International Journal of Computer (IJC)
ISSN 2307-4523 (Print & Online)
https://ijcjournal.org/index.php/InternationalJournalOfComputer/index

A Method for Identifying and Assessing Phishing Attacks in Communication Messages

Modestas Krištaponis[a], Jevgenijus Toldinas[b]*

[a,b]*Kaunas University of Technology, K. Donelaičio 73, Kaunas 44249, Lithuania*
[a]*Email: modestas.kristaponis@ktu.edu*
[b]*Email: eugenijus.toldinas@ktu.lt*

Abstract

Phishing attacks have become a significant threat in online communication platforms. These attacks exploit human vulnerabilities by using deceptive messages to steal sensitive information or distribute malicious content. This paper presents a comprehensive phishing detection system, leveraging machine learning and multi-layered analysis of URLs, files, and message content. The proposed system integrates URL analysis, file analysis, and text analysis services to identify potential threats effectively. Experimental results demonstrate the efficacy of the approach, achieving high accuracy in detecting phishing attempts. This research contributes to the field of cybersecurity by providing a robust framework for identifying and mitigating phishing risks in real-time communication.

Keywords: Phishing; Communications; Chat Applications; Email; Machine Learning.

1. Introduction

Phishing attacks are disseminated through multiple channels, including email, online social networks, SMS, and blogs. Attackers establish counterfeit websites, and despite ongoing blacklisting initiatives, these sites persist in inflicting financial losses.

Received: 3/27/2025
Accepted: 5/10/2025
Published: 5/20/2025

---

* Corresponding author.

In Q3 of 2024, there were 932,923 phishing assaults recorded by the Anti-Phishing Working Group (APWG), as noted social media platforms remained the most targeted sector, accounting for 30.5% of all phishing attacks [1]. Various types of phishing attacks exist, with the taxonomy presented in [2]:

- Email Phishing – this is a widely recognized form of phishing assault in which perpetrators mimic real brands in their emails
- HTTPS Phishing – is frequently seen as a safe link to click due to its use of encryption to bolster security.
- Spear Phishing – attackers target victims with individualized messages that appear to be from a trustworthy source.
- Vishing – voice phishing is a deceptive activity in which cybercriminals make phone calls to generate a sense of urgency in the victim to execute potentially harmful acts.
- Smishing – by making it seem like time is running out, attackers use text messages to get people to share private information or click on harmful links.
- Angler Phishing – social media users are the focus of the attackers, who use notifications or direct messages on social media apps to fool users into clicking on a malicious link or disclosing their login information.
- Pharming – is a form of phishing attack in which internet traffic is diverted to a counterfeit website by manipulating the victim's computer or DNS settings through malicious code or malware.
- Pop-up Phishing – to fool users into divulging personal information or putting malicious software onto their devices, cybercriminals fabricate pop-up windows that mimic authentic prompts from reputable websites or applications.
- Clone Phishing – involves making a fake website or email that looks like the real one by copying it and giving it the same logo and style.

In recent years, numerous researchers have focused on deep learning (DL) and machine learning (ML) techniques for phishing detection to rectify the limitations of conventional methods and enhance detection accuracy [3]. Most researchers retrieved and selected features from the email body text, whilst other researchers utilized features from the header, subject, body, URL, attachments, structure, and attachment files to train the model. In this research, we present a method for evaluating and calculating the total phishing probability score by taking into account the attachment, URL, and message text. As a result, an automatic system for phishing attack detection is developed. The rest of the paper is organized as follows. Section 2 discusses the related works. Section 3 presents and explains the methodology. Section 4 presents and discusses the experimental results. Concluding remarks are formulated in Section 5.

2. Related Work

Personal and professional contact, as well as the exchange of information, have been substantially facilitated by email. Due to their continued use and dependence on email, consumers are more vulnerable to cybersecurity threats like spam, phishing attacks, and other forms of exploitation. According to the findings of a virus investigation, it was found that 94% of malware was spread by email, and machine learning (ML) methods have been employed in the detection of email spam as a result of their exceptional classification performance in identifying email spam [4]. A major threat comes from phishing attacks, which attempt to deceive victims into

clicking on malicious URLs. The use of ML algorithms to examine several elements of incoming URLs is a common component of multimodal strategies that aim to detect and prevent these types of attacks [5].

