social Informatics Journal
Business, Politics, Law, Environmental Sciences & Technology

Vol 10, No 4, December, 2024 Series

Development of an Hybrid Machine Learning Model to Detect Email-based Social Engineering Attacks

Oguntunde, T. & Momoh, M.S.
Department of Computer Science
University of Ibadan, Ibadan, Nigeria
E-mails: tantos557@gmail.com, mercymomoh160@gmail.com

ABSTRACT

Social engineering is the deliberate manipulation of human behaviour to gain unauthorized access to information, systems or resources. Social engineering attacks continue to exploit human vulnerability and bypass traditional security controls, resulting in financial loss, identity theft, operational disruptions, and wider security risks. Existing email-based detection systems often struggle with high false-negative rates as phishing techniques evolve, especially within local contexts. This study addresses this gap by developing a hybrid machine learning model, the Randomized Logistic XGBoost (RL-XGBoost) model, designed to improve the detection and minimization of email-based social engineering attacks. The dataset used for the study was gathered from individual email inboxes from the University of Ibadan email server. The collected emails were preprocessed. The preprocessed email was partitioned into 80% (training) and 20% (testing). An hybrid Machine Learning model for detection and minimisation of email-based social engineering attack was developed from Logistic regression(LR), Random Forest(RF) and Extreme Gradient boost called Randomized Logistic XGBoost (RL-XGBoost). The RL-XGBoost was trained with the training dataset. The RL-XGBoost predicts with Logistic Regression, Random Forest and Extreme Gradient boost with soft voting. The evaluation performance of RL-XGBoost was done by comparing it with Logistic Regression, Random Forest and Extreme Gradient boost individually using accuracy,precision, recall, F1-score, ROC-AUC as metrics. Seven thousand, two hundred and seventeen emails were harvested from the university of Ibadan email server. Logistic Regression, Random Forest, Extreme Gradient boost and RL-XGBoost had accuracy of 0.86 ,0.89 ,0.92 and 0.94, respectively; precision of 0.71, 0.84 ,0.88 and 0.90, respectively. Moreover, Logistic Regression, Random Forest, Extreme Gradient boost and RL-XGBoost had recall 0.74, 0.86, 0.90, and 0.96, respectively; F1-score of 0.73, 0.85, 0.89, and 0.93, while they have ROC-AUC of 0.78, 0.82, 0.90 and 0.92, respectively. This indicates that RL-XGBoost outperformed Logistic Regression, Random Forest and Extreme Gradient boost having the best performance in accuracy, precision, recall, F1-score and ROC-AUC. The Randomized Logistic Extreme Gradient Boost outperformed the existing, individual models and can be deployed as and add-on into an existing email filtering systems within organizational mail gateways to automatically flag suspicious messages and assist security teams in early threat detection.

Keywords: Randomized Logistic Extreme Gradient Boost, Ensemble voting modeling, Email-based Social engineering, Phishing detection, Machine learning, Cybersecurity.

Journal Reference Format:
Oguntunde, T. & Momoh, M.S. (2023): Development of an Hybrid Machine Learning Model to Detect Email-based Social Engineering Attacks. Social Informatics, Business, Politics, L:aw, Environmental Sciences & Technology Journal. Vol. 10, No. 4. Pp 31-46.
www.isteams/socialinformaticsjournal. dx.doi.org/10.22624/AIMS/SIJ/V10N4P4

1. INTRODUCTION

Social engineering (SE) attacks exploit human psychology and email is one of their most common way. Phishing, business email compromise, financial scams, and malicious attachments, trick users into revealing sensitive information or clicking harmful links. Traditional defenses like firewalls and spam filters are no longer enough, as attackers constantly adapt their techniques to bypass existing safeguards. As a result, machine learning approaches have been widely adopted for phishing detection due to their ability to learn complex patterns from historical data and adapt to evolving threats. However, individual machine learning classifiers exhibit limitations such as overfitting, sensitivity to data imbalance, and inadequate recall performance. In phishing detection, false negatives are particularly dangerous because undetected malicious emails can result in significant security breaches. This study focuses specifically on email-based social engineering attacks within the Nigerian email environments, by developing a hybrid ensemble learning approach that combines multiple classifiers to improve detection robustness, accuracy and to minimize SE attacks.

2. RELATED WORKS

Mouton (2018) proposed a Social Engineering Attack Detection Model (SEADM) as part of his PhD thesis. The study provided much of importance to the standardisation of the vocabulary of social engineering and the development of a multi-level model of social engineering attack detection. It was based on the social engineering cycle that was developed by Kevin Mitnick and had been modified three times. The first one was bi-directional communication in environments such as call centers. The second one added the area of the first one with unidirectional and indirect communication channels, and the third one introduced a finite state machine (FSM), to have more abstraction and expandability. The methodology presented by Mouton aimed at theoretical mapping of the real-life attack cases, and normative ethical analysis, which entails the use of virtue ethics, deontology and utilitarian is used to examine possible outcomes. SEADM was to be an adaptation-based training, education and theoretical detection. However, it was not compared with any machine learning algorithms and datasets or predictive models.

Despite his theoretically sound foundation, Mouton was not using automated or data-driven analytical procedures in his research. The FSM had been manual and the model had not been trained with real data. As a result, despite the fact that the research by Mouton led to the theoretical and ethical knowledge of social engineering detection, nonetheless, it was not implemented on machine learning algorithms and real-life data. The FSM was also not programmable hence could not be applied to dynamic and data driven applications such as email security. Even though this paper has contributed to the field of knowledge in theory, it left an open gap in automated detection using ML. This study aimed at filling this gap by developing a machine learning model that is deployed as email-based social engineering attacks real-time and dynamic detection model, which is absent in SEADM.

