Journal of Computer Science

Research Article

A Novel Hybrid Machine and Deep Learning Model for Detecting Arabic Phishing Emails

Fahad Ghabban

Department of Information Systems, College of Computer Science and Engineering, Taibah University, Medina, Saudi Arabia

Article history
Received: 29-05-2025
Revised: 22-08-2025
Accepted: 28-08-2025

Email: Fghaban@taibahu.edu.sa

Abstract: Phishing emails are becoming an increasingly popular type of cybercrime on the internet, affecting both businesses and individuals. The attackers generally use various methods to trick victims and extract personal information from them, such as bank details, home addresses, and account information. Many attempts have been proposed to tackle this issue by using filtering mechanisms or automated classification methods which require human intervention. However, this issue still remains a significant challenge. Additionally, attackers previously used manual methods to write phishing emails, however, recent AI tools have been used in means to generate such phishing emails. Therefore, this study aims to propose a hybrid machine and deep learning model to distinguish between content based phishing emails to categorize such emails into either real or fake, particularly, of the Arabic language. This model consists of BiLSTM (Bidirectional Long Short-Term Memory), GRU (Gated Recurrent Unit), and RF (Random Forest). In addition to this, a novel Arabic phishing email dataset has been developed. This imbalanced dataset consists of 418 phishing emails. Several experiments have been conducted in order to evaluate the proposed model. In addition, the sentence structure has been considered through the use of N-gram methods. Moreover, the experimental results show that the proposed model outperforms traditional machine learning classifiers and deep learning models. The model's performance achieved an accuracy of 98.81%.

Keywords: Machine Learning, Deep Learning, Spam, Generative AI, Phishing Emails

Introduction

Internet and technology have increased the popularity of online services, which are used by both individuals and commercial entities. Because of this, Internet fraud threatens their privacy and security more and more every day (Khan et al., 2023; Al-Charchafchi et al., 2020). Email has become increasingly important in both personal and business communication. Among the most common issues on the internet is phishing. Phishing is a sort of cyber-attack that involves sending deceptive emails, texts, or phone calls. In this method, phishing attacks and sophisticated methods are used to manipulate email messages. The attacker is trying to trick people into revealing sensitive information, such as passwords or credit card numbers (Al-Otaibi and Alsuwat, 2020; Yasin and Abuhasan, 2016; Chetioui et al., 2022). This has resulted in an increase of email traffic. Due to an increase in unsolicited emails, users may lose track of important messages due to an inundation of emails. In light of the increasing volume of spam and unwanted emails inundating inboxes, the conventional approach of blacklisting spam emails or relying solely on filtering mechanisms is no longer sufficient (Chetioui et al., 2023). These traditional methods often fail to keep up with the sheer volume of unsolicited messages, resulting in a barrage of unwanted content.

An important component of efficient email management is the automation for classification of emails. Such methods organize incoming messages and prioritize them based on predefined categories. Therefore, researchers have proposed numerous solutions to counter phishing attacks and search into the issue of phishing. There are two main methods, the first is adding a security layer, and the second is detecting phishing emails. In the first method, extra security can be achieved by adding a second authentication factor to the login process. This means that even if someone steals your password, they still can't access the account without the second factor, like a code sent to your phone. In the second method, researchers have developed improved methods for detecting and handling spam by utilizing Deep Learning (DL) (Mohammed et al., 2019; Ghourabi et al., 2020; AbdulNabi and Yaseen, 2021; Srinivasan et al., 2021 Kaddoura et al., 2020; Magdy et al., 2022; Debnath and Kar, 2022), Machine Learning (ML) (Gangavarapu et al., 2020; Alsaidi et al., 2022; Bountakas et al., 2021; Salahdine et al., 2021; Zamir et al., 2020; Ripa et al., 2021), and transformers (Karki et al., 2022; Gogoi and Ahmed, 2022; Giri et al., 2022; Somesha and Pais, 2024). Using these approaches, Distinguishing spam from legitimate emails can be done more accurately with minimal human intervention. Recently, generative AI can make new content based on the patterns and data it has been trained on, including images, text, and music. A generational AI algorithm can create complex emails as well as simple responses. These algorithms are able to create emails that mimic human communication by analyzing vast datasets of emails, helping them to learn the patterns of language, tone, and structure (Brynjolfsson et al., 2025; Fui-Hoon Nah et al., 2023).

Therefore, the aim of this study is to propose a hybrid model that uses ML and DL to distinguish between fake and real emails for the Arabic language. This model consists of BiLSTM (Bidirectional Long Short-Term Memory), GRU (Gated Recurrent Unit), and RF (Random Forest). A new Arabic dataset with 418 emails, divided into two classes (real and fake), was introduced for evaluation. The main research questions are "how do bigram-based features influence the performance of ML classifiers in distinguishing AI-generated phishing emails from human-generated phishing emails in Arabic?", and "how effective is a hybrid model combining BiLSTM, GRU, and Random Forest in distinguishing between AI-generated phishing emails from human-generated phishing emails in Arabic compared to standalone ML and DL models?". Several experiments were conducted using ML classifiers and DL models. The proposed model outperformed both ML classifiers and standalone DL models in several experiments. In terms of accuracy, the model was 99% accurate in recognizing real phishing emails from fake phishing emails. The contributions of this research can be viewed in three ways:

1- Proposing a hybrid ML and DL model to detect real and fake phishing emails. The model consists of BiLSTM, GRU, and RF
2- Introducing a new Arabic email dataset containing 418 emails that can be classified as binary emails
3- Presenting a comparison between the experimental results of ML learning classifiers and standalone DL models for evaluating the performance of the proposed model
4- Comparing real phishing emails with fake emails based on the sentence structure using N-gram

Related Studies

This section provides an overview of the related studies and methods used to detect and classify phishing emails using machine learning and deep learning. Yasin and Abuhasan (2016), proposed a method to differentiate between phishing emails and legitimate ones using data mining techniques. The process involved extracting details, from email headers, contents and word frequencies supported by WordNet ontology and stemming methods. Relevant features were chosen using Information Gain and overfitting was prevented through 10 fold cross validation. The authors tested five data mining algorithms (Random Forest, J48, SVM, MLP and Bayes Net) on two datasets resulting in improved accuracy compared to models. The best accuracy was achieved with Random Forest (0.991). Likewise, Salloum et al. (2023), presented a new parallel corpus for phishing emails in English and Arabic. It was built using IWSPA-AP 2018 dataset. The main goal of the research is to detect phishing emails better. For this purpose, Random Forests, SVM and Logistic Regression through machine learning techniques were applied in which TF-IDF feature extraction method was used while training on the corpus. Based on the findings, the Arabic corpus outperforms the English counterpart where MLP classifier achieved highest accuracy rate of 94.63% for English emails and 96.82% for Arabic emails respectively. Identically however with the use of a neural network model, (Hassanpour et al., 2018), introduced a neural network model that uses MATLAB and Python for detecting and classifying phishing e-mails. The weights of the words in the emails were calculated by TF-IDF method (Term Frequency-Inverse Document Frequency). In order to train the model, a dataset has been created which contains the phishing emails. 600 emails have been taken for this study out of which there were 300 spam (phishing) emails and 300 ham (legitimate communications). The dataset was divided into two parts: training data comprising 80% of the total records, and testing data containing 20%. To see how good is our model at recognizing phishing mails, we compare its performance with other machine learning based models on this problem.

Following up with this idea but exploring a broader range of techniques and algorithms, (Bountakas et al., 2021; Abu-Nimeh et al., 2007), compared Machine Learning (ML) algorithms, such as logistic regression, decision trees, random forests, gradient boosting trees and naive bayes – against Natural Language Processing (NLP) techniques, including TF-IDF, Word2vec and BERT. To find out which ones would be best for identifying phishing emails. The authors carried out the evaluation on balanced and imbalanced datasets consisting of phishing corpus emails from Nazario and Enron. The analysis of the study was done with Apache Spark on Ubuntu 20.04 and the source code was made available to the public. The top NLP/ML combinations for balanced data were Word2Vec/Random Forest while for unbalanced data they were Word2Vec/Logistic Regression. Metrics like accuracy, precision, recall and F1-score were used in this study where English language emails were given more weight during model evaluation performance measurement. In the experiment with the balanced dataset method; Random Forest had 98.95% accuracy rate followed by Gradient Boosting Trees at 97.48%, Logistic Regression 96.77%, Decision Tree 96.25% then Naive Bayes had 95.64%. For imbalanced dataset experiments: Logistic Regression gave an F1-score of 89.96% while using TF-IDF method gave Gradient Boosting Trees an F1-score of 81.83%.