A web-based application was created by the authors in [6] to differentiate between legitimate and fraudulent emails by utilizing insights obtained from recent phishing trends. The LIME technique for explainable artificial intelligence was applied by the authors to provide insights into the decision-making process that had been implemented by the model. Through the utilization of LIME visualizations, an analysis was conducted to determine which characteristics (words) contributed the most to the classification of an email as spam or not spam. An investigation [7] was conducted into the potential security risks of phishing email attacks, comparing the performance of three primary classification algorithms: random forest, SVM, and a combination of k-fold cross-validation with the XGBoost model. The creation of email security solutions that are more efficient and dependable is where the practical applications of the proposed research lie.

The importance of feature selection in machine learning-based phishing detection cannot be emphasized, as suggested in the article [8]. For the purpose of facilitating robust trials and assembling important parts for successful phishing detection, the study that was completed was motivated by the goal of detecting critical qualities, which were gained from a comprehensive assessment of the available literature. Experiments that were provided included a wide variety of feature selection procedures, culminating in the utilization of three techniques that proved to be quite effective: correlation feature selection, RFE feature selection, and UFS feature selection.

It is possible to gain an understanding of the structural distinctions that exist between genuine and phishing websites through the use of the visual representation techniques that have been proposed [9]. The system appears to have the potential to detect phishing attackers in a short amount of time while maintaining a high level of accuracy, as indicated by the preliminary trial results that were achieved. Moreover, the proposed strategy enhances its efficiency by gaining knowledge from the incorrect classifications that it has made.

Based on the reasoning skills of large language models (LLMs), a new approach to document vectorization called prompted contextual document vectors was suggested [10]. Other document classification jobs could benefit from it as well, particularly when faced with adversarial situations like spear-phishing detection using the suggested strategy. A novel dataset of high-quality spear-phishing emails produced by an LLM-powered system was introduced and published by the authors. Introduced in [11], chat spam detector is a system that can identify phishing emails using LLMs. The suggested approach accurately determines if an email is phishing by transforming email data into a format appropriate for LLM analysis. To help users make educated judgments about how to deal with questionable emails, it provides extensive reasoning for its phishing determinations.

After reviewing the most recent research on the topic, the authors of [12] compared Naive Bayes to various ML and DL state-of-the-art (SOTA) algorithms for phishing detection tasks. Naive Bayes's capabilities in detecting phishing attempts in comparison to other DL and machine learning-based SOTA methods were examined and evaluated. A comparative analysis of Naive Bayes performance against other ML and DL SOTA algorithms for phishing detection, based on a review of recent research publications, reveals that Random Forest, Decision

Tree, CNN, and XGBoost achieved individual mean accuracies of 97.1%, 95.2%, 94.2%, and 94.1%, respectively, ranking as the top four performers in URL properties-based phishing detection tasks. Conversely, Naive Bayes, SVM, and RNN exhibited individual mean accuracies of 80.4%, 89.4%, and 91.6%, respectively, representing the lowest three performances in the same classification task.

An innovative method for identifying phishing emails utilizing big language models, particularly emphasizing the functionalities of ChatGPT described in [13]. The authors examine the customization and deployment of modern AI algorithms to enhance the precision and efficiency of phishing email detection. The research delineates the advantages and drawbacks of GPT-3.5, GPT-4, and customized ChatGPT, highlighting their distinct appropriateness for actual applications in email security. The findings of this study highlight the significance of implementing sophisticated LLMs for specific security applications as a means of bolstering defenses against the unrelenting stream of phishing attempts. A novel design utilizing a synthetic minority over-sampling technique (SMOTE) for data preparation aimed at recognizing phishing attacks is presented in [14]. This new approach to data imbalance ensures the framework's resistance to various types of phishing behavior scenarios. The study presents a model incorporating ResNeXt within a Gated Recurrent Unit (RNT) framework, aimed at facilitating real-time detection of phishing attacks. This lightweight strategy aims to significantly reduce processing time, thereby enhancing the effectiveness of the defense system. The proposed methodology, situated within the wider framework of digital forensics, signifies significant progression in combating phishing attempts, resulting in enhanced security and operational efficiency.

