P-ISSN 1991-8941
E-ISSN 2706-6703
Vol. 20, No. 1 (2026)

Journal of University of Anbar for Pure Science
(JUAPS)

journal homepage: juaps.uoanbar.edu.iq

Open Access

Original Paper

A hybrid stacked bidirectional long short-term memory and Bidirectional Encoder Representations from Transformers model for enhanced phishing email detection

Marwan B. Mohammed [1*]

[1]Department of Computer Science, College of Science, Al-Nahrain University, Baghdad, Iraq

ARTICLE INFO

Received: 16/04/2025
Accepted: 17/08/2025
Available online: 24/03/2026
April Issue
10.37652/juaps.2025.158955.1369

Cite @ JUAPS

Corresponding author
Marwan B. Mohammed
marwan.badran@nahrainuniv.edu.iq

ABSTRACT

Phishing attacks have been among the latest cybersecurity threats as they exploit people's weaknesses and compromise critical information. Traditional methodologies of phishing detection practices based on machine learning have generally been unsuccessful in offering high performance due to limited generalizability and very high false positive rates. In this paper, a novel hybrid deep learning model is proposed that incorporates Bidirectional Encoding Representation from Transformers (BERT) with Bidirectional Long Short-Term Memory (BLSTM) to increase the detection of suspicious emails within the framework of the proposed model. The model utilizes the contextual understanding of texts from BERT and sequential learning from BLSTM to enhance classification accuracy. Experimental results using an available phishing email dataset show that the proposed approach significantly outperforms the baseline models in F1-score, recall, and precision metrics. Those results highlight the advantages of combining transformer-based embedding and recurrent architectures in phishing detection, providing avenues towards a more robust framework for email security enhancement.

Keywords: *BERT, BLSTM, Cybersecurity, Dataset, Phishing email detection*

1 INTRODUCTION

In recent years, the number of people who have faced cyber-attacks has increased. Many cases are related to the phishing attack in which hackers have utilized phishing email attacks to deceive individuals from government organizations, websites from social organizations and politicians, and reputable companies [1]. These types of attacks are increasing dramatically in volume, and they are starting to use more sophisticated approaches. Such attacks have a destructive effect on big companies, as they could cause leakage of personal data, resulting in an adverse effect on countries or industries [2]. In such attacks, the attacker applies social engineering, in contrast to other types of attacks in which hackers use software or take advantage of certain protocols. First, the attacker sends an email that contains links, a request to enter a password, or clicking on links without having knowledge that this action could be harmful [3]. As this attack does not require a substantial amount of technical expenses and zero-day vulnerabilities, it can be more straightforward to pass the security mechanisms [4]. On the other hand, it could be challenging to prevent this attack by following only certain rules or gaining holistic security. Simultaneously, due to the fact that email is the prevalent way of formal communication, phishing attacks' popularity is also growing. As a result, the conventional techniques have failed to decide whether the emails are phishing emails or not [5]. This kind of attack is considered one of the main attacks to gain access to personal, companies, and government data, because there is no direct solution for this attack [6]. For the reasons mentioned above, many researchers have struggled to develop a system

that detects and prevents such attacks. Many systems have been proposed to prevent such attacks, including filtering botnet email, linguistic attribution, sandbox, sender reputation analysis, and behavior blacklist [5]. Those methods fail to combat the sophisticated phishing attack, which requires considering many aspects, such as coping measures, social engineering, consciousness, and psychology. This kind of system is typically used for attacks that lead victims to malicious websites via links, and they are challenging to apply for activity tracking and recognition [7]. One of the most effective ways to cope with these attacks is machine learning based models. Nevertheless, it could be challenging to implement them to resolve such attacks. On the other hand, phishing emails can be divided into varied classes derived from its pattern or technique, like copying IP, disguising as a public domain name, and using short links, each of which has its own features [8]. Whereas the models described above can identify phishing emails, in some cases, such as cloud attachment and forgery, systems like sandbox cannot prevent these attacks properly [9]. Furthermore, there is a huge difference between datasets that are based on real-world phishing or just created for testing phishing email detection and are synthetic [10]. Therefore, in this paper, a new approach is suggested that relies on a deep learning architecture based on a stacked bidirectional long-short term memory enhanced by a BERT layer for phishing email detection. This model uses the capabilities of bidirectional LSTM to extract contextual information from the future and the past sequences, while the Bidirectional Encoder Representations from Transformers benefits the model by focusing on the most important features of the email's content, which increases the model's accuracy significantly. This model is situated in the domain of natural language processing for creating security systems for the detection and prevention of phishing emails.

The main contribution of this paper is as follows:

- Introducing a new deep learning architecture that includes layers such as BERT and Stacked BLSTM.
- Utilizing the deep learning method to identify suspicious emails.
- Comparing the proposed model with other methods and different criteria.
- Integrating a comprehensive data preprocessing pipeline tailored for phishing email characteristics, which significantly enhances model robustness.
- All hyperparameters, including the optimizer choice and learning rate, were systematically optimized using the Optuna hyperparameter tuning framework, enabling superior performance compared to manually tuning or default settings.