Bokhonko et al. (2024) designed a custom machine learning models which were applied in detecting spam, spear phishing, and Trojan email-based social engineering attacks. The authors explained that BotGRABBER framework had come up with a variety of classifiers, which included Random Forest, K-Nearest Neighbor (KNN), XGBoost, and Decision Tree. These classifiers were trained following certain characteristics such as details of email, behavior of interaction and embedded attachment behavior. The Trojan model was based on using sandbox environment to measure the degree of risk of malicious email attachments, but the spear phishing model was based on metadata of a sender and false hyperlinks.

The authors used feature-targeted methodology to identify and validate their findings using practical tests, whose accuracy was 99 percent and their false positive rate was 6 percent. Each of the social engineering vectors had its own machine learning pipelines. They correctly developed their system, although it was not generalized by using multi-modal attacks and adaptive learning or explainable AI methodologies. There is also a lack of security against malicious attacks introduced by the authors. Even though they focused on detection systems with regards to individual threats, they failed to provide a general or dynamically adaptive detection system that could be useful in addressing the dynamic attacks. This study makes one of the adaptive ML model, which will decrease a greater range of email-based social engineering attacks at the cost of flexibility and resiliency.

Lansley et al. (2024) proposed SEADer and implemented both artificial neural networks (ANNs) and the natural language processing (NLP) to classify social engineering in Internet chat systems. They counted on their feature extraction pipeline, such as urgency and intent, and third-party tools, such as the WOT API, to retrieve URL credibility and SymSpellpy to analyze spelling. The classification was performed on multi-layer perceptron with weighted features input. This model has had the ability to detect the manipulation in the chat based interaction but was not that broad because it only took into consideration the platforms of conversation. It was also not resistant to adversarial manipulation, not to mention that it was not part of larger cybersecurity frameworks. Even though SEADer is an important contribution to NLP-based detection, it has some limitations in regards to implementation as it can be applied to particular applications. The concepts in this paper are generalized to the email setting where phishing is the most utilized SE channel, and using ML models that are continually updated with evolving trends of email fraud.

According to Lin et al. (2021), their investigation of machine learning system vulnerabilities concentrated on data poisoning and adversarial attacks as dangers to the integrity of ML models. They clarify the way adversaries can break the results of models by corrupting the training data (poisoning) or corrupting the input data (adversarial). The white box-based approaches to attacks were considered including DeepFool, Projected Gradient Descent (PGD) and Fast Gradient Sign Method (FGSM) whereas the black-box ones that included the score-based attacks and transfer attacks were examined in the limited-access environment. They pointed out that the issue of ML security was not about the strength of the algorithms, but also the strength against manipulation, and they provided explicit mathematical statements about the risks.

In relation to the methodology, they have employed strict definitions of algorithm and theoretical frameworks not the implementation. They wanted to give the scientific community knowledge about the possible risky hostility particularly in safety-critical systems and smart systems. Despite the fact that the work is an important input to the security-awareness area of machine learning, its main drawback was the fact that it did not have practical defensive implementations. The paper builds on this knowledge by analyzing at length the concept of resilience and adaptability of ML models to phishing email detection in order to ensure that the system is not only accurate but also inoperable.

Krombholz et al. (2014) conducted a comprehensive review of advanced social engineering assaults, categorizing them by channel, operator, and technique. A taxonomy was created to describe how social engineering attacks have evolved in email, social media, and cloud services and phone based-vectors. Unlike model-driven studies, their article highlighted the fact that the current communication networks, such as Dropbox, Skype, LinkedIn, and mobile apps endangered a user (and particularly a knowledge worker) due to the more advanced threats, such as Advanced Persistent Threats (APTs).

That is why their study was so effective due to their contextual analysis of contemporary threats of SE and the psychological tricks they exploit to mcbreak the trust. The current study, however, lacks the machine learning-based solution and even does not strive to measure the effectiveness of countermeasures even though the variety of different scenarios and attack surfaces is extensive. The research being presented will be able to address these gaps by suggesting a email and machine learning-based SE detection model to conduct descriptive analysis rather than proactive mitigation.

Almseidin et al. (2019) in their study “Phishing Detection Based on Machine Learning and Feature Selection Methods”, suggested a comparison framework for classifying phishing web pages using a variety of machine learning algorithms. In their model, they focused on applying the Random Forest (RF), the J48, and the Multilayer Perceptron (MLP) of a balanced set of 10 000 webpages given by Alexa, PhishTank and OpenPhish, 5 000 of which were phishing and 5 000 of which were legitimate. They reduced their features to 20 by applying feature selection algorithms on feature selection, including InfoGain and ReliefF in order to improve performance. RF selected 20 features only and this led to an accuracy of 98.11 indicating that the method drastically reduced the training time without compromising the detection accuracy. They were 10-fold cross-validation in order to maintain consistency and Weka 3.8.3 to test the model. The measures were evaluated in terms of standard binary classification, and hybrid feature selection improved efficiency and speed in detection. The effectiveness of selecting features to enhance the machine learning ability of phishing detection was highlighted in the paper.

The web-based phishing pages were restricted to this paper and other forms of social-engineering vectors, including email spear phishing or impersonation, were not covered, even though this paper had an upper hand in classifying phishing using machine learning, it only tries to address this gap by not only directly focusing on email phishing detection but also, through the development of a lightweight ML model, which focuses more on flexibility as phishing techniques keep changing.

Lee et al. suggested a framework named CATBERT (2020) Context-Aware Tiny Bert To identify Social Engineering Emails. The CATBERT model proved to be a useful yet easy to apply model in the detection of phishing in emails. The article is a resolution to the accuracy of detection and computational efficiency which is a huge concern on the enterprise systems that deal with large amount of emails per day. On the methodological end, CATBERT built upon the efforts of systems based on transformers, yet dialing down a smaller-sized version of BERT on phishing data. The model has used the linguistic and metadata characteristics that incorporated semantic embeddings of email body (linguistic) and sender characteristics and the email subject line (metadata).

