Enhancing Arabic Phishing Email Detection: A Hybrid Machine Learning Based on Genetic Algorithm Feature Selection

Amjad A. Alsuwaylimi
Department of Information Technology-Faculty of Computing and Information Technology,
Northern Border University, Rafha 91911, Saudi Arabia

***Abstract*—Recently, owing to widespread Internet use and technological breakthroughs, cyber-attacks have increased. One of the most common types of attacks is phishing, which is executed through email and is a leading cause of the recent surge in cyber-attacks. These attacks maliciously demand sensitive or private information from individuals and companies. Various methods have been employed to address this issue by classifying emails, such as feature-based classification and manual verification. However, these methods face significant challenges regarding computational efficiency and classification precision. This work presents a novel hybrid approach that combines machine learning and deep learning techniques to improve the identification of phishing emails containing Arabic content. A genetic algorithm is employed to optimize feature selection, thereby enhancing the performance of the model by effectively identifying the most relevant features. The novel dataset comprises 1,173 records categorized into two classes: phishing and legitimate. A number of empirical investigations were carried out to assess and contrast the performance outcomes of the proposed model. The findings reveal that the proposed hybrid model outperforms other machine learning classifiers and standalone deep learning models.**

Keywords*—Machine learning; phishing email; BiLSTM; Arabic content-based

I. INTRODUCTION

Phishing emails are powerful instruments for scammers who want to make money by taking advantage of their feelings and trust. Attackers can achieve this by pretending to be trustworthy organizations such as banks or government authorities, and creating a sense of urgency or panic in their victims [1-3]. Phishing attacks offer scammers the possibility of high returns on investment because they are inexpensive and easy to perform. These assaults exploit human susceptibilities, such as the need for self-preservation and curiosity, while avoiding security systems through emotional manipulation. Therefore, one must learn how phishing emails operate if they are to protect themselves from falling victims to these dishonest campaigns. Phishing emails come in all shapes and sizes but share the same end, outsmarting the target to perform their bidding. One type of phishing email is scam email, which disguises itself as a legitimate organization, often a bank or another well-known business, requesting the target to click on links or share personal information [4,5]. The other type is spear-phishing, which uses intimate and personal information about the recipient to appear more authentic and trustworthy. Furthermore, another type exists, which is called the clone phishing email, which copies a real email the victim has already received, and then sends it back directly with a malicious attachment or URL. Furthermore, cybercriminals imitate high-ranking officers through CEO fraud emails to deceive employees and send urgent payments or sensitive data to them. Meanwhile, phishing emails direct their victims to fake websites, where they attempt to acquire their financial details or log-in credentials. These kinds of deceitful electronic mail play on either trust, urgency, or curiosity to make unsuspecting individuals compromise their own safety.

Researchers have been investigating various techniques, such as Natural Language Processing (NLP) [6-8], Machine Learning (ML) [9-16] and Deep Learning (DL) [17-21] to deal with the significant challenge posed by phishing emails in the field of cybersecurity. NLP methods are used for analyzing the text of emails and detecting linguistic indications or patterns indicative of phishing attempts. ML algorithms can be trained with large datasets containing examples of phishing emails to identify the commonalities between them and subsequently automate the recognition and classification of new ones into predefined categories based on these observed regularities. DL, specifically Convolutional Neural Networks (CNNs) alongside Recurrent Neural Networks (RNNs), makes it possible to detect phishing emails more precisely by learning complex features from both email content and its metadata.

Despite these efforts, achieving time efficiency and accuracy with these techniques remains a significant challenge in the field. The processing of extensive features necessitates substantial memory and computational time, further complicating the development of effective email classification techniques. Furthermore, the proliferation of big data presents also significant challenges for these techniques, mainly due to increased training durations from the heightened computational demands of processing large datasets. Both the larger sample size and greater dimensionality of the data contribute to this issue. High dimensionality particularly affects inference times due to the added computational burden of feature extraction. In real-time phishing detection models, these factors can negatively impact user experience and compromise the effectiveness of deployed techniques. Consequently, optimizing these techniques to balance accuracy and computational efficiency is an ongoing area of research. To address these challenges, researchers and practitioners often use feature reduction or feature selection techniques. By

strategically selecting a subset of features, these methods reduce computational costs [22].

The existing literature describes several feature selection algorithms that are commonly used, such as tree-based methods [23], selectKBest [24], Recursive Feature Elimination [25], LASSO [26], Principal Component Analysis [27], and Evolutionary Algorithms [28]. Evolutionary algorithms utilize a population of candidate solutions that progressively adapt over time via number of operators: selection, mutation, and recombination mechanisms to identify optimal solutions. Their robustness, flexibility, ability address complex, non-differentiable, and non-linear problems with versatility and resilience, parallelization capabilities, and multi-objective optimization potential make them particularly advantageous for various optimization tasks, including feature selection.

Genetic Algorithms (GAs), a subset of EAs, emulate natural selection to solve optimization problems. GAs have demonstrated significant success in addressing diverse optimization challenges, including feature selection. GAs have been employed in feature selection tasks, whereby the features are represented as chromosomes, and various genetic operators, including selection, mutation, and crossover are applied to evolve candidate solutions. The efficacy of each proposed solution is evaluated using a predefined objective function, and the optimization procedure continues iteratively until a satisfactory feature subset is obtained.

In the context of our research, the primary aim is to enhance detection accuracy and recall while simultaneously reducing processing time by minimizing the feature set. This work introduces a new hybrid approach for email classification. Specifically, we propose a method that integrates machine learning (ML) and deep learning (DL) methodologies to detect and categorize content-based phishing emails in the Arabic language, utilizing a GA to identify and select the best features. Our research significantly advances the current state of knowledge in this domain through several key contributions. First, we develop a hybrid model combining Random Forest (RF) and Bidirectional Long Short-Term Memory (BiLSTM) techniques, enhancing the detection of Arabic-based phishing emails. Additionally, we create and introduce a novel dataset comprising 1,173 Arabic content-based emails, providing a valuable resource for future research in Arabic phishing email detection. Furthermore, we innovatively apply a genetic algorithm to optimize feature selection and reduce feature dimensionality, thereby improving the efficiency and accuracy of the model. Finally, we conduct a comprehensive evaluation of the impact of genetic algorithms on model performance, demonstrating their effectiveness in enhancing accuracy relative to ML classifiers and DL models.

The remainder of this paper is organized as follows: Section II reviews related studies on phishing detection techniques, emphasizing recent advancements and the integration of machine learning (ML) and deep learning (DL) methodologies. Section III provides a detailed formulation of the problem addressed in this research. Section IV outlines the methodology and materials used, describing the dataset, pre-processing techniques, and the proposed hybrid model combining Random Forest and BiLSTM with Genetic Algorithm Feature Selection. Section V presents the experimental results and discussion, comparing the performance of the proposed model with baseline machine learning classifiers and deep learning models. Finally, Section VI concludes the paper by summarizing the key findings and suggesting directions for future research in the field of phishing email detection.

II. Related Studies

A. Phishing Email Detection Approaches

The ongoing evolution and increasing prevalence of phishing attacks necessitate continued research in detection methodologies. As these threats become more diverse, studies examining detection methods are concurrently updated and enhanced to address emerging challenges. The advent of ML and DL techniques, having proven their effectiveness in various problem domains, has led to their adoption in phishing detection research [16]. Researchers in this field have increasingly employed these advanced computational methods due to their competence in detecting complex patterns and addressing emerging attack strategies. Current studies primarily concentrate on identifying phishing emails and websites, utilizing ML and DL methods to enhance detection accuracy and minimize false positives. This transition to more advanced analytical approaches reflects the field's adaptation to the growing sophistication of phishing attacks and highlights the need for ongoing innovation in cybersecurity defences.

The authors of [1] explored the effectiveness of a transformer model named Bidirectional Encoder Representations from Transformers (BERT) and word embeddings for spam email classification. The findings were compared with those of Deep Neural Network (B-DDN) model, which includes Naive Bayes classifiers, k-nearest Neighbors, the BiLSTM layer. The model was tested and trained using two public datasets. The BERT transformer model with English Wikipedia and BookCorpus was used as the training data, with F1-score of 98.66% and an accuracy of 98.67%. This work investigated spam email detection using contextual word embeddings, attention layers, and deep learning techniques.