The survey [15] introduces a new taxonomy of phishing that categorizes the approach, its intended targets, and the mediums of circulation. Without being overly complicated, the proposed taxonomy addresses all facets of phishing attempts. No other article that the authors could find attempted to provide a more comprehensive explanation of the many subtypes of phishing circulation channels than this one. The primary objective of this survey was to categorize phishing attacks and examine attack methodologies, while also evaluating phishing countermeasures and the numerous detection methods offered by researchers. A deep learning framework for the identification of phishing emails is conducted through the use of word embedding methods, as proposed in [16]. It has been introduced that a new model called DeepEPhishNet has been developed for the purpose of classifying phishing emails. This model combines deep learning and word embedding techniques. A neural network with two hidden layers was offered as the final model proposed in [17]. It was able to obtain good results in detecting mass disseminated phishing emails (about 88%) without flagging a significant number of customer accounts (less than approximately 5 percent). The false positive rate should be significantly lower than the false negative rate since each potential positive requires administrative involvement to block the user account if it is required. It is essential that the false positive rate be far lower than the false negative rate. It is important to take note that this trade-off may be readily modified at any moment because the neural network generates a probability. The threshold can be changed to something other than 0.5 if that is what the user desires. An analysis of the similarities and differences between deep learning models for sophisticated phishing email detection is presented in [18]. Not only was the possibility of employing deep learning for the detection of email phishing researched but also the problem of deep learning model complexity in comparison to performance was investigated. A convolutional neural network (CNN) in conjunction with a bi-directional gated recurrent unit (Bi-GRU) looks to be an effective strategy for spotting phishing emails, according to the results

that were obtained. This is the conclusion that can be drawn from the findings. With the assistance of Bi-GRU, it is possible to accomplish the goal of minimizing time while preserving performance. The findings of the research presented in [19] indicate that a phishing email detection technique that makes use of big-data technologies can be utilized to generate a large-scale phishing email dataset, detect phishing emails, visualize additional attributes, and recognize phishing emails as quickly as feasible. It stands to reason that the utilization of big data AI could be one of several new options, given the steady advancement of artificial intelligence. Cross-referencing networks and network slowness with big data might be a technique to discover compromised and controlled devices much sooner, as well as provide more insight on how they operate and spread. Going further, designers could potentially incorporate more than just email datasets. This would be a feasible option.

3. Methodology and proposed method

The proposed method is designed as a multi-layered architecture (see Figure 1) that integrates three main services: URL analysis, file analysis, and text analysis. Each service operates independently but communicates through a message queue to ensure seamless data flow.

