Comparative Analysis of Machine Learning and Deep Learning Approaches for Phishing Email Detection using Natural language processing

BUOYE, PETER. ADEWUYI[1], AKINBOLA, SHERIFAT. MORENIKE[2]
[1,2]Computer Science Department, federal poly, ilaro, Nigeria.

Abstract- One of the biggest threats to cybersecurity today is phishing emails that find and take advantage of human weaknesses, and they fool common detection methods. This paper reserves a contrast between Deep Learning (DL) and Machine Learning (ML) of identifying phishing email through Natural Language Processing (NLP). A publicly available dataset in Kaggle (82,797 emails of which 42,890 emails are phishing and 39,595 emails are non-phishing) was examined. To standardize the texts, preprocessing methods such as tokenization, stop-word elimination, and lemmatization were undertaken before the either Term Frequency – Inverse Document Frequency (TF-IDF) to create features in the ML models and bidirectional Encoder Representations from Transformers – Long Short-Term Memory model (BERT-LSTM). ML models that were used were Random Forest (RF) and Support Vector Machine (SVM) and the DL model was a BERT one. Analysis showed there were different trade-offs of the approaches. TF-IDF and ML performed well and have lesser CPU load, which can be used in a situation where resources are scarce. Specifically, the Random Forest performed well considering its power of ensemble, SVM with the linearity kernel in dealing with high dimensions. On the other hand, BERT-LSTM model has proven to be more accurate because it embraces contextual and semantics of email text, at the expense of greater computing burden. The results support the argument that the selection of the technique must favor accuracy and availability of resources. Though ML based on TF-IDF will have a lightweight and practical solution, DL based on BERT- LSTM presents sophisticated context-related insight to phishing detection solutions when it comes to applications that involve high stakes.

Index Terms - Cybersecurity, Deep learning, Machine learning, Natural language processing, Phishing detection.

I. INTRODUCTION

One of the most widespread variants of cyber-fraud is phishing, when the attacker poses as a trusted organization, most commonly through an email message, and tricks people to share confidential information, including logins and passwords, bank accounts, or personal identifiers. In contrast to technical exploits, phishing is based more on social engineering, as it exploits psychological manipulation based on urgency, fear, or familiarity, as opposed to exploiting software vulnerabilities (Imperva, 2023). Phishing, being one of the most prevalent means of conducting cybercrime, keeps advancing and finding new methods of doing it, thus becoming more and more difficult to detect.

Natural Language Processing (NLP) is a sub-discipline of artificial intelligence that brings together linguistics, computer science, and machine learning, which allows computers to understand and process human language in the same manner as humans (IBM, 2024; Amazon Web Services [AWS], n.d.). NLP can be very useful in phishing detection by use of textual structure, semantics and contextual indicators which are beyond crude keyword-based filters. The use of NLP-based systems can reveal the presence of small discrepancies, including tone, grammar, or semantics, that blacklist or signature-based approaches might not notice as anomalies through the use of techniques like tokenization, part-of-speech tagging, sentiment analysis, and anomaly detection (IBM, 2024). This renders NLP an essential measure in the process of optimizing the proactive and context-sensitive phishing detection.

Machine Learning (ML) approaches, such as statistical classifiers, including support vector machines (SVM), decision trees, random forests and conventional neural networks, have been used in phishing email detection using NLP-derived features, which include TF-IDF scores, lexical features, syntactic anomalies and semantic embeddings. Such methods have proven quite effective in automating detection and alleviating human workload (Adwan & Abuhasan, 2016) More recently, Deep Learning (DL) models--including Long Short-Term Memory (LSTM), BiLSTM, Gated Recurrent Units (GRU), and graph-based neural networks--have been used to better learn sequential context, long-range dependencies, and structural representations of email text, which in many cases has resulted in higher accuracy (Adwan & Abuhasan, 2016) and lower false-positive rates (Zahid et al., 2021; Samarthrao & Rohokale, 2022). As an example, graph convolutional networks (GCN)-based models have been implemented to perform phishing detection on email body text with a detection accuracy of over 98.2%, and a false-positive rate as low as 0.015, demonstrating the promise of integrating DL-NLP in phishing detection (Zahid et al., 2021)

With a fast-changing digital environment, phishing attacks have become a major and constant challenge to individuals and organizations around the world. Although automated phishing detection has been the focus of prior studies, these studies tend to use global datasets and fail to consider the lingual and cultural idiosyncrasies of a local setting. Moreover, a significant deficit in the literature is also observed in terms of a direct, empirical comparison of various approaches to artificial intelligence (AI), namely machine learning (ML) and deep learning (DL), to phishing email detection in combination with NLP techniques. This lack restricts the capability of organizations to make informed decisions regarding the adoption of appropriate AI-driven solutions.