The study by [11] explored the use of ML-based spam detection models. Various ML methods were employed to categorize SMS messages as legitimate or spam, including Naive Bayes (NB), Decision Trees (DT), Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Random Forest (RF), and Artificial Neural Networks (ANN). The research utilized several datasets, such as the Spam SMS Dataset and UCI SMS database, along with a custom-crawled dataset. For real-world data, both English and Indonesian languages were considered. Application of these models to the datasets yielded promising accuracy rates: SVM achieved 97.4% precision, CNN demonstrated 99.19% accuracy, and Weka SVM exhibited 99.3% accuracy for spam classification. Preprocessing techniques, including tokenization, removal of stop words, and feature extraction, were identified as methods to enhance accuracy. Similarly, focusing on SMS spam detection, the authors in [17] introduced a Hybrid Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) approach for the detection of spam within SMS

messages in Arabic and English languages. This approach was compared to conventional machine learning algorithms, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM) and DL models such as LSTM and CNN individually. The study utilized two datasets: the SMS Spam data collection obtained from the UCI Machine Learning Repository containing a collection of Arabic messages and English messages sourced from local mobile phones. This hybrid model was designed to handle mixed linguistic content in Arabic and English communications. The proposed approach achieved a notable accuracy of 98.37% in categorizing SMS messages as spam or legitimate, outperforming all other machine learning algorithms tested in the study.

The study of [12] introduced two novel algorithms, FMMPED and FMPED, specifically designed to enhance phishing email detection capabilities. These algorithms were developed utilizing undersampling techniques and ensemble learning methodologies. To simulate real-world email environments, the researchers employed a dataset with a 10:1 ratio of malicious to legitimate emails. The study, published in a highly technical journal focused on machine learning and cybersecurity, employs sophisticated terminology commensurate with its computer science orientation. The proposed algorithms demonstrated superior performance compared to traditional machine learning and deep learning approaches, accomplishing accuracy and an impressive F1-score of 0.9945. The authors of [16] employed 13 machine learning classifiers to differentiate between spam and non-spam emails, including Bayesian methods, Decision Trees, Random Forest, Support Vector Machines, Decision Tables, and Bagging. The evaluation methodology incorporated two datasets: Spam Corpus and Spambase. The Random Forest classifier demonstrated superior performance on the Spam Corpus dataset, surpassing all other classifiers implemented using the Python programming language, with an accuracy rate of 99.91%.

The study of [15] investigated the implementation of diverse machine learning approaches for spam email classification. The study utilized traditional spam detection methods, which include Support Vector Machine, Naïve Bayes, Random Forest, and K-Nearest Neighbour models. The research incorporated a comprehensive collection of email datasets and real-life scenarios of varying sizes and formats from multiple sources, such as Kaggle and Sklearn. The study emphasized document pre-processing, encompassing cleaning, integration, transformation, and reduction. Additionally, tokenization and stop word removal were considered. The problem statement effectively elucidated the significance of this research.

In the study by [29], they utilized email samples as the dataset and employed federated learning approaches with THEMIS and BERT models for phishing email detection. Due to architectural constraints, the BERT model focused specifically on the email body. Both training and evaluation of the models were conducted in English. THEMIS achieved a test accuracy of 97.9% for federated learning with five clients at epoch 45, while BERT attained a test accuracy of 96.1% for federated learning with five clients at epoch 15. It is noteworthy that accuracy rates varied depending on the client. The authors in [18] introduced a new Intelligent Cybersecurity Phishing Email Detection and Classification (ICSOA-DLPEC) model that leverages n-gram feature extraction, a Gated Recurrent Unit (GRU) model, and a Compressive Sensing (CS) algorithm for optimal parameter tuning. They evaluated its performance using a standard dataset, achieving impressive accuracy rates between 98.46% and 99.72% across different training and testing data volumes. This study also discusses online safety terms and deep learning concepts. When compared to other models such as LSTM, CNN, and THEMIS, the ICSOA-DLPEC model demonstrated superior performance in being correct, precise, able to recall, and in its F-score.

Tong et al. 2021 [30], proposed a capsule network model with long-short attention for Chinese spam detection. The authors used the Trec06c dataset and received an accuracy of 98.72% on an unbalanced dataset and a 99.30% accuracy on a balanced dataset. Similarly, Li et al. [31], introduced an LSTM based phishing email detection method and tested it on a dataset of 29,942,735 emails from an enterprise mail server with both Chinese and English content. The model got nearly 100% accuracy in classifying phishing emails. Wu et al. [32], evaluated ChatGPT's spam detection against baseline models like SVM, LR, NB and BERT on the English Email Spam Detection (ESD) dataset and the low-resourced Chinese Spam Dataset (CSD) and the results were achieved in accuracy was 83% and 86% accuracy on two different experiments.

B. Feature Selection Methods

The process of feature selection is a critical component in the fields of ML and data analytics. It involves identifying and extracting the most salient subset of attributes from the complete set of features present within a given dataset. The primary goals of feature selection are to reduce dimensionality, improve model performance, enhance generalization, and provide better interpretability of results [33]. In the context of phishing email detection, the feature selection process involves the identification and selection of the most informative characteristics of emails that effectively differentiate between legitimate and phishing messages. This process is essential for developing accurate and efficient phishing detection systems that can adapt to evolving threats. The selected features must be relevant to phishing indicators, adaptable to new techniques, and effective across different languages and cultural contexts, especially in multilingual environments [34-36].

Common types of features in phishing email detection include linguistic features, such as writing style and urgency indicators; structural features, like email header information and HTML content; contextual features, including sender reputation and domain age; and behavioral features, such as user interaction patterns and email sending times [37-40]. The selection process must strike a balance between content-based and metadata-based features to create a comprehensive and robust feature set.

Various methods are employed for feature selection in phishing detection. These include filter methods, which use statistical approaches to select features according to their relationship with the target variable; wrapper methods, which evaluate feature subsets using specific machine learning

models; embedded methods, which integrate feature selection into the model training process; and evolutionary algorithms, such as Genetic Algorithms, which optimize feature subsets based on multiple objectives [41,42].

Recent advancements in deep learning introduced methods that have the ability to automatically derive pertinent features directly from raw email data, sometimes mitigating the requirement for manual feature selection [43]. However, in many applications, especially those dealing with multilingual content or requiring model interpretability, careful feature selection remains a critical step in developing effective phishing detection systems. By focusing on the most discriminative features, researchers and cybersecurity professionals can develop more accurate, efficient, and adaptable phishing detection models. This approach not only enhances overall email security but also helps protect users from increasingly sophisticated phishing attacks. The ongoing challenge in this field is to continuously refine feature selection methods to keep pace with evolving phishing techniques and maintain robust detection capabilities across diverse linguistic and cultural contexts.

C. Genetic Algorithm-Based Feature Selection Approaches for Phishing Email Detection

Genetic Algorithms (GAs) are evolutionary optimization techniques that draw inspiration from the mechanisms of natural selection and genetics. GAs have become powerful tools for solving complex optimization problems across various domains [44]. In the context of ML and DL, GAs have shown remarkable efficacy in feature selection tasks, offering a robust approach to identifying optimal subsets of features from large and complex datasets [45]. The core premise of genetic algorithms (GAs) is their capacity to iteratively refine a population of candidate solutions over successive generations. Each solution, or ``chromosome" denotes a possible subset of features. The algorithm evaluates these chromosomes based on a fitness function, which typically measures the performance of a machine learning model using the selected features. Through processes mimicking genetic inheritance – selection, crossover, and mutation – GAs iteratively refine the population, converging towards an optimal or near-optimal feature subset [46].

In the domain of phishing email detection, the application of GAs for feature selection presents a promising approach to enhancing detection accuracy while optimizing computational efficiency. Phishing emails often contain subtle and evolving characteristics, making the selection of relevant features a critical and challenging task. Given these challenges, GAs can effectively navigate this complex feature space, considering various combinations of linguistic, structural, and behavioral email attributes to identify the most discriminative feature set [47]. The use of GAs in phishing email detection typically involves encoding email features as binary strings, where each element indicates the presence or absence of a specific feature. Moreover, the fitness function may incorporate multiple objectives, such as maximizing detection accuracy, minimizing false positives, and reducing the overall number of features used. Consequently, this multi-objective optimization approach allows for a balanced solution that meets the often-conflicting goals of high accuracy and computational efficiency [48].

Several studies have demonstrated the effectiveness of GA-based feature selection in phishing detection systems. For instance, the authors in [49] employed a GA to optimize feature selection for their email classification system, resulting in improved accuracy and reduced computational complexity. Similarly, the study in [50] utilized a GA-based approach to identify the most pertinent features for their phishing website detection model, achieving high accuracy rates with a compact feature set.

The adaptability of GAs makes them particularly suitable for the dynamic nature of phishing attacks. As cybercriminals continually evolve their techniques, GA-based feature selection can be periodically re-run on updated datasets, ensuring that the selected features remain relevant and effective against new phishing strategies [51]. This adaptability is crucial in maintaining the long-term effectiveness of phishing detection systems. Moreover, the interpretability of GA-selected feature subsets can provide valuable insights into the most significant indicators of phishing attempts. This transparency can aid security professionals in understanding evolving phishing tactics and developing more targeted prevention strategies [52]. Therefore, the application of Genetic Algorithms to feature selection in phishing email detection offers a powerful means of enhancing detection accuracy, improving computational efficiency, and adapting to evolving threats. As phishing attacks continue to grow in sophistication, the role of advanced feature selection techniques like GAs becomes increasingly critical in developing robust and effective defence mechanisms.