A brief outline of the structure of this paper is as follows. Section 2 discusses related research in the area of phishing detection. In Section 3, the suggested method is elaborated. In Section 4, the efficiency of the suggested method is evaluated. Finally, Section 5 discusses the future work and concluding remarks.

2 LITERATURE REVIEW

Nowadays, phishing attacks have been occurring increasingly, making them one of the most important types of attacks. Numerous models have been introduced to date, and conventional methods have generally struggled with the dynamic nature of such attacks, leading to the advancement of more efficient systems like deep learning. Hybrid deep learning architectures have demonstrated encouraging results by dynamically finding phishing emails, although challenges such as efficiency and accuracy remain.

Many studies have reviewed and compared their current models and systems, identifying their advantages and disadvantages. However, most authors mainly focus on reviews based on NLP applications for phishing detection models. The authors in [11] covered over a hundred papers published between 2006 and 2022, along with their benefits and drawbacks. A wide variety of key studies were analyzed, addressing different aspects of phishing attacks, such as models utilizing machine learning, NLP applications, resources and datasets, text features, and other criteria. The authors concluded that feature selection, extraction, and their clustering methods for identifying phishing emails are key areas of focus. They also mentioned that one of the methods frequently used in this domain is Support Vector Machines (SVM), with word embeddings and TF-IDF being other commonly applied methods. Moreover, they found that the most widely used phishing dataset is the Nazario phishing corpus, while Python is the most popular programming language for developing suspicious email recognition models. Additionally, studies with publicly available resources and tools were thoroughly reviewed. They observed that the Arabic language is the least studied in terms of developing phishing email detection systems

As email is one of the most used means of com-

munication, it has faced serious threats from phishing emails. At first glance, they may look legitimate, but they try to gain access to the computer's data. Many statistical-based models have been introduced to tackle such attacks. In one study [12], this research utilized distributional representation. Also, many machine learning methods, including DT, AdaBoost, Naive Bayes, SVM, and Random Forest, were evaluated in this paper and then compared to the model. One of the studies that utilized machine learning was a model that suggested an ensemble learning model for suspicious email detection [13]. It was named HELPED, responsible for detecting phishing emails via ensemble learning based on hybrid features. In this model, hybrid features make it possible to accurately represent emails. In this paper, two different models were proposed, with the first one using the Stacking Ensemble Learning method, whereas the other one applied the Soft Voting Ensemble Learning. Each model proposed various machine learning methods to address the fusion of features individually. However, its accuracy increased by decreasing the intricacy of the features. The experimental evaluation conducted in this paper illustrated that it has relatively good performance compared to other methods. In another research [14], the authors proposed three novel machine learning-based models, each with its own cues, namely loss persuasion cues, loss persuasion cues, and combined gain. Then, these models were examined and compared with the base system. The outcome of this research demonstrated that all three models performed significantly better than the base model in F-score criteria, with improvements ranging from 5 percent to 20 percent.

Machine learning methods have been widely applied for phishing email identification, and the SVM is a commonly used approach for attack detection. Nevertheless, adjusting the SVM kernel parameter can be challenging. In [15], a hybrid model that utilized the Cuckoo Search algorithm and SVM was developed. In this model, 23 features were retrieved, and Cuckoo Search was integrated to optimize the parameters of the Radial Basis Function. Experimental evaluation using a dataset comprising 20,071 legitimate emails and 1,384 phishing emails showed that their method can achieve high accuracy.

Properly analyzing the context of the email for model creation can be beneficial. In [16], the structure of the emails was evaluated in the first step, then a model applying an RNN integrating an attention mechanism and multilevel vectors was proposed, named THEMIS. THEMIS was applied to model emails at different levels, such as the email context, email header, word stage, and character stage. An unbalanced dataset was utilized to analyze the model's efficiency, and experimental evaluation revealed that the THEMIS model could achieve good accuracy. However, dimensionality reduction can be useful when there is a high dimensional data, such as text, through feature selection and extraction techniques. In [17], various dimension reduction methods were tested to identify which one is the best for contexts like emails. Identifying and classifying emails can be challenging; therefore, dimension reduction can be utilized to retain the most discriminative and informative features. To evaluate the models, two feature extraction methods, namely Latent Semantic Analysis and Principal Component Analysis, and two feature extraction models, such as Information Gain Ratio and Chi-Square, were utilized. The authors declared that feature extraction techniques were more suitable for boosting the classification accuracy.

