or unintentionally installing harmful applications. As mobile devices are used everywhere, detecting smishing messages has become a very important yet difficult task in cybersecurity. In the present study, the authors conduct a comparative analysis of several deep learning models, namely, the Long Short-Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), Convolutional Neural Network (CNN), and a combination of CNN-LSTM, for the detection of smishing. The experiments are conducted on publicly available SMS datasets, and performance is evaluated using accuracy, precision, recall, F1 Score, and a confusion matrix. The findings indicate that deep learning-based approaches yield significantly better results than traditional methods, with hybrid architectures leading in overall performance.

***Keywords*: Cybersecurity, Deep Learning, LSTM, Smishing.**

Nomenclature:
LSTM: Long Short-Term Memory
Bi-LSTM: Bidirectional Long Short-Term Memory
CNN: Convolutional Neural Network
CNN-LSTM: Convolutional Neural Network - LSTM
SMS: Short Message Service
SVM: Support Vector Machines
TD-IDF: Term Frequency Inverse Document Frequency
URL: Uniform Resource Locator
NLP: Natural Language Processing
RF: Random Forest
AdaBoost: Adaptive Boosting
XGBoost: Extreme Gradient Boosting
CTI: Cyber Threat Intelligence
OFVA: Optimal Feature Extraction Algorithm
MTD: Mobile Threat Defence
QML: Qt Modelling Language
CTI-MURLD: Cyber Threat Intelligence-based Malicious URL Detection
IndRNN: Independent Recurrent Neural Network
TSV: Tab-Separated Values

Manuscript received on 01 March 2026 | Revised Manuscript received on 09 March 2026 | Manuscript Accepted on 15 March 2026 | Manuscript published on 30 March 2026.
*Correspondence Author(s)
Aqsa Shaikh, Student, Department of Information Technology, University of Mumbai, Mumbai (Maharashtra), India. Email ID: aqsazshaikh786@gmail.com, ORCID ID: 0009-0000-8640-3270
Mariya Shaikh, Student, Department of Information Technology, University of Mumbai, Mumbai (Maharashtra), India. Email ID: shaikhmariya2909@gmail.com, ORCID ID: 0009-0003-1182-2640
Srivaramangai R.*, Head, Department of Information Technology, University of Mumbai, Mumbai (Maharashtra), India. Email ID: rsrimangai@gmail.com, ORCID ID: 0000-0003-2723-6067.

I. INTRODUCTION

The meteoric rise of mobile communication technologies has turned Short Message Service (SMS) into a necessary means of communication in the modern world. Banks, delivery services, and even government agencies communicate through SMS because it is instant, straightforward, and easy to use, just like personal messaging. Nevertheless, the very fact that SMS is everywhere has made it the most favoured platform for scammers, who use it to deceive and trick people. For instance, by sending fake messages that appear to come from a legitimate source, these perpetrators deceitfully lure unsuspecting victims to expose confidential information, such as bank user IDs, passwords, PINs, or other private data. They might also direct victims to click on harmful links that compromise users' devices and the organisation's data security. The increasing sophistication of smishing attacks, to the point that they even use personalised content and context-aware messaging, and the fact that they are hard to detect with traditional security measures, make the situation dire. Deep learning models, however, represent a breakthrough in text classification, as they relieve researchers of the noisy text preprocessing and instead perform the task of finding representative features through brute force. Different architectures, including Long Short-Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), Convolutional Neural Networks (CNN), and CNN-LSTM hybrids, can extract both major and minor features in text. The primary aim of the present research is to provide a comprehensive assessment and comparative analysis of the aforementioned deep learning models for smishing detection, highlighting their key aspects and weaknesses in detecting fake SMS messages. The application of deep learning methods in research is to construct a robust, automated framework capable of recognising malicious messages, adapting to evolving attack methods, and reducing the need for manual feature engineering, thereby improving the security of SMS communication in real-world situations.

II. RELATED WORK

Numerous studies have investigated the use of machine learning techniques such as Naive Bayes, Support Vector Machines (SVMs), and Random Forests for spam and phishing detection. Although these methods demonstrate adequate performance, they still rely on manual feature engineering. Shaikh et al. This paper provides a thorough investigation into detecting smishing (SMS phishing) attacks using machine learning techniques. It