III. Problem Formulation

This study involves a problem-formation process for binary text classification. The problem can be framed as classifying emails into two distinct classes: phishing emails and legitimate emails. Let $\{D\}$ be a collection of emails $\{E\}$, known as a dataset. Let $\{D\}$ consist of E_{phishing} and $E_{\text{legitimate}}$. Feature matrices to be used as inputs for the models are derived from *D*, where rows represent emails (content-based) and columns represent features. Let X be the input feature and Y be the target variable, which can be represented as $D = \{X_1, Y_1, X_2, Y_2, X_3, Y_3 \dots \dots \dots \dots X_n, Y_n\}$, where n denotes the total number of words in V, X_1 is the feature vector (V), and Y_1 is the label.

The D model is divided into two parts, Dtraining and Dtesting. The Dtraining is used for training the model, whereas the Dtesting is used to assess and test it. In both cases, the learn a function F(X), which makes the decision in to detect emails whether the input E is legitimate or phishing, as shown in the mathematical formula (1).

$$F(X) = f(X, Pa) \tag{1}$$

where X is the input features and the Pa is the parameter of the model.

IV. METHODOLOGY AND MATERIALS

This section details the five-phase methodological framework employed in this study. Each phase contributes to the development and evaluation of the machine learning model for email classification. A visual representation of these stages is provided in Fig. 1.

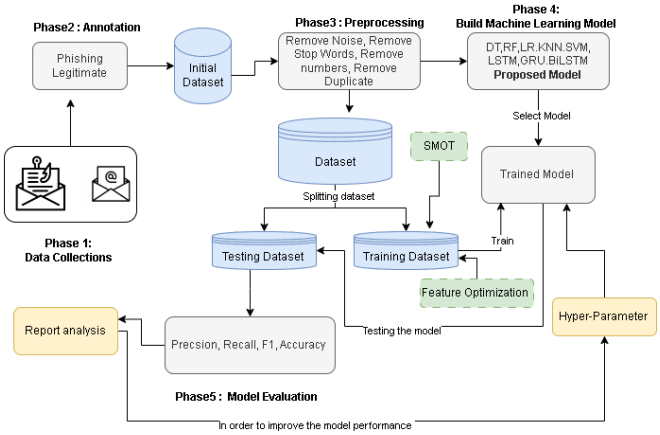


Fig. 1. Methods and Phases of this study.

A. Phase 1: Data Collection

Due to the scarcity of Arabic datasets for phishing emails, this study employed an initial method to construct the study's dataset. Data was gathered from a variety of personal email collections, ensuring a diverse range of sources. To further enrich the dataset, phishing emails were generated using ChatGPT, complemented by additional emails crafted manually. This compilation process ensured a diverse and representative dataset. Upon finalizing the collection, all emails were organized and stored in a CSV file, which was subsequently sent to annotators for detailed processing.

B. Phase 2: Annotation Process

In the annotation process, three native Arabic speakers assisted in the annotation process by classifying the emails into two categories: phishing and legitimate. In addition, Cohen's Kappa was utilized in order to measure the degree of agreement between the annotators [53,54] based on the mathematical formula (2).

$$K = \frac{P_o - P_e}{1 - P_e} \quad (2)$$

P_o is the proportion agreement between judges, while P_e is the expected agreement proportion by chance. Thus, the K is 81% which indicates prefect agreement between the judges (raters). In the end, Table I provides information on the size of the final dataset.

TABLE I. DATASET DESCRIPTION

Factors	No. of Emails	Maximum Words	Minimum Words
Phishing	861	166	23
Legitimate	312	178	46
Total	**1,173**		

C. Phase 3: Pre-Processing Steps

Pre-processing phase were performed on the Arabic dataset before it was used in the models. These steps included the removal of stop words from the Arabic language, noise such as HTML tags, and duplicate messages or emails.

All Arabic-specific stop words were systematically eliminated to enhance the efficacy of subsequent text analysis procedures. Stop words, defined as high-frequency lexical items that typically carry minimal semantic content, are routinely excluded to optimize the quality and relevance of the dataset. Table II presents a representative sample of common Arabic-specific stop words. Additionally, for noise removal, all irrelevant text, including HTML tags, was removed from the emails. This step ensured that only meaningful textual information was retained, making the data more suitable for analysis. Furthermore, duplicates were identified and removed to ensure that each email in the dataset was unique, thereby preventing redundancy and improving the accuracy of the analysis.

TABLE II. COMMON EXAMPLES OF ARABIC-SPECIFIC STOP WORDS

Arabic-specific stop words	Meaning in English
و	and
في	in
من	from
على	on
إلى	to
أو	Or

D. Phase 4: Build Machine Learning Model

Prior to detailing the process of this phase, we provide a brief overview of the ML classifiers and DL models and GA-based feature selection employed in this investigation to assess the efficacy of our proposed approach.

1) Machine learning classifiers: Support Vector Machine (SVM): SVM is a supervised learning algorithm that analyze data to perform regression and classification tasks. The algorithm operates by identifying the optimal hyperplane that best separates the given dataset into distinct classes. SVMs are particularly adept in high-dimensional feature spaces and have been widely employed in text categorization tasks including sentiment analysis and spam detection [55,56].

a) Decision Tree (DT): DT is a non-parametric supervised learning algorithm used for regression and classification. DT works by splitting the data into a number of subsets according to the most significant differentiators in the input features. Decision Trees are highly interpretable and easily understood, making them indispensable tools in decision-making processes [57].

b) Logistic Regression (LR): LR is a statistical technique that employs a logistic function to analyze a binary dependent variable in a model. LR is a widely adopted technique for binary classification problems, including applications such as email spam detection and medical diagnosis [58-60].

c) Random Forest (RF): RF is a widely utilized machine learning algorithm that generates numerous decision trees during the training process and predicts the class that is the statistical mode of the classes output by the individual trees. RF helps overcome the tendency of individual decision trees to overfit to the training data, resulting in a robust and accurate model [61,62].

2) Deep learning models:

a) Long Short-Term Memory (LSTM): LSTM is a specialized form of Recurrent Neural Network (RNN) that can effectively model and capture long-term temporal dependencies within sequential data. It addresses the vanishing gradient issue that conventional RNNs face by leveraging memory cells to retain information over extended durations. LSTMs have demonstrated effectiveness in tasks involving sequential data, including time-series forecasting and natural language processing [63].

b) Gated Recurrent Units (GRU): GRU is another variety of RNN which is similar to LSTM but with a simplified architecture. GRU integrates the forget and input gates into a single update gate, resulting in improved computational efficiency without compromising its overall performance. GRUs are used in various applications requiring sequence modeling [64].

c) Bidirectional Long Short-Term Memory (BiLSTM): BiLSTM is a bidirectional variant of the LSTM architecture, which enhances performance by analyzing input sequences in both the backward and forward temporal directions. BiLSTM enables the model to have access to future and past context information, making it particularly useful in tasks like machine translation and text generation [65,66].

3) Genetic algorithm for feature selection: Feature selection tasks for content-based email message analysis have proven to be highly effective when using GAs [67, 68], which are optimization techniques. As part of detecting phishing emails, feature selection involves recognizing the most relevant attributes present within the email content, including textual patterns, keywords, and structural characteristics. A genetic algorithm then generates a population of potential solutions and iteratively evolves them through processes such as selection, crossover, and mutation.

The GA efficiently narrows down the feature set to those features that contribute most significantly to distinguishing phishing emails from legitimate ones, by evaluating the fitness of each solution. This process enhances the efficiency of the model by reducing dimensionality and eliminating irrelevant or redundant features, leading to improved accuracy and faster computation. In the proposed model, the GA for feature selection played an important role in optimizing the input features, thus enhancing the overall performance and robustness of the Random Forest and BiLSTM hybrid approach.

To start the GA for detecting normal and phishing emails, a dataset containing both types of emails is loaded, and relevant features are extracted. An initial population of candidate individuals is generated, with each individual denoting a set of features. The classification accuracy is evaluated based on the fitness of these chromosomes. The algorithm selects parents, performs mutations and crossovers to create offspring, and evaluates their fitness, replacing the old population with new offspring until a satisfactory fitness level is achieved or a specified number of generations is reached. Ultimately, the GA outputs the highest-performing chromosome, representing the optimal feature set for email classification, as depicted in Fig. 2.