They compared benchmark phishing datasets and the model was identified to be more precise (87 percent) and with less false rate (1 percent) in comparison to logistic regression and LSTM baselines. Their adversarial resistance to perturbations was also a strength including intentional misspellings to prevent detection. However, the training of CATBERT is performed using English data only, which casts doubt on multilingual ability and problem transferability. It did not also state about the problems of life long learning in dynamic environments. The study showed that techniques, which were founded on transformers, may significantly improve the detection of phishing, but it did not reach so far as adaptive or cross-lingual detection systems. The paper builds on it by developing a model more reinforced with reinforcement learning characteristics that do not merely examine the contents of emails but also evolves as the years and language change to provide real-time, dynamic security against any of the several phishing offenses.

Keelan et al. (2021) developed RAIDER, which is Reinforcement-Aided Spear Phishing Detector. RAIDER framework solved the issue of spear phishing detection particularly the challenge of determining that the emails were malicious and it was sent to the users by seemingly genuine email senders. The reinforcement learning suggested in the model as a way of dynamically maximizing the choice of features in the training process and therefore dimensionality reduction without interfering with the predictive power. RAIDER applied methodologically reinforcement agents and conventional supervised classifier to train the model using labeled spear phishing samples which utilized sender information, message content and embedded links. Results showed that RAIDER reduced the feature space by 55 per cent, as well as, raised the detection rates to 94 per cent over the base classes of the ML.

The main contribution of the study was that it was able to overcome the known sender problem that is one of the weaknesses of most phishing detection models that attackers employ trusted identities. However, RAIDER was also evaluated on small-scale datasets, which also casts doubt upon it in the large-scale or multilingual enterprise email, and its generalizability. It was also lacking in flexibilities to both invisible phishing tricks that were not part of its training. Though the concept of reinforcement learning was found useful in the context of email security by RAIDER, its constraints were due to the constraint of the dataset. This approach is enhanced in the proposed research through enacting reinforcement-based learning to larger email phishing cases, where an ongoing retraining on real organizational emails is deployed to increase resilience and adaptability to evolving threat environment.

Koide et al. (2024) examined whether large language models (LLM) based on GPT-4 can detect phishing email without being trained on the subject matter. The article reported the zero-shot classification ability of the LLAMs in which phishing identification would be carried out by devising prompts that would know manipulable or deceptive each time. The researchers used API to request GPT-4 and evaluated its responses in methodological terms on different benchmark phishing datasets. The model also achieved high precision of 99.7 percent that by far is more precise than the traditional classifiers and the deep learning baselines. Among the greatest contributions is the fact that the system was explainable, GPT-4 could deliver justifications in terms of human readables, and thus it was more transparent in decision-making. However, the study was also associated with several issues that entailed high reliance on proprietary APIs, expenses incurred, privacy threat to data and susceptibility to adversarial prompt injection attack. These limitations make it less feasible to use in large-scale adoption in any organization. The proposed study is such an alternative that it takes into consideration certain concepts of the LLM but intends to produce a lightweight locally executable ML model that is extremely accurate and transparent and does not need to be based on using costly external APIs or cloud computing services. This enables high-end scaling, administration and stability to enterprise email configurations.

Jabbar and Al-janabi(2025) The present paper introduced the principle of reinforcement learning into phishing detection with Deep Q-Networks (DQNs), which offered the dynamic approach to the detection of the adaptive changing email threat. The authors took a simulated agent-based model whereby the model was trained to distinguish between phishing and authentic email messages by playing within the environment repeatedly. Datasets were also made up of both actual email corpora and artificially generated phishing messages, which allowed covering a broader training.

The accuracy, and precision and recall rates of DQN were 95, 96, and 94, respectively, which demonstrates its capability of generalizing between the traditional and the new strategy of phishing. Among the greatest contributions was the fact that it eliminated the restrictions of the static datasets given that the model was free to learn upon being presented with their feedback loops of reinforcement, which reduced overfitting. However, the system is also highly computer-intensive and has a lengthy training time and thereby restricts its use, real-time in small and medium-sized organizations. The paper also has not succeeded in fully describing the phenomenon of explainability or integration into enterprise workflows. Though it did the usefulness of the reinforcement learning in detecting phishing, there are still barriers to deployment. The concept in my research is adjusted by developing more computationally efficient model of reinforcement learning, that is, optimized to operate in enterprise scale email detection where real-time flexibility and resource efficiency is a priority.

Alsufyani and Alzahrani (2021) investigated machine learning methods in identifying phishing emails, and specifically, linguistic and psychological cues to phishing e-mails. They used 6,224 labeled emails as their data, 3,000 phishing and 3,224 legitimate. Authors have employed support vector machines, random forests and logistic regression as classifiers that were implemented methodologically with the use of classifier features that were derived on the basis of urgency words, manipulation cues and deceptive phrasing. Measuring competitive accuracy with small differences between classifiers was done. The value addition of the research was that it focused on the semantic and psychological manipulation strategies of phishing keeping in mind the fact that attackers are likely to exploit the urgency and authority in the email body. However, the size of the dataset was quite small that limited the ability of the model to be extrapolated to other kinds of attacks. Also, the emails that had been written negatively, though similar to the original communication, were not tested. Though this work supported the utilization of the linguistic features in phishing detection, this was not a scalable and robust work. This paper contributes to the latter with the addition of massive multilingual data and transformer-based embeddings to the latter, which allows a further semantic understanding and flexibility to the adversarially manipulated emails.

