Multi-layered Phishing Email Detection Using Convolutional Neural Networks Approach

Mutahir Shaukat
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
mutahirshaukat120@gmail.com

Saqlain Raza
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
abidh5791@gmail.com

Sana Tariq
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
sanatariq@iub.edu.pk

Iram Haider
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
iramhaider765@yahoo.com

***Abstract*—Phishing attacks carried out through deceptive emails sustain a relationship of critical risk and threat to personal and organizational cybersecurity. To assess techniques used to target and identify phishing emails, this study integrates the analysis of Emails, URLs, NLP, and the associated Metadata. In addition to Random Forest and XGBoost, the outreach of the analysis to deep learning Neural Network, specifically Convolution Neural Network, was critiqued in detail. The analyses were carried out on a public Phishing Email Dataset from Kaggle provided by user Naserabdullahalam. CNN Model performance results indicate an accuracy of 91.2% with a 92.1% precision, 89.7% recall, and an F1 score of 90.9%. The performance results are indicative of the model detecting phishing emails based on structural and linguistic features. However, the results are indicative of simplicity and potential bias of the dataset rather than an actual reflection of real-world performance. There was an 80:20 train-test split without a separate validation set, potential redundancy in the dataset, and manual feature fusion with embeddings, without clear details and methodology were downsides. Although the potential is clear, the approach captures better than the false positive single detection method theories.**

Index Terms—Phishing Detection, **Convolutional Neural Networks (CNN)**, **Deep Learning**, **Cybersecurity**, **Multi-layered Detection**

I. Introduction

Phishing is still one of the most common and highly damaging types of cybercrime. It exploits trust and social engineering toward individuals and companies alike [1]. This type of cybercrime usually involves fraudulent emails that mimic and pass off as legitimate correspondence, tricking people into disclosing sensitive information such as passwords, social security numbers, and even bank information [2]–[5]. Because phishers are constantly regrouping and rationalizing new attacks, detecting phishing attempts has become more and more difficult [6]. Historically, phishing scams have been detected through the use of blacklisting, rule-based, and pattern-matching heuristics [7]–[9]. While effective in dealing with known threats, these approaches largely miss zero-day phishing attacks and other concealed malicious emails that bear no resemblance to known patterns [10]. Moreover, with the adaptive nature of attackers, this scenario is best described as an arms race [11]. More recent work attempts to analyze a phishing email in parts, be it the message body, the URLs, or the email headers [12]. Unfortunately, this approach is limited in its ability to detect phishing attacks, as attackers are free to manipulate one detection target while the rest of the email stays benign [13]. This leads to a gap in the intersection of phishing research pertaining to the minimum integrated multi-dimensional approaches that collaboratively assess the different aspects of phishing emails, which can potentially enhance accuracy and strength simultaneously [14]. To face this issue, we provide a phishing detection framework that is multilayered and includes the use of Natural Language Processing (NLP) for content analysis, metadata in emails, and URL tokens. This strategy aimed to uncover advanced phishing attacks that approached the problem from many different angles. To find the optimal method for detection, we compared the efficacy of convolutional neural networks (CNN) with other standard machine learning frameworks. The current paper outlines these contributions. Firstly, we conducted a comparative analysis for phishing detection, evaluating standard machine learning and cutting-edge deep learning frameworks to identify the most effective method [15]. Second, we proposed the first multi-layered detection approach that concurrently evaluates textual content, metadata, and URL framework [16]. Third, we proved our method in extensive research using authentic phishing datasets, showing that our method offered marginally better detection with respect to the baseline models [17].

II. Literature Review

Phishing detection methods have evolved a lot over time, and all current research is focused on deep learning and machine learning approaches [18]–[20]. Machine learning

DOI: https://doi.org/10.67535/tsp.000001.007
Published under the ICACNC 2025 Proceedings

methods for phishing detection have been subject to many studies, using classifiers such as Support Vector Machines (SVM), Random Forest (RF), and Naive Bayes [1], [12]. These models use characteristics of textual data and metadata. They then determine if the email is phishing or legitimate. These approaches are successful as they need wide-ranging feature engineering and work to create effective detection systems. Salloum et al. [2] made a deep study of phishing email IDs via natural language processing, noting the strong effect of machine learning types if mixed with great feature extraction ways. Likewise, Rastenis et al. [12] demonstrated some potential of multi-language spam/phishing classification using email body text to help automate security incident investigation.