examines and contrasts several ML and deep learning models, such as Logistic Regression, Random Forest, SVM, CNN, and LSTM, for sorting SMS messages. Several feature extraction techniques, such as TF-IDF, N-grams, NLP, and URL analysis, are used to improve detection accuracy. The paper shows that ensemble and deep learning methods outperform classical techniques and address the main issues that arise in smishing detection. Shaikh et al. The research presented in this paper seeks to determine how well various machine learning and deep learning models detect smishing (i.e., SMS phishing) attacks. A variety of traditional classifiers (such as Logistic Regression, Random Forest, and SVM) and deep learning models (such as CNN and LSTM) are evaluated on the UCI SMS Spam Collection dataset. Preprocessing of the text and feature extraction methods, including TF-IDF and word embeddings, are used in the study. The findings indicate that deep learning models, especially LSTMs, are not only more accurate but also better suited for building robust smishing detection systems. According to Timko et al. [3], the study evaluated users' employability in distinguishing genuine messages from smishing messages through an online survey involving 187 participants. The implications of the findings suggest that attention and security behavioural scores significantly affect users' accuracy when identifying smishing messages. The author states that fake messages' accuracy was 67.1%, while real messages' accuracy was 43.6%, indicating the difficulty in the matter concerning user awareness. Mahmood and Hameed [4] report an increasing threat of smishing, which involves phishing via SMS messages in the mobile communication context. The study proposes a way to detect smishing that uses a combination of URL inspection by Google Engine and VirusTotal, along with analysis of the SMS content. The research used four machine-learning classifiers to continue classifying messages into legitimate vs smishing, using Support Vector Machine (SVM), Random Forest (RF), Adaptive Boosting (AdaBoost), and Extreme Gradient Boosting (XGBoost). The model thereby attained an accuracy of 98.5%, which is superior to existing methods concerning smishing message detection. Ankit et al. [5] report that Smishing—a growing threat to Cyber Security—deceives users into installing malicious software. The approach is a two-phase machine learning methodology for spam and smishing text detection: first, spam is differentiated from ham, and afterwards, smishing detection is performed on spam. Achieving 96% accuracy with different classifiers, notably neural networks. The study affirms the necessity of unique phishing features and information-gain values for improved quality classification. Sharif et al. [6] note that the detection of suspicious text is a vital topic in cybersecurity, particularly in Bengali language processing. In his studies, he proposed a machine-learning-based model for classifying text into suspicious and non-suspicious. With a collection of 7,000 Bengali text documents, the model, which includes classifiers such as logistic regression and random forest, achieved 84.57% accuracy. His study emphasizes that both unigram and bigram features are significant for enhancing classification performance. Identifying the cyber threats posed by suspicious URLs is vital for internet security. Mohanty et al. [7] are also creating unique models for multi-class ML-based classification to detect spam, phishing, defacement attacks, and malware. A model using classifiers such as decision trees and logistic regression can achieve an accuracy of 94.1% using URL lexical features. It is shown that feature extraction and classification techniques would significantly improve the current state of threat detection. Shinde et al. [8] present a hybrid model that extracts text from images using OCR technology and employs K-means, RNN, LSTM, and Bi-LSTM machine learning and deep learning methods. The RNN-Flatten model achieved the highest accuracy of 94.13%, indicating improved efficiency in detecting both text- and image-based smishing attacks. Schlette et al. [9] argue that Cyber Threat Intelligence (CTI) is important for managing security incident responses and their associated threat data in an organisation. The study evaluates the six CTI formats identified as key concepts for increased efficiency in incident response. This involves discussing playbooks, automation, and the standardisation of the format in cybersecurity defence. The study results show that organisations can improve their response capabilities across multiple CTI formats, thereby helping them better protect against cyberattacks. Chong et al. [10], The need to detect malicious URLs arises because more people use mobile devices, which creates new web security vulnerabilities. The researchers used machine learning techniques that combined URL textual characteristics, JavaScript code elements, and payload dimensions to detect malicious URLs. The system achieved 81% accuracy and 74% F1 score using an SVM with a polynomial kernel. The work confirms that security systems can use real-time URL threat detection to enhance protection through feature extraction methods, including URL pattern recognition and JavaScript obfuscation detection. Jain et al. [11] report that spam and phishing attacks are critical threats in cybersecurity. His work proposes machine-learning-based detection of spam and phishing links. The model uses previously identified phishing websites and spam content to generate new content classifications based on their distinctive attributes. The researchers tested multiple algorithms, including Random Forest and Support Vector Classifier, and found that both methods performed well at identifying security threats. Jaiswal and Raut [12] report that URLs pose a significant cybersecurity risk because they enable scams, data theft, and the distribution of malicious software. The study recommends using machine learning to detect and block URLs. The proposed model improves detection capabilities through web crawling content analysis combined with sentiment analysis methods. Aljabri et al. [13] establish that blacklists fail to detect emerging threats, whereas supervised and deep learning methods succeed in identifying harmful URLs. The research paper addresses two primary issues: insufficient dataset quality and the need for detection methods that can adapt to evolving attack patterns. Tamal et al. [14] demonstrate how machine learning techniques help identify phishing threats by explaining their detection methods and highlighting the challenges of dataset evaluation. The study found that 41 crucial features exist within URLs after researchers examined 274,446 URLs. Rifah et al. [15] note that URL detection is crucial for safeguarding against a range of cyber threats, including