In the study by Rathee and Mann (2022), the authors gave a comparative analysis of the traditional ML algorithms and deep learning architecture in identifying email phishing. They made a contribution on the growing debate that classical models have become insufficient to cope with modern phishing. The trained algorithms of the authors were Decision Trees and SVMs, along with CNNs and RNNs, and needed a set of benchmarked phishing and legitimate emails. The results proved that deep learning was significantly superior to classical ML in terms of recall and accuracy in identifying subtle patterns in the linguistic structure and metadata. Performance of CNNs and RNNs proved to be the most effective because it is possible to model contextual relation of text sequences. However, the authors have noted that deep learning models are computationally intensive and large in terms of data requirements that might not be practically applicable in small organizations with limited resources. They are supposed to empirically demonstrate that DL is optimal in preventing phishing, but are also meant to highlight problems with its deployment. The paper underpins the benefits of deep learning, yet the application of reinforcement learning to reduce dependence on large labelled datasets, even more useful and adaptable phishing detection models can be created to apply to enterprise email environments.

Senturk et al. (2017) finished one of the previous surveys on the machine learning in the detection of phishing emails, in which data mining techniques are employed to understand the fraudulent behavior of the sender addresses, the subject lines, and the content of the email bodies. Their data were the labeled phishing and legitimate email, the features of which were made manually to trigger suspicious linguistic and structural information. Naive Bayes, Decision Trees and k-NN algorithms were applied to this dataset and the outcomes of the algorithms were moderate detection accuracy. Even though the study was the first to apply ML to phishing, its application of manually-written rules and non-varying features undermined its ability to respond to evolving threats. The authors were aware of how the attackers could easily improve to avoid the detection. However, the research proved helpful in establishing the base on the future models of phishing detection as it revealed that it was feasible to employ data-based approaches. The present work builds up to these initial findings, yet with a different method whereby Rule-based detection has been substituted by dynamic and adaptive ML-based methods which use a combination of NLP and reinforcement learning to offer real-time phishing protection in email services.

Atlam & Oluwatimilehin (2022) reviewed the systematic literature of Business Email Compromise (BEC) attacks with the help of which machine learning might assist in eliminating this increasing threat. BEC is not equivalent to ordinary phishing as it does not necessarily involve malicious links and attachments, it involves impersonation and abuse of trust. Among the existing detection methods, which were referenced in the review, are the ML-based communication pattern analysis, linguistic profiling and metadata inspection. The discovery of the absence of BEC-specific datasets was a significant contribution to the body of knowledge as one of the most critical impediments to the further development because most of the literature on ML focused on general phishing, and not impersonation phishing. The authors also found that adversarial evasion, privacy concerns, and explainable detection models were also a challenge. In spite of the fact that the review described the landscape of the BEC research, it did not propose a new detection model and empirical analysis. My study considers them by constructing a model that transcends the concept of general phishing to incorporate impersonation based fraud, where contextual characteristics and a reinforcement learning model are employed in a proactive protection against email fraud on the enterprise level.

In the case of the problem of phishing detection scalability, (Butt et al., 2023) developed a framework that employed both the ML and DL models, thereby allowing them to employ the cloud-based framework. They used CNNs to perform textual analysis and metadata feature classification with the Random Forest classifier on high volumes of enterprise email traffic to execute on a distributed cloud-based infrastructure. This solution was quite precise and demonstrated that it was possible to work with the real-time scale detection. However, the deployment of cloud environments introduced latency, cost and privacy of the information. In addition, a system might not be feasible in those organizations which lack a robust cloud infrastructure. The research value of the work is that it shows how the problem of email phishing can be brought to a larger scale when utilizing cloud computing. On the other hand, this paper outlines a lightweight locally-deployable ML model which will balance scalability and privacy by minimizing the use of third-party cloud computing systems by preserving its detection performance and flexibility.

Shahrivari et al. (2020) contrasted a number of machine learning based classifiers that can be used in the identification of phishing like Decision Trees, Random Forest and Gradient Boosted Trees. They tested them using the datasets that had such characteristics like URLs, contents of email messages and metadata. The results revealed that an amalgamation between various learners proved to be more efficient in each and every case than solitary one in identifying phishing. The benefits in approaches were cross-validation and comparative performance of different algorithms.

However, the study still had limitations of fixed data sets and did not consider the potential of adversarial phishing methods and need to maintain a model on a fixed timeline. In as much as the study established the strength of the ensemble learning as an effective phishing detection strategy, it was not adaptable to the evolving email threats. This paper expands on this by introducing dynamic retraining and reinforcement learning processes to offer continuous resilience in dynamic enterprise environments, which mediate the gap between the two concepts, i.e., the static detection and adaptive defense.

3.0 METHODOLOGY

3.1 Dataset Collection

The dataset used in this research consists of 7,217 emails collected from the University of Ibadan email server. The dataset includes both legitimate and phishing emails, providing a realistic representation of organizational email traffic. Primary dataset is employed due to the fact that publicly available dataset will have samples of data that might not represent local phishing styles. To maintain the message headers, links, and structure of the body, the participants will be instructed to send the emails in the form of attachments (in.eml format).