Deep learning is also popular for phishing email detection models. Hybrid deep learning architectures can be noted in previous research as commonly robust models with high detection accuracy. In [18], a hybrid deep learning architecture based on different layers, such as RNN, CNN, BERT, and LSTM, was proposed. The authors utilized NLP to collect a set of relevant features. The numerical results illustrated that the combined approach could achieve high accuracy. In [19], the authors also analyzed the efficiency of deep learning and machine learning methods. In this study, phishing emails were considered spam emails. The outcomes of several classification techniques and the evaluation of these outcomes for detecting phishing emails by utilizing deep learning and machine learning were illustrated. In [20], the authors demonstrated how to distinguish between phishing emails and trusted emails. Two types of datasets were used, which included contexts with and without a header. In addition, to develop the phishing detection model, Convolutional Neural Network and Keras Word Embedding methods were utilized.

Phishing is a widespread cyber threat exploiting malicious URLs, emails, and websites. Recent studies using large datasets show that advanced machine learning models improve detection accuracy. Among these, a BERT-LSTM hybrid model [21] achieved the highest performance, outperforming traditional methods like SVM and Naive Bayes. The model also illustrates solid generalization with minimal overfitting, indicating its promise for real-time dangerous email detection. Another research [18] explores deep learning models, including

CNN, LSTM, RNN, and BERT, applied to phishing email detection using NLP-extracted features. Among these, a hybrid BERT-LSTM model obtained the best accuracy. These findings highlight deep learning's effectiveness in enhancing phishing detection and defense.

Protecting wireless communications from interference is crucial because of the rising volume of data transmission and rising cybersecurity threats like phishing attacks. Recent studies have leveraged deep learning methods, combining CNN and recurrent units, to enhance real-time phishing detection and digital forensics. An approach in [22] employs a novel ResNeXt-based model with embedded GRU (RNT), improved by SMOTE for data imbalance and feature selection using ResNet (EARN) and autoencoders. Optimized via the Jaya algorithm, the RNT method outperforms many methods by 11% to 19% in accuracy, achieving 98% accuracy with efficient processing times. Traditional phishing detection methods have primarily relied on machine learning models enhanced by handcrafted, domain-specific feature engineering. More recently, research has shifted towards deep learning-based, end-to-end models that automatically extract features without manual intervention. However, many of these approaches are heavily dependent on large, specific datasets and lack extensive validation across diverse real-world scenarios, which limits their practical applicability. To address these challenges, one study [23] proposed a multimodal suspicious URL detection framework integrating a fine-tuned BERT method for URL analysis, incorporating additional external features sourced from public Internet sources. This study also introduced the PhishMail dataset, containing 8,937 phishing samples collected from real-world email traffic, and demonstrated improved detection effectiveness and generalization through extensive cross-dataset evaluations. Another study [24] introduced PhishGuard, an Android-based application aimed at combating the increasing prevalence of voice phishing scams in India. The system runs on the Android OS stage to detect, notify, and prevent suspicious behaviors instantly by leveraging speech recognition and monitoring runtime misuse of Android APIs. When potential call redirection or phishing behavior is detected, the application issues active warning alerts to prevent further exploitation. By focusing on runtime monitoring and adaptive detection, PhishGuard offers a robust defense mechanism against evolving phishing techniques that circumvent conventional security solutions. This approach demonstrates significant potential for safeguarding individuals and organizations from advanced social engineering attacks.

3 MATERIALS AND METHODS

3.1 The stacked auto-encoder

The stacked neural network [25] is a network with multiple autoencoders. In this model, the output of a layer is connected to the next layer as its input. In the case of a single-layer autoencoder, it can be considered a supervised learning model with three layers: the output layer, the hidden layer, and the input layer. The input data is converted into a hidden representation by the encoder, while the decoder retrieves it from this hidden layer. This procedure can be mathematically represented as follows,

$$h_n = f\left(W_1 x_n + b_1\right) \tag{1}$$

$$x'_n = g\left(W_2 h_n + b_2\right) \tag{2}$$

In (1), f denotes the encoding function, while g represents the decoding function. In (1) and (2), w represents the weight matrix, while the bias vector is represented by b. The encoded value is represented by h_n, the initial value by x_n, the bias of the decoding function by b_2, and the bias of the encoding function by b_1. Training a single-layer autoencoder aims to extract feature representations while reducing the difference between the original and reconstructed input. The fitness function for minimizing the error can be calculated as

$$\arg\min_{w,b}[\ \mathrm{J}] = \arg\min_{w,b} \frac{1}{n} \sum_{i=1}^{n} L\left(x^i, x'^i\right) \tag{3}$$

In (3), L represents the loss function, which can be modeled by $L\left(x, x'\right) = \|x - x'\|^2$. A stacked autoencoder can be created by organizing a progression of single-layer autoencoders. In such a neural network, the output of one layer is arranged so that it serves as the input to the next one. Greedy techniques are used to train each layer in this type of algorithm. In this method, hidden features in the initial layer are considered as the input for the next AE layer. This procedure continues for the following layers.