phishing and malware attacks. They propose a machine learning detection framework based on logistic regression to classify URLs as safe or unsafe. The system relies on analysing URL structure and behaviour to effectively identify links that may pose a threat, without using webpage content. This method protects cyberspace via rapid and effective threat detection. Li and Dib [16] propose a new machine-learning architecture for detecting both known and unknown malicious URLs in real time. The system classifies URLs into three major classes, namely phishing, malware, and others, leveraging tree-based algorithms and CL_K-means. It achieves 92.54% accuracy on zero-day attacks, completing a single classification in under 14 milliseconds. Yuan et al. [17] proposed a parallel neural joint model for malicious URL detection. The above system converts URLs into word embeddings and grayscale images, then extracts semantic and visual features and processes them with IndRNN and CapsNet with an attention mechanism. This achieves a very high classification accuracy and outperforms traditional techniques. Xuan et al. [18] propose a machine-learning-based method for detecting malignant URLs. The system extracts features from lexical features, host-based features, and correlated groups to create new URL attributes for training Random Forest, SVM, etc., to categorise the URL as safe or malignant. The outcome proved that the Random Forest Model can achieve high accuracy on larger datasets, which makes this approach not only efficient but also practical. Ravindra et al. [19] note that phishing poses a significant security threat by fooling users into disclosing sensitive information. The study presents a Random Forest-based machine learning detection system that classifies a URL as legitimate or phishing. The system uses URL features, including length, suspicious characters, and subdomains, to enhance its detection performance. The model achieved 86% accuracy after testing 4000 URLs, helping it detect and block phishing attacks. Shoaib et al. [20] present a detection system that implements machine learning and deep learning techniques, along with TF-IDF feature extraction and preprocessing. Deep learning models such as LSTM and Bi-LSTM outperform traditional methods in accuracy while producing fewer false positives. Reyes-Dorta et al. [21] evaluated ML and QML techniques for detecting fraudulent URLs. More than 90% true-positive rates were achieved using classical ML algorithms such as decision trees, logistic regression, and neural networks. Subsequently, QML was explored using methods such as the Variational Quantum Classifier, yielding promising results that appear to match those of classical models. The authors highlight the promise of QML in cybersecurity, albeit limited due to the constraints of current quantum hardware and datasets. Sonowal [22] shows that improving smishing message (SMS phishing detection) can be achieved through feature selection and machine learning. Conducted five different ranking algorithms and concluded that the one with the best performance is Kendall rank correlation with an AdaBoost classifier, which showed the highest accuracy of 98.40% by reducing the feature set to 61.53%. Thus, indicating an efficient way to improve smishing detection. Lee et al. [23] propose a model that uses OCR to extract text from images and implements machine learning and deep learning methods, including RNN, LSTM, and Bi-LSTM. The results demonstrate that deep learning models achieve superior accuracy for complex smishing attacks because RNN-based systems outperform other detection methods. Tabassum et al. [24] The URLs that pose security risks direct users to phishing websites and distribute malware. The traditional blacklisting methods have proved, at best, only marginally effective against new threats. The study investigates machine learning because it needs better accuracy to detect harmful URLs. According to the study results, Random Forest and Neural Networks achieved above 90% accuracy in various tests. About what the research had to offer, it also highlighted the evolving nature of threats and challenges posed by zero-day attacks, as well as proposed adaptations to improve future results. Ghaleb et al. [25] Note That Malicious websites are a major cybersecurity threat, often used for phishing, malware, and fraud. It's really hard to detect new kinds of attacks with old-style detection approaches. In this paper, we present Cyber Threat Intelligence-based Malicious URL Detection (CTI-MURLD), an ensemble-learning model that improves detection accuracy. It extracts URL-based cyber threat intelligence (CTI) and Whois-based features based on a web search. The best classification method improves detection accuracy by using a two-stage approach with Random Forest (RF) and Multilayer Perceptron (MLP). Improvement in accuracy is 7.8%, with a 6.7% decrease in false positives compared with previous methods. Das Gupta et al. [26], An increase in the instances of SMS being used as an avenue for fraud or sending unsolicited messages has made this study relevant. Conventional rule-based detection methods often fail to handle new spam patterns. This study proposes using machine learning to classify SMS messages as legitimate or not. Classifiers, spam or ham, including Random Forest (RF), Naïve Bayes, and Support Vector Machine (SVM), were tested to compare spam-detection accuracy. Results showed that Random Forest had the best accuracy in classifying spam messages. The study shows that spam-detection success is significantly influenced by feature selection and text preprocessing. Palwankar et al. [27] detected malicious links to mitigate phishing and malware attacks. Traditional methods have become a hiccup, as they have been unsuccessful in keeping up with the threats occurring. Bollam et al. [28] propose an NLP-based spam detection model that uses multiple machine learning algorithms and feature extraction techniques. The Support Vector Machine (SVM) model achieved the best results, with 98.49% accuracy, while the ensemble voting classifier achieved 98.6% accuracy. The research demonstrates that machine learning algorithms, when combined with ensemble methods, yield better spam-detection performance. Haritha et al. [1] propose a Phishing-Alarm system that uses visual similarities and colour (hue) features to track phishing websites through an automatically updating database. The system achieves high performance by maintaining complete precision, and its detection abilities have been improved. Manish et al. [2] present a phishing URL detection model that uses data mining and machine learning techniques to analyse the PhishTank dataset. The system implements association rule mining using the Apriori and FP-Tree algorithms, demonstrating that the FP-Tree method provides superior performance and precise

results for phishing URL classification.

III. DATA DESCRIPTION

The SMS Spam Collection dataset, available on GitHub, was used in this study. It is a publicly available dataset comprising approximately 5,574 English text messages, each marked as either “ham” (legitimate message) or “spam” (unwanted or malicious message).

The data is in TSV (tab-separated values) format, with two columns: the class label and the corresponding SMS text content. The SMS messages were taken from actual SMS conversations, so the dataset reflects mobile messaging behaviour in real-world situations very well.

Table I: Summary of the Dataset

Class Label	Description	Number of Messages
Ham	Legitimate SMS messages	4,825
Spam	Smishing / Spam messages	747
Total		**5,572**

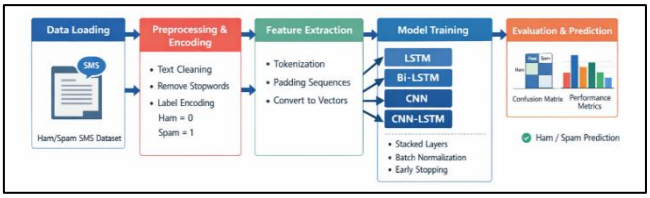
The SMS Spam Collection dataset comprises 5,572 SMS messages, of which 4,825 are marked as ham (legitimate) and 747 as spam. This suggests a discrepancy in the classes, with legitimate messages much more frequent than spam. This kind of imbalance is common in real-world SMS communications. Also, it makes it harder for smishing detection systems to correctly identify minority-class smishing messages, as models have to do so without increasing the rate of false positives.