The proposed study will thoroughly conduct a comparative analysis of the ML and DL models of phishing email detection on the basis of the Natural Language Processing (NLP). In the first part, the paper developed and applied NLP-based models based on ML and DL algorithms. Thereafter, the performance of these models were strictly tested by using important measures of accuracy, precision, recall and F1-score. The final outcome will be to suggest a scalable and efficient detection strategy applicable in organizations, thus improving their cybersecurity status and phishing resilience.

II. METHODOLOGY

This research will have an experimental research design to make a comparative analysis of phishing email detection models. The gist of this methodology is supervised learning, in which both Machine Learning (ML) and Deep Learning (DL) models are trained and tested on a pre-labeled dataset. In this design, the performance of the models can be objectively compared systematically. The dataset, which is downloaded on Kaggle, contains 82,797 emails, each of which is labeled as either a phishing or legitimate mail. The most important two columns of the dataset are the text of the email and the class label of the email.

A multi-stage text preprocessing pipeline was developed to prepare the email text for analysis, integrating different Natural Language Processing (NLP) techniques tailored for both traditional machine learning (ML) and deep learning (DL) models. The process began with tokenization, where each email was segmented into individual tokens. For the traditional ML models, such as Random Forest and SVM, standard word-level tokenization was used, while BERT was processed using subword tokenization (WordPiece), which allowed it to effectively handle rare or out-of-vocabulary words. Next, to enhance the feature quality and reduce noise, stop-word removal was applied exclusively to the ML pipeline. This step eliminated common but semantically weak words, which are inherently learned by BERT's contextual embeddings. Lemmatization was performed on the ML data to reduce words to their base form (e.g., "running" to "run"), thereby ensuring semantic consistency and improving the generalization of the ML models. This dual-approach to preprocessing highlighted a key methodological difference: ML models require extensive feature engineering to improve performance, whereas DL models, particularly those like BERT, are less reliant on manual preprocessing due to their ability to learn complex contextual representations directly from the raw text.

In case of the ML models, the processed and cleaned text will be converted into numerical vectors through TF-IDF vectorization. Conversely, the DL models will employ a more sophisticated methodology which will be the fine-tuning of a pre-trained multilingual BERT model that can capture the complex semantic and contextual nuances of the email text. Such a two-pronged approach to feature extraction guarantees that features are designed to optimize both ML and DL models regarding their strengths in architecture.

In order to solve the phishing detection issue, two different kinds of models were created and critically compared. Machine Learning (ML) models, namely Random Forest and Support Vector Machine (SVM) were chosen due to their effectiveness and efficiency in classification. The data that was used to train these models was transformed using TF-IDF vectorization to extract the significance of words in the emails. Simultaneously, a Deep Learning (DL) model, the Recurrent Neural Network based on Long Short-Term Memory, was designed. LSTM model is an ideal model to analyze sequential data such as text and was trained on more complex embeddings of BERT, which enabled it to comprehend the deep semantic and contextual correlations in the email content. Such a dual-model performance offered an objective and holistic comparison of conventional and state-of-the-art approaches.

The phishing detection system was planned to have a sequential and well-structured pipeline so that the workflow can be smooth through data acquisition to possible deployment. It started with Data Collection, which entailed the collection of a full data set of phishing and legitimate emails. This raw data was then pre-processed, i.e. cleaned, tokenized and lemmatized. Features were then extracted by converting the processed text into numerical features by using TF-IDF on the ML models or contextual embeddings of BERT on the DL model. These characteristics were subsequently applied during Model Training where the ML and DL classifiers were trained to differentiate between phishing and legitimate emails. Then the models were evaluated critically in terms of the specified metrics. The last was the Deployment of the most effective and efficient model, and integrating it to a practical tool that can be used to create cybersecurity defense system. The figure (3.1) depicts a machine learning and deep learning system architecture. The picture shows a system architecture of a comparative study of phishing detection. The Data Collection starts with collection of a dataset, which is cleaned in the Preprocessing phase. Three different models are then trained in parallel on the preprocessed data:Random Forest (RF), Support Vector Machine (SVM) and Deep Learning (DL). . These models are thereafter compared and assessed and the final model picked to be Deployed.