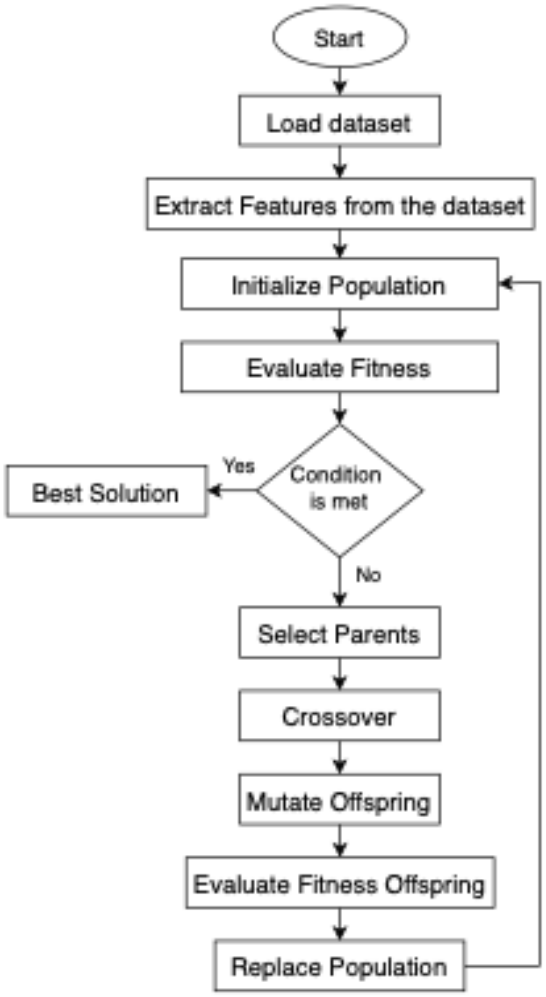


Fig. 2. Genetic algorithm flowchart.

4) The process of phase 4: In phase 4, the proposed model for Arabic detecting phishing emails is designed with a multi-stage process, leveraging both machine learning and deep learning techniques optimized through feature selection using a genetic algorithm, as shown in Fig. 3. The proposed hybrid approach provides a number of significant benefits. It integrates RF and BiLSTM with a GA for feature optimization. The model takes advantage of the advantages of both methods by combining the outputs of RF and BiLSTM, improving prediction performance by capturing a variety of data features. Sequential data is best captured by BiLSTM, unlike RF's ensemble approach reduces variance and minimizes overfitting to produce robust predictions. By

choosing the most pertinent features, the addition of GA for feature optimization further improves the model's efficiency and reduces its dimensionality, resulting in quicker training durations. By combining the advantages of both sequential pattern expertise from BiLSTM and structured data handling from RF, this hybrid technique also enhances generalization. The model performs well against noise, and is suitable for various data types. Its scalability and ability to handle large datasets make it a valuable tool in machine learning.

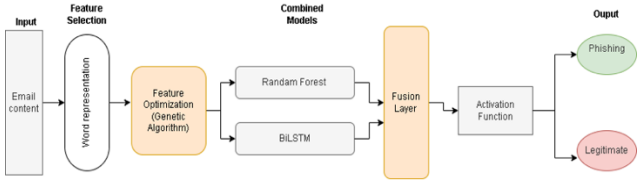


Fig. 3. Flowchart of the proposed model.

Initially, the process begins with the input of email content, which serves as the raw data for analysis. This content undergoes transformation into a word representation format, capturing the essential textual features needed for further processing. In the feature selection stage, a GA is employed to optimize the features derived from the word representation. This optimization step is crucial as it enhances the model's capacity to recognize relevant features that are indicative of phishing attempts.

The process begins with the input of email content, which is transformed into numerical vectors through word representation. This transformation allows the textual data to be processed effectively by subsequent machine learning algorithms. In the next stage, feature selection is carried out using a genetic algorithm. This step is crucial as it identifies and selects the most relevant features from the word representations, enhancing the model's capability to distinguish between phishing and legitimate emails.

Following feature optimization, the refined features are fed into two different machine learning models: BiLSTMb and RF. The RF model brings robustness and the capability to handle a large number of features effectively, while the BiLSTM model leverages its capacity to capture sequential dependencies within the email content, making it well-suited for analyzing the context and order of words [69].

The outputs from both the RF and BiLSTM models are then integrated in a fusion layer. This layer combines the strengths of both models, creating a more comprehensive understanding of the email content. The combined data is subsequently processed through an activation function, which refines the decision-making mechanisms employed by the model.

Finally, the model produces its output by classifying the email as either phishing or legitimate. This classification is the culmination of the entire process, providing a clear and actionable result based on the sophisticated analysis of the email content. Through this structured approach, the model effectively identifies phishing emails, ensuring robust detection by integrating feature optimization, machine learning, and deep learning methodologies.

E. Model Evaluation

This section introduces the most widely-used metrics to evaluate the performance of the proposed model. The metrics employed include Precision, Recall, F1-score, Accuracy, and Area Under the Receiver Operating Characteristic (AUC-ROC) curve.

Precision evaluates the accuracy of positive classifications, specifically the proportion of emails categorized as phishing that were correctly identified. This metric is crucial for evaluating the model's capacity to minimize the occurrence of false positive results. Mathematically, Precision is defined in the mathematical formula (3):

$$Precision = \frac{\text{Correct emails retrieved}}{All\ retrieved\ emails} \quad (3)$$

In this formula, “Correct emails retrieved” represents the number of emails accurately identified as phishing, and “All retrieved emails” represents the total number of emails classified as phishing by the model. The model's high level of precision is evidenced by a diminished rate of false positive outcomes, thus ensuring that most of the emails flagged as phishing are indeed malicious.

Recall metric assesses the model's capability to detect all pertinent instances of phishing emails. It represents the proportion of correctly identified phishing emails relative to the total number of actual phishing emails. A high recall value indicates that the model effectively identifies most phishing emails, thereby minimizing the occurrence of false negatives. The mathematical formula (4) is for calculating Recall:

$$Recall = \frac{\text{Correct emails retrieved}}{\text{All relvent emails}} \quad (4)$$

The F1-score is a composite metric that synthesizes the trade-off between precision and recall. It represents the harmonic mean of these two measures, offering a holistic evaluation that is particularly valuable in the context of unbalanced datasets. A high F1-score indicates that the model effectively balances the identification of phishing emails while minimizing both false negative and false positive classifications. The mathematical formula (5) calculates the F1-score:

$$F1 - Score = 2 \text{ x } \frac{Recall \text{ x } Precision}{Recall + Precision} \quad (5)$$

This metric is highly useful for comprehensively evaluating the model's performance in scenarios where both incorrectly identifying positive cases and failing to detect true positive cases are crucial considerations.

Accuracy metric evaluates the overall correctness of the model by quantifying the ratio of correctly classified emails (both phishing and legitimate) to the total number of emails. Accuracy is calculated by the mathematical formula (6):

$$Accuracy = \frac{Number\ of\ correct\ emails}{Total\ number\ of\ predication} \quad (6)$$

In this context, “Number of correct emails” includes both correctly identified phishing and legitimate emails, while “Total number of predictions” is the total number of emails that have been classified by the model. High accuracy denotes

that the model performs well in classifying both phishing and legitimate emails correctly.

In addition to the above metrics, the Area Under the Receiver Operating Characteristic (AUC-ROC) is employed to measure the model performance. The AUC-ROC provides a single value that represents the capability of the model to differentiate between classes across various threshold settings. The ROC curve graphically depicts the balance between the true positive rate and the false positive rate across varying decision criteria. A higher AUC value signifies improved model performance, with an AUC of 1.0 denoting a model with perfect classification performance.

V. Results and Discussion

This section details the experimental settings, findings, and analysis. The proposed model is evaluated in comparison to the baseline ML classifiers and DL models to distinguish between phishing and authentic emails for future identification.

A. Experimental Settings

The experiments were conducted using Google Colaboratory, which has been utilized for all the experiments on ML classifiers, DL models, and the proposed model. We used three Python libraries to conduct the experiments: Matplotlib, Scikit-learn (sklearn), and DEAP. Using Matplotlib, we created plots and charts to visualize the data and results. Sklearn was used for ML classifiers, pre-processing steps, and splitting the dataset. Finally, DEAP was used for the genetic algorithm. The hyperparameters for the machine learning (ML) classification algorithm are enumerated in Table III, while Table IV delineates the architectural and training parameters employed in the deep learning (DL) model.