3.2 The bidirectional LSTM

RNNs are a kind of NN that utilize a series of connected nodes to build a directed graph. RNN can evaluate time-based dynamic behavior for a time series. Nevertheless, LSTM was proposed in 1997 [26], and is considered a type of recurrent neural network. This approach mitigates the vanishing gradient issue, where

it is relatively impossible for a regular recurrent neural network to absorb long-term events. It is constructed with different units, including a forget gate, an output gate, and a memory cell. This neural network applies a function that is hidden within every gate and layer. The memory cell condition is safeguarded through the latent function. Mathematically,

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \tag{4}$$

$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + b_i\right) \tag{5}$$

$$o_t = \sigma\left(W_{co}c_t + W_{ho}h_{t-1} + b_o\right) \tag{6}$$

$$\tilde{c}_t = \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \tag{7}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \tag{8}$$

$$h_t = o_t\sigma\left(c_t\right) \tag{9}$$

In (4) to (9), the o_t represents the activation output gate, i_t and f_t signify the activations of the forget and input at the t time step. Hence, it can be controlled by the proportion of the input, and the prior state will be determined, and the amount of the cell will be integrated in the activation, which is hidden in the network. Although x_t and h_t denote the input and output vectors, $\tilde{c}_t$ and c_t represent the potential condition, the safeguarded memory state array in the LSTM unit. Moreover, b and W denote the bias and weight matrices, respectively. The bias and weight matrices represent the trainable parameters that we need to adjust to minimize the loss function. To increase the amount of data accessible within the network, a new module, namely bidirectional, was introduced in [27]. Whereas the conventional LSTM absorbs representation from prior information, BLSTM is able to access data from both the future and the past at the same time. Fig. 1 illustrates two hidden layers moving in reverse directions, both linked to a common output.

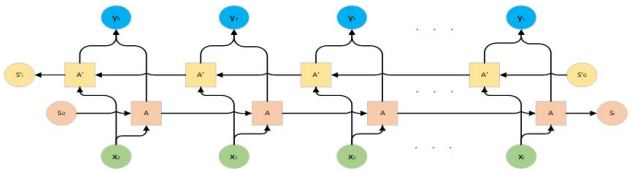

Fig. 1 Bidirectional neural network

3.3 The bidirectional encoder representations from transformers (BERT)

Different types of word embedding methods, like Word2Vec [28] and GloVe [29], have been utilized within sequence tagging frameworks to boost their efficiency. Nevertheless, as many pre-trained language models are being proposed these days, numerous drawbacks, in contrast to the contextualized word embeddings, are illustrated by traditional word embeddings. One of the recent developments in contextualized word representations is BERT. This method is capable of mitigating this issue significantly. Therefore, the proposed method can feed contextualized word embeddings into the input of the subsequent BLSTM layer by utilizing BERT. This mechanism is constructed with multiple layers of transformers. On the other hand, each transformer is made up of a self-attention sub-layer that has several attention heads. Thus, by comparing each pair of input elements, the self-attention scores can be calculated mathematically using (10).

$$e_{ij} = \frac{\left(h_i W^Q\right)\left(h_i W^K\right)^T}{\sqrt{d_z}} \tag{10}$$

The sum of input elements transformed through linear weighting in every output of a self-attention subcomponent can be calculated by (11).

$$z_i = \sum_{j=1}^{N} \frac{\exp e_{ij}}{\sum_{k=1}^{N} \exp e_{ik}} \left(h_j W^V\right) \tag{11}$$

In (10) and (11), h_i denotes the result of the preceding self-attention subcomponent, d_z represents the dimension of the output, and W^K, W^Q, and W^V are the parameters of the BERT.

4 PROPOSED METHOD

In this paper, a new hybrid deep learning framework is devised that includes a Stacked BLSTM layer with BERT. This model is utilized to boost text classification for phishing email detection. This model uses BERT's contextualized embeddings, whereas the main aim of incorporating BLSTM is to capture long-range dependencies in textual data due to the BLSTM's ability in sequential processing. In the first step, the input raw textual data is processed by applying the BERT tokenizer to convert it into tokenized representations. The data is then transferred via a pre-trained BERT encoder. This

encoder generates a pooled output with a fixed-length embedding of size 768. In the next step, the features are passed through a stacked BLSTM network. The first BLSTM layer has a certain number of units, which are identified by the Optuna optimizer and configured to return sequences. This method allows for temporal feature extraction among the sequences. The second BLSTM layer will process the sequential features to return a fixed-length output. To reduce overfitting, we apply a dropout layer with a rate of 0.4 after both BLSTM layers. Finally, a fully connected layer identified by Optuna and a ReLU activation is applied. This layer is used to modify the extracted features for classification. The output of the model consists of a neuron with a sigmoid activation function. This function enables binary classification as either phishing or legitimate. The model undergoes end-to-end training where the BERT encoder and BLSTM layers are updated via backpropagation to maximize classification performance. The general flow of the suggested solution is demonstrated in Fig.2.