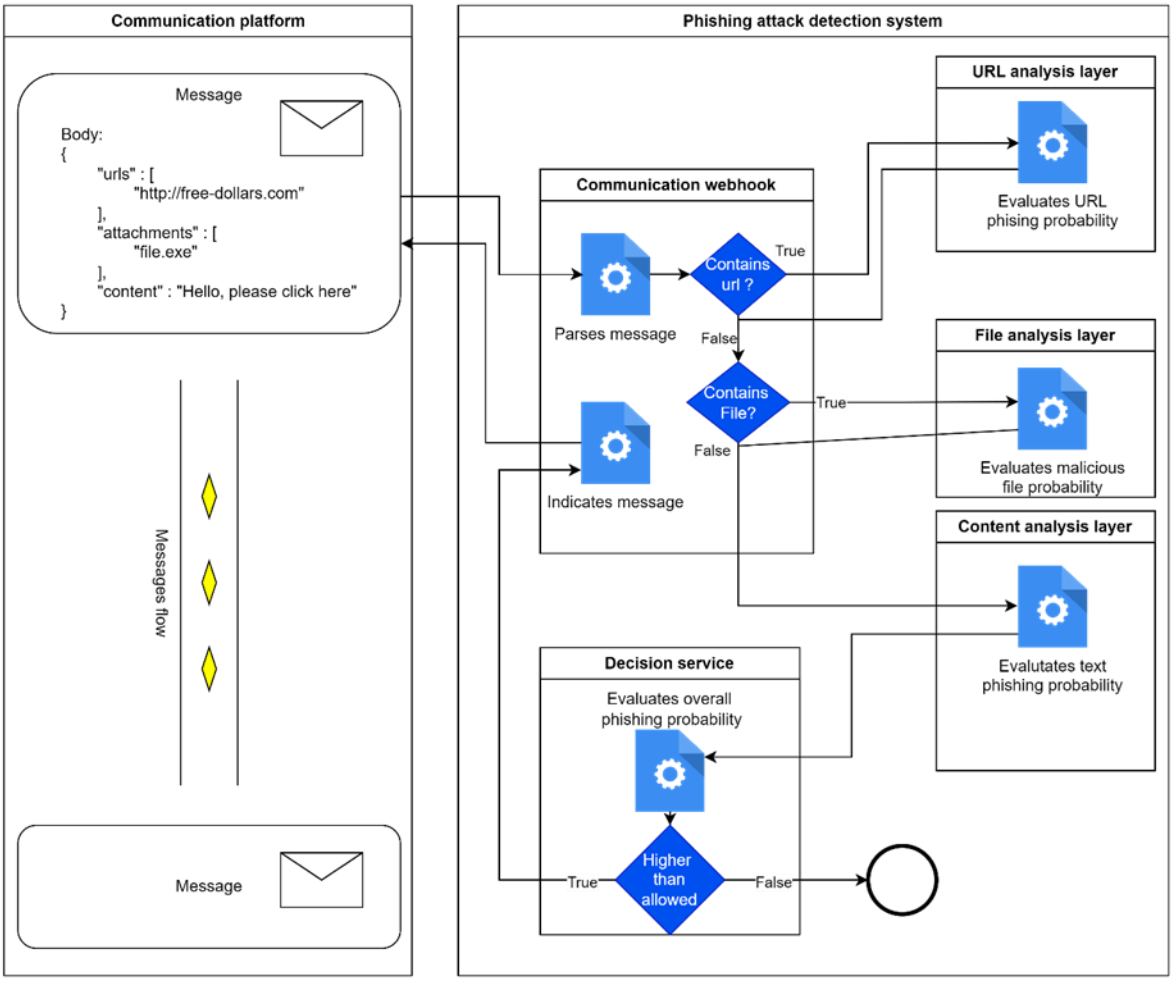

Figure 1: The proposed method is designed as a multi-layered architecture

The data collection process involves monitoring messages using a bot implementation via the API. The bot gathers key information from each message, including content, embedded URLs, file attachments, user information, timestamp, and channel context. These raw data form the foundation of our phishing detection system.

The URL analysis service is implemented as an independent microservice that receives URLs from a RabbitMQ message queue. The service performs several steps:

- URL Parsing and Validation: Extract constituent parts of the URL using *urllib.parse*.
- Domain Analysis: Checks domain length, subdomains, and SSL certificate validity.
- Pattern Matching: Identifies suspicious patterns using regular expressions.
- Reputation Check: Uses third-party services like Google Safe Browsing API to verify domain reputation.

The file analysis service processes file attachments from a dedicated RabbitMQ queue. The service performs:

- File Type Identification: Determines the file type using python-magic.
- Metadata Extraction: Extracts file name, size, and SHA-256 hash.
- Risk Factor Analysis: Checks for suspicious file extensions, small executable files, and mismatched MIME types.
- Integration with VirusTotal: Queries the VirusTotal database to check for known malicious files.

The text analysis service uses machine learning models to identify potential risks in message content. The service supports both Random Forest and Transformer models:

- Model Prediction: Generates a probability score indicating the likelihood of phishing.
- Text Matching Against Static Rules: Identifies known phishing phrases, URL patterns, and financial indicators.
- Contextual Analysis: Evaluates urgency indicators, coercive language, and promises of rewards.