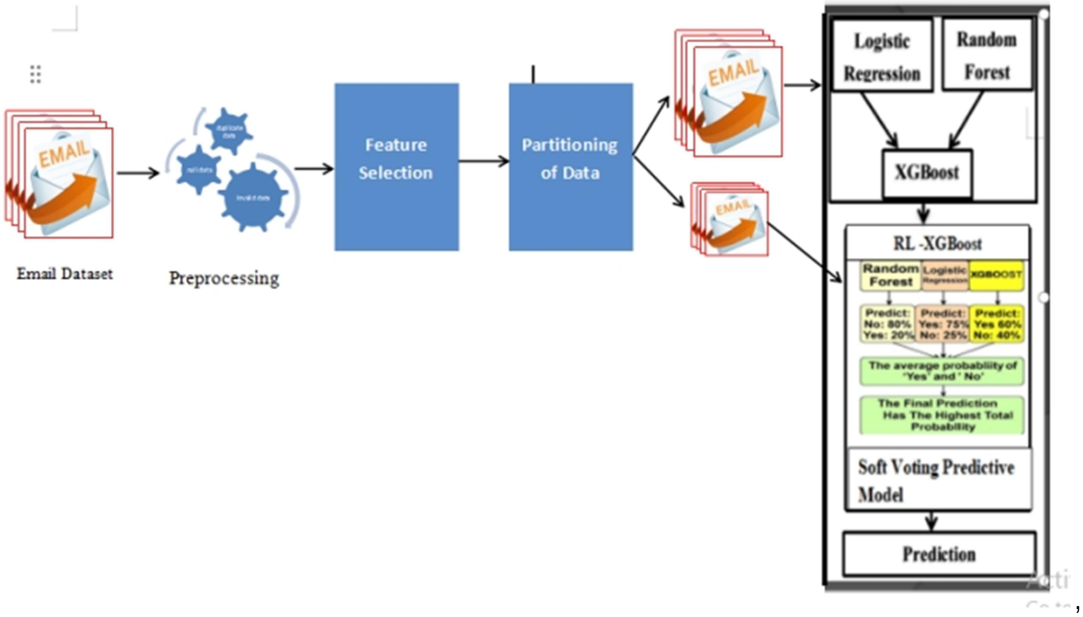

Figure 1: General Architecture for a Hybridized Model for Minimizing phishing email-based Social Engineering Attacks

3.2 Data Preprocessing

Data preprocessing involves parsing emails to ensure that the emails are fit in machine learning. Every .eml file will be separated into the corresponding sections: subject line, body data, email headings (From, To, Reply-To, etc.), embedded links and attachments.

There will be no unnecessary symbols, HTML tags and stop words. The email will then have features that will be extracted both in the textual and technical components. Examples are: frequency of trigger words such as urgent, verify, or password; message length and text complexity; Number of embedded URLs and attachments and Mismatch between sender and reply-to domains.

3.2.1 Data Cleaning

This is preliminary cleaning that consisted of the extraction of inconsistencies and conversion of the raw email files into a structured form. The cleaning processes included the following:
Duplicate Removal, HTML to Text Conversion, Normalization of Date Formats, Noise Removal.

3.2.2 Dataset Partitioning

The preprocessed and cleaned dataset emails was divided into the training and testing data using the 80:20 stratified sampling plan. The stratification was used to maintain the initial class difference, such that phishing and legitimate emails would be represented in the two subsets equally.

3.2.3 The deployment of the Hybrid Model(Randomized Logistic ExtremeGradientboost)

This research made use of a hybrid model where all three models were used as opposed to depending on a single model. All the models were trained on the dataset but they emphasized different aspects of the dataset. The process of processing a new email will involve each of the models making its own prediction and the results were added together as a form of a voting system to finalize the decision. The text-related patterns which can include the existence of popular phishing scams will be addressed by Logistic Regression and XGBoost and the structural patterns will be handled by the Random Forest, such as the disparity of the sender and reply-to address, or the number of embedded links and attachments. Such co-operation is beneficial in reducing the number of false positives (legitimate emails marked as phishing) and in reducing the number of false negatives (phishing emails overlooked). When these models are applied together, one will compensate for the weaknesses of the other.

4. RESULTS AND DISCUSSION

Across all evaluation metrics in Table 1, the Logistic Regression was found to be performing well with an accuracy of 0.86 and AUC of 0.78, it can be stated that the textual features, which were extracted from the body and the subject of the email messages, do not have no significant discriminative power. Nonetheless, its lower precision (0.71) and recall (0.74) means that it was not always able to deal with more profound, structural anomalies like malicious URLs, spoof sender domains and suspicious attachments. This is the worst choice of all since it is a linear model and therefore, it does not have the capacity to show the non-linear, and complex relationships which are present in phishing emails. Random Forest is more effective (accuracy of 0.89) because it deals with non-linear trends and it uses more than one decision rule, its results were better in all measures especially recall (0.86) and precision (0.84) which demonstrates that phishing emails have distinct non-linear signatures which can be identified by tree-based models. The higher score of F1 (0.85) is more balanced and it lowers the false positive and false negative. XGBoost is even more effective (accuracy, 0.92) due to the fact that it constructs the trees in sequence and corrects the errors of the previous trees.

In that way, it can recognises malicious phishing behaviour concealed in both technical metadata and textual information. It performed the best on the predictive performance when compared to the individual models. XGBoost with a precision of 0.88, recall of 0.90, and an accuracy of 0.92 showed excellent results in phishing email detection with low false-positive. The AUC score of 0.90 goes on to underscore its better discrimination power.

The hybrid RL-XGBoost model is the most successful as it is a combination of the three models that have complementary strengths. The Hybrid RL-XGBoost model is a weighted soft-voting ensemble model that is a combination of the predictive capability of Logistic Regression, Random Forest, and XGBoost. It was noted that the model was specifically optimized to achieve the highest recall and high precision in order to lower false alarms.