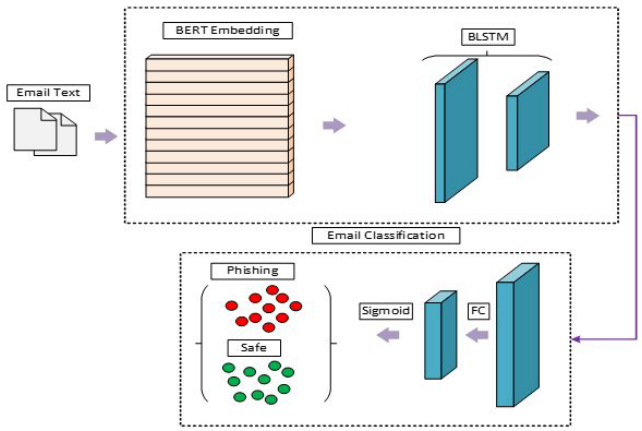

Fig. 2 Overall structure of the proposed method.

5 RESULTS AND DISCUSSION

5.1 System specification

All experimental evaluations in this paper were developed using the Python programming language, and all experimental evaluations were conducted on a laptop with 24GB RAM, an Intel Core i7-13650HX CPU, and an RTX 4050 graphics card.

5.2 Parameter settings

All hyperparameters in our model were optimized using the Optuna hyperparameter tuning framework to ensure optimal performance. The tuning process included the selection of the optimizer, learning rate, dropout rates, and the count of units in the BLSTM layers. As a result of this process, the Adam optimizer was selected, with an optimal learning rate of 2.2658×10^{-5}. The optimal dropout rates were determined to be 0.1211 and 0.413365, while the BLSTM layers were configured with 189 and 47 units, respectively. These optimized settings were applied throughout all experiments to ensure robust and reproducible results.

The method was trained for 10 epochs with a batch size of 32 for both validation and training. The maximum token sequence length was set to 110. To ensure balanced class representation, the dataset was divided using stratified sampling, eighty percent for training, 16% for validation, and 4% for testing. These optimized and standardized parameter settings were consistently applied throughout all experiments to ensure robust and reproducible results. Class imbalance was addressed using stratified sampling during the train-validation-test split, ensuring balanced class distribution across all subsets. Additionally, mini-batching and shuffling were applied during training to further maintain balanced representation and mitigate bias toward the majority class.

5.3 Dataset description

To analyze the suggested solution, we applied the phishing emails dataset accessible on the Kaggle website (https://www.kaggle.com/datasets/subhajournal/phishingemails). This dataset contains a total of 18,101 instances, which are separated into two subcategories: phishing emails and legitimate emails. 61% of this dataset consists of legitimate emails, while the rest are phishing emails. Datasets with empty email content were removed during the preprocessing stage, and the emails were labeled as either phishing or safe.

5.4 Competitor models

To compare the results obtained from the proposed method, we selected four deep learning architectures: CLSTM [30], RCNN [31], CNN [32], and LSTM [33]. These models show the ability to handle both temporal and spatial dependencies, making them great competitors against the proposed method.

5.5 Evaluation metrics

To assess the efficiency of the proposed approach and the other models, we used a variety of metrics, one of

which is the accuracy, which is described as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

Where TP is the true positives, TN is the true negatives, whereas FP and FN represent false positives and false negatives, respectively. The second metric is the Precision which is represented as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{13}$$

The third metric is the Recall

$$\text{Recall} = \frac{TP}{TP + FN} \tag{14}$$

The fourth metric is the F1, which is mathematically calculated as:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

5.6 Numerical results

In this subsection, a comprehensive experimental evaluation will be carried out to evaluate how well the proposed approach performs. In Fig. 3, the convergence curve obtained by the proposed method is illustrated.

The left diagram demonstrates the loss function versus the number of epochs, which shows good performance in optimization. In this diagram, minimal overfitting can be seen by both of the curves. This behavior is essential for creating a powerful model. On the other hand, while the binary accuracy was chosen as the objective function for training, the pattern in this diagram shows a steady improvement in the classification task. Although the validation accuracy is relatively high even in the first epoch, there was a dramatic rise in the training accuracy. This suggests that the solution is capable of adapting to the data rapidly using the representational capabilities of the stacked BLSTM and BERT layers. The slight fluctuations in the validation accuracy might result from variability in the dataset. However, it can be concluded that the overall trajectory represents stable learning. In Table 1, results achieved by different initial conditions are illustrated in which the effects of the L2 Regularization and Dropout are evaluated.