TABLE III. Hyper-Parameter of ML Models

Model	Hyper-parameter	Default Value
Random Forest	n_estimators	100
	criterion	“gini”
	max_depth	None
	min_samples_split	2
	min_samples_leaf	1
Decision Tree	criterion	"gini"
	splitter	"best"
	max_depth	None
	min_samples_split	2
	min_samples_leaf	1
SVM	C	1.0
	kernel	"rbf"
	gamma	"scale"
Naive Bayes	priors	None
Logistic Regression	solver	"lbfgs"
	max_iter	100

TABLE IV. Hyper-Parameter of DL Models

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

B. Experimental Results

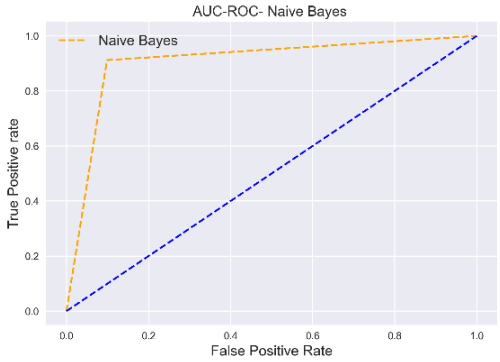
Numerous experiments have been conducted using ML, DL, and the proposed model. In these experiments, genetic algorithms are used to select the best features. Tables V and VI present a comparison of precision, recall, F1-score, and accuracy, respectively. Table V presents a comparative analysis of the experimental results of the ML classifiers without using genetic algorithms. The performance of various ML classifiers is used to detect phishing and legitimate emails with a focus on their accuracy. Among the classifiers evaluated, the Naive Bayes (NB) model demonstrated the most robust performance, attaining the highest accuracy rate of 90.91%, which underscores its substantial reliability and effectiveness. RF also performed exceptionally well, with an accuracy of 90.06%, making it a strong contender for detecting phishing emails. DT, LR, and SVM showed moderate accuracy, with values of 85.51%, 83.81%, and 84.38%, respectively, indicating that they are fairly accurate, but not as high-performing as NB and RF. In contrast, KNN showed the lowest accuracy among the classifiers, with an accuracy of 67.90%, indicating that it is less effective for this detection task. In summary, NB and RF are the top-performing models, whereas KNN has the lowest accuracy.

TABLE V. Experimental Results of ML Classifiers

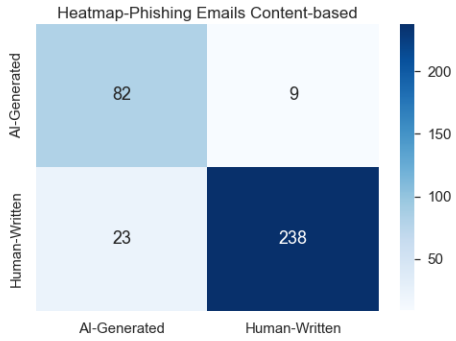
Classifier	Precision	Recall	F1-score	Accuracy
DT	91.83%	71.98%	76.08%	85.51%
KNN	65.35%	69.76%	64.76%	67.90%
LR	88.61%	69.40%	72.95%	83.81%
SVM	88.04%	70.85%	74.51%	84.38%
RF	92.68%	81.48%	85.25%	90.06%
NB	87.23%	90.65%	88.69%	**90.91%**

Fig. 4 shows the AUC-ROC and confusion matrix for the NB classifier, which achieved the highest accuracy. The figure illustrates the performance of a Naive Bayes classifier in detecting phishing emails based on content, consisting of a confusion matrix and an AUC-ROC curve. The AUC-ROC curve (a) shows the classifier's true positive rate (y-axis) against the false positive rate (x-axis), with the Naive Bayes model (orange dashed line) performing significantly better than random guessing (blue dashed line), indicating high sensitivity and specificity. The confusion matrix heatmap (b) highlights

classification results, with 82 AI-generated and 238 human-written emails correctly identified, while 9 AI-generated and 23 human-written emails were misclassified. The color intensity reflects the number of instances, demonstrating the classifier's overall accuracy and effectiveness.



(a)



(b)

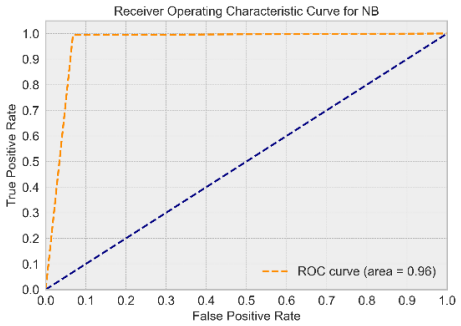
Fig. 4. AUC-ROC and confusion matrix of naive bayes. (a) AUC-ROC curve. (b) Confusion matrix showing correct (82 AI, 238 human) and incorrect (9 AI, 23 human) classifications.

Table VI summarizes the performance metrics for various classifiers used to detect phishing and legitimate emails, this time utilizing genetic algorithms for optimization. Notably, NB emerged as the top performer, with an accuracy of 97.87%, significantly improving from its previous accuracy of 90.91%. LR also saw a substantial increase in performance, achieving an accuracy of 94.47% compared to its earlier 83.81%. RF recorded an improved accuracy of 92.77%, up from 90.06%, thus maintaining its position as a strong classifier. DT demonstrated a marked improvement, with its accuracy increasing from 85.51% to 90.64%, indicating that genetic algorithms significantly enhanced its performance. In contrast, SVM showed a marginal improvement, with its accuracy slightly increasing from 84.38% to 85.53%. KNN improved its accuracy from 67.90% to 74.89%, achieving perfect recall; however, it remained less effective overall compared to other classifiers. In summary, the application of genetic algorithms for optimization led to performance improvements across most classifiers, particularly boosting the accuracy of NB and LR. Despite these enhancements, NB remains the top-performing model, while KNN still lags behind the others.

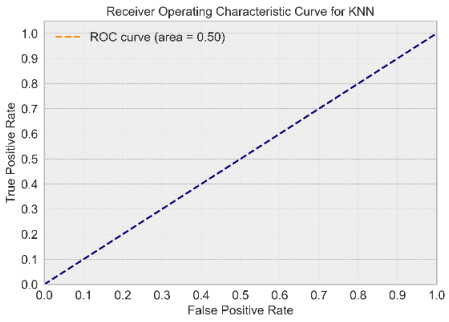
TABLE VI. EXPERIMENTAL RESULTS OF ML CLASSIFIERS USING GENETIC ALGORITHM

Classifier	Precision	Recall	F1-score	Accuracy
DT	87.55%	87.55%	87.55%	90.64%
KNN	74.89%	100.00%	85.64%	74.89%
LR	96.56%	88.98%	92.03%	94.47%
SVM	91.90%	71.19%	75.36%	85.53%
RF	94.68%	86.16%	89.43%	92.77%
NB	97.99%	96.33%	97.12%	**97.87%**

Fig. 5 shows the AUC-ROC for the best ML classifier and worst ML classifiers NB and KNN. The figure compares the ROC curves for two classifiers: NB and KNN. In subplot (a), the ROC curve for NB shows a high area under the curve (AUC) of 0.96, demonstrating outstanding performance with a high rate of correctly identified positives and a low rate of incorrectly identified positives. In contrast, subplot (b) displays the ROC curve for KNN, which has an AUC of 0.50, signifying performance equivalent to random guessing, as indicated by the diagonal line. This comparison highlights the superior effectiveness of the Naive Bayes classifier over the KNN classifier in this context.



(a)

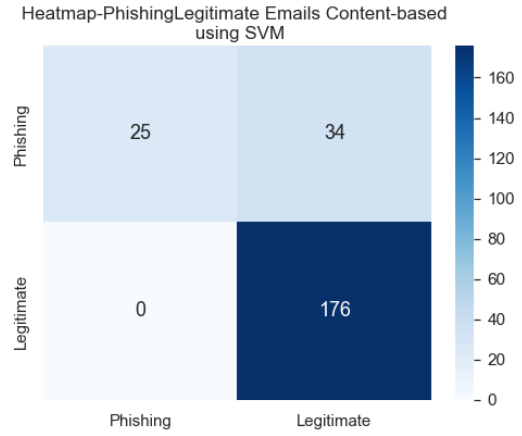


(b)

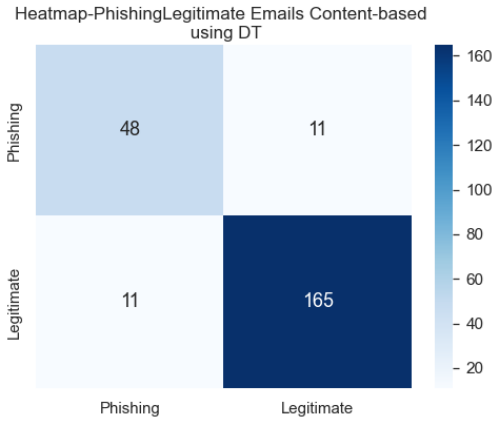
Fig. 5. AUC-ROC Curves for naive bayes and k-nearest neighbors; (a) Naive bayes (b) K-nearest neighbors.