4. Experimental Settings and Results

In this section, experimental settings and results are presented.

4.1. The dataset and model parameters

The dataset for model training was collected from publicly available sources on Kaggle: Email Spam or Not (Classification) [20], SMS Spam Detection Dataset [21], SMS Spam Collection Dataset [22] and Spam SMS Classification Using NLP [23]. The data aggregation class combines multiple datasets, because current used datasets are not in the same format. For that python script created, using *pandas* library to standardize and merge datasets into one dataset. Created configuration for each different dataset and provided rules on how they must be standardized. Standardized format ensures consistency and reducing redundancy. The main parameters of the used datasets are presented in Table 1. Text normalization involves removing whitespace, truncating long

messages, and standardizing labels.

Table 1: Selected datasets for the experiment

Dataset	File name	Year	Classes
Email Spam or Not	spam_dataset.csv	2024	Spam / Ham
SMS Spam Detection Dataset	spam_sms.csv	2024	Spam / Ham
SMS Spam Collection Dataset	spam.csv	2017	Spam / Ham
Spam SMS Classification Using NLP	Spam_SMS.csv	2024	Spam / Ham

The number of classes and records in the aggregated dataset for the experiment is presented in Table 2.

Table 2: The aggregated dataset for the experiment

Classes	Number of records in final_training_datset.csv	Number of records in final_test_datset.csv	Number of records in final_validate_datset.csv
Ham	3838	822	823
Spam	620	133	133

The Accuracy, precision, recall, and F1 score are the primary qualities that have typically been utilized in the process of evaluating computer-based learning techniques. To determine the level of accuracy, the following equation is utilized:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN'} \quad (1)$$

where *TP* is a number of true positive predictions, *TN* is a number of true negative predictions, *FP* is a number of false positive predictions, and *FN* is a number of false negative predictions. Specifically, precision refers to the proportion of cases that are tagged as positive and are, in fact, positive.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

The percentage of true positives that are expected to be positive is referred to as recall, and it is the fraction of genuine positive predictions that are designated as positive overall.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

In terms of precision and recall, the F1 score is the harmonic mean of the two.

$$F1\ score = \frac{2TP}{2TP + FP + FN} \quad (4)$$

4.2. Experimental result

The Random Forest classifier is trained using TF-IDF features extracted from the text content. The model is configured with parameters optimized for phishing detection:

- n_estimators: 200 trees.
- max_depth: 20.
- min_samples_split: 5.
- min_samples_leaf: 2.
- random_state: 42.

As shown in Figure 2, The performance of the model during training is compared across all epochs, during the training process, the performance of the model is evaluated and compared over all epochs.