A. Deep Learning-Based Approaches

Deep learning advancements have led to the use of Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and also Transformer-based architectures for phishing detection [7], [14]. The algorithms improve detection accuracy as they recognize phishing traits inside of email content. Still, they often get help from key computer tools, and big-tagged data sets [15]. Do et al. [6] developed a mostly thorough taxonomy of many deep learning methods for phishing, recognizing pose difficulties as well as future paths. Their studies focused on how deep learning models found some trends within phishing emails. Kyaw et al. [10] did a systematic review of several deep learning techniques. It was great for phishing email detection, presenting the progression of methods from common machine learning to deeper neural architectures.

B. Hybrid Models with Multi-Layer Analysis

Recent research proposes hybrid models with the integration of several phishing detection methods, including URL inspection, into metadata analysis, as well as NLP-based text categorization [21], [16]. Multi-model methods incorporate some phishing clues. They do improve detection precision and lower false positives. Bountakas and Xenakis [13] introduced almost all HELPHED, a hybrid ensemble learning approach for phishing email detection that combines several analysis layers. Their study showed worth for a mix of many feature sets with analytic methods. Likewise, Atlam and Oluwatimilehin [17] did furnish a complete literature review of business email compromise phishing detection with machine learning, highlighting the value in thorough, multi-layered approaches.

Prior studies for detecting phishing centered mainly on single methods, looking at emails, headers, and also URLs alone. It produces a certain gap using insufficient multi-layered detection systems. This gap exists in thorough forms. Furthermore, much of the research now skips real-time detection capabilities, still important for the identification of complex phishing efforts that bypass conventional filters [22], [23]. Our literature assessment showed the necessity for a phishing detection strategy incorporating URL inspection, as well as email metadata and NLP-based email body analysis, in a framework to increase accuracy plus lower false positives [11].

III. Methodology

Figure .1 illustrates the overall architecture of the proposed methodology, showing the systematic flow from dataset acquisition through preprocessing, feature engineering, and CNN model development.

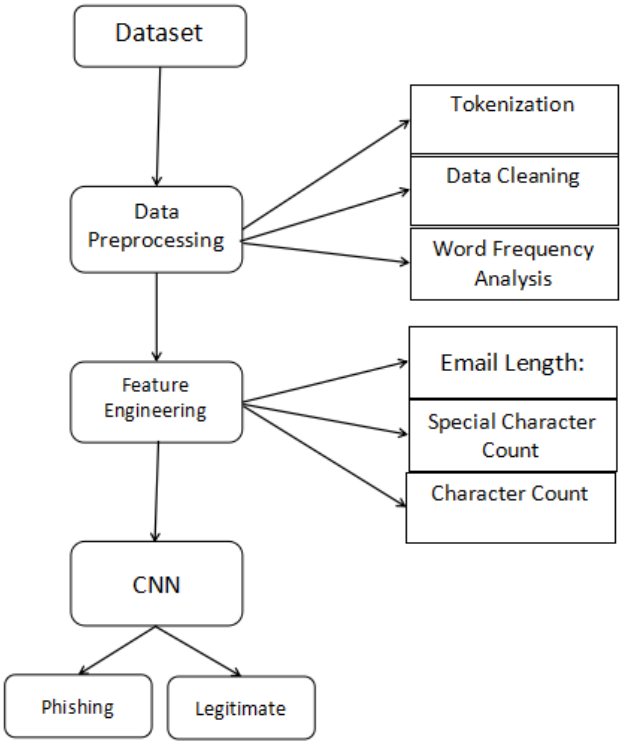

Fig. 1. CNN model architecture used for phishing email classification.