Fig. 6 illustrates the confusion matrices for two machine learning classifiers, SVM and DT, applied to the classification of phishing and legitimate emails. Panel (a) shows the performance of the SVM classifier, which correctly identified 25 phishing emails and 176 legitimate emails. However, it misclassified 34 legitimate emails as phishing, with no false negatives (phishing emails classified as legitimate). Panel (b)

displays the results for the DT classifier, which correctly identified 48 phishing emails and 165 legitimate emails, with 11 false positives (legitimate emails classified as phishing) and 11 false negatives. The color intensity in both matrices represents the number of instances, providing a visual comparison of the classification accuracy and error distribution between the SVM and DT models. This analysis highlights the strengths and weaknesses of each classifier in distinguishing between phishing and legitimate emails.



(a)



(b)

Fig. 6. Confusion matrices (a) SVM, (b) DT.

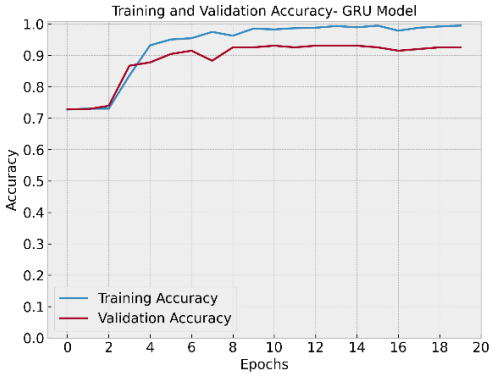
In the DL experiments, various models were evaluated based on precision, recall, F1-score, and accuracy in detecting phishing and legitimate emails. The models tested included LSTM, BiLSTM, GRU, and the proposed model utilizing Genetic Algorithm Feature Selection, as detailed in Table VII. The LSTM model achieved a precision of 94.97%, recall of 96.59%, F1-score of 95.77%, and accuracy of 93.62%. The BiLSTM model recorded a precision of 93.76%, recall of 88.98%, F1-score of 91.04%, and accuracy of 93.62%. The GRU model demonstrated an accuracy of 95.12%, with a precision of 95.28%, recall of 95.32%, and F1-score of 95.28%.

TABLE VII. EXPERIMENTAL RESULTS FOR THE DL MODELS AND PROPOSED MODEL

Classifier	Precision	Recall	F1-score	Accuracy
LSTM	94.97%	96.59%	95.77%	93.62%
BiSLTM	93.76%	88.98%	91.04%	93.62%
GRU	95.28%	95.32%	95.28%	95.12%
Proposed Model	96.77%	100.00%	98.36%	**97.90%**

In comparison, the proposed model outperformed the ML classifiers and DL models in terms of the common matrix and achieved a precision of 96.77%, recall of 100.00%, F1-score of 98.36%, and accuracy of 97.90% using Genetic Algorithm Feature Selection. Thus, the proposed model can select the most relevant features, thereby increasing its efficiency and accuracy. By combining RF with BiLSTM, we leveraged the strengths of both algorithms: RF's ability to handle high-dimensional data and reduce overfitting through ensemble learning, and BiLSTM's ability to capture sequential dependencies. This combination results in a robust model that can accurately detect phishing emails. In addition to the superior performance of the proposed model, it has also been demonstrated that when RF and BiLSTM are combined with Genetic Algorithm Feature Selection in order to enhance email security, this combination results in a superior performance that is reflected in all metrics evaluated.

Fig. 7 and 8 present the training and validation accuracy and loss functions for the GRU, BiLSTM, and proposed BiLSTM-RF models with Genetic Algorithm Feature Selection over 20 epochs. Fig. 7(a) and 7(c) show that both the GRU and BiLSTM models achieve high training and validation accuracy, with the GRU showing steady improvement and the BiLSTM demonstrating rapid early gains. Fig. 7(b) and 7(d) illustrate the loss functions for these models, indicating good generalization with minimal divergence between training and validation losses. In comparison, Fig. 8 highlights the proposed BiLSTM-RF model's performance, showing superior results: Fig. 8(a) demonstrates that it quickly reaches near-perfect accuracy for both training and validation, while Fig. 8(b) shows a sharp decline and stabilization in loss values, indicating efficient learning and minimal overfitting. Overall, the proposed model outperforms the GRU and BiLSTM models in accuracy and robustness, making it the most effective in detecting phishing and legitimate emails.



(a)

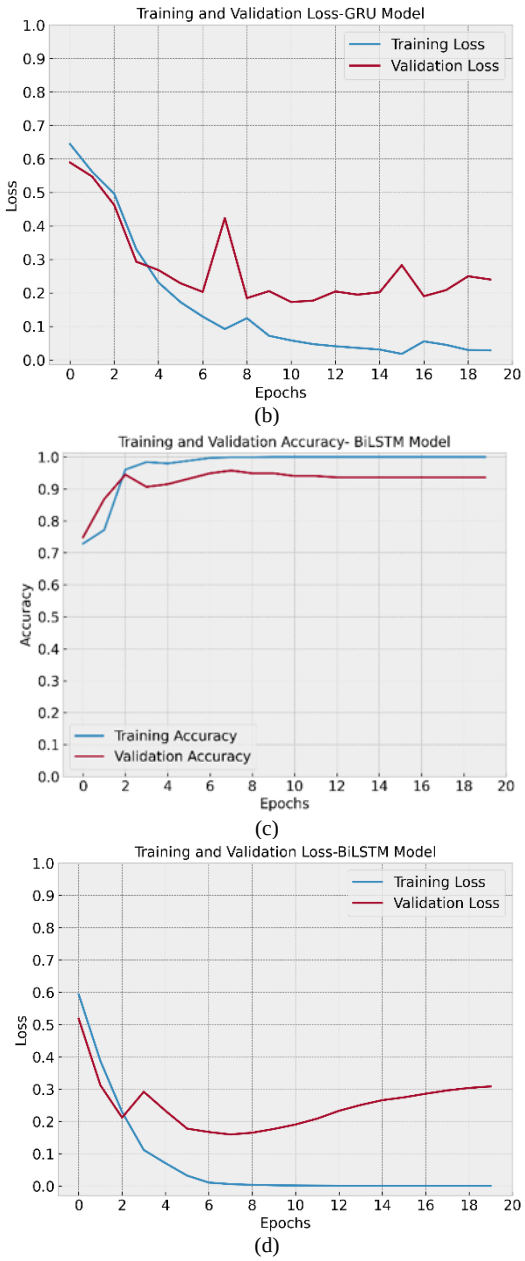


Fig. 7. Training accuracy, validation, and loss function of GRU and BiLSM.

In terms of precision, recall, and F1-score, the LSTM model achieved 94.97% precision, 96.59% recall, and 93.77% accuracy. A precision of 93.76%, recall of 88.98%, F1-score of 91.04%, and accuracy of 93.62% were recorded by the BiLSTM model. The GRU model demonstrated an accuracy of 95.12%, precision of 95.28 percent, recall of 95.32%, and F1-score of 95.28 percent.

The proposed model outperformed the ML classifiers and DL models in terms of the common matrix and achieved a precision of 96.77%, recall of 100.00%, F1-score of 98.36%, and accuracy of 97.90% using Genetic Algorithm Feature Selection. Thus, the proposed model can select the most relevant features, thereby increasing its efficiency and accuracy. By combining RF with BiLSTM, we leveraged the strengths of both algorithms: RF's ability to handle high-dimensional data and reduce overfitting through ensemble learning, and BiLSTM's ability to capture sequential dependencies. This combination results in a robust model that can accurately detect phishing emails. In addition to the superior performance of the proposed model, it has also been demonstrated that when RF and BiLSTM are combined with Genetic Algorithm Feature Selection in order to enhance email security, this combination results in a superior performance that is reflected in all metrics evaluated. Fig. 7 shows the training accuracy, validation, and loss function of the proposed model.

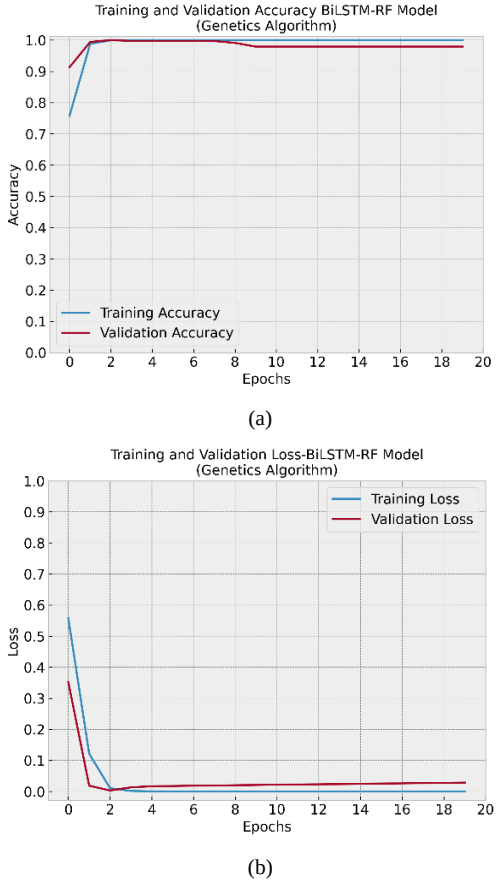


Fig. 8. Training accuracy, validation, and loss function of proposed model.