Similarly, however with a larger dataset, (Abu-Nimeh et al., 2007; Mehdi Gholampour et al., 2018), analyzed and assessed the predictive power of six classifiers in distinguishing phishing emails. Random Forests, Support Vector Machines, Bayesian Additive Regression Trees, Logistic Regression, Classification and Regression Trees, and Neural Networks are among these methods. The authors employed a dataset with 1171 phishing emails and 1718 genuine ones. To train and test the models, 43 attributes were used in the data set. In this study error rates were estimated via 10-fold-cross-validation which is an unbiased method of accuracy estimation. Naive Bayes had not been taken into consideration due to its poor performance on this dataset in terms of prediction abilities. Identically, (Kaddoura et al., 2020), presented a machine learning approach for detecting phishing emails by using TF-IDF representation, SVD, NMF and several machine learning algorithms such as Decision Tree, Random Forest, Logistic Regression, Naive Bayes, KNN, AdaBoost and SVM. The collection consists of numeric representations of emails marked as legitimate or phishing. Machine learning principles are the main focus in technical language used. Several techniques achieve high accuracy rates with SVM recording 98.7% accuracy on validation data. Feature selection together with dimensionality reduction and classification is combined in these methods to effectively identify whether an email is genuine or fake based on learnt patterns.

Further focusing on the email aspect of malicious activity, (Mohammed et al., 2019), presented a Multi-Natural Language Anti-Spam (MNLAS) model. This system uses machine learning techniques to protect emails effectively. The model has many stages of processing to enhance spam detection such as feature extraction, presentation, selection, identification of short words and email classification. In this research, the authors took 200 emails in HTML and text forms as a dataset; among them 100 were spam emails while other 100 were non-spam emails. Hence it can be applicable to English as well as Arabic languages too. The accuracy rate for distinguishing between junk mails and legitimate ones by the MNLAS model is remarkable-91%.

In the same way but with a Hybrid approach, (Ghourabi et al., 2020), suggested a CNN-LSTM hybrid model for spotting SMS spam in English and Arabic messages. Aside from this proposed deep learning model, conventional machine learning methods such as SVM, Naive Bayes, KNN and Decision Trees were utilized. A set of two datasets were used; one constituted Arabic messages collected from smartphones around the area while another was comprised of UCI Repository's SMS Spam dataset. In terms of precision, recall, accuracy, f1-score, ROC AUC and other metrics used for measuring performance on classification tasks, the hybrid CNN-LSTM model showed better results than any other algorithm tested according to all measures employed in this study. In the same way, (AbdulNabi and Yaseen, 2021), provided a spam email detection system for English text emails based on deep learning. The authors applied Feed Forward Neural Network (FFNN) and Bidirectional Encoder Representations from Transformers (BERT) in modeling while Term Frequency-Inverse Document Frequency (TF-IDF) was used for feature extraction. Evaluation was done using the Spam Assassin, TREC and Ling Spam datasets were used for experiments. Count-Vectorizer was outperformed by TF-IDF feature extraction method with 99.15 average F1 score which shows high precision of the model. According to F1 scores, machine learning algorithms and deep learning models of the study achieved accuracies between 94% – 99.89%. BERT as model training method along with FFNN, technique employed were using TF-IDF for feature extraction while evaluation was through F1 scores. Also addressing the issue of spam emails however employing a different technique for feature extraction, (Masri and Al-Jabi, 2023), delved into the deep learning methods and the BERT transformer model for email spam classification while applying word embedding. For training and testing, the study makes use of two public datasets: the first one is the Spam base dataset which is publicly available from the UCI machine learning repository, and it consists of 5569 emails, 745 of which are spam, and the second one is the Spam filter dataset which is publicly available on the Kaggle website, and it contains 5728 emails, 1368 of which are spam. The English language is the foundation of this class and the main aspect of the course. The proposed algorithm surpasses the k-NN and a Naive Bayes classifiers with the highest accuracy of 98.67% and F1-score of 98.66% achieved by applying the BERT transformer model.

In Business Email Detection, (Yafooz et al., 2021), proposed a new and efficient method for Arabic email classification using a deep learning models. Word-based lexicon is used for email classification by means of a Convolutional Neural Network (CNN). In addition to the analysis, 63,257 emails subject, sentiment and urgency are also included from the dataset. This study is a particular area of the Arabic language. The simulations demonstrated the viability and adaptability of the approach the models reached an accuracy rate of up to 92% with no more than 8% of loss. Utilizing data preparation steps, training the model during 20 epochs with the use of the Binary Crossentropy loss function and the Keras Adam optimizer as well as assessment of the results based on the metrics such as the F1 score, precision, recall and accuracy were the techniques used (Mohammed et al., 2019).

Li et al. (2024), implemented a BERT-based deep learning method to get around the problems that traditional rule-based methods have when it comes to finding complex and changing social work email phishing attacks. The authors wanted to make real-time detection more accurate by letting the system learn new ways to trick people and new types of tricks. The authors did this by training and updating the BERT model so that it could look at all the contextual features of email content. Al Daoud et al. (2024), proposed using transformer-based models to improve the detection of phishing emails and social media scams, which would help with the problems that traditional rule-based and machine learning methods have with new cyber threats. The authors looked into four different ways: zero-shot learning with big pre-trained language models (LLMs) like GPT-4o, feature extraction with transformers followed by random forest classification, fine-tuning Small Language Models (SLMs) on new datasets, and an ensemble method that puts together the best models.

Jamal and Wimmer (2023), developed a phishing and spam detection framework called IPSDM. It uses fine-tuned Large Language Models (LLMs) like DistilBERT and RoBERTA to make detection more accurate and reliable, especially when dealing with imbalanced datasets. They used transformer-based models with optimization techniques, hyper-parameter tuning, and data augmentation strategies like oversampling with ADASYN to get around the problems with traditional machine learning methods. Uddin et al. (2024), suggested a transformer-based method for detecting phishing emails using a refined DistilBERT model, resolving issues with model interpretability and unbalanced datasets that are typical of cybersecurity tasks. They achieved high accuracy (up to 98.48%) by optimizing the model through data cleaning, balancing techniques, and hyperparameter tuning, proving its efficacy in differentiating between phishing and authentic emails.

Park and Kim (2025), presented a model for automatically creating anti-phishing training scenarios using a group of generative AIs. They developed a system that combines outputs from models such as ChatGPT and LLaMA to create realistic, customized training scenarios in order to combat the increase in AI-generated phishing attacks. To choose the best outcomes, these scenarios are assessed using both human (feasibility, personalization, completeness) and automatic (BLEU, ROUGE) metrics.

Problem Formulation

The problem of detecting phishing emails can be described as a binary classification problem which is made up of two classes, namely E_{Real} and E_{Fake}. Each email message E can be represented as a tuple x_i and y_i in E, where E represents the set of all email messages. x is represented the email content and y represents the label of the classes, i is the number of emails. The goal is to design and train a model that function of f: $E \rightarrow \{0,1\}$, can detect and assign the emails to Real or Fake classes.

The given dataset D = $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\ldots\ldots (x_n, y_n)\}\}$ Consists of n labeled email messages. Each x is the email content which represent by the features that extracted from the content. Using the classification model, distinguish between E_{Real} and E_{Fake}. E_{Real} and E_{Fake} can be viewed as subsets of email messages in E as being real emails and fake emails, respectively.

The dataset D divided into parts, one for training process $\{D_{train}\}$ and the second for testing process $\{D_{Test}\}$. Then, classification error must be minimized by finding a classification function f(x) as represented in mathematical Equation 1:

$$objective\,function = \min \textstyle\sum_{i=1}^{n} l(y_i\,, F(x_i) \qquad (1)$$

Where n is the number of emails, and l is the loss function. Then, the model's performance can be evaluated by comparing its predictions to the ground truth labels of real and fake emails, using metrics such as accuracy, precision, recall, and F1-score. Algorithm 1 shows the steps of the detecting and classifying the phishing emails.