Table 1 analyzes the proposed method's performance with different initial conditions for phishing email detection. The proposed method with both L2 regularization and dropout achieves the highest accuracy, precision, recall, and F1 score. The model incorporating L2 regularization and dropout obtains an accuracy of 96.96%, outperforming other configurations. L2 regularization and dropout also achieve the highest precision, recall, and F1 score, at 0.9385, 0.9871, and 0.9622, respectively. This shows that the model is able to balance both false positives and false negatives effectively. Moreover, the proposed method with only dropout exhibits relatively lower accuracy at 95.95%, with a precision of 0.9262, a recall of 0.9741, and an F1 score of 0.9496. This reveals that while dropout alone can improve the performance of the proposed method, L2 regularization is also a valuable option for increasing efficiency. On the other hand, the lowest accuracy and other metrics, including precision, recall, and F1 score, belong to the method with only L2 regularization among the three configurations, achieving an accuracy of 95.28%. While this model performs well with high accuracy and recall of 0.9655 and an F1 score of 0.9412, its performance is relatively lower compared to the other configurations. The outcomes in the table show that integrating L2 regularization and dropout can improve the efficiency of the suggested model for phishing email detection, making their incorporation necessary.

Table 2 compares various models across several parameters of classification performance showing a detailed statistical description of deep learning classification models such as CNN, LSTM, CLSTM, RCNN-based, and the proposed method for suspicious email identification. The results reveal that the proposed method outperforms all of the benchmarks in precision, recall, accuracy, and F1 score, making it the best among existing methods. Performance improved significantly due to the addition of stacked bidirectional LSTM (BLSTM) and the inclusion of BERT layers in the suggested method.

The proposed method achieves the maximum accuracy of 96.96%, which is much higher than that of CNN (93.57%), LSTM (94.08%), CLSTM (94.94%), and RCNN (94.92%). The improved performance can also be observed in the F1 scores, recall, and precision. To be specific, the proposed method achieves a precision of 0.9385, a recall of 0.9871, and an F1 score of 0.9622, all of these demonstrating its capability to effectively reduce FP and FN. In comparison to this, the CNN ranked the lowest among the existing methods with an accuracy of 93.57% and an F1 score of 0.9198. The LSTM model slightly outperforms CNN with respect to the recall, with a value of 0.9563, a result indicating a certain strength in processing sequential dependencies. Meanwhile, both the CLSTM and RCNN models showed competitive

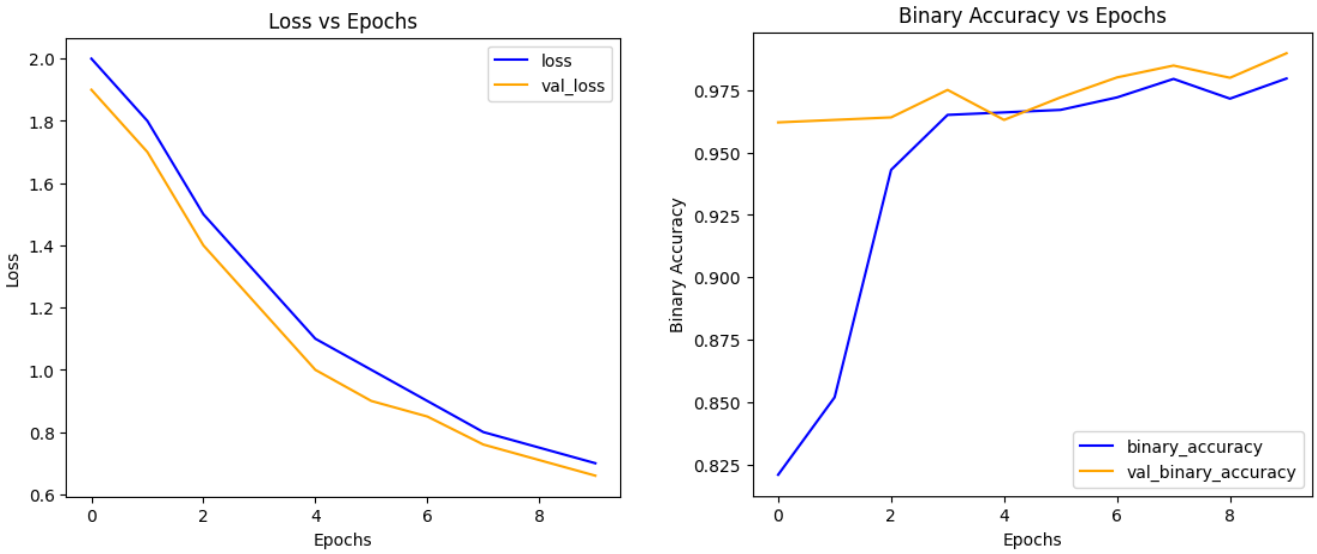

Fig. 3 Convergence curve: Analyzing loss and binary accuracy of various models during training

Table 1 Effectiveness comparison with different initial conditions.

Model	Accuracy	Precision	Recall	F1 Score
Proposed Method with only L2 Regularization	0.9528	0.9180	0.9655	0.9412
Proposed Method with only Dropout	0.9595	0.9262	0.9741	0.9496
Proposed Method with L2 and Dropout	0.9696	0.9385	0.9871	0.9622

results regarding their accuracies (94.94% and 94.92%, respectively), and F1 scores (0.9370 and 0.9367). They still do not compete well with the hybridized framework employed in the proposed method. Furthermore, the significant improvement of the proposed method was due to its capability to capture both local and global dependencies in text data through deep learning architectures. Thus, generalizing complicated patterns in phishing emails and achieving robustness in classification are accomplished through the combination of BERT contextualized embeddings with the sequential processing capabilities of BLSTM. Moreover, adding dropout and L2 regularization to the overall model enhances generalization and reduces overfitting.