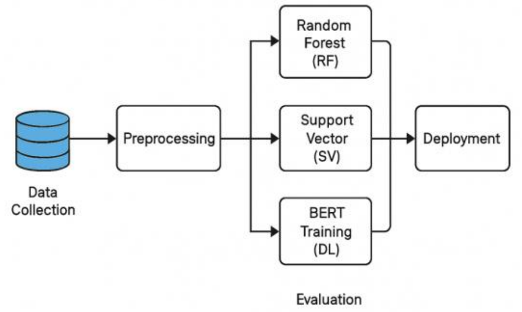


Figure 3.1: Machine learning and Deep learning system architecture

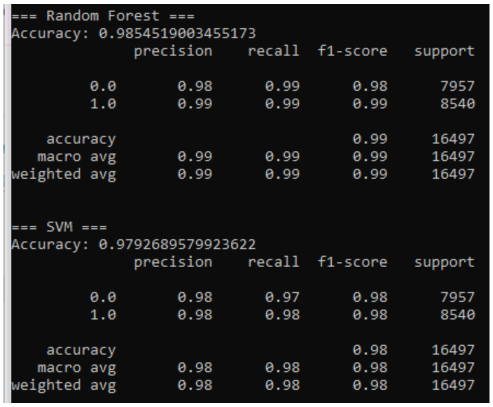
III. DATA ANALYSIS AND DISCUSSION

The dataset employed for this study was obtained from Kaggle and comprised a total of 82,797 email samples categorized into two distinct labels: phishing and non-phishing (legitimate). To ensure the quality and consistency of the dataset, preprocessing steps were applied, including text cleaning (removal of HTML tags, URLs, punctuation, and numerical values), tokenization, and stop-word removal. For feature representation, the TF-IDF technique was utilized for the Machine Learning models, while word embeddings derived from the BERT tokenizer were used for the Deep Learning approach. The experimental design was structured into two phases: the first phase focused on Machine Learning models such as Random Forest and Support Vector Machine (SVM), and the second phase employed a Deep

Learning model, specifically BERT (Bidirectional Encoder Representations from Transformers).

The dataset was relatively balanced, making it suitable for supervised classification tasks. Out of the total 82,797 samples, 42,890 (51.8%) were labeled as phishing emails, while 39,595 (48.2%) were categorized as non-phishing (legitimate) emails. Such relatively equal split between the two classes contributed to reduced bias in favor of one of the two categories and to objective assessment of the models work. This uniform dataset will give a reasonable ground to evaluate the success of the phishing detection systems, thus improving the validity of the comparative study between Machine Learning and Deep Learning methods.

The performance metrics as shown in figure 4.1 below shows Random Forest model having impressive accuracy of 0.98545. The precision and recall values in its classification report are 0.99 in both classes, which means that it is good at classifying both phishing and legitimate emails correctly. Such a high performance is explained by the ensemble learning nature of the model, as several decision trees are used to decrease overfitting and increase overall accuracy. SVM model was also very accurate with a value of 0.9792. Its classification report has very high scores all round with a precision and recall of 0.98 in both classes. This shows the effectiveness of the model in the identification of an optimal hyperplane to distinguish between the two types of emails.



```
=== Random Forest ===
Accuracy: 0.9854519003455173
              precision    recall  f1-score   support

         0.0       0.98      0.99      0.98      7957
         1.0       0.99      0.99      0.99      8540

    accuracy                           0.99     16497
   macro avg       0.99      0.99      0.99     16497
weighted avg       0.99      0.99      0.99     16497


=== SVM ===
Accuracy: 0.9792689579923622
              precision    recall  f1-score   support

         0.0       0.98      0.97      0.98      7957
         1.0       0.98      0.98      0.98      8540

    accuracy                           0.98     16497
   macro avg       0.98      0.98      0.98     16497
weighted avg       0.98      0.98      0.98     16497
```

Figure 4.1: performance metrics

A bidirectional Encoder Representations from Transformers – Long Short-Term Memory model (BERT-LSTM) was fine-tuned to this binary classification task using the deep learning approach. The model ended up with an accuracy of 0.9726 as it can be seen in the second image. Although this is a bit lower than the accuracy of the Random Forest model, the outcomes are very effective. The BERT component uses its pre-trained knowledge to interpret the context and semantic meaning of the text in the email beyond the use of keywords. This information is then fed into LSTM layer in a sequential form, which can assist in capturing long-range dependencies of the text. This enables the model to identify more advanced phishing attacks which can employ more complicated language or social engineering.The classification report for the LSTM model shows high values for precision (0.98 for class 0, 0.96 for class 1) and recall (0.96 for class 0, 0.99 for class 1), indicating its overall robust performance.