Algorithm 1: Classifying fake and real Phishing.

```
Input: D as dataset
  Set of email messages represented by E
  Subset of phishing emails $E_{Real}$
  Subset of emails generated by $E_{Fake}$
For each email $e_i$ in E:
  a. If $e_i$ belongs to $E_{Real}$ or $E_{Fake}$
     Set $f'(e_i)=1f'(e_i)=1$ (Real email)
  b. Otherwise:
  Set $f'(e_i)=0f'(e_i)=0$ (Fake email)
Output:
Model output $f'$ for each email $e_i$ indicating whether email is fake or real.
```

Methods and Materials

This section describes the methods used to achieve the study's objectives. The methods consist of several steps, namely data collection, pre-processing, feature selection, the development of a proposed model, and the evaluation of the proposed model. All the phases are illustrated in Fig. 1.

Data Collection Phase

In this first phase of the study, data were collected. There were two types of emails in the dataset: real phishing emails and fake phishing emails. On the basis of that, two methods were used to collect data. There are two sets of emails: those sent by attackers last year and some being real. Also, ChatGPT was used to generate fake emails. Recently, attackers have increasingly leveraged generative AI tools to produce convincing phishing content. ChatGPT, as one of the most popular generative AI tools, was incorporated in this study. These samples generated by ChatGPT enhance the realism of the experiments. This approach ensures that the proposed model is robust against both traditional and modern threats. Three Arabic native speakers have read and evaluated phishing emails manually. Additionally, the dataset size is 418 emails for both classes. Moreover, this dataset is considered as an imbalance dataset. Finally, Table 1 shows the result of the dataset. A sample of an AI generated phishing email in Arabic can be viewed in Table 2. Additionally, a sample of a Human generated phishing email can be seen in Table 3.

Pre-Processing Phase

There are several essential steps in the pre-processing phase that ensure that the dataset is ready for the feature selection process, and that the model is fed effectively during the feature selection phase. In these steps, the text is first tokenized, which allows further processing to take place. The next step is to remove punctuation marks that do not contribute to the meaning of the text. Additionally, to exclude common words that do not carry significant information, therefore, Arabic stop words are removed. Moreover, numerical values that are irrelevant are removed, and special characters are also removed. Both ML and DL got the same cleaned input data. For ML models, features were extracted using N-gram and TF-IDF representations. For DL models, the same cleaned dataset was tokenized and converted into word embeddings before being fed into the network.

Table 1: The proposed dataset description

Classes	Description	Max	Min
Real	228	168 Words	15 Words
Fake	190		
Total	418		

Table 2: Sample of AI generated Phishing email in Arabic

AI-Arabic	Translation in English
مرحبًا جون،	Hello Jone,
أتمنى أن تكون بخير. أحتاج إلى تحويل مبلغ من البنك المحلي إلى بنك خارجي، وأتطلع إلى وجود شريك يمكنني التعاون معه في هذا الصدد. نسبة العمولة المالية ستكون محل مناقشة بيننا وفقًا للاتفاق الذي نتوصل إليه.	I hope you're doing well, I need to transfer an amount from a national bank to an international bank, and I need a partner that I can work with in this matter. The commission rate will be discussed to reach an agreement between us.
يرجى الرد إذا كنت مهتمًا بالتعاون، وسأكون ممتنًا للغاية لأي استفسارات أو توضيحات تحتاجها.	Please respond if you're interested in partnering, and I will be happy to answer any inquiries or questions you may have.
شكرًا لاهتمامك وتعاونك، وأنا في انتظار ردك. تحياتي	Thank you for your interest and collaboration, and I will be waiting to hear back from you. Regards

Table 3: Sample of Human written Phishing Email in Arabic

Human-Arabic	Translation in English
مرحبا، أنا مدير توظيف في أمازون ونبحث حاليا عن موظفا عبر الانترنت بدوام جزئي بالعمل من المنزل باستخدام الهاتف المحمول، يمكنك بسهولة كسب 1000 الى 3000 جنيه مصري في اليوم، ويتم دفع الراتب في نفس اليوم. مهام العمل بسيطة ويمكن القيام بها في أي وقت وفي أي مكان. يرجى الاتصال بنا عبر أو الضغط على الرابط لاضافة Telegram:vip347 Telegram والاتصال بنا https://t.me/vip347ملاحظة: يجب الا يقل عمر المتقدمين عن 20 عاما، ولا يمكن للطلاب المشاركة	Hello, I am the manager of employment in amazon and we are currently looking for a part-time employee that can work online from home by using a mobile phone, you can earn 1000 to 3000 Egyptian pounds in a day, the work is simple and can be done anytime and anywhere. Please call us through or by clicking the link to join Telegram:vip347 and calling us https://t.me/vip347 Note: The age of the appliers should not be less than 20 years, and students cannot participate.

Feature Selection Phase

The purpose of this section is to describe how words and sentences are represented before they are used as inputs for the model. There are three types of representations: TF-IDF, n-Gram, and word embedding. A TF-IDF measure measures the significance of a word in an email relative to a dataset (class), it includes two components: Term Frequency (TF) and Inverse Document Frequency (IDF). A calculation of the TF-IDF is presented in the mathematical formula no. 1:

$$W_{i,j} = tf_{i,j} X \log \left(\frac{N}{df_i}\right) \quad (1)$$

Where $tf_{i,j}$ is represent the amount of words (i) in the class (j) , and df_i represent the number of emails contain word I and N is the overall total number of emails.

An N-gram is a sequence of n words from a given email. It can represent one word on UniGram, two words on BiGram, three words on TriGram, four words on 4-Gram, and five words on 5-Gram. By using such representation, the performance of the model can be improved in identifying the most important words in each sentence.

Word embeddings are representations of dense vectors, in which words are in a continuous vector space. It's a method for expressing words in vector spaces that map semantically related words to nearby points in the space.

The Proposed Model

Several ML classifiers and DL models have been used in several experiments to evaluate the performance of the proposed model. These classifiers are used in machine learning experiments: Naive Bayes (NB), K-Nearest Neighbors (KNN), Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), AdaBoost (ADA), Gradient Boosting (GB), XGBoost (XGB), and Stochastic Gradient Descent (SGD) (Salloum et al., 2023). While in the DL experiments, BiLSTM, GRU, LSTM model has been ustilized. The most popular and efficient DL models for handling textual data in sequence form is to use these models. The proposed model consists of three models combined in the form of a fusion of three different types of models: BiLSTM, GRU, and Random Forest as shown in Figure 2.

RF component, we plan to conduct a detailed feature importance analysis using Gini importance and permutation-based techniques. This will allow us to identify which word- or phrase-level features. BiLSTM captures context from both past and future words, which is essential for understanding the full meaning of Arabic words in a sentence. This bidirectional context helps detect subtle patterns typical in phishing emails.