Table 1: Model performance evaluation

Metric	**Logistic Regression**	**Random Forest**	**XGBoost**	**RL-XGBoost (Hybrid)**
Accuracy	0.86	0.89	**0.92**	**0.94**
Precision	0.71	0.84	**0.88**	**0.90**
Recall	0.74	0.86	**0.90**	**0.96**
F1-Score	0.73	0.85	**0.89**	**0.93**
AUC	0.78	0.82	**0.90**	**0.92**

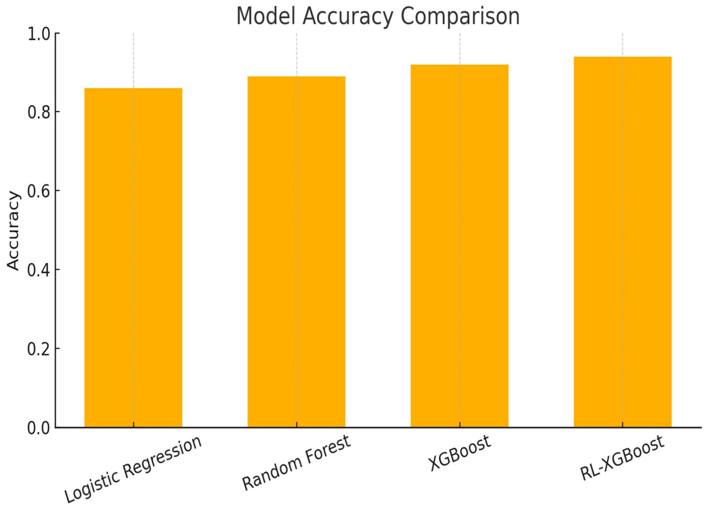

Figure 2: Accuracy Visualisation of five performance metrics

In Figure 2, Logistic Regression, Random Forest, XGBoost and RL-XGBoost had accuracy of 0.86, 0.89, 0.92 and 0.94, respectively, which indicated that the ensemble RL-XGBoost had the highest accuracy.

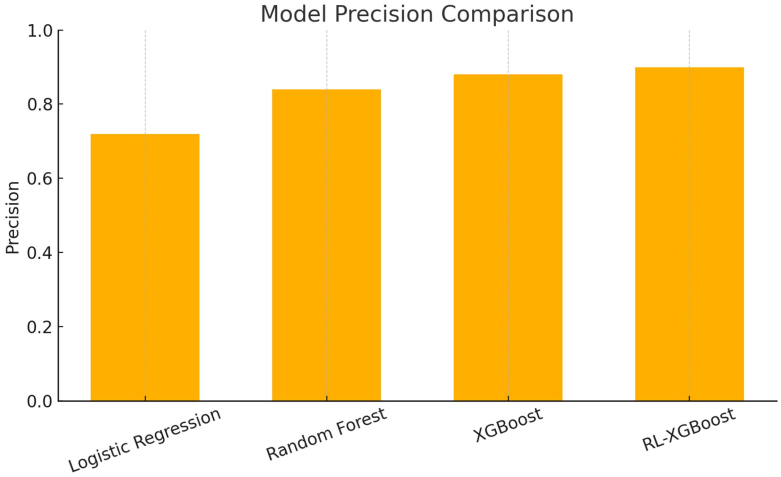

Figure 3: Precision Visualisation of five performance metrics

Moreover, in Figure 3, Logistic Regression, Random Forest, XGBoost and RL-XGBoost had precisions **of** 0.71, 0.84, 0.88 and 0.90, respectively. This is an indication that the ensembled model had the best precision value.

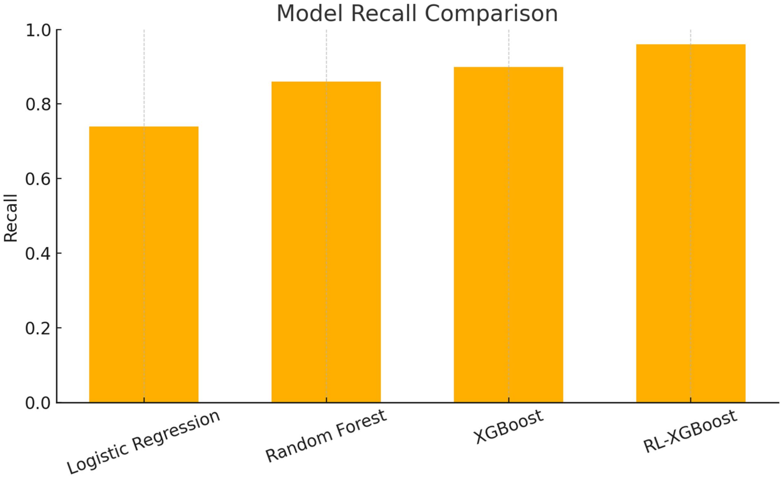

Figure 4: Recall Visualisation of five performance metrics

In Figfure 4, the recall values for Logistic Regression, Random forest, XGBoost and RL-XGBoost models were 0.74, 0.86, 0.90 and 0.96, respectively. In recall, RL-XGBoost outperformed the constituent models.

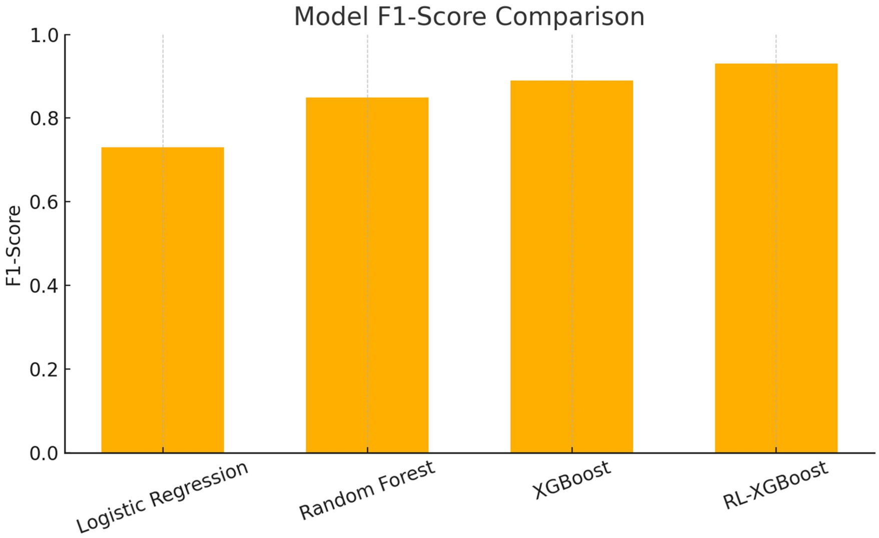

Figure 5: F1-score Visualisation of five performance metrics

Figure 5 presents the visualisation of F1-score of Logistic Regression (0.73), Random Forest (0.85), XGBoost (0.89) and RL-XGBoost. It is evident here that RL-XGBoost has the highest F1-score.