C. Discussions

The experiment findings demonstrate the effectiveness of feature optimization with a GA in identifying Arabic phishing emails. The classifiers performed differently in the first experiments conducted without GA. Certain models outperformed others in terms of precision, recall, and total accuracy, while others had difficulty reaching high effectiveness.

However, in the second set of experiments, all classifiers showed notable gains when the Genetic Algorithm was used for feature optimization. By picking the most pertinent features, the GA improved the models and improved their prediction power. This resulted in increased scores on all metrics overall. The significance of the GA in machine learning workflows, especially for intricate jobs like phishing

email detection, is highlighted by its efficacy in improving features. In addition to improving each classifier's performance, the GA also helped create detection systems that are more dependable and resilient by cutting down on noise and concentrating on the most important data points.

The proposed model has several advantages over conventional deep learning models like LSTM, BiLSTM, and GRU. It combines RF and BiLSTM with GA for feature optimization. This hybrid technique achieves improved precision, recall, F1-score, and accuracy in phishing email detection, demonstrating superior performance. A balanced and thorough feature representation is produced by combining the power of BiLSTM to capture sequential patterns with the capabilities of RF to evaluate feature relevance. By streamlining the feature set, lowering noise, and boosting overall effectiveness, the addition of GA improves the model even more. This results in a detection system that is more precise, dependable, and flexible and can adjust to the changing strategies used by phishing attempts. The proposed model's outstanding performance metrics underscore its effectiveness as a robust tool for enhancing cybersecurity.

VI. Conclusion

In this study, we proposed a hybrid model for phishing email detection, combining Random Forest (RF) and Bidirectional Long Short-Term Memory (BiLSTM) networks, augmented with Genetic Algorithm Feature Selection. The experimental results demonstrated that the proposed model significantly outperformed conventional approaches, including traditional machine learning classifiers, LSTM, BiLSTM, and Gated Recurrent Unit (GRU) models, across multiple performance metrics: Precision, Recall, F1 Score, and Accuracy. Specifically, the proposed model achieved an accuracy of 97.90%, recall of 100.00%, F1 score of 98.36%, and precision of 96.77%, illustrating its exceptional capability in correctly classifying phishing emails. The integration of RF and BiLSTM leveraged RF's proficiency in handling high-dimensional data and BiLSTM's capacity to capture sequential relationships, while the Genetic Algorithm Feature Selection ensured optimal feature subset identification.

Future research directions include expanding the dataset to encompass a broader range of phishing and legitimate emails, incorporating diverse linguistic and cultural variations. Additionally, we plan to explore advanced feature selection techniques such as Particle Swarm Optimization or Ant Colony Optimization. Furthermore, to capture more complex patterns in the data, we intend to investigate the integration of additional deep learning architectures, such as Transformers.

Acknowledgment

The authors extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA for funding this research work through the project number NBU-FFR-2024-1197-05.

References

[1] 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.

[2] Salloum, S., Gaber, T., Vadera, S., & Shaalan, K. (2022). A systematic literature review on phishing email detection using natural language processing techniques. IEEE Access, 10, 65703-65727. https://doi.org/10.1109/ACCESS.2022.3172553.

[3] Wang, J., Li, Y., & Rao, H. R. (2016). Overconfidence in phishing email detection. Journal of the Association for Information Systems, 17(11), 1. https://doi.org/10.17705/1jais.00443.

[4] Valecha, R., Mandaokar, P., & Rao, H. R. (2021). Phishing email detection using persuasion cues. IEEE Transactions on Dependable and Secure Computing, 19(2), 747-756. https://doi.org/10.1109/TDSC.2021.3055228.

[5] Form, L. M., Chiew, K. L., & Tiong, W. K. (2015, August). Phishing email detection technique by using hybrid features. In 2015 9th International Conference on IT in Asia (CITA) (pp. 1-5). IEEE. https://doi.org/10.1109/CITA.2015.7349825.

[6] Egozi, G., & Verma, R. (2018, November). Phishing email detection using robust NLP techniques. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 7-12). IEEE. https://doi.org/10.1109/ICDMW.2018.00010.

[7] Alhogail, A., & Alsabih, A. (2021). Applying machine learning and natural language processing to detect phishing email. Computers & Security, 110, 102414. https://doi.org/10.1016/j.cose.2021.102414.

[8] Verma, P., Goyal, A., & Gigras, Y. (2020). Email phishing: Text classification using natural language processing. Computer Science and Information Technologies, 1(1), 1-12.

[9] Harikrishnan, N. B., Vinayakumar, R., & Soman, K. P. (2018, March). 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, pp. 455-468).

[10] Alhogail, A., & Alsabih, A. (2021). Applying machine learning and natural language processing to detect phishing email. Computers & Security, 110, 102414.

[11] Teja Nallamothu, P., & Shais Khan, M. (2023). Machine Learning for SPAM Detection. Asian Journal of Advances in Research, 6(1), 167-179. https://doi.org/10.9734/ajarr/2023/v6i113039.

[12] Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. Applied Sciences, 13(15), 8756. https://doi.org/10.3390/app13158756.

[13] Atlam, H. F., & Oluwatimilehin, O. (2022). Business email compromise phishing detection based on machine learning: A systematic literature review. Electronics, 12(1), 42. https://doi.org/10.3390/electronics12010042.

[14] Unnithan, N. A., Harikrishnan, N. B., Akarsh, S., Vinayakumar, R., & Soman, K. P. (2018). Machine learning based phishing e-mail detection. Security-CEN@ Amrita, 65-69.

[15] Kumar, N., & Sonowal, S. (2020, July). Email spam detection using machine learning algorithms. In 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 108-113). IEEE. https://doi.org/10.1109/ICIRCA48905.2020.9183145.

[16] Ghosh, A., & Senthilrajan, A. (2023). Comparison of machine learning techniques for spam detection. Multimedia Tools and Applications, 82(19), 29227-29254. https://doi.org/10.1007/s11042-023-14693-3.

[17] 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.

[18] Brindha, R., Nandagopal, S., Azath, H., Sathana, V., Joshi, G. P., & Kim, S. W. (2023). Intelligent Deep Learning Based Cybersecurity Phishing Email Detection and Classification. Computers, Materials & Continua, 74(3). https://doi.org/10.32604/cmc.2023.032386.

[19] Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E. A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), 3819-3828. https://doi.org/10.1007/s10586-022-03672-0.

[20] Fang, Y., Zhang, C., Huang, C., Liu, L., & Yang, Y. (2019). Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism. IEEE Access, 7, 56329-56340. https://doi.org/10.1109/ACCESS.2019.2909314.

[21] Alsaidi, R. A., Yafooz, W. M., 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). https://doi.org/10.14569/IJACSA.2022.0131132.

[22] Hasan, B. M. S., & Abdulazeez, A. M. (2021). A review of principal component analysis algorithm for dimensionality reduction. Journal of Soft Computing and Data Mining, 2(1), 20-30.

[23] Shobana, G.; Bushra, S.N. Classification of Myopia in Children using Machine Learning Models with Tree Based Feature Selection. In Proceedings of the 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 5-7 November 2020; pp. 1599-1605. https://doi.org/10.1109/ICECA49313.2020.9297528.

[24] Gupta, K. Optimizing Performance: SelectKBest for Efficient Feature Selection in Machine Learning. 2020. Available online: https://medium.com/@Kavya2099/optimizing-performance-selectkbest-for-efficient-feature-selection-in-machine-learning-3b635905ed48 (accessed on 11 July 2024).

[25] Jeon, H.; Oh, S. Hybrid-Recursive Feature Elimination for Efficient Feature Selection. Appl. Sci. 2020, 10, 3211. https://doi.org/10.3390/app10093211.

[26] Li, F.; Lai, L.; Cui, S. On the Adversarial Robustness of LASSO Based Feature Selection. IEEE Trans. Signal Process. 2021, 69, 5555-5567. https://doi.org/10.1109/TSP.2021.3102136.

[27] Greenacre, M., Groenen, P. J., Hastie, T., d'Enza, A. I., Markos, A., & Tuzhilina, E. (2022). Principal component analysis. Nature Reviews Methods Primers, 2(1), 100.

[28] Rey, C.C.T.; García, V.S.; Villuendas-Rey, Y. Evolutionary feature selection for imbalanced data. In Proceedings of the 2023 Mexican International Conference on Computer Science (ENC), Guanajuato, Mexico, 11-13 September 2023; pp. 1-7. https://doi.org/10.1109/ENC57540.2023.10177468.