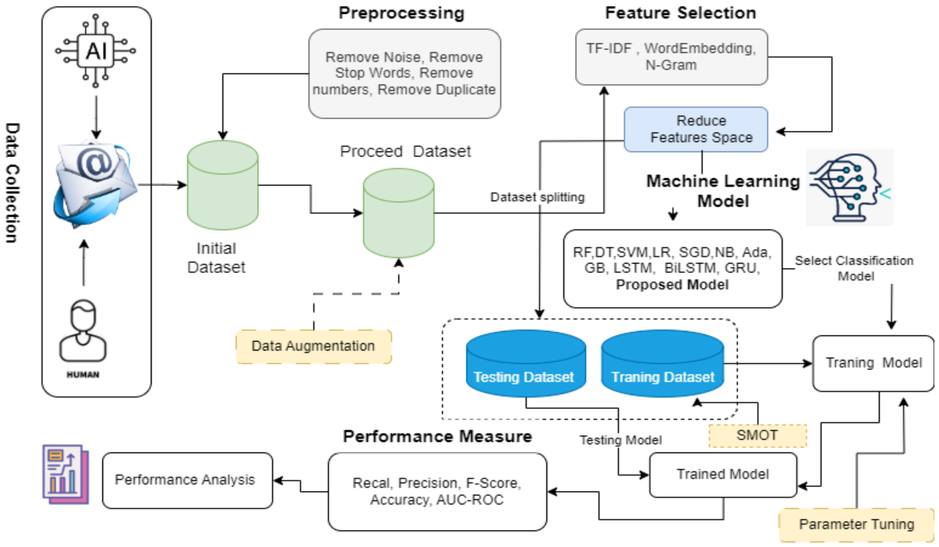


Fig. 1: Research methods

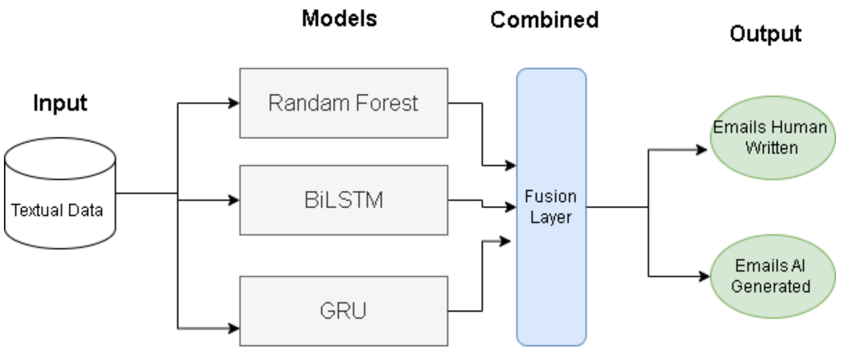


Fig. 2: Architecture of propose model

Figure 2 illustrates the architecture of the proposed model. Input data is processed differently by each of these models and the final prediction is made using each of them. BiLSTM is Recurrent Neural Networks (RNNs) capable of capturing sequential information from input data. There are three layers in this model: the embedded layer, the bidirectional LSTM layer, and the dense layer. It can capture sequential information in text with BiLSTM and GRU models, while complex nonlinear relationships can be captured with Random Forest. Random Forest models extract features based on statistical relationships between words and their labels, while BiLSTM and GRU models extract features based on the sequential nature of the text.

The BiLSTM and GRU models are concatenated with the Random Forest model. The final prediction is then generated using a Dense layer with a sigmoid activation function. With this combined model, text classification tasks are performed more efficiently by leveraging the strengths of each individual model. This combination of models can result in a more comprehensive set of features, leading to a better classification result.

Model Performance Phase

In this section, we present our experimental results to demonstrates the model performance in order to assess the reliability and validity of the results, different metrics for the training and testing processes were used. Moreover, the performance of the model was evaluated using the F1 score, precision, accuracy, confusion matrix, and Area Under Curve and Receiver Operating Characteristics (AUR-ROC).

Precision is the measure of the ratio of True Positives (TP) to the sum of true positives and False Positives (FP). Moreover, percentage of emails classified as fake are actually fake can by shown by Precision, as shown in Eq. (2):

$$recison = \frac{Number\ of\ the\ correct\ phishing\ emails\ classified}{Total\ number\ of\ relevant\ phishing\ emails} \quad (2)$$

Recall is the measure of the ratio of true positives to the sum of true positives and false negatives. Through recall, the percentage of fake emails that were correctly identified can be found, as represented in Eq. (3):

$$Recall = \frac{Number\ of\ the\ correct\ phishing\ emails\ classified}{Total\ number\ of\ phishing\ emails\ classified} \quad (3)$$

The harmonic mean of precision and recall was calculated through the F1 score as shown in Eq. (4). Moreover, the AUC-ROC level is calculated by plotting the True Positive Rate (TPR) by the False Positive Rate (FPR). Therefore, this is a measure of how well the model can distinguish between real and fake emails:

$$F1 - Score = 2\ X\ \frac{Precision + Recall}{Precision\ X\ Recall} \quad (4)$$

Results and Discussion

This section presents the experimental setup, including the settings and hyper-parameters that were used in the conducted experiments of ML, DL and the proposed model. The results and description of these experiments for the three different types of aforementioned models/classifiers (ML, DL and the proposed model) can be viewed in this section.

Experiment Settings

The settings of the experiments for the machine learning classifiers, deep learning, and the proposed model are presented in this section. In all experiments, Google Colab was used, specifically the sklearn package (splitting datasets, extracting features, machine learning classifiers, and evaluating confusion matrix and models), NLTK package (tokenization and stop words removal for the machine learning classifiers, and TensorFlow for the deep learning). In all experiments the SMOTE has been used to handle the imbalance class issue. Table 4 shows the hyper parameters for the ML classifiers and Table 5 shows the hyper parameters for the deep learning models. While Table 6 shows the hyper-parameters for the proposed model based on the experiments.

Machine Learning Experiments

In the first experiment, ML classifiers were used to detect whether the email was real or fake. Different N-gram features were used to evaluate machine learning classifiers (UniGram, BiGram, TriGram, 4-Gram, 5-Gram). The performance of the ML classifiers is measured in terms of accuracy as shown in Table 6. We have done cross-validation for all experiments, using five-fold validation, the difference for all the classifiers was between 2-4%.

Table 6 shows that the best classifiers consistently achieved high accuracy rates, with the SVM and RF classifiers achieving 84.25 to 95.24% and SGD classifiers achieving 84.25 to 96.83%. K-Nearest Neighbors (KNN) performed worse than other classifiers, with accuracy ranging from 68.50 to 89.68%. Despite having a reasonable accuracy with smaller N-gram sizes, KNN's accuracy dropped significantly with larger N-gram sizes, suggesting that it may not be the best method for this kind of text classification. It was found that SVM, RF, and SGD were the most accurate classifiers for this text classification task, while KNN was the least accurate. Figure 3 shows the confusion matrix and proves that the best accuracy was achieved by RF, SVM, and SGD in distinguishing between the phishing emails on whether they were fake or real, and the worst accuracy that was recorded was through the KNN and LR classifiers. Additionally, the AUC-ROC is presented in Figure 4.

Table 4: Parameters of ML classifiers

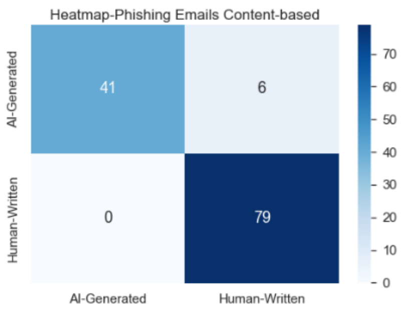
Classifier	Default Parameters
NB	No specific default parameters to set
KNN	n_neighbors=5, weights='uniform', algorithm='auto', leaf_size=30, p=2 (Euclidean distance)
LR	penalty='l2', dual=False, tol=1e-4, C=1.0,, max_iter=100, multi_class='auto',
SVM	C=1.0, kernel='rbf', degree=3, gamma='scale', coef0=0.0, shrinking=True, , tol=1e-3, cache_size=200
DT	criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1
RF	n_estimators=100, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', bootstrap=True
Ada	n_estimators=50, learning_rate=1.0, algorithm='SAMME.R'
GB	learning_rate=0.1, n_estimators=100, subsample=1.0, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_depth=3
XGB	n_estimators=100, max_depth=3, learning_rate=0.1, objective='binary:logistic',
SGD	loss='hinge', penalty='l2', alpha=0.0001, l1_ratio=0.15, fit_intercept=True, max_iter=1000, tol=1e-3, learning_rate='optimal', validation_fraction=0.1, n_iter_no_change=5,

Table 5: Hyper-parameters for LSTM BiLSTM and GRU

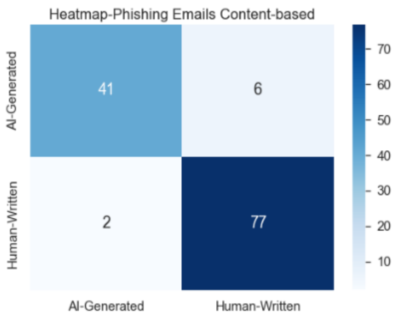
Hyper-parameter	Value
Embedding Dimension	32
LSTM/GRU Units	32
Batch Size	32
Sequence Length	100
Optimizer	"Adam"
Loss Function	"binary_crossentropy"
Metrics	["accuracy"]
Number of Epochs	30