A. Dataset

We utilized the “Phishing Email Dataset” which was available on Kaggle and was published by the user Naserabdullahalam. The dataset contained full email body texts along with their classification either as phishing (1) or legitimate (0) emails. Out of a total of 39,154 emails, 21,842 were classified as phishing and 17,312 were classified as legitimate. After the initial preprocessing, we performed extensive exploratory data analysis (EDA). This helped us uncover the specific features that are indicative of a phishing email such as a pattern of using language that expresses a sense of urgency, the presence of potentially malicious hyperlinks, and and odd succession of characters. Legitimacy, however, was associated with a wider range of positive and neutral vocabulary. Class distribution, as shown in figure 2, illustrates that the phishing category slightly exceeded the legitimate emails, which constitutes a minor class imbalance. In order to retain the class distribution, we performed stratified sampling during the dataset split, assigning 80% of the data for training and the remaining 20%

for testing. We allocate 10% of the training data to a validation set so we can track a model's performance throughout training and study any potential overfitting.

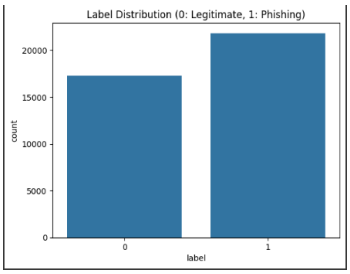

Fig. 2. Label Distribution (0: Legitimate, 1: Phishing

B. Data Preprocessing

There were multiple steps involved in our preprocessing. For the first step of text cleaning, we made all the text lowercase, stripped the text of HTML tags, and took out special characters, leaving only intended punctuation for feature extraction. For the handling of missing values, we defined and dealt with the null values in our dataset, and then removed incomplete records, which represented under 0.5% of our overall data. For feature extraction, we derived many useful features, which included the email length (in words), total character count, distribution of special characters, and total digits. During email text tokenization, we converted cleaned email text into numerical sequences using a specific tokenizer. This tokenizer helped reduce overall dimensionality, utilizing a vocabulary of 5000, which included the most frequently occurring words. We utilized a padding length of 200 tokens for the sequences, thereby facilitating uniform sequences for ease of batch processing in the neural network.

C. Feature Engineering

Feature engineering included several components. Text Normalization: We discarded stopwords, numbers, and unnecessary punctuation; however, we kept context-relevant terms that could signify phishing attempts. Statistical Feature Extraction: We explored the distribution of characters and words in phishing and legitimate emails to understand the features that differentiate the two. Visualization: We created distribution graphs and word clouds to identify different patterns. We found that phishing emails had terms associated with urgency, such as "verify," "urgent," and "account suspended." Feature Representation: For the CNN model, we implemented an embedding layer to derive dense word vector representations from the data. This enabled the network to understand the semantic relationships between words, bypassing the use of pre-trained embeddings. Table I describes the main features that were extracted from the emails and their importance in identifying phishing attempts. While these features were used during EDA, the CNN learned them implicitly through its embedding and convolutional layers instead of having them manually intertwined with the embeddings.

TABLE I
FEATURE CATEGORIES AND THEIR ROLE IN PHISHING DETECTION

Feature Category	**Description and Significance**
Email Length	Number of words and characters; phishing emails tend to be shorter.
Vocabulary Diversity	Lexical richness and variety; legitimate emails show greater diversity.
Special Characters	Frequency and Distribution; phishing emails often contain unusual patterns.
URL Characteristics	Domain age, length, and suspicious patterns; phishing emails contain malicious URLs.
Text Sentiment	Emotional tone of content; phishing emails often evoke urgency or fear.

D. Model Architecture

We developed a Convolutional Neural Network (CNN) architecture for phishing detection due to its effectiveness in identifying spatial patterns in sequential text data. It started with an embedding layer that transformed text tokens into a dense, 128-dimensional vector, thus learning representations of words for phishing detection. Then, we implemented a 1D convolutional layer with 128 filters of kernel size 5, wherein the ReLU activation function helped in identifying suspicious n-gram patterns in the text. The global max pooling layer helped in reducing the dimensions spatially while retaining salient features, thus ensuring that patterns detection was position invariant within the email while the most spatial features were detected. The fully connected part of the network had 64 neurons in a layer with ReLU activation to transform features and learn patterns in a non-linear fashion. To combat overfitting, we added a dropout layer with a 0.3 rate. The final layer provided the binary output with a sigmoid function, generating a score indicating the phishing probability of the email—0 for legitimate and 1 for phishing.