Table II: Sample Messages Taken from the SMS Spam Collection Dataset

Label	SMS Message
Ham	You don't say so early… You can already then say…
Ham	I'm gonna be home soon, and I don't want to talk about this stuff anymore tonight, k? I’ve cried enough today.
Ham	As per your request, 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all callers.
Ham	Oh k…I'm watching here :)
Spam	Free entry in 2 weekly comps to win FA Cup final tickets on 21st May 2005. Text "FA" to 87121 to receive the entry question (standard text rate). T&Cs apply. 08452810075 over 18s.
Spam	WINNER!! As a valued network customer, you have been selected to receive a £900 prize reward! To claim, call 09061701461. Claim code KL341. Valid 12 hours only.
Spam	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply. Reply HL 4 info.
Spam	URGENT! You have won a 1-week FREE membership in our £100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C…

The dataset comprises ham (legitimate) and spam messages, illustrating the informal character of legitimate conversations and the promotional or scam-like attributes of spam. These cases indicate the preprocessing difficulties, like special symbols, URLs, and mixed-case words, that need to be solved before model training.

IV. METHODOLOGY



[Fig.1: Methodology]

A. Data Preprocessing

The first stage in the SMS spam detection system consists of data preprocessing, which involves cleaning and standardising the textual content. The SMS Spam Collection dataset is imported directly from the web and contains two columns: label (denoting "ham" or "spam") and message (the SMS text). The preprocessing step initially converts all messages to lowercase to ensure uniformity. All URLs are replaced with a "URL" placeholder string to avoid variability from different web addresses. Noise is reduced by removing all non-alphabetic characters, punctuation marks, and numeric values; common English stop words such as “the,” “is”, and “at” are filtered out. This way, only semantically meaningful words remain, allowing the models to focus on relevant textual features. In addition, the categorical labels are encoded as numerical values using label encoding, where ham is marked as 0 and spam as 1, enabling deep learning models to incorporate the labels during training.

B. Feature Extraction

After preprocessing, numerical sequences are generated by feature extraction from the textual data. The tokeniser assigns each word a unique integer, and the 5000 most common words are selected as the vocabulary to improve performance. Each text message is then converted into a number string that refers to the words in it. To make all the inputs the same size for the models, the sequences are extended with zeros to a fixed length of 100 tokens. Tokenisation, combined with sequence padding, provides structured numerical representations of SMS messages suitable for feeding to deep learning architectures. Meanwhile, the data is partitioned into training (80%) and testing (20%) sets, with one subset used by the models for learning and the other for performance evaluation on *previously unseen data*.

C. Deep Learning Models

In this research paper, text messages sent to people can be easily classified as ham or spam using four modern deep learning architectures—LSTM, Bi-LSTM, CNN, and CNN-LSTM. Each model is created to elucidate the characteristics of the textual data and thus help achieve a thorough understanding of the message's content. The Long Short-Term Memory (LSTM) model is one of the most effective models for handling sequential data, as it traces dependencies over time. A multi-stacked LSTM is used, where the first layer feeds its output, in the form of sequences, to the subsequent layer. This type of construction allows the network to capture both the short-term and the long-term simultaneously; for instance, in the case of spammy keywords and phrases that recur within a message. LSTMs are a major

reason text classification has become so effective: they do not lose contextual information even when the sequence is very long, which is often the case when spam hints span multiple words or phrases. The Bidirectional LSTM (Bi-LSTM) extends the basic LSTM by processing the input sequence in both the forward and backward directions. This double processing allows the model to present the conversation's context from both the beginning and the end simultaneously.

The system detects spam signals that appear either at the start of a message, such as "Congratulations, you win," or at the end, such as "click this URL now." The stacked Bi-LSTM layers not only improve the model's ability to differentiate features across levels but also help detect subtle spam characteristics that unidirectional models easily miss. The Convolutional Neural Network (CNN) starts as a system for image recognition and then transforms into a 1D CNN, serving as an effective instrument for text classification. The convolutional layers in this system scan word embeddings to identify local patterns and n-grams, such as "free money" and "call now". A stack of convolutional layers is followed by Batch Normalisation, which stabilises the learning process and reduces internal covariate shift, allowing the network to train faster and better. Global max pooling is applied to capture the most important features across the sequence, thereby reducing dimensionality, although the key spam indicators remain. CNNs are good at spotting repeated keyword patterns and phrases that are characteristic of spam, even if they appear in different parts of the message. The CNN-LSTM model, a hybrid, capitalises on the strengths of both CNN and LSTM architectures. Initially, the convolutional layers extract local features and identify significant n-gram patterns in the embedded word sequences. After this step, the extracted features are passed through an LSTM layer, which captures sequential dependencies and contextual relationships among features across the message. The model exploits local-textual patterns and global-sequence information simultaneously through this hybrid technique; thus, it is highly effective for SMS spam detection, where even individual keywords and their contextual sequences are important. Dropout layers are included between these models to prevent overfitting, while the best-performing model weights are preserved through early stopping during training. The incorporation of dropout layers in all models helps minimise overfitting. At the same time, early stopping is a training process enhancement that halts training if validation loss does not improve for a set number of consecutive epochs and restores the weights of the best-performing model. These architectures are designed to detect spammy words and sequences, such as “free,” “win,” “urgent,” “call now,” “URL,” etc., thereby enabling accurate message classification.