Table 6: Accuracy using the ML classifiers

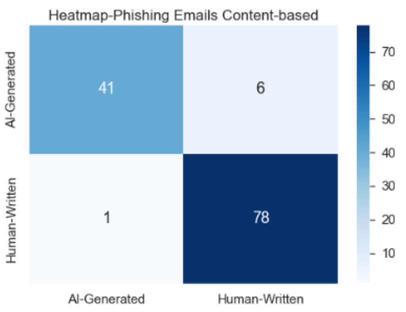
ML Classifiers	UniGram	BiGram	TriGram	4-Gram	5-Gram
NB	94.44%	91.27%	90.48%	85.83%	83.46%
KNN	89.68%	73.02%	69.84%	68.50%	69.29%
LR	92.06%	91.27%	84.92%	77.17%	75.59%
SVM	95.24%	93.65%	91.27%	85.04%	84.25%
DT	90.48%	85.71%	84.13%	75.59%	78.74%
RF	96.83%	95.24%	95.24%	84.25%	83.46%
ADA	95.24%	93.65%	94.44%	84.25%	85.04%
GB	95.24%	89.68%	92.86%	86.61%	85.04%
XGB	97.41%	92.06%	88.10%	83.46%	77.95%
SGD	94.44%	96.83%	91.27%	85.83%	84.25%



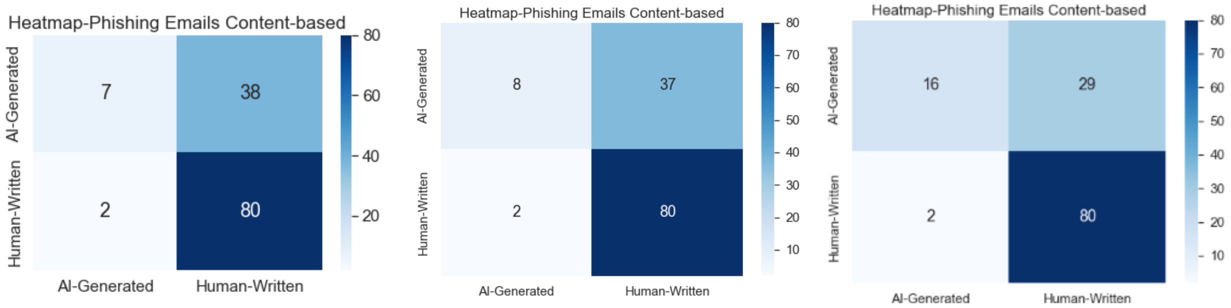
A. UniGram using RF



B. Unigram using SVM

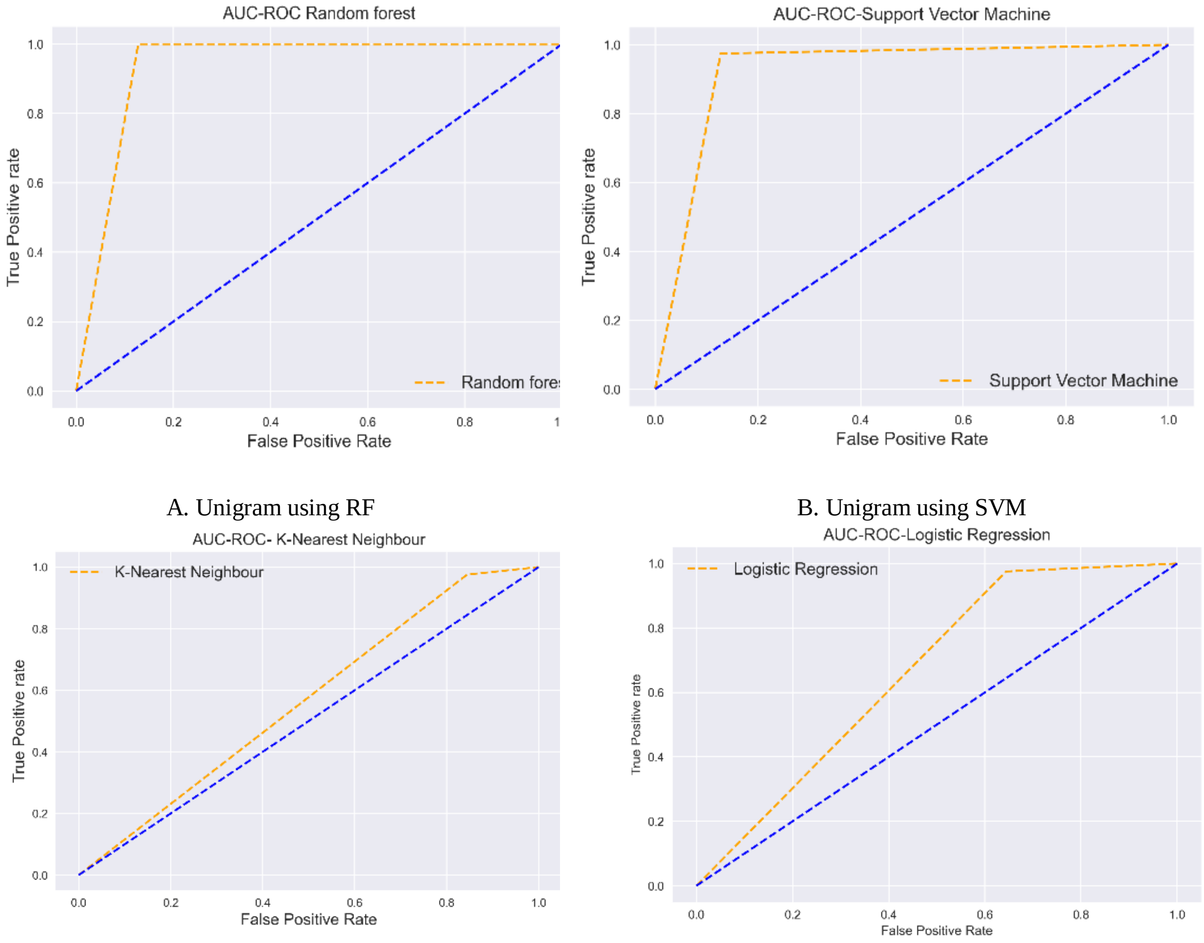


C. BiGram using SGD



D. 4-Gram using KNN E. 5-Gram using KNN F. 5- Gram using LR

Fig. 3: Confusion matrices show the best and worst model accuracy for ML Classifiers