This architecture aimed to detect phishing attempts by analyzing text sequences at a local level. The convolutional layer examined the text by emphasizing potentially phishing phrases and flagged suspicious phishing characteristics. The global max pooling layer aided robustness by ensuring that key patterns were captured regardless of their location within the email.

This, in turn, enabled the model to handle diverse structures and phishing tactics used in emails. We used the Adam optimizer with a learning rate of 0.001 to optimize the CNN model and used binary cross-entropy for the loss function because it's suitable for binary classification. We trained the model for 15 epochs with 32 samples in each batch. In order to evaluate generalization during training, we used the validation set.

For assessing the performance of the CNN architecture, we built three baseline models to compare against. The LSTM network with 128 LSTM units was able to learn the sequential dependencies in the text. For the traditional machine learning approach, XGBoost, the gradient boosted algorithm, was trained on the TF-IDF extracted email text. As per convention,

Random Forest, the ensemble algorithm with 100 decision trees, was also trained on the TF-IDF to provide another baseline. To enable a fair comparison, all models used the same train-validation-test split. Hyperparameter tuning for the baseline models was done through a grid search coupled with 5-fold cross-validation on the training set.

Four regular metrics applied for binary classification enabled evaluation for all models. Total emails accounted included all correctly classified emails for determining accuracy. Proportion of classified phishing emails that were true phishing emails assessed precision. This assessed ability of the model to avoid false alarms. Proportion of phishing emails that were true phishing emails assessed recall.This focused on the capability of the model to identify threats. The F1-Score was defined by the harmonic mean of precision and recall. This provided a balanced perspective of model performance. Using these metrics, all models could be evaluated for phishing detection. For all assessments, the false-positive and false-negative rates were considered, which are extremely important in cybersecurity use cases.

IV. Experiments and Results

A. Experimental Setup

- **Environment**: The experiments were performed in a Google Colab environment as well as with necessary libraries such as sci-kit-learn and Keras or TensorFlow. These libraries provided a number of tools for model creation for the training and detection of content.
- **Evaluation Metrics**: Every model underwent assessment according to four different metrics pertaining to performance in classification. When measuring accuracy, we calculated the proportion of all made predictions that were correct. To measure precision, we looked at the actual phishing emails in the pool of emails identified as phishing and determined the extent to which phishing emails were incorrectly classified and counted as false positives. For recall, we measured the total phishing attempts that the model correctly identified, which correlates directly to the model's capability to determine phishing attempts that are genuine. The F1 score describes the balance in the metrics to fully represent the extent of the dataset which is only mildly imbalanced. As the harmonic mean of the precision and recall, it directly describes the balance.
- **Train-Test Split**: We implemented stratified sampling to maintain class distribution for all sets to record training, validation, and testing sets as 80/10/20 percentages, respectively. This balances the distribution of phishing and legitimate emails throughout the experiment. The validation set allowed for monitoring of training to implement early stopping, and the test set, which was for model development, provided a fair assessment for performance evaluation
- **Training Details**: All models were assessed based on the four common definitions of excellence: the proportion of classified emails that were accurate, precision— the ratio of actual phishing emails within those flagged as phishing, recall— the ratio of actual phishing emails that were identified, and the F1 score— the harmonic mean of precision and recall. Having all these definitions of excellence together forms the basis for a comprehensive assessment of performance. This is important for applications in cybersecurity due to the critical implications of false positives and false negatives.

B. CNN Model Performance

The CNN model was exceptional in all dimensions of evaluative assessment, confirming it as the most successful method in our research. As stated in Table II, the model registered 91.2% accuracy, 92.1% precision, 90.9% F1-score, and 89.7% recall on the test dataset. With these results, one can appreciate the magnitude of the success with which the CNN discriminated between phishing and legitimate emails, all the while keeping a fair balance between false positives and false negatives.