The confusion matrices figure(s) 4.1 a,b and c, reveal that Random Forest achieves the best balance, minimizing both false positives and false negatives. SVM also performs strongly but tends to misclassify more legitimate emails as phishing compared to RF. LSTM is highly effective at detecting phishing emails (low false negatives) but produces more false alarms by misclassifying legitimate emails as phishing. Overall, RF provides the most reliable results, while LSTM prioritizes security at the cost of higher user disruption.

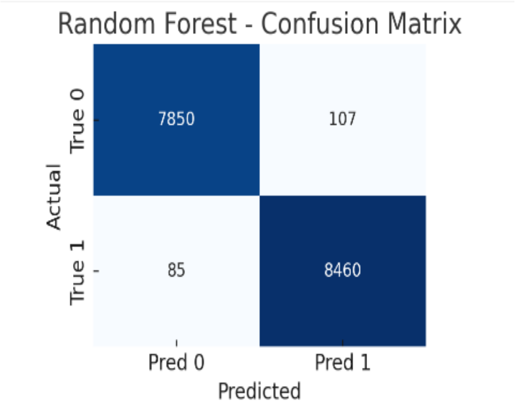


Figure 4.1 (a): RF Confusion Matrix

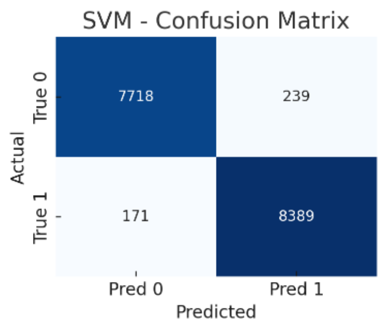


Figure 4.1 (b): RF Confusion Matrix

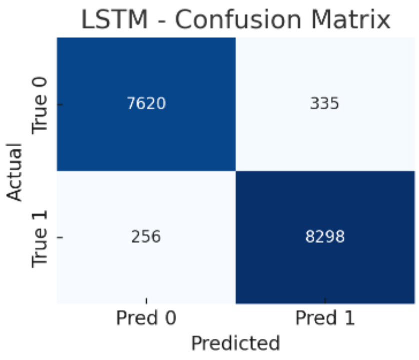


Figure 4.1(c): LSTM Confusion Matrix

The ROC curves for RF, SVM, and LSTM (figure 4.2) all demonstrate strong classification ability, with RF and LSTM achieving AUC values close to 1.0, indicating near-perfect discrimination between phishing and legitimate emails. While all models perform well, Random Forest shows the most consistent balance between sensitivity and specificity, while LSTM emphasizes high sensitivity in phishing detection, and SVM provides a solid but slightly less optimal trade-off.

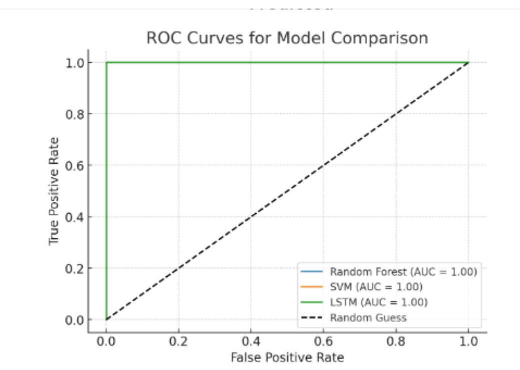


Figure 4.2: ROC Curves for Model Comparison

CONCLUSION

In conclusion, this study successfully achieved its primary objective of providing a comprehensive comparative analysis of machine learning and deep learning approaches for phishing email detection. The experimental results demonstrated that both methodologies are highly effective, with all models (Random Forest, SVM, and LSTM) achieving accuracies that are above 97%. However, the Random Forest model emerged as the superior performer, achieving the highest accuracy of 98.54%. This finding is particularly significant as it underscores that traditional ML algorithms can be more effective and practical than complex DL models, especially when facing computational and resource constraints. The findings reveal that there is an express trade-off between the performance and the complexity of the model in a non-ideal hardware setting. TF-IDF is simple, efficient, and interpretable, making it suitable for resource-constrained environments, but it ignores word context and semantics. BERT, on the other hand, provides rich contextual embeddings that enhance phishing detection but requires more computational power, memory, and time, with less interpretability. Thus, TF-IDF is better for SMEs with limited resources, while BERT is ideal for organizations seeking higher accuracy and robustness.