D. Evaluation Metrics

Model performance is evaluated using standard binary classification metrics. The predictions are probability expressions between 0 and 1, with, as a rule of thumb, 0.5 used to classify the messages as ham (the message is likely to be non-spam) or spam (≥0.5). To assess the effectiveness of the suggested SMS spam detection models, an evaluation using standard metrics, including Accuracy, Precision, Recall, and F1 score, was conducted. Accuracy is the metric obtained by dividing the total number of TP and TN cases (correctly classified messages) by the total number of tests performed, and thus giving an overall idea of the model's performance through a simple measure. Mathematically, it is defined as:

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

Thus, (TP) indicates actual positives (spam classified correctly as spam), (TN) indicates actual negatives (ham classified correctly as ham), (FP) indicates false alarms (ham put wrongly in the spam category), and (FN) indicates misses (spam put wrongly in the ham category). Even though accuracy provides an overall assessment, it may not fully reflect the model's performance on skewed datasets, where the number of ham messages is usually much larger than the number of spam messages. Precision is the ratio of correctly identified robotic spam messages to all messages predicted as spam. The model's ability to prevent false alarms serves as a vital assessment metric, protecting authentic messages from being misclassified in real-world field operations.
The formula for precision is:

$$\text{Precision} = \frac{\mathbf{TP}}{\mathbf{TP + FP}}$$

The model measures its performance by recall, the percentage of actual spam messages it successfully identifies. The system achieves high recall by detecting most spam messages, reducing the risk of undetected dangerous messages.
Recall is calculated as:

$$\text{Recall} = \frac{\mathbf{TP}}{\mathbf{TP + FN}}$$

The F1 score provides a comprehensive assessment of model performance by combining precision and recall into a single metric. It is defined as the harmonic mean of precision and recall:

$$\text{F1} - \text{score} = 2 \text{ x } \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-score is of great importance in cases where precision and recall compete, as it provides a single figure that reflects both the errors (false positives and false negatives) in one go. The metrics, when taken together, have provided a thorough assessment of the models, the quality of spam detection, and the reduction in misclassifications, and have measured these aspects properly. To depict the correct and incorrect predictions for ham and spam, visualisation tools such as confusion matrices are used. Meanwhile, bar charts show the performance of allfour models across different metrics.

V. EXPERIMENTAL RESULT

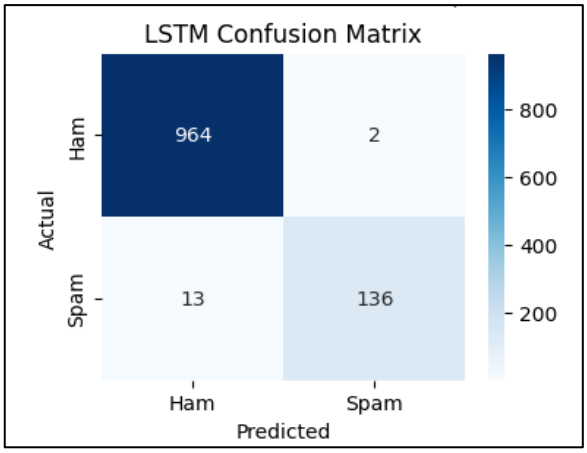
The proposed SMS spam detection framework was evaluated on the SMS Spam

Collection dataset, with 80% of the data used for training and the remaining 20% for testing. Four deep learning models—LSTM, Bi-LSTM, CNN, and CNN-LSTM—were developed and incorporated into the architecture, with stacked layers and batch normalisation (for CNN-based models), in addition to dropout and early stopping to control overfitting. The performance of each model was measured using standard metrics: Accuracy, Precision, Recall and F1-score.

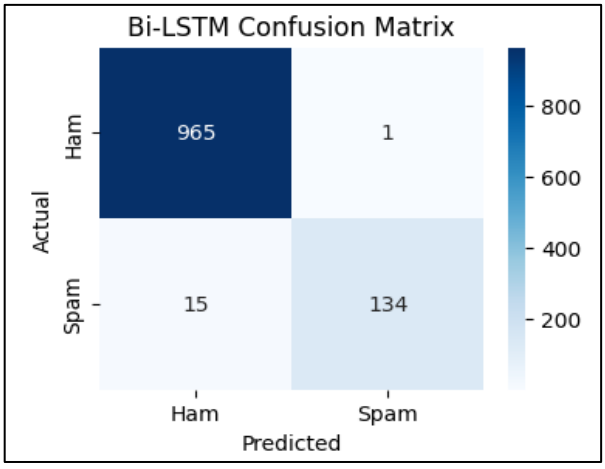
Table III: Summarized Result (in Percentage)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
LSTM	98.65	98.55	91.28	94.77
Bi-LSTM	98.57	99.26	89.93	94.37
CNN	97.58	89.10	93.29	91.15
CNN-LSTM	98.65	99.26	90.60	94.74

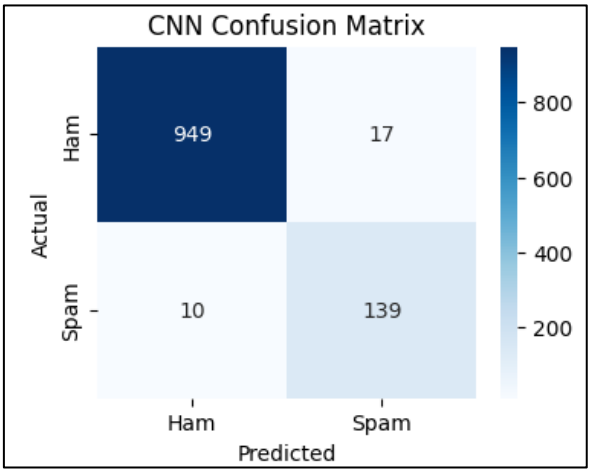
To visualise the results, a comparative performance table and confusion matrices are used. Confusion matrices were produced for all models to provide a graphical depiction of classification performance. The confusion matrices showed that all these models performed very well at classifying 'ham' messages (i.e., normal messages), with nearly all of them yielding no false positives. Spam detection also performed well, although some spam messages were misclassified as 'ham', particularly with the Bi-LSTM and CNN-LSTM models.