Table 2 Comparison of classification results.

Model	Accuracy	Precision	Recall	F1 Score
CNN	0.9357	0.8934	0.9478	0.9198
LSTM	0.9408	0.8975	0.9563	0.9260
CLSTM	0.9494	0.9139	0.9612	0.9370
RCNN	0.9492	0.9098	0.9652	0.9367
Proposed method	0.9696	0.9385	0.9871	0.9622

In summary, the experimental evaluation yields results that clearly demonstrate that the approach promises a more accurate and reliable phishing email detection mechanism as compared to traditional deep learning models. Such improvements in all the classification metrics support the incorporation of advanced hybrid architectures.

6 CONCLUSION

The modern world still has an open door for phishing attacks, one of the world's most dangerous cyberattacks, which exploit the greatest human weaknesses in stealing sensitive information. We propose a new hybrid suspicious email recognition model that stacks BLSTM and BERT to improve classification accuracy. Our approach successfully captures the semantic and syntactic structures contained in phishing emails by combining BERT's deep contextual embeddings with sequential learning capabilities offered by BLSTM. Numerical results showed that the proposed model outperformed the traditional methods with respect to accuracy, recall, F1 score, and false positive rate. Future work would include scaling the model for multilingual datasets and optimizing it for real-time deployment through model pruning and quantization. Furthermore, one would aim to improve

adversarial robustness and would include explainability methods to make the system more adaptive, transparent, and more resilient to changing phishing attacks.

Acknowledgement

I would like to thank the researchers whose work contributed to this study. I would also like to thank Kaggle for providing us with the phishing attack database we used in our research.

Funding source

No funds received.

Data availability

N/A

DECLARATIONS

Conflict of interest

Author affirm that no conflict of interest exists.

Consent to publish

N/A

Ethical approval

N/A

REFERENCES

[1] Carroll F, Adejobi JA, Montasari R. How Good Are We at Detecting a Phishing Attack? Investigating the Evolving Phishing Attack Email and Why It Continues to Successfully Deceive Society. SN Computer Science. 2022;3(2). 10.1007/s42979-022-01069-1

[2] Alkhalil Z, Hewage C, Nawaf L, Khan I. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. Frontiers in Computer Science. 2021;3. 10.3389/fcomp.2021.563060

[3] Halaseh RA, Alqatawna J. Analyzing CyberCrimes Strategies: The Case of Phishing Attack. In: 2016 Cybersecurity and Cyberforensics Conference (CCC). IEEE; 2016. p. 82–88. 10.1109/ccc.2016.25

[4] Wang L, Jajodia S, Singhal A, Cheng P, Noel S. k-Zero Day Safety: A Network Security Metric for Measuring the Risk of Unknown Vulnerabilities. IEEE Transactions on Dependable and Secure Computing. 2014;11(1):30–44. 10.1109/tdsc.2013.24

[5] Gupta BB, Arachchilage NAG, Psannis KE. Defending against phishing attacks: taxonomy of methods, current issues and future directions. Telecommunication Systems. 2017;67(2):247–267. 10.1007/s11235-017-0334-z

[6] Okokpujie K, Kennedy CG, Nnodu K, Noma-Osaghae E. Cybersecurity Awareness: Investigating Students' Susceptibility to Phishing Attacks for Sustainable Safe Email Usage in Academic Environment (A Case Study of a Nigerian Leading University). International Journal of Sustainable Development and Planning. 2023;18(1):255–263. 10.18280/ijsdp.180127

[7] Alabdan R. Phishing Attacks Survey: Types, Vectors, and Technical Approaches. Future Internet. 2020;12(10):168. 10.3390/fi12100168

[8] Putra FPE, Ubaidi U, Zulfikri A, Arifin G, Ilhamsyah RM. Analysis of Phishing Attack Trends, Impacts and Prevention Methods: Literature Study. Brilliance: Research of Artificial Intelligence. 2024;4(1):413–421. 10.47709/brilliance.v4i1.4357

[9] Yeboah-Boateng EO, Amanor PM. Phishing, SMiShing & Vishing: an assessment of threats against mobile devices. Journal of Emerging Trends in Computing and Information Sciences. 2014;5(4):297-307

[10] Ayiku D. Comparative Analysis: The increase in phishing activities. PhD thesis. 2023

[11] Salloum S, Gaber T, Vadera S, Shaalan K. A Systematic Literature Review on Phishing Email Detection Using Natural Language Processing Techniques. IEEE Access. 2022;10:65703–65727. 10.1109/access.2022.3183083