A. Unigram using RF B. Unigram using SVM

4-Gram using KNN 5-Gram using LR

Fig. 4: AUC-ROC for best and worst ML classifiers

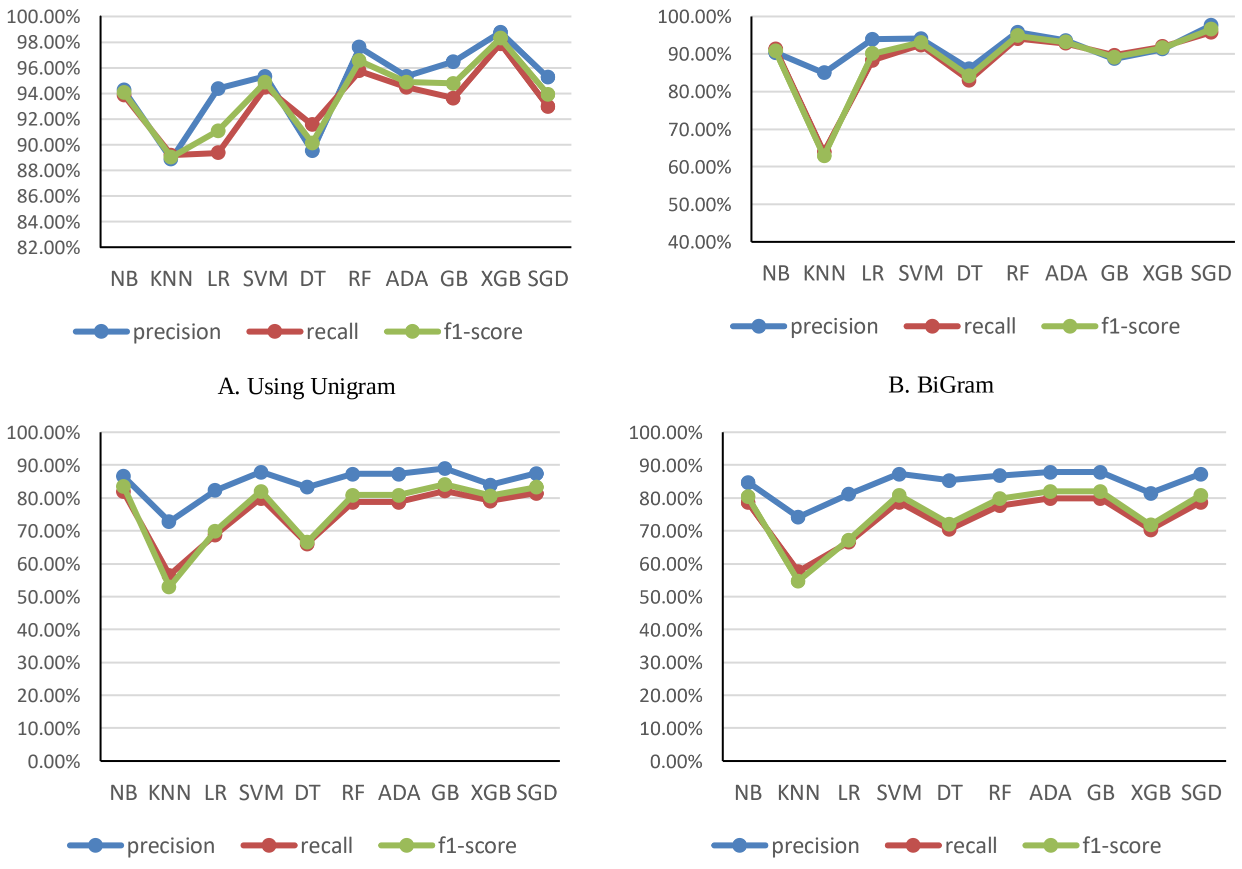
Figure 5 shows that the XGB classifier generally had the best and precision, recall, and F1 Score values across all N-gram features, followed by the RF classifier. In comparison to the other classifiers, these two can be considered better for this task. In general, the KNN classifier has lower precision, recall, and F1Scoe values than the other classifiers, especially for higher N-gram features (4-Gram and 5-Gram). Overall, in ML experiments, the XGB performed better in distinguishing between real emails than fake emails.

In the second experiment, the DL models were used to conduct the experiments. Comparing the experimental results as shown in Table 7. The different deep learning architectures are LSTM, BiLSTM, GRU, and the proposed Model are compared across several metrics as presented in Table 7.

Table 7 shows that the LSTM classifier recorded a Precision of 94.44%, a Recall of 92.73%, an F1-score of 93.58%, and an Accuracy of 91.76% have been achieved. In terms of performance, BiLSTM demonstrates an improvement with a Precision of 94.64%, Recall of 96.36%, F1-score of 95.50%, and Accuracy of 94.12%. The GRU model demonstrates a precision of 94.55%, a recall of 94.55%, an accuracy of 92.94%, and a F1-score of 94.55%. With the Proposed Model, Precision is 100.00%, Recall is 98.15%, F1-score is 99.07%, and Accuracy is 98.81%. This indicates that the proposed model is highly accurate in classifying phishing email for both classes, surpassing other architectures in precision, recall, and overall performance with only 20 epochs. In addition, the cross-validation has been done and the results show that the difference was between 2%-5% as a maximum to all classifiers. Thus, indicating there was no overfitting issues, compared to the accuracy of the testing. The accuracy and validation of the proposed model can be seen in Figure 7, which indicates a high degree of accuracy, while the loss function for both was reduced. On the other hand, Figure 6 shows the lower accuracy and validation of the LSTM and GRU models compared to the proposed model. There is also no close relationship between the loss functions of the accuracy and validation.

Table 7: Comparison between the DL models and proposed model

DL Models	Precision	Recall	F1-score	Accuracy
LSTM	94.44%	92.73%	93.58%	91.76%
BiLSTM	94.64%	96.36%	95.50%	94.12%
GRU	94.55%	94.55%	94.55%	92.94%
Proposed Model	99.22%	98.77%	98.99%	98.81%



A. Using Unigram

B. BiGram

C. Using 4-Gram

D. Using 5-Gram

Fig. 5: Precision, recall, F1 of ML classifiers

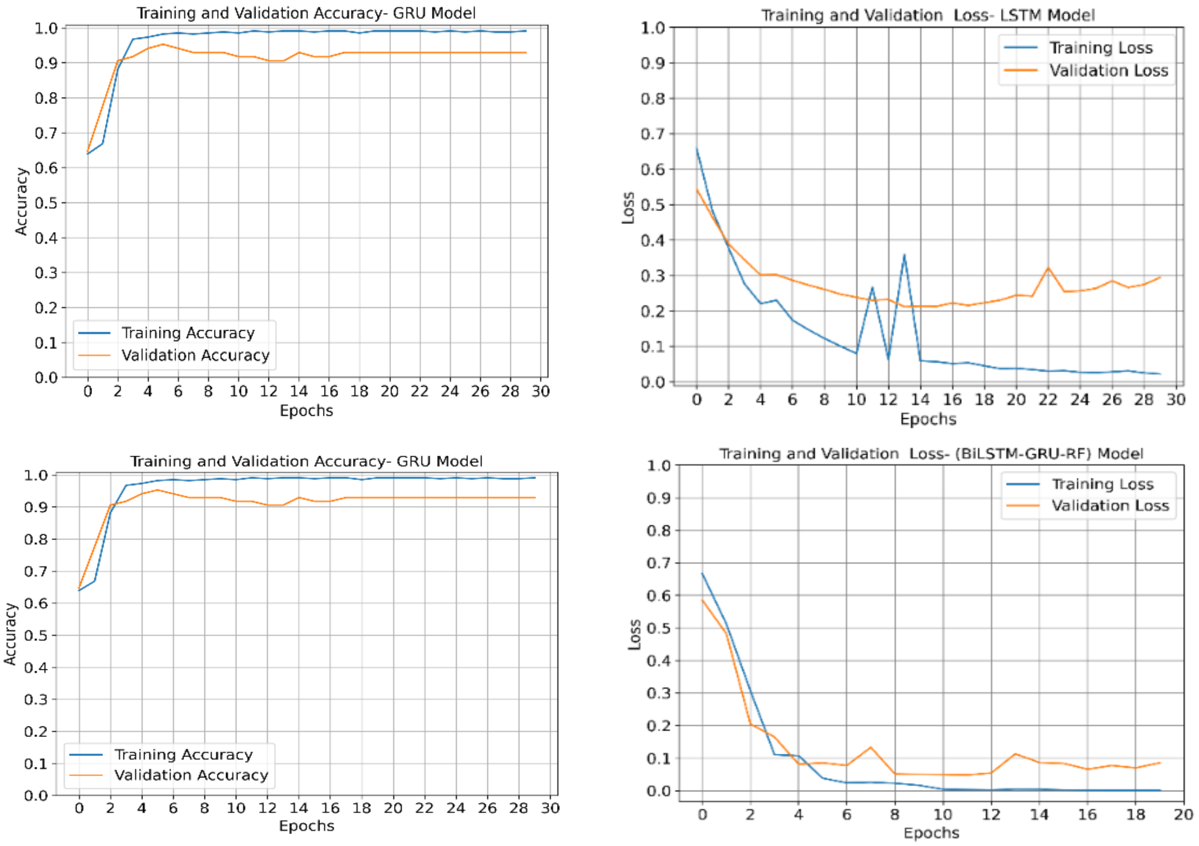
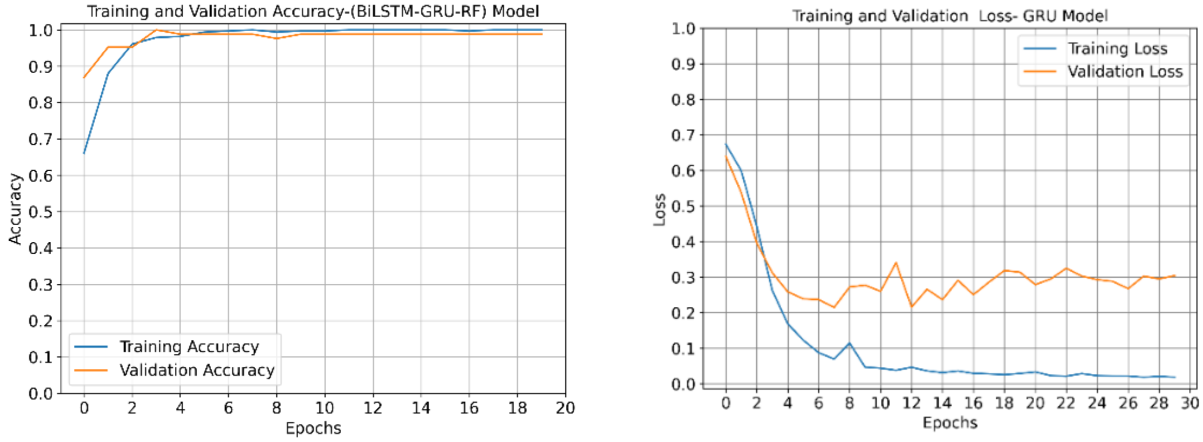