TABLE II
CNN MODEL PERFORMANCE

Metric	Score
Accuracy	91.2%
Precision	92.1%
F1-Score	89.7%
Recall	90.9%

A precision of 92.1% shows that, when the model classified an email as phishing, it was correct most of the time. This helps reduce the likelihood of false alarms. An 89.7% recall shows that the model identified nearly 90% of the phishing emails, thus providing useful coverage against harmful emails. The 90.9% F1 score means that the model was able to achieve an appropriate balance between precision and recall, which is extremely important when deploying these types of systems in the area of cybersecurity.

For the training, early stopping was triggered at epoch 12, which restored the optimal weights of epoch 7, where validation loss was at its minimum. This was a means for avoiding overfitting as it stopped training before the model started learning the training data patterns and was losing the ability to generalize to unseen data.

Figure 3 shows the training and validation accuracy over the epochs, while Figure 4 shows the corresponding loss. The dashed vertical line in both figures shows the epoch during which early stopping was triggered. This shows that the validation metrics did not change very much over the epochs and that the model did not overfit the training data.

C. Results Summary

As part of the CNN architecture evaluation, we also included comparisons to the LSTM, XGBoost, and Random Forest models as baselines. The general comparison of all the evaluation criteria is presented in Table III.

Based on all the metrics, it was evident the CNN model surpassed the performance of the baseline methods. While

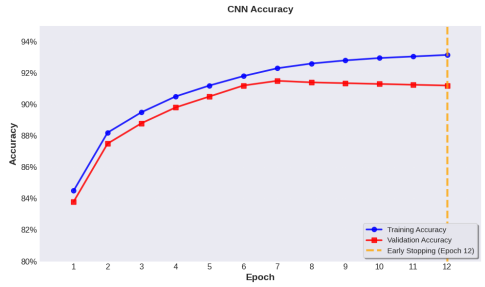

Fig. 3. CNN Accuracy Curve with Early Stopping Indicator

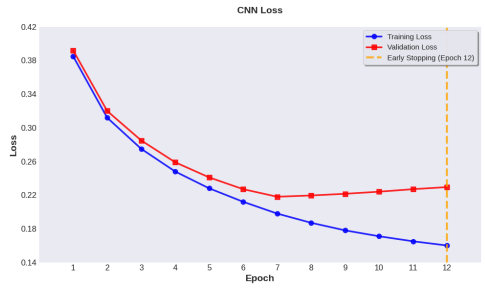

Fig. 4. CNN Loss Curve with Early Stopping Indicator

TABLE III
MODEL PERFORMANCE COMPARISON

Model	Accuracy	Precision	F1-Score	Recall
CNN	91.2%	92.1%	89.7%	90.9%
LSTM	89.5%	90.2%	87.8%	87.0%
XGBoost	85.7%	86.4%	83.9%	85.1%
Random Forest	83.9%	84.7%	82.0%	83.3%

the LSTM model did attain an accuracy of 89.5%, CNN had an accuracy of 91.2% and outperformed it on almost all metrics. The difference was even greater relative to the previous machine learning models. For example, XGBoost achieved an accuracy of 85.7%, while Random Forest achieved 83.9%. The difference described exemplifies the advantages deep learning models have over older methods, which require significant manual feature construction. Deep learning models utilize the ability of automated feature construction on raw text data.

V. Discussion

The findings demonstrate that the Convolutional Neural Network (CNN) performed exceptionally well in identifying phishing attacks, achieving the highest accuracy, recorded at 91.2%, and the highest precision of 92.1%, in the study. Among all the baseline models in the performance analysis, CNN not only excelled but also automated advanced feature extraction, thereby removing the need for costly and time-consuming manual feature engineering. Such feature extraction is crucial for identifying phishing attempts since they mainly disguise their attempts in text, and phishing detection methods have mainly overlooked this. The model's ability to automatically learn features means that it can accurately and consistently identify phishing text by detecting suspicious patterns such as abnormal text and links.

The design of the CNN has an impact on performance as well. When the embedding layer transforms tokens to dense vectors, it captures the different contexts and complexities of the texts. Convolutional layers, having 128 filters with 5 kernel sizes, identify suspicious and phishing texts. The global max pooling layer filters and retains phishing text patterns and is agnostic to the email portions in which they occur. This attribute increases the system's resilience against adversaries who conceal the malicious content in different parts of the communication. The multilayered architecture comprising the email content analysis, URL analysis, and consideration of the metadata certainly enhances the detection. This layer approach greatly reduces false positives and phishing signals on a single element. This showed the resilience to phishing surfaces that are designed to target evasively.