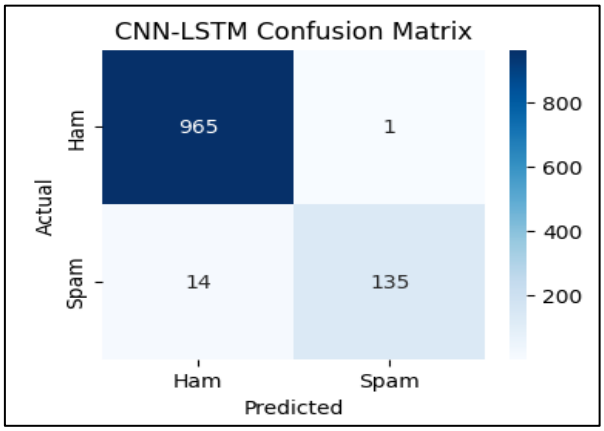
[Fig.2: LSTM Confusion Matrix]



[Fig.3: Bi-LSTM Confusion Matrix]

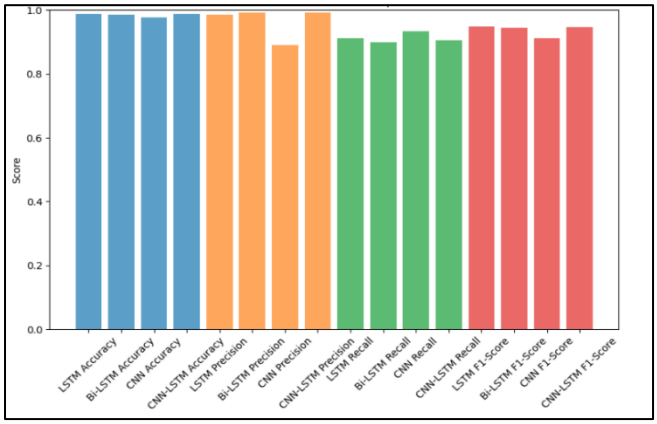


[Fig.4: CNN Confusion Matrix]



[Fig.5: CNN-LSTM Confusion Matrix]

Moreover, a bar graph comparing Accuracy, Precision, Recall, and F1-Score across all models is another way to visualise performance differences. The LSTM and CNN-LSTM models yielded the best accuracy and balanced F1 Scores, whereas the Bi-LSTM model achieved the highest precision but slightly lower recall. The CNN model achieved the highest recall, meaning it captured most spam messages, but at the cost of slightly lower precision.



[Fig.6: Model Performance Comparison]

VI. DISCUSSION

The experimental results demonstrate the advantages and limitations of the deep learning architectures

employed by researchers for SMS spam detection. The LSTM model achieved the highest accuracy and F1 Score because it effectively captured both sequential dependencies and contextual information across the entire message. The Bi-LSTM model processed sequences in both directions, resulting in very high precision for spam detection. Still, it failed to identify some spam messages, causing a small drop in recall. The CNN model demonstrated its strongest performance when identifying spam keywords and local patterns through its convolutional layers, which functioned as n-gram detectors. The system achieved high recall because it detected most actual spam emails, but its lower precision rating showed that it incorrectly classified some genuine messages as spam. CNNs exhibit this behaviour because they do not process long-range dependencies but rather focus on nearby textual patterns. The CNN-LSTM hybrid model combined its two components because convolutional layers captured essential local patterns while LSTM layers processed sequential information. The combination achieved maximum performance by achieving the highest accuracy and F1 Score while maintaining strong precision and recall. The hybrid model demonstrates that local feature extraction, together with sequence modelling, yields effective results for SMS spam detection. Spams can appear through the repetition of phrases or through the use of context-specific sequences. The implementation of stacked layers, batch normalisation, dropout, and early stopping contributed to model stability and effectively mitigated overfitting.

VII. CONCLUSION AND FUTURE WORK

In this study, a detailed, thorough comparative analysis of deep learning models for SMS-based smishing detection was conducted, in which the performance of LSTM, Bi-LSTM, CNN, and CNN-LSTM architectures was evaluated. The outcomes showed that all models could differentiate ham from spam messages with high accuracy, excellent precision, strong recall, and very high F1-scores. The CNN-LSTM hybrid model performed best overall among the assessed architectures. The system achieved this result by effectively merging convolutional and LSTM layers to detect local patterns, comprehend sequential patterns, and capture contextual information. The combined approach effectively captured all main spam indicators, including repeated phrases and keywords, and their temporal and spatial relationships within the messages. To sum up these promising achievements, however, there are still some ways for improvement and research that will be pursued. The use of transformer-based models BERT and RoBERTa represents one potential method for improving text comprehension by enhancing understanding of complex contextual and semantic relationships. Even though transformer models have already proven very effective on various natural language processing tasks, their use may further facilitate the detection of subtle or context-dependent smishing attempts. The other significant step forward is the addition of multilingual smishing datasets that would allow not only detection but also the entire system to operate in different languages, including regional variations, making it more widely applicable. Moreover, the current investigation can be extended by developing real-time smishing detection systems for mobile devices.