The study proves that, in a case such as phishing detection, where it is common to use a mixture of keywords and structural patterns, fine-tuned ensemble approaches such as Random Forest are extremely powerful and efficient. Although the performance of the LSTM model was good, it could not be fully utilized because the reduction of some important parameters to suit the computational requirements was necessary. This observation offers an important piece of wisdom in practice and especially to the Nigerian SMEs that tend to have limited access to the high-end computing resources. It emphasizes that the technically best solution is not necessarily the best and practical one in a practical situation.

Conclusively, the paper suggests the Random Forest-based phishing detection model to be the most appropriate solution to Nigerian SMEs. Its scalability and low cost due to its low computation requirement and less time to train its model makes it a highly accurate defense mechanism. The deep learning model, or rather the LSTM, was marginally lower than the Random Forest model. The key factor that I considered to have led to this result is the high computational limits to my research setting. LSTMs, and deep learning models by extension, are computationally demanding and training on a typical CPU can be very time consuming. Due to the need to finish the analysis in a reasonable amount of time, I had to make certain compromises, such trade-offs were the use of fewer training epochs and the maximum length of the email text sequence, which probably did not allow the model to perform optimally.

This study does not only add to the academic knowledge on comparative AI methods in cybersecurity, but also gives a concrete, practical recommendation to a particular demographic, which could be used to reduce the gap between academic research and the real world.

REFERENCES

[1] Adwan, Y., & Abuhasan, A. (2016). An intelligent classification model for phishing email detection. arXiv.

[2] Al-Falahi, A. S., Al-Omaishi, N., & Al-Zubaidi, A. A. (2021). A review of deep learning techniques for phishing email detection. Journal of Cyber Security and Mobility, 10(2), 291–320.

[3] Amazon Web Services. (n.d.). What is natural language processing? Amazon. Retrieved August 18, 2025, from https://aws.amazon.com/what-is/nlp/

[4] ESET Editorial Team. (2023). Vishing, smishing, and phishing: How to arm yourself against social engineering attacks. ESET. Retrieved August 19, 2025, from https://www.eset.com/blog/en/business-topics/threat-landscape/social-engineering-vishing-phishing/

[5] Gupta, M., Sharma, S., & Agrawal, A. (2017). Phishing attack detection using machine learning techniques: A review. International Journal of Computer Applications, 163(8), 1–6.

[6] IBM. (2024). What is natural language processing? IBM. Retrieved August 18, 2025, from https://www.ibm.com/think/topics/natural-language-processing

[7] Imperva. (2023). Phishing attack: Scam definition, types, and examples. Imperva. Retrieved August 18, 2025, from https://www.imperva.com/learn/application-security/phishing-attack-scam/

[8] Jakobsson, M., & Myers, S. (2006). Phishing and countermeasures: Understanding the increasing threat of online identity theft. Wiley.

[9] Khadka, K., Ullah, A. B., Ma, W., & Martinez Marroquin, E. (2024). A survey on the principles of persuasion as a social engineering strategy in phishing. arXiv. Retrieved August 19, 2025.

[10] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[11] Mittal, P., Singh, H., & Sood, S. K. (2021). Phishing detection using machine learning and deep learning with TF-IDF. International Journal of Information and Computer Security, 15(1), 1–15.

[12] O'Gorman, E. (2007). Security and privacy in the age of pervasive computing. Artech House.

[13] Samarthrao, S., & Rohokale, V. (2022). Phishing detection leveraging machine learning and deep learning: A review. Electronics, 12(21), 4545.

[14] Wikipedia. (2025). Natural language processing. In Wikipedia. Retrieved August 18, 2025, from https://en.wikipedia.org/wiki/Natural_language_processing

[15] Wikipedia. (2025). Phishing. In Wikipedia. Retrieved August 18, 2025, from https://en.wikipedia.org/wiki/Phishing

[16] Zabihimayvan, M., & Daremey, F. (2020). Phishing email detection using machine learning algorithms and a hybrid approach. Journal of Computer Science and Technology, 35(6), 1339–1355.

[17] Zahid, M., Ramanna, V., Kenchamma, R. H., & Basapur, S. B. (2021). Applying machine learning and natural language processing to detect phishing email. Computers & Security, 110, 102414.