The model showcases numerous strengths in practical applications. With increased accuracy comes a lower false positive rate, which means that legitimate emails are less likely to be misclassified. This improves the efficiency of detecting emails and minimizes interruptions caused to people. Consequently, this increases usability for organizations focused on enhancing their email security systems.

Some of the limitations need to be mentioned, too. While the dataset is useful for research purposes, it still does not capture the complete nuance of phishing in the real world. The assessment of the model was conducted on mostly English phishing attacks that had straightforward phishing indicators, so performance results may be overly optimistic. Practical application will face more complex challenges, including emails in multiple languages, highly developed social engineering, and direct attack strategies. Furthermore, phishing practices change quickly, and it is unknown if the model can cope with such generalized new practices. For production environments to continue being effective, the model will need to be routinely updated and retrained on new data to avoid stagnation.

The proposed CNN model has been created and tested in a controlled experimental environment in a phishing detection scenario. Feasibility has been demonstrated. The model's lightweight architecture and efficient inference are conducive to integration in environments where real-time applications are required. With suitable modifications, model implementation as an email client plug-in or as a backend service is a possibility, showing real-world potential for the model in practical cybersecurity.

VI. Conclusion

This research integrated multiple strategies to identify phishing attacks through URL checks, content analysis via natural language processing, and evaluation of email metadata. In these analyses, deep learning, and in particular, CNN-based frameworks, were far superior to classical machine learning, with the techniques employed attaining an accuracy of 91.2% and a precision of 92.1%. The CNN architecture was able

to learn text patterns and feature hierarchies, and to identify the structures of phishing emails and phishing patterns. Moreover, it outperformed other deep learning structures, including LSTMs, as well as classical structures like XGBoost, Random Forest classifiers, and other deep learning models, demonstrating the applicability of deep learning in solving challenges in this particular area of cybersecurity.

This is the first framework to offer a phishing detection system that maintains a low false positive rate. That said, there are some limitations in the research. For example, the dataset does not include some types of phishing, particularly those in different languages around the world, adversarial phishing, and phishing email construction. Also, the CNN models need more research, particularly regarding the emerging phishing techniques, and the need for constant retraining directly affects the real-world applicability of the system.

Future work approaches could prioritize the use of transformer models like BERT and RoBERTa to improve the understanding and contextualization of the target phishing languages. The creation of continuous learning methods is essential for the mostly static detection systems. Detection system multilingualism, adversarial system robustness, and real-world system deployment studies will boost system applicability, robustness, and practical cybersecurity performance against sophisticated phishing email attacks.

Figure 5 presents a confusion matrix used to understand the performance of the CNN model on the test set. The model classified 3,933 phishing emails and 3,109 legitimate emails correctly, so the model performance was fairly accurate. There were, however, 452 phishing emails classified as legitimate which created the most significant problem. 337 legitimate emails were flagged as phishing which is far less serious. Overall, the model demonstrates balanced performance in identifying phishing emails while providing a good experience for the user with low false alarm rates, making the model suitable for use.

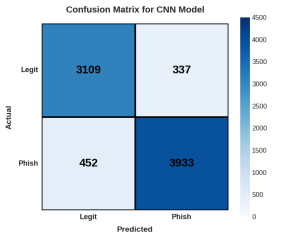

Fig. 5. CNN Confusion Metrics

REFERENCES

[1] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "A systematic literature review on phishing email detection using natural language processing techniques," *IEEE Access*, vol. 10, pp. 65 703–65 727, 2022.

[2] ——, "Phishing email detection using natural language processing techniques: a literature survey," *Procedia Computer Science*, vol. 189, pp. 19–28, 2021.

[3] I. Haider, M. A. Haider, and A. Saeed, "Big data in internet of things: Architecture and open research challenges," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, 2020, pp. 1–6.

[4] M. K. Khan, K. Nisar, Y. Farooq, S. Habib, M. Danish, and I. Hyder, "An improved banking application model using blockchain," 2021.