DECLARATION STATEMENT

As the article's author, I must verify the accuracy of the following information after aggregating input from all authors.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted objectively and without external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical approval or consent to participate with supporting documentation.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Haritha Rajeev, Midhun Chakkravarthy (2023). Detection of Malware using Phishing Alarm. In Indian Journal of Artificial Intelligence and Neural Networking (Vol. 3, Issue 4, pp. 1–4). DOI: https://doi.org/10.54105/ijainn.a1077.124123
2. Manish Tiwari, Tripti Arjariya (2021). A Phishing URL Classification Technique using Machine Learning Approach. In International Journal of Innovative Technology and Exploring Engineering (Vol. 10, Issue 3, pp. 73–79). DOI: https://doi.org/10.35940/ijitee.c8338.0110321
3. Daniel Timko, Daniel Hernandez Castillo, Muhammad Lutfor Rahman, "A Quantitative Study of SMS Phishing Detection," Unpublished Manuscript (arXiv Preprint), 2024, 16 pages, DOI: https://doi.org/10.48550/arXiv.2311.06911
4. Ameen R. Mahmood, Sarab M. Hameed, "A Smishing Detection Method Based on SMS Contents Analysis and URL Inspection Using Google Engine and VirusTotal," Iraqi Journal of Science, vol. 64, no. 10, 2023, 16 pages, DOI: http://doi.org/10.24996/ijs.2023.64.10.41
5. Ankit Kumar Jain, Sumit Kumar Yadav, Neelam Choudhary, "A Novel Approach to Detect Spam and Smishing SMS using Machine Learning Techniques," International Journal of E-Services and Mobile Applications, vol. 12, no. 1, January-March 2020, 21 pages, DOI: https://doi.org/10.4018/IJESMA.2020010102
6. Sharif Omar, Mohammed Moshiul Hoque, A. S. M. Kayes, Raza Nowrozy, and Iqbal H. Sarker, "Detecting Suspicious Texts using Machine Learning Techniques," Applied Sciences, vol. 10, no. 18, 2022, 23 pages, DOI: https://doi.org/10.3390/app10186527
7. Sanjukta Mohanty, Sourav Nanda, Rupayan Rout, Arpan Kumar, Vansam Agrawal, Arup Abhinna Acharya, Namita Panda, "Detection of Cyber Threats from Suspicious URLs Using Multi-Classification Approach" ResearchGate / Book Chapter, 2024, 14 pages, DOI: http://doi.org/10.4018/979-8-3693-1186-8.ch007
8. Anjali Shinde, Essa Q. Shahra, Shadi Basurra, Faisal Saeed, Abdulrahman A. AlSewari, and Waheb A. Jabbar, "SMS Scam Detection Application Based on Optical Character Recognition for Image Data Using Unsupervised and Deep Semi-Supervised Learning," Sensors 2024 19 pages DOI: https://doi.org/10.3390/s24186084
9. Daniel Schlette, Marco Caselli, and Gunther Pernul, "A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," IEEE Communications Surveys & Tutorials, 2021, DOI: http://doi.org/10.1109/COMST.2021.3117338
10. Christophe Chong, Daniel Liu (Stanford), and Wonhong Lee (Neustar), "Malicious URL Detection", Unspecified publication, 4 pages,