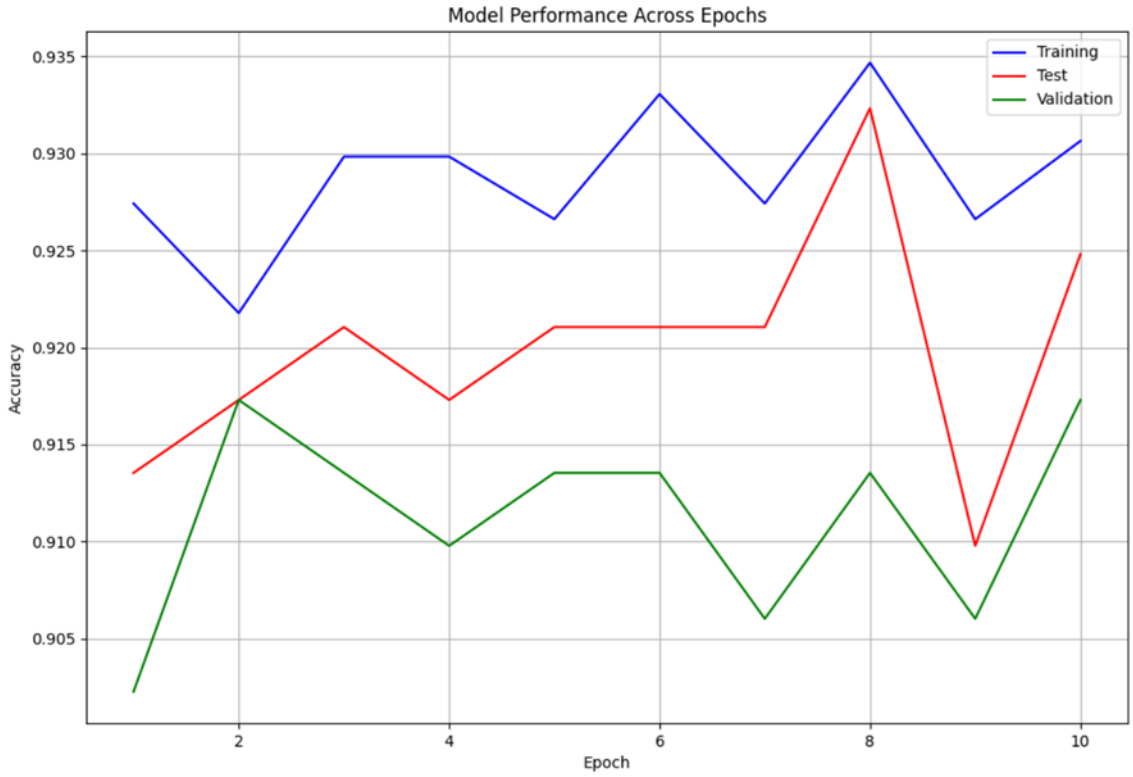

Figure 2: The performance of the model during training is compared across all epochs.

An investigation and evaluation are being conducted to see how effectively the trained model is presented in Table 3.

Table 3: Training, validation, and test accuracy across epochs

Epoch	Training accuracy	Validation accuracy	Test accuracy
1	0.9274	0.9023	0.9135
2	0.9218	**0.9173**	0.9173
3	0.9298	0.9135	0.9211
4	0.9298	0.9098	0.9173
5	0.9266	0.9135	0.9211
6	0.9331	0.9135	0.9211
7	0.9274	0.906	0.9211
8	**0.9347**	0.9135	**0.9323**
9	0.9266	0.906	0.9098
10	0.9306	**0.9173**	0.9248

At the eighth epoch, the best training accuracy was reached, which was 0.9347. At the second and tenth epoch, the best validation accuracy was reached, which was 0.9173. At the eighth epoch, the best test accuracy was reached, which was 0.9323. The precision, recall, and F1 score across all epochs are presented in Table 4.

Table 4: Precision, Recall, F1-score across epochs

Epoch	Training Precision	Training Recall	Training F1-score
1	0.9241	0.9135	0.913
2	0.9269	0.9173	0.9168
3	0.9298	0.9211	0.9206
4	0.9269	0.9173	0.9168
5	0.9298	0.9211	0.9206
6	0.9298	0.9211	0.9206
7	0.9298	0.9211	0.9206
8	**0.9387**	0.9323	**0.9321**
9	0.9213	0.9098	0.9092
10	0.9327	**0.9248**	0.9245

At the eighth epoch, the best precision was reached, which was 0.9387. At the tenth epoch, the best recall was reached, which was 0.9248. At the eighth epoch, the best F1-score was reached, which was 0.9321. Validation of a model is the process of determining whether or not the model is successful in accomplishing the goals that it was designed to accomplish. In the majority of cases, this will include establishing that the model is predictive under the circumstances that are required for its intended application. It is determined that the model with the highest validation accuracy is the one that performs the best, and it achieves the highest validation accuracy of 0.9173. The model testing result presented using confusion matrices that are depicted in Figure 3.