[5] I. Haider, M. A. Qureshi, A. Saeed, and M. A. Haider, "Enhanced model for data security in cyber space using combined steganographic and encryption techniques," in *2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEST)*, 2023, pp. 1–6.

[6] N. Q. Do, A. Selamat, O. Krejcar, E. Herrera-Viedma, and H. Fujita, "Deep learning for phishing detection: Taxonomy, current challenges and future directions," *Ieee Access*, vol. 10, pp. 36 429–36 463, 2022.

[7] B. Janet, R. J. A. Kumar *et al.*, "Malicious url detection: a comparative study," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*. IEEE, 2021, pp. 1147–1151.

[8] I. Haider, M. A. Qureshi, A. Amin, and A. Saeed, "An efficient cyber security framework for network intrusion detection using hybrid classifier," in *2023 25th International Multitopic Conference (INMIC)*, 2023, pp. 1–6.

[9] Z. Ahmad, A. Shahid Khan, K. Nisar, I. Haider, R. Hassan, M. R. Haque, S. Tarmizi, and J. J. Rodrigues, "Anomaly detection using deep neural network for iot architecture," *Applied Sciences*, vol. 11, no. 15, p. 7050, 2021.

[10] P. H. Kyaw, J. Gutierrez, and A. Ghobakhlou, "A systematic review of deep learning techniques for phishing email detection," *Electronics*, vol. 13, no. 19, p. 3823, 2024.

[11] F. Salahdine, Z. El Mrabet, and N. Kaabouch, "Phishing attacks detection a machine learning-based approach," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE, 2021, pp. 0250–0255.

[12] J. Rastenis, S. Ramanauskaitė, I. Suzdalev, K. Tunaitytė, J. Janulevičius, and A. Čenys, "Multi-language spam/phishing classification by email body text: Toward automated security incident investigation," *Electronics*, vol. 10, no. 6, p. 668, 2021.

[13] P. Bountakas and C. Xenakis, "Helphed: Hybrid ensemble learning phishing email detection," *Journal of network and computer applications*, vol. 210, p. 103545, 2023.

[14] A. Vazhayil, N. Harikrishnan, R. Vinayakumar, K. Soman, and A. Verma, "Ped-ml: Phishing email detection using classical machine learning techniques," in *Proc. 1st antiphishing shared pilot 4th acm int. workshop secur. privacy anal.(iwspa)*. Tempe, AZ, USA, 2018, pp. 1–8.

[15] S. Smadi, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "Detection of phishing emails using data mining algorithms," in *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*. IEEE, 2015, pp. 1–8.

[16] S. Magdy, Y. Abouelseoud, and M. Mikhail, "Efficient spam and phishing emails filtering based on deep learning," *Computer Networks*, vol. 206, p. 108826, 2022.

[17] H. F. Atlam and O. Oluwatimilehin, "Business email compromise phishing detection based on machine learning: A systematic literature review," *Electronics*, vol. 12, no. 1, p. 42, 2022.

[18] S. Tariq and A. Amin, "Detection of dna-protein binding using deep learning," in *2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICES&T)*. IEEE, 2023, pp. 1–4.

[19] ——, "Deepctf: transcription factor binding specificity prediction using dna sequence plus shape in an attention-based deep learning model," *Signal, Image and Video Processing*, vol. 18, no. 6, pp. 5239–5251, 2024.

[20] A. Farooq and K. H. Memon, "Kernel possibilistic fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering," *Fuzzy Sets and Systems*, vol. 477, p. 108792, 2024.

[21] A. Alhogail and A. Alsabih, "Applying machine learning and natural language processing to detect phishing email," *Computers & Security*, vol. 110, p. 102414, 2021.

[22] N. Yadav and S. P. Panda, "Feature selection for email phishing detection using machine learning," in *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021, Volume 2*. Springer, 2022, pp. 365–378.

[23] I. Ul Haq, P. Black, I. Gondal, J. Kamruzzaman, P. Watters, and A. Kayes, "Spam email categorization with nlp and using federated deep learning," in *International Conference on Advanced Data Mining and Applications*. Springer, 2022, pp. 15–27.