Fig. 6: Training and validation accuracy for DL models



A. Training and validation accuracy

B. Loss of training and validation

Fig. 7: Training and validation accuracy for proposed model

The experiments show that the combination of BiLSTM, GRU, and Random Forest models can improve the accuracy of identifying whether text was fake or real. Random Forest is good at recognizing patterns and relationships in data, while BiLSTM and GRU are good at understanding context and meaning of text. Using this approach, we can also extract a wider range of features from the text, which can make classification more effective. As a result, ensemble methods like this can also lead to more accurate and robust models by reducing overfitting and enhancing generalization to new, previously unknown data. On a standard CPU, the average inference time for LSTM, BiLSTM, and GRU was 0.25–0.5 seconds. The proposed hybrid model required 0.6–0.8 seconds. This shows only a slight increase in runtime. On average, each email took about 2–6 milliseconds to process. Moreover, Table 8 shows similar studies which applied state-of-the-art techniques to phishing emails and phishing email datasets. In addition, Figure 8 shows the top features that extracted using the proposed model.

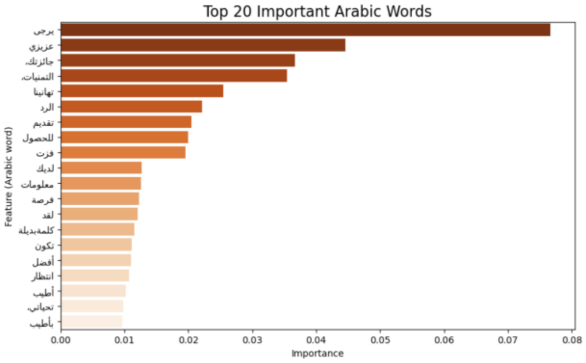


Fig. 8: Top features from the proposed dataset

Table 8: Comparison of state-of-the-art and the proposed model

Model/Approach	Dataset Size	Language	Accuracy	Authors
TF IDF + Multilayer Perceptron	1 258 emails (balanced phishing/legitimate)	Arabic	96.82%	Salloum et al. (2023)
OSINT enhanced Random Forest	– not numerically specified	English & Arabic	97.37%	An et al. (2025)
Bi LSTM	Enron + PhishingCorpus (~3 000 emails)	English	95.4%	Divakarla and Chandrasekaran (2023)
RAPH Model: NLP-based word-/sentence-matching with custom phishing vocabularies	1 250 emails (1 000 legitimate, 250 phishing)	Arabic	98.4%	Al-Yozbaky and Alanezi (2023)
BERT (pre trained)	UCI ML + SpamFilter combined (~5 000 emails)	English	98.67%	AbdulNabi and Yaseen (2021)
BERT (feature extractor) + CNN classifier	Kaggle “phishing email” dataset (exact count not specified)	English	97.5%	Gupta et al. (2024)
Proposed Model	Proposed dataset	Arabic	98.81%	This study

Conclusion

In this paper, a hybrid model is presented to distinguish between fake and real phishing emails, for the Arabic language. This proposed model is a combination of BiLSTM, GRU, and RF models. Combining these models improves classification accuracy and robustness by leveraging their complementary strengths. This combined approach achieves a higher accuracy rate when comparing two types of text when compared to individual models and traditional ensemble methods. Moreover, the interpretability of the RF model enhances the understanding of the classification process by providing insight into the most important features. A novel dataset has been introduced, the experiment results show that the proposed model outperformed ML classifiers and the DL models. The sentence structure also plays an important role in detecting and classifying the text on whether it is fake or real. In the future work, the dataset size will be increased and a bilingual dataset will be introduced for English and Arabic due to the fact that a significant number of phishing emails are also written in the English language. In addition, BERT, AraBERT, and transformer models could also be applied and evaluated as future work.

Acknowledgment

Thank you to the publisher for their support in the publication of this research article. We are grateful for the resources and platform provided by the publisher, which have enabled us to share our findings with a wider audience. We appreciate the efforts of the editorial team in reviewing and editing our work, and we are thankful for the opportunity to contribute to the field of research through this publication.

Funding information

The authors have not received any financial support or funding to report.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

Data Availability Statement

The dataset is available upon request.

References

AbdulNabi, I., & Yaseen, Q. (2021). Spam Email Detection Using Deep Learning Techniques. *Procedia Computer Science*, *184*, 853–858. https://doi.org/10.1016/j.procs.2021.03.107

Abu-Nimeh, S., Nappa, D., Wang, X., & Nair, S. (2007). A comparison of machine learning techniques for phishing detection. *Proceedings of the Anti-Phishing Working Groups 2nd Annual ECrime Researchers Summit*, 60–69. https://doi.org/10.1145/1299015.1299021

Al Daoud, E., Al Daoud, L., Asassfeh, M., Al-Shaikh, A., Al-Sherideh, A. S., & Afaneh, S. (2024). Enhancing Cybersecurity with Transformers: Preventing Phishing Emails and Social Media Scams. *Proceeding of the Conference on Dependable and Secure Computing (DSC)*, 31–36. https://doi.org/10.1109/dsc63325.2024.00017

Al-Charchafchi, A., Manickam, S., & Alqattan, Z. N. M. (2020). Threats Against Information Privacy and Security in Social Networks: A Review. *Advances in Cyber Security*, *1132*, 358–372. https://doi.org/10.1007/978-981-15-2693-0_26

Al-Otaibi, A. F., & Alsuwat, E. S. (2020). A study on social engineering attacks: Phishing attack. *International Journal of Recent Advances in Multidisciplinary Research*, *7*(11), 6374–6380.

Alsaidi, R. A. M., Yafooz, W. M. S., Alolofi, H., Taufiq-Hail, G. A.-M., Emara, A.-H. M., & Abdel-Wahab, A. (2022). Ransomware Detection using Machine and Deep Learning Approaches. *International Journal of Advanced Computer Science and Applications*, *13*(11), 95–102. https://doi.org/10.14569/ijacsa.2022.0131112

Al-Yozbaky, R. S., & Alanezi, M. (2023). Phishing Emails Detection Models: A Comparative Study. *JMCER*, *4*(3), 58–67.