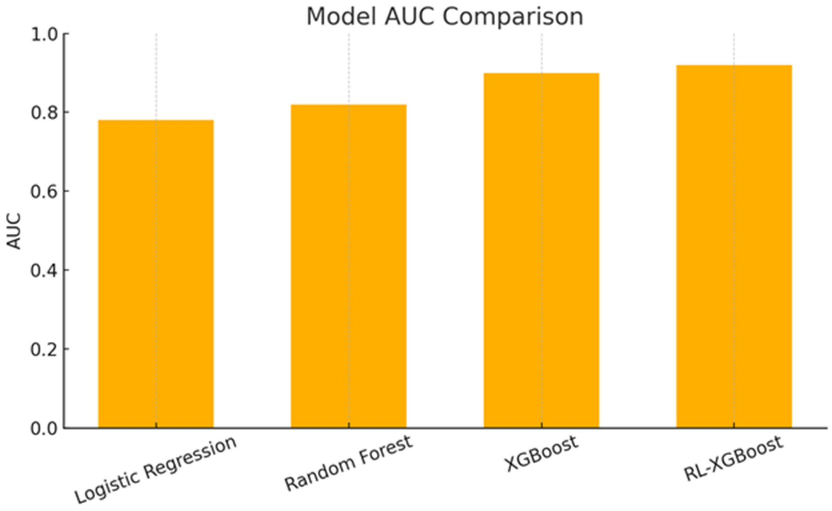

Figure 6: AUC Visualisations of four models

In Figure 6, the Area Under Curve for Logistic Regression, Random Forest, XGBoost and RL-XGBoost are 0.78, 0.82, 0.90 and 0.92, respectively. The confusion matrices of the three base models (Figures 7-9) are characterized by a distinct advancement in the detection ability with each model rectifying the limitations of the earlier model.

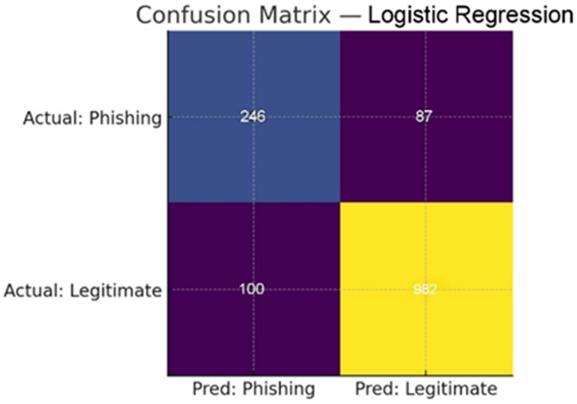

Figure 7: Confusion matrix for Logistics Regression model

In Figure 7, Logistic Regression makes the poorest showing, identifying 246 phishing emails correctly, but failing to identify 87 (false negatives) false emails, which is also suggestive of the highly variant patterns that phishing emails employ.

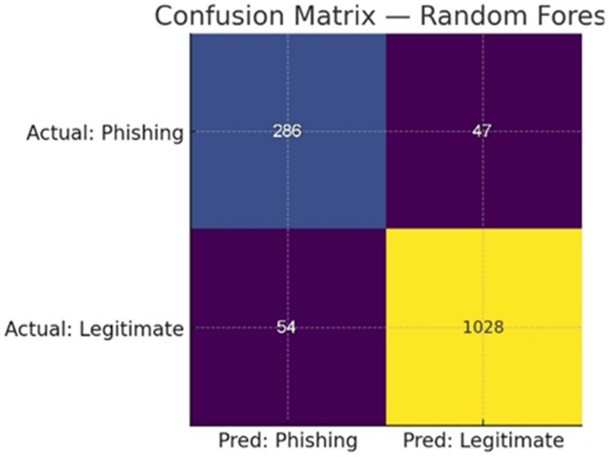

Figure 8: Confusion matrix for Random Forest mode

According to Figure 8, Random Forest offers a much better outcome because it achieves a lower false negative of 47 and a higher true positive of 286 which shows that it is a strong tool at identifying non-linear structural patterns like domain mismatch and abnormal URLs.