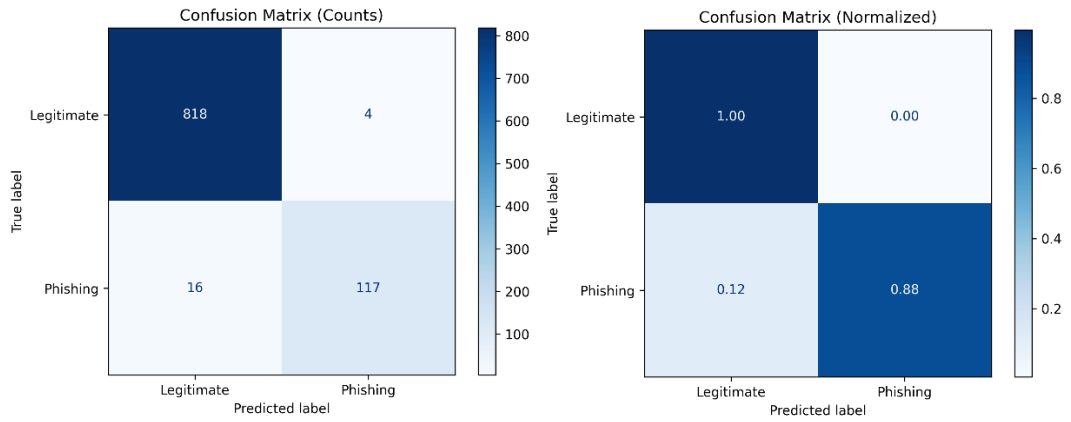

Figure 3: The confusion matrices that are generated during the model validation process.

According to the confusion matrix, the rows represent the actual class, while the columns indicate the class that is anticipated to be present. On the other hand, off-diagonal cells indicate observations that have been incorrectly classified, whereas diagonal cells represent observations that have been accurately classified. A row-normalized summary provides a presentation of the percentages of observations that were correctly recognized and those that were incorrectly identified for each real class. A column-normalized summary provides a presentation of the percentages of observations that were correctly recognized and those that were incorrectly identified for each projected class. It is evident from the data presented in Figure 3 that 88 percent of all true phishing messages belong to this class.

5. Discussion

Phishing attacks sometimes include embedded elements such as links, photos, and videos to mislead visitors into engaging with harmful content. Malefactors frequently utilize compromised domains to render their phishing websites seemingly authentic. Compromised domains are authentic domains that have been seized by attackers to facilitate phishing websites. Identifying these compromised domains can be difficult as they may seem authentic, and conventional blacklisting methods may prove ineffective. This is why the proposed method's first layer is an independent microservice called URL analysis service. It takes in URLs from a RabbitMQ queue, parses them for their constituent parts, checks the validity of the domain, subdomains, and SSL certificate, looks for suspicious patterns using regular expressions, and uses services like Google Safe Browsing API to verify the reputation of the domain.

Attachments are common in spam emails; as a result of this, the second layer of the proposed method performs the following functions: It determines the file type by utilizing Python-magic; it extracts the file name, size, and SHA-256 hash; it checks for suspicious file extensions, small executable files, and mismatched MIME types; and it queries the VirusTotal database to check for known malicious files.

The advantage of the proposed method is the multi-layered phishing detection system that combines data collection, URL analysis, file analysis, content analysis, and ML techniques to identify malicious activities. The

limitation of the proposed method is based on using datasets that are not in the regional Lithuanian language. As a direction for feature study, artificial intelligence explainability is a developing research issue. There is continuing effort to investigate the explainability of deep learning algorithms in the context of phishing detection, which is a topic that is worthy of exploration. It is essential to research methods for expanding datasets, such as text augmentation methods in regional languages, because there are not enough genuine phishing datasets. One example of this is a distinct Lithuanian language that exhibits its own grammatical and lexical exclusivity.

6. Conclusion

To address the shortcomings of traditional approaches and improve the accuracy of detection, researchers have turned their attention to DL and ML methods for phishing detection in the past few years. Researchers trained their models using features extracted from various parts of emails; some focused on the body text alone, while others looked at the header, subject, body, URL, attachments, structure, and attachment files.