An, P., Shafi, R., Mughogho, T., & Onyango, O. A. (2025). Multilingual email phishing attacks detection using OSINT and machine learning. *ArXiv, DBLP*, 1–10. https://doi.org/10.48550/arXiv.2501.08723

Bountakas, P., Koutroumpouchos, K., & Xenakis, C. (2021). A Comparison of Natural Language Processing and Machine Learning Methods for Phishing Email Detection. *Proceedings of the 16th International Conference on Availability, Reliability and Security*, 297–308. https://doi.org/10.1145/3465481.3469205

Brynjolfsson, E., Li, D., & Raymond, L. (2025). Generative AI at work. *The Quarterly Journal of Economics*, *140*(2), 889-942. https://doi.org/10.3386/w31161

Chetioui, K., Bah, B., Alami, A. O., & Bahnasse, A. (2022). Overview of Social Engineering Attacks on Social Networks. *Procedia Computer Science*, *198*, 656–661. https://doi.org/10.1016/j.procs.2021.12.302

Debnath, K., & Kar, N. (2022, May). Email spam detection using deep learning approach. In *2022 international conference on machine learning, big data, cloud and parallel computing (COM-IT-CON)* (Vol. 1, pp. 37-41). IEEE. https://doi.org/10.1109/com-it-con54601.2022.9850588

Divakarla, U., & Chandrasekaran, K. (2023). Predicting Phishing Emails and Websites to Fight Cybersecurity Threats Using Machine Learning Algorithms. *Proceeding of the International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)*, 1–10. https://doi.org/10.1109/smartgencon60755.2023.10442775

Fui-Hoon Nah, F., Zheng, R., Cai, J., Siau, K., & Chen, L. (2023). Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, *25*(3), 277–304. https://doi.org/10.1080/15228053.2023.2233814

Gangavarapu, T., Jaidhar, C. D., & Chanduka, B. (2020). Applicability of machine learning in spam and phishing email filtering: review and approaches. *Artificial Intelligence Review*, *53*(7), 5019–5081. https://doi.org/10.1007/s10462-020-09814-9

Ghourabi, A., Mahmood, M. A., & Alzubi, Q. M. (2020). A Hybrid CNN-LSTM Model for SMS Spam Detection in Arabic and English Messages. *Future Internet*, *12*(9), 156. https://doi.org/10.3390/fi12090156

Giri, S., Banerjee, S., Bag, K., & Maiti, D. (2022). Comparative Study of Content-Based Phishing Email Detection Using Global Vector (GloVe) and Bidirectional Encoder Representation from Transformer (BERT) Word Embedding Models. *2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, 1–6. https://doi.org/10.1109/iceeict53079.2022.9768612

Gogoi, B., & Ahmed, T. (2022). Phishing and Fraudulent Email Detection through Transfer Learning using pretrained transformer models. *Proceeding of the IEEE 19th India Council International Conference (INDICON)*, 1–6. https://doi.org/10.1109/indicon56171.2022.10040097

Gupta, B. B., Gaurav, A., Arya, V., Attar, R. W., Bansal, S., Alhomoud, A., & Chui, K. T. (2024). Advanced BERT and CNN-Based Computational Model for Phishing Detection in Enterprise Systems. *Peer-Reviewed Journal Article*, *141*(3), 2165–2183. https://doi.org/10.32604/cmes.2024.056473

Hassanpour, R., Dogdu, E., Choupani, R., Goker, O., & Nazli, N. (2018). Phishing e-mail detection by using deep learning algorithms. *Proceedings of the ACMSE 2018 Conference*, 208–213. https://doi.org/10.1145/3190645.3190719

Jamal, S., & Wimmer, H. (2023). *An improved transformer-based model for detecting phishing, spam, and ham: A large language model approach*. https://doi.org/10.48550/arXiv.2311.04913

Kaddoura, S., Alfandi, O., & Dahmani, N. (2020). A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach. *Proceeding of the IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 193–198. https://doi.org/10.1109/wetice49692.2020.00045

Karki, B., Abri, F., Namin, A. S., & Jones, K. S. (2022). Using Transformers for Identification of Persuasion Principles in Phishing Emails. *Proceeding of the International Conference on Big Data (Big Data)*, 2841–2848. https://doi.org/10.1109/bigdata55660.2022.10020452

Khan, N. A., Brohi, S. N., & Zaman, N. Z. (2023). Ten deadly cyber security threats amid COVID-19 pandemic. *TechRxiv Preprint*, 1–19.

Li, H., Yang, J., Li, Y., & Li, K. (2024, November). Email phishing attack detection based on BERT transformer model. In *International Conference on Optics, Electronics, and Communication Engineering (OECE 2024)* (Vol. 13395, pp. 1040-1045). SPIE.

Magdy, S., Abouelseoud, Y., & Mikhail, M. (2022). Efficient spam and phishing emails filtering based on deep learning. *Computer Networks*, *206*, 108826. https://doi.org/10.1016/j.comnet.2022.108826

Masri, A., & Al-Jabi, M. (2023). A novel approach for Arabic business email classification based on deep learning machines. *PeerJ Computer Science*, *9*, e1221.

Mehdi Gholampour, H. N., Vinayakumar R, Harikrishnan NB, Vinayakumar R, Soman KPHarikrishnan NB, Vinayakumar R, Soman KP, & Verma, R. M. (2018). A Machine Learning approach towards Phishing Email. *Proceedings of the Anti-Phishing Pilot at ACM International Workshop on Security and Privacy Analytics (IWSPA-AP)*, 455–468.

Mohammed, M. A., Mostafa, S. A., Obaid, O. I., Zeebaree, S. R. M., Ghani, G., Mustapha, A., Fudzee, M. F. M., Jubair, M. A., Hassan, M. H., Ismail, A., Ibrahim, D. A., & AL-Dhief, F. T. (2019). An Anti-Spam Detection Model for Emails of Multi-Natural Language. *Journal of Southwest Jiaotong University*, *54*(3), 358–369. https://doi.org/10.35741/issn.0258-2724.54.3.6

Park, J. Y., & Kim, T.-S. (2025). An Automated Scenario Generation Model for Anti-phishing using Generative AI. *Proceeding of the IEEE International Conference on Big Data and Smart Computing (BigComp)*, 368–370. https://doi.org/10.1109/bigcomp64353.2025.00073

Ripa, S. P., Islam, F., & Arifuzzaman, M. (2021). The Emergence Threat of Phishing Attack and the Detection Techniques Using Machine Learning Models. *Proceeding of the International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, 1–6. https://doi.org/10.1109/acmi53878.2021.9528204

Salahdine, F., El Mrabet, Z., & Kaabouch, N. (2021). Phishing Attacks Detection A Machine Learning-Based Approach. *Proceedings of the Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 1–6. https://doi.org/10.1109/uemcon53757.2021.9666627

Salloum, S., Gaber, T., Vadera, S., & Shaalan, K. (2023). A New English/Arabic Parallel Corpus for Phishing Emails. *ACM Transactions on Asian and Low-Resource Language Information Processing*, *22*(7), 1–17. https://doi.org/10.1145/3606031

Somesha, M., & Pais, A. R. (2024). *Phishing Classification Based on Text Content of an Email Body Using Transformers*. *1075*, 343–357. https://doi.org/10.1007/978-981-99-5091-1_25

Srinivasan, S., Ravi, V., Alazab, M., Ketha, S., Al-Zoubi, A. M., & Kotti Padannayil, S. (2021). *Spam Emails Detection Based on Distributed Word Embedding with Deep Learning*. *919*, 161–189. https://doi.org/10.1007/978-3-030-57024-8_7

Uddin, M. A., Sarker, I. H., & Mahiuddin, M. (2024). *An explainable transformer-based model for phishing email detection: A large language model approach*. https://doi.org/10.48550/arXiv.2402.13871

Yafooz, W. M. S. (2024). Enhancing Business Intelligence with Hybrid Transformers and Automated Annotation for Arabic Sentiment Analysis. *International Journal of Advanced Computer Science and Applications*, *15*(8), 1–7. https://doi.org/10.14569/ijacsa.2024.0150821

Yafooz, W. M. S., Hizam, E. A., & Alromema, W. A. (2021). Arabic Sentiment Analysis on Chewing Khat Leaves using Machine Learning and Ensemble Methods. *Engineering, Technology & Applied Science Research*, *11*(2), 6845–6848. https://doi.org/10.48084/etasr.4026

Yasin, A., & Abuhasan, A. (2016). An intelligent classification model for phishing email detection. *arXiv preprint arXiv:1608.02196*. https://doi.org/https://doi.org/10.48550/arXiv.1608.02196

Zamir, A., Khan, H. U., Iqbal, T., Yousaf, N., Aslam, F., Anjum, A., & Hamdani, M. (2020). Phishing web site detection using diverse machine learning algorithms. *The Electronic Library*, *38*(1), 65–80. https://doi.org/10.1108/el-05-2019-0118