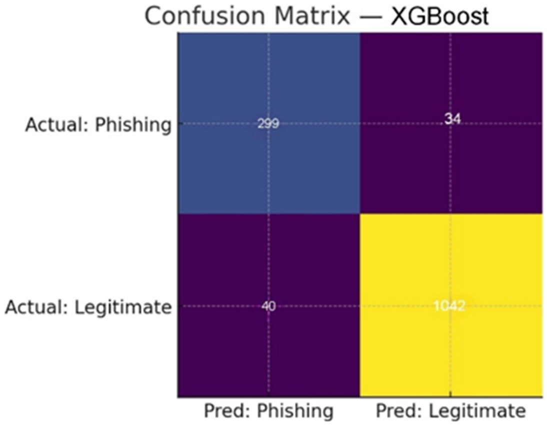

Figure 9: Confusion matrix for XGBoost model

In Figure 9, the XGBoost is the most effective of the individual models with 299 true positives yet with 34 false negatives since it has a boosting mechanism that learns previous errors and offers more detailed feature interactions. False positives also reduce gradually between the Logistic Regression and the Random forest and XGBoost. The confusion matrix of the RL-XGBoost model in Figure 10 better explained its high performance. It rightly detected 320 phishing mails and 975 genuine emails, 13 False Negatives, and generated 107 false positives. A middle range of false positive is acceptable in most cybersecurity systems. False alarms might be inconvenient to users, but they are a significantly less perilous problem than false negatives. This trade-off is satisfactory in a low-security environment like a university, a financial institution, and a small business that does not have dedicated IT people. The outcome of the hybrid model was thus very much consistent with the real-life cybersecurity priorities. The hybrid RL-XGBoost model is evidently the best in terms of all models.

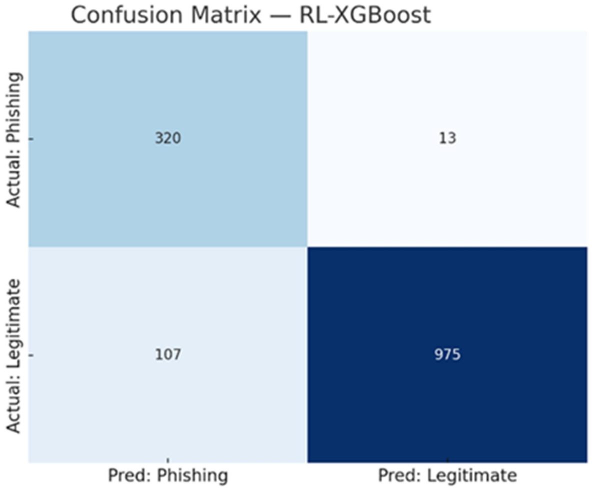

Figure 10: Confusion matrix for RL-XGBoost model

5. CONCLUSION

The results of this research indicate that machine learning is an effective and useful method of identifying email based social engineering attack. The various models were good individually, but they were not comprehensive enough to solve the problem individually. The hybrid model can effectively identify phishing emails with minimum false negative rate. This is more so in a cybersecurity setting where one misplaced phishing attack can cause severe breaches. The system designed in this case forms a solid base that could be implemented, upgraded or further developed into actual use, particularly as an add-on that could be downloaded to the enterprise email servers to label emails that pass across organisations as phishing or legitimate.

REFERENCES

1. Almseidin, M., Al-Jarrah, M., Al-Qerem, A., & Al-Abdallah, A. (2017). Evaluation of machine learning models for intrusion detection systems. Procedia Computer Science, 170, 1167–1174.
2. Alsufyani, A. A. & Alzahrani, S. M. (2021). Social engineering attack detection using machine learning: Text phishing attack. Indian Journal of Computer Science and Engineering, 12(3), 743–751.
3. Atlam, H. F. & Oluwatimilehin, O. (2023). Business email compromise phishing detection based on machine learning: A systematic literature review. Electronics, 12(1), Article 42. Available at: https://doi.org/10.3390/electronics12010042
4. Beridze, T., Sheng, O., & Wang, Y. (2022). Logistic regression for email threat prediction in enterprise environments. IEEE Access, 10, 52234–52248.
5. Bokhonko, O., Nosovsky, A., & Shelestov, A. (2024). Detecting phishing emails using optimized machine learning classifiers. Journal of Cybersecurity Research, 12(1), 1-18.
6. Butt, U. A., Amin,R., Aldabbas, H., Mohan, S.,Alofi, B., & Ahamadian, A. (2022). Cloud-based email phishing attack using machine and deep learning algorithm. Complex & Intelligent Systems, 9, 3043–3070
7. Jabbar, H. & Al-Janabi, S. (2023). AI-driven phishing detection: Enhancing cybersecurity with reinforcement learning. Journal of Cybersecurity and Privacy, 3(1), 1–18. Available at: https://doi.org/10.3390/jcp3010001
8. Koide, T., Fukushi, N., Nakano, H., & Chiba, D.(2024). ChatSpamDetector: Leveraging Large Language Models for Effective Phishing Email Detection. 20th EAI International Conference on Security and Privacy in Communication Networks (SecureComm 2024), October 28–30, 2024, Dubai, United Arab Emirates
9. Krombholz, K., Hobel, H., Huber, M., & Weippl, E. (2014). Advanced social engineering attacks. Journal of Information Security and Applications, 22, 113–122.
10. Lee, Y., Saxe, J. & Harang, R. (2020). CATBERT: Context-aware Tiny BERT for detecting social engineering emails. arXiv preprint arXiv:2010.03484. Available at: https://arxiv.org/abs/2010.03484
11. Lansley, C., O'Connor, M., & Harrop, S. (2024). Real-world phishing detection using behavioral and machine learning signals. Computers & Security, 129, 103235.
12. Lin, H., Chen, M., & Li, Y. (2021). Phishing email detection using natural language processing and machine learning. Expert Systems with Applications, 185, 115672.
13. Mouton, F., Leenen, L., & Venter, H. S. (2014). Social engineering attack frameworks. South African Computer Journal, 56, 2–24.
14. Rathee, D. & Mann, S. (2022). Detection of e-mail phishing attacks – using machine learning and deep learning. International Journal of Computer Applications, 183(47), 1–6.

15. Senturk, S., & Sogutpinar,I. (2017). Email phishing detection and prevention by using data mining techniques. doi: 10.1109/ubmk.2017.8093510
16. Shahrivari, V., Darabi, M. M. & Izadi, M. (2020). Phishing detection using machine learning techniques. arXiv:2009.11116