Phishing via email is a well-known type of phishing attack in which the attackers imitate actual brands in their emails. When it comes to phishing, https phishing is generally considered to be a secure link to click on because it employs encryption to strengthen security.

This paper presents a multi-layered phishing detection system that combines data collection, URL analysis, file analysis, content analysis, and ML techniques to identify malicious activities. Experimental results demonstrate the system's effectiveness in achieving high accuracy and reliability in detecting phishing attempts. The proposed approach contributes to the advancement of cybersecurity measures by providing a comprehensive solution for real-time threat detection.

References

[1] APWG, Phishing Activity Trends Reports, 2024. Internet: https://apwg.org/trendsreports/ [Apr. 28, 2025]

[2] T. Singh, M. Kumar, S. Kumar. “Walkthrough phishing detection techniques”. *Computers and Electrical Engineering* 118 (2024). https://10.1016/j.compeleceng.2024.109374.

[3] P.H. Kyaw, J. Gutierrez, A. Ghobakhlou. “A Systematic Review of Deep Learning Techniques for Phishing Email Detection”. *Electronics* 13(19):3823 (2024). https://10.3390/electronics13193823.

[4] E. H. Tusher, M. A. Ismail, M. A. Rahman, A. H. Alenezi and M. Uddin. “Email Spam: A Comprehensive Review of Optimize Detection Methods, Challenges, and Open Research Problems”. *IEEE Access* (2024) vol. 12, pp. 143627-143657, https://10.1109/ACCESS.2024.3467996.

[5] A. Gunjan and R. Prasad. “Phishing Email Detection Using Machine Learning: A Critical Review” in: *Proceedings of the 2024 IEEE International Conference on Computing, Power and Communication*

Technologies (IC2PCT), Greater Noida, India, 2024, pp. 1176-1180, https://10.1109/IC2PCT60090.2024.10486341.

[6] A. Al-Subaiey, M. Al-Thani, N. A. Alam, K. F. Antora, A. Khandakar, SM A. U. Zaman. "Novel interpretable and robust web-based AI platform for phishing email detection". *Computers and Electrical Engineering*, Vol 120, 2024, https://doi.org/10.1016/j.compeleceng.2024.109625.

[7] D. Sugianto, R. A. Putawa, C. Izumi, and S. A. Ghaffar. "Uncovering the Efficiency of Phishing Detection: An In-depth Comparative Examination of Classification Algorithms". *Int. J. Appl. Inf. Manag.*, vol. 4, no. 1, pp. 22–29, Apr. 2024. https://doi.org/10.47738/ijaim.v4i1.72.

[8] J. Tanimu, S. Shiaeles, and M. Adda. "A Comparative Analysis of Feature Eliminator Methods to Improve Machine Learning Phishing Detection". *JDSIS*, vol. 2, no. 2, pp. 87–99, Dec. 2023, https://10.47852/bonviewJDSIS32021736.

[9] A. O. Taofeek. "Development of a Novel Approach to Phishing Detection Using Machine Learning". *ATBU Journal of Science, Technology and Education*, vol. 12, n. 2, p. 336-351, jun. 2024. URL: https://www.atbuftejoste.com.ng/index.php/joste/article/view/2107.

[10] D. Nahmias, G. Engelberg, D. Klein, A. Shabtai. "Prompted Contextual Vectors for Spear-Phishing Detection". *Computer Science, Machine Learning*, 2024, https://doi.org/10.48550/arXiv.2402.08309.

[11] T. Koide, N. Fukushi, H. Nakano, and D. Chiba. "ChatSpamDetector: Leveraging Large Language Models for Effective Phishing Email Detection". In: *20th EAI International Conference on Security and Privacy in Communication Networks* (SecureComm 2024), October 28–30, 2024, Dubai, United Arab Emirates. https://doi.org/10.48550/arXiv.2402.18093.

[12] T. Ige, C. Kiekintveld, A. Piplai, A. Waggler, O. Kolade, B. H. Matti. "An investigation into the performances of the Current state-of-the-art Naive Bayes, Non-Bayesian and Deep Learning Based Classifier for Phishing Detection: A Survey