[29] Thapa, C., Tang, J. W., Abuadbba, A., Gao, Y., Camtepe, S., Nepal, S., ... & Zheng, Y. (2023). Evaluation of federated learning in phishing email detection. Sensors, 23(9), 4346. https://doi.org/10.3390/s23094346.

[30] Tong, X., Wang, J., Zhang, C., Wang, R., Ge, Z., Liu, W., & Zhao, Z. (2021). A content-based chinese spam detection method using a capsule network with long-short attention. IEEE Sensors Journal, 21(22), 25409-25420.

[31] Li, Q., Cheng, M., Wang, J., & Sun, B. (2020). LSTM based phishing detection for big email data. IEEE transactions on big data, 8(1), 278-288.

[32] Wu, Y., Si, S., Zhang, Y., Gu, J., & Wosik, J. (2024). Evaluating the performance of chatgpt for spam email detection. arXiv preprint arXiv:2402.15537.

[33] Sonowal, G. (2020). Phishing email detection based on binary search feature selection. SN Computer Science, 1(4), 191.

[34] Ablel-Rheem, D. M., Ibrahim, A. O., Kasim, S., Almazroi, A. A., & Ismail, M. A. (2020). Hybrid feature selection and ensemble learning method for spam email classification. International Journal, 9(1.4), 217-223.

[35] Saber, W. M., Ding, W., Sonne, C., & Abdelsalam, H. M. (2022). Email phishing detection: A systematic literature review. ACM Computing Surveys, 55(2), 1-37. https://doi.org/10.1145/3491207.

[36] [35] 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.

[37] [36] Valecha, R., Mandaokar, P., & Rao, H. R. (2021). Phishing email detection using persuasion cues. IEEE transactions on Dependable and secure computing, 19(2), 747-756. https://doi.org/10.1109/TDSC.2021.3050803.

[38] [37] Thapa, C., Tang, J. W., Abuadbba, A., Gao, Y., Camtepe, S., Nepal, S., ... & Zheng, Y. (2023). Evaluation of federated learning in phishing email detection. Sensors, 23(9), 4346. https://doi.org/10.3390/s23094346.

[39] [38] Harikrishnan, N. B., Vinayakumar, R., & Soman, K. P. (2018, March). 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, pp. 455-468). https://doi.org/10.1145/3180445.3180635.

[40] [39] Unnithan, N. A., Harikrishnan, N. B., Akarsh, S., Vinayakumar, R., & Soman, K. P. (2018). Machine learning based phishing e-mail detection. Security-CEN@ Amrita, 65-69.

[41] Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E. A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), 3819-3828. https://doi.org/10.1007/s10586-022-03613-6.

[42] Yafooz, W., & Alsaeedi, A. (2024). Leveraging User-Generated Comments and Fused BiLSTM Models to Detect and Predict Issues with Mobile Apps. Computers, Materials & Continua, 79(1). https://doi.org/10.32604/cmc.2024.027108.

[43] Brindha, R., Nandagopal, S., Azath, H., Sathana, V., Joshi, G. P., & Kim, S. W. (2023). Intelligent Deep Learning Based Cybersecurity Phishing Email Detection and Classification. Computers, Materials & Continua, 74(3). https://doi.org/10.32604/cmc.2023.025628.

[44] Holland, J. H. (1992). Genetic algorithms. Scientific American, 267(1), 66-73. https://doi.org/10.1038/scientificamerican0792-66.

[45] Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2016). A survey on evolutionary computation approaches to feature selection. IEEE Transactions on Evolutionary Computation, 20(4), 606-626. https://doi.org/10.1109/TEVC.2015.2504420.

[46] Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. Addison-Wesley Longman Publishing Co., Inc.

[47] Hamid, I. R. A., & Abawajy, J. (2011). Hybrid feature selection for phishing email detection. In International Conference on Algorithms and Architectures for Parallel Processing (pp. 266-275). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-24669-2_25.

[48] Deb, K. (2001). Multi-objective optimization using evolutionary algorithms. John Wiley & Sons. https://doi.org/10.1002/9780470496947.

[49] Zareapoor, M., & Seeja, K. R. (2015). Feature extraction or feature selection for text classification: A case study on phishing email detection. International Journal of Information Engineering and Electronic Business, 7(2), 60-65. https://doi.org/10.5815/ijieeb.2015.02.08.

[50] Akinyelu, A. A., & Adewumi, A. O. (2014). Classification of phishing email using random forest machine learning technique. Journal of Applied Mathematics, 2014. https://doi.org/10.1155/2014/425731.

[51] Chowdhury, M., Colbert, J., Kabir, M., Sait, S. M., & Aslam, N. (2020). A multi-optimization based feature selection method for phishing detection using neural networks. IEEE Access, 8, 219616-219626. https://doi.org/10.1109/ACCESS.2020.3042717.

[52] Zou, Y., & Schaub, F. (2019). Beyond mandatory: Making data breach notifications useful for consumers. IEEE Security & Privacy, 17(2), 67-72. https://doi.org/10.1109/MSEC.2019.2905629.

[53] Wang, J., Yang, Y., & Xia, B. (2019). A simplified Cohen's Kappa for use in binary classification data annotation tasks. IEEE Access, 7, 164386-164397.

[54] Yafooz, W. M., Alsaeedi, A., & Emara, A. H. M. (2023, February). AraDS: Arabic datasets for text mining approaches. In 2023 International Conference on Smart Computing and Application (ICSCA) (pp. 1-6). IEEE. https://doi.org/10.1109/ICSCA57840.2023.10087675.

[55] Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In Machine learning (pp. 101-121). Academic Press.

[56] Yafooz, W. M., Alsaeedi, A., Alluhaibi, R., & Abdel-Hamid, M. E. (2022). Enhancing multi-class web video categorization model using machine and deep learning approaches. Int. J. Electr. Comput. Eng, 12, 3176. https://doi.org/10.11591/ijece.v12i3.pp3176-3191.

[57] Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01), 20-28.

[58] Song, X., Liu, X., Liu, F., & Wang, C. (2021). Comparison of machine learning and logistic regression models in predicting acute kidney injury: A systematic review and meta-analysis. International journal of medical informatics, 151, 104484.

[59] Alhejaili, R., Alhazmi, E. S., Alsaeedi, A., & Yafooz, W. M. (2021, September). Sentiment analysis of the COVID-19 vaccine for Arabic tweets using machine learning. In 2021 9th International conference on reliability, infocom technologies and optimization (Trends and Future Directions)(ICRITO) (pp. 1-5). IEEE. https://doi.org/10.1109/ICRITO51393.2021.9596517.

[60] Nusinovici, S., Tham, Y. C., Yan, M. Y. C., Ting, D. S. W., Li, J., Sabanayagam, C., ... & Cheng, C. Y. (2020). Logistic regression was as good as machine learning for predicting major chronic diseases. Journal of clinical epidemiology, 122, 56-69.

[61] Boateng, E. Y., Otoo, J., & Abaye, D. A. (2020). Basic tenets of classification algorithms K-nearest-neighbor, support vector machine, random forest and neural network: A review. Journal of Data Analysis and Information Processing, 8(4), 341-357.

[62] Antoniadis, A., Lambert-Lacroix, S., & Poggi, J. M. (2021). Random forests for global sensitivity analysis: A selective review. Reliability Engineering & System Safety, 206, 107312.

[63] Yahya, A. E., Gharbi, A., Yafooz, W. M., & Al-Dhaqm, A. (2023). A novel hybrid deep learning model for detecting and classifying non-functional requirements of mobile apps issues. Electronics, 12(5), 1258. https://doi.org/10.3390/electronics12051258.

[64] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555. https://doi.org/10.48550/arXiv.1412.3555.

[65] Hameed, Z., & Garcia-Zapirain, B. (2020). Sentiment classification using a single-layered BiLSTM model. Ieee Access, 8, 73992-74001.

[66] Deng, J., Cheng, L., & Wang, Z. (2021). Attention-based BiLSTM fused CNN with gating mechanism model for Chinese long text classification. Computer Speech & Language, 68, 101182.

[67] Saibene, A.; Gasparini, F. Genetic algorithm for feature selection of EEG heterogeneous data. Expert Syst. Appl. 2023, 217, 119488. https://doi.org/10.1016/j.eswa.2023.119488.

[68] Catak, F.O. Genetic algorithm based feature selection in high dimensional text dataset classification. WSEAS Trans. Inf. Sci. Appl. 2015, 12, 290-296.

[69] Xue, B., Zhu, C., Wang, X., & Zhu, W. (2022, March). The study on the text classification based on graph convolutional network and BiLSTM. In Proceedings of the 8th International Conference on Computing and Artificial Intelligence (pp. 323-331).