https://cs229.stanford.edu/proj2012/ChongLiu MaliciousURLDetection.pdf
11. Ms Shilpi Jain, Dr Madhur Jain, Ridhi Kalia, Divyansh Rampal, "A Comprehensive Model for Spam Detection and Phishing Link Detection," International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 10, no. 3, 2024, 5 pages, DOI: http://doi.org/10.32628/CSEIT24103109
12. Muskaan V. Jaiswal and Anjali B. Raut, "Detecting and Blocking of Malicious URL," International Journal of Science and Research (IJSR), vol. 10, no. 6, 2021, 3 pages, DOI: http://doi.org/10.21275/SR21610230148
13. Malak Aljabri; Hanan S. Altamimi, Shahd A. Albelali, Maimunah Al Harbi, Haya T. Alhuraib, Najd K. Alotaibi, "Detecting Malicious URLs Detection Using Machine Learning Techniques: Review and Research Directions," IEEE Access, vol. 10, 2022, 23 pages, DOI: http://doi.org/10.1109/ACCESS.2022.3222307
14. Maruf A. Tamal, Md K. Islam, Touhid Bhuiyan, Abdus Sattar, Nayem Uddin Prince, "Unveiling Suspicious Phishing Attacks: Enhancing Detection with an Optimal Feature Vectorization Algorithm and Supervised Machine Learning," Frontiers in Computer Science, vol. 6, no. 1428013, 2024, 16 pages, DOI: http://doi.org/10.3389/fcomp.2024.1428013
15. Amar Palwankar, Rifah Solkar, Afiya Borkar, Shreya Khedaskar, and Pranali Shingare, "Malicious Link Detection System," International Research Journal of Engineering and Technology (IRJET), vol. 9, no. 11, 2022, 5 pages, https://www.irjet.net/archives/V9/i11/IRJET V9I1165.pdf
16. Shiyun Li and Omar Dib, "Enhancing Online Security: A Novel Machine Learning Framework for Robust Detection of Known and Unknown Malicious URLs," Journal of Theoretical and Applied Electronic Commerce Research, vol. 19, no. 4, 2024, 42 pages, DOI: https://doi.org/10.3390/jtaer19040141
17. Yuan Jianting, Chen Guanxin, Tian Shengwei, Pei Xinjun, "Malicious URL Detection Based on a Parallel Neural Joint Model," IEEE Access, vol. 9, 2021, 9 pages, DOI: http://doi.org/10.1109/ACCESS.2021.3049625
18. Cho Do Xuan, Hoa Dinh Nguyen, Tisenko Victor Nikolaevich, "Malicious URL Detection Based on Machine Learning," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 1, 2020, 6 pages, DOI: http://dx.doi.org/10.14569/IJACSA.2020.0110119
19. Salvi Siddhi Ravindra, Shah Juhi Sanjay, Shaikh Nausheenbanu Ahmed Gulzar, Khodke Pallavi, " Phishing Website Detection Based on URL," IJSRCSEIT, vol. 7, no. 3, 2021, 6 pages, DOI: https://doi.org/10.32628/CSEIT2173124
20. Mohd Shoaib, Mohammad Sarosh Umar, " An investigation in detection and mitigation of smishing using machine learning techniques," Springer Nature Link vol 13, article 135, 2023, 15 pages, DOI: https://doi.org/10.1007/s13278-023-01142-4
21. Nuria Reyes-Dorta, Pino Caballero-Gil, Carlos Rosa-Remedios, "Detection of Malicious URLs Using Machine Learning," Wireless Networks, vol. 30, 2024, 18 pages, DOI: https://doi.org/10.1007/s11276-024-03700-w
22. Gunikhan Sonowal, "Detecting Phishing SMS Based on Multiple Correlation Algorithms," SN Computer Science, vol. 1, no. 361, 2020, 9 pages, DOI: https://doi.org/10.1007/s42979-020-00377-8
23. Hwabin Lee, Sua Jeong, Seogyeong Cho, and Eunjung Choi, "Visualisation Technology and Deep-Learning for Multilingual Spam Message Detection," *Electronics*, 2023, Vol. 12, Article 582, 17 pages, DOI: https://doi.org/10.3390/electronics12030582
24. Tasfia Tabassum, Md. Mahbubul Alam, Md. Sabbir Ejaz, Mohammad Kamrul Hasan, "A Review on Malicious URLs Detection Using Machine Learning Methods," Journal of Engineering Research and Reports, vol. 25, no. 12, 2023, 13 pages, DOI: http://doi.org/10.9734/JERR/2023/v25i121042
25. Fuad A. Ghaleb, Mohammed Alsaedi, Faisal Saeed, Jawad Ahmad, Mohammed Alasli, "Cyber Threat Intelligence-Based Malicious URL Detection Model Using Ensemble Learning," Sensors, vol. 22, no. 9, 2022, 19 pages, DOI: https://doi.org/10.3390/s22093373
26. Suparna Das Gupta et al., "SMS Spam Detection Using Machine Learning," Journal of Physics: Conference Series, vol. 1797, no. 1, 2021, 6 pages, DOI: http://doi.org/10.1088/1742-6596/1797/1/012017
27. Prof. Amar Palwankar, Afiya Borkar, Pranali Shingare, Rifah Solkar, Shreya Khedaskar, "Suspicious Link Detection Using AI," International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), vol. 3, no. 3, 2023, 8 pages, DOI: http://doi.org/10.48175/IJARSCT-9171
28. Bollam Pragna, M. Rama Bai, Spam Detection using NLP Techniques. (2019). In International Journal of Recent Technology and Engineering (Vol. 8, Issue 2S11, pp. 2423–2426).

DOI: https://doi.org/10.35940/ijrte.b1280.0982s1119

AUTHOR'S PROFILE

Aqsa Shaikh is a student in the M.S. (Cyber Security) program at the University Department of Information Technology, University of Mumbai. She is proficient in programming languages such as Java and Python and is familiar with software development principles. The combination of her interest in cybersecurity and the groundwork laid by her hands-on experience with tools such as Wireshark and Nmap led her to pursue a master's degree in cyberattacks and detection techniques. Her specialization includes Cryptography, Network Security, and Information Security. She has been introduced to concepts in security analysis, risk management, and vulnerability assessment. She has already published one research paper related to cybersecurity. Now, her research focuses on Smishing Detection, the practice of identifying and preventing SMS-based phishing attacks.

Mariya Shaikh, a postgraduate student, is pursuing her M.S. in Cybersecurity at the University Department of Information Technology, University of Mumbai, India. She has completed her bachelor's degree in Computer Science with a CGPA of 9.17 (A+). She has gained practical knowledge with internships in Cybersecurity, SOC operations, Digital Forensics, and OSINT. She has attended well-recognised industry training from Microsoft and Cisco and is currently holding ISC2 Candidate status. She has a TryHackMe ranking of Top 5%. Her deep interest in cybersecurity, together with practical experience with tools such as Wireshark and Nmap, has led her to consider further studies in cyberattack detection and analysis. Her interests cover Network Security, Information Security, Digital Forensics, and OSINT. Her research interests include SMS phishing detection, machine-learning-based security solutions, and cyber threat analysis.

Srivaramangai R., Head, Department of IT, University of Mumbai, India. Having 24 years of experience in teaching and 6 years in industry. The specialization areas are artificial intelligence, security, image processing. Has industry experience in web development and report code generators. Has published more than 35 International journal papers, 25 conference papers, and served as a resource person for various workshops and chaired sessions. The papers relevant to Cyber Security include "Assessment of Deep Packet Inspection System of Network traffic and Anomaly Detection", Enhancing Security using ECC in Cloud Storage", "Recapitulation of the Use of Machine Learning for Prevention of DDoS Attack on SDN Controller" and "Unmasking Deceptive Websites: Harnessing Machine Learning For Phishing Detection".