[12] Harikrishnan N, Vinayakumar R, Soman K. A machine learning approach towards phishing email detection. In: Proceedings of the anti-phishing pilot at ACM international workshop on security and privacy analytics (IWSPA AP). vol. 2013; 2018. p. 455-68

[13] Bountakas P, Xenakis C. Helphed: Hybrid Ensemble Learning Phishing Email Detection. SSRN Electronic Journal. 2022. 10.2139/ssrn.4147334

[14] Valecha R, Mandaokar P, Rao HR. Phishing Email Detection using Persuasion Cues. IEEE Transactions on Dependable and Secure Computing. 2021:1–1. 10.1109/tdsc.2021.3118931

[15] Niu W, Zhang X, Yang G, Ma Z, Zhuo Z. Phishing Emails Detection Using CS-SVM. In: 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC). IEEE; 2017. p. 1054–1059. 10.1109/ispa/iucc.2017.00160

[16] Fang Y, Zhang C, Huang C, Liu L, Yang Y. Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism. IEEE Access. 2019;7:56329–56340. 10.1109/access.2019.2913705

[17] Zareapoor M, K R S. Feature Extraction or Feature Selection for Text Classification: A Case Study on Phishing Email Detection. International Journal of Information Engineering and Electronic Business. 2015;7(2):60–65. 10.5815/ijieeb.2015.02.08

[18] Atawneh S, Aljehani H. Phishing Email Detection Model Using Deep Learning. Electronics. 2023;12(20):4261. 10.3390/electronics12204261

[19] Hassanpour R, Dogdu E, Choupani R, Goker O, Nazli N. Phishing e-mail detection by using deep learning algorithms. In: Proceedings of the ACMSE 2018 Conference. ACM SE '18. ACM; 2018. p. 1–1. 10.1145/3190645.3190719

[20] Hiransha M, Unnithan NA, Vinayakumar R, Soman K, Verma A. Deep learning based phishing e-mail detection. In: Proc. 1st AntiPhishing Shared Pilot 4th ACM Int. Workshop Secur. Privacy Anal.(IWSPA). Tempe, AZ, USA; 2018. p. 1-5

[21] Chinta PCR, Moore CS, Karaka LM, Sakuru M, Bodepudi V, Maka SR. Building an Intelligent Phishing Email Detection System Using Machine Learning and Feature Engineering. European Journal of Applied Science, Engineering and Technology. 2025;3(2):41–54. 10.59324/ejaset.2025.3(2).04

[22] Alsubaei FS, Almazroi AA, Ayub N. Enhancing Phishing Detection: A Novel Hybrid Deep Learning Framework for Cybercrime Forensics. IEEE Access. 2024;12:8373–8389. 10.1109/access.2024.3351946

[23] Wei Y, Nakayama M, Sekiya Y. Enhancing Generalization in Phishing URL Detection via a Fine-Tuned BERT-Based Multimodal Approach. IEEE Access. 2025;13:131197–131216. 10.1109/access.2025.3591843

[24] Benny A, Saji AM, Joseph CJ, Christina PB, Antony MA. In: Real-Time Voice Phishing Detection Using BERT. Springer Nature Switzerland; 2025. p. 410–426. 10.1007/978-3-031-90482-0_33

[25] Mi G, Gao Y, Tan Y. In: Apply Stacked Auto-Encoder to Spam Detection. Springer International Publishing; 2015. p. 3–15. 10.1007/978-3-319-20472-7_1

[26] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Computation. 1997;9(8):1735–1780. 10.1162/neco.1997.9.8.1735

[27] Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing. 1997;45(11):2673–2681. 10.1109/78.650093

[28] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv preprint arXiv:13013781. 2013

[29] Pennington J, Socher R, Manning C. Glove: Global Vectors for Word Representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics; 2014. p. 1532–1543. 10.3115/v1/d14-1162

[30] Mustaqeem, Kwon S. CLSTM: Deep Feature-Based Speech Emotion Recognition Using the Hierarchical ConvLSTM Network. Mathematics. 2020;8(12):2133. 10.3390/math8122133

[31] Bharati P, Pramanik A. Deep Learning Techniques R-CNN to Mask R-CNN: A Survey. In: Computational Intelligence in Pattern Recognition. Springer Singapore; 2019. p. 657-68. 10.1007/978-981-13-9042-5_56

[32] Li Z, Liu F, Yang W, Peng S, Zhou J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Transactions on Neural Networks and Learning Systems. 2022;33(12):6999-7019. 10.1109/tnnls.2021.3084827

[33] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Computation. 1997;9(8):1735–1780. 10.1162/neco.1997.9.8.1735

How to cite this article

Mohammed MB. A Hybrid Stacked Bidirectional Long Short-Term Memory and Bidirectional Encoder Representations from Transformers Model for Enhanced Phishing Email Detection. Journal of University of Anbar for Pure Science. 2026; 20(1):282-291. doi:10.37652/juaps.2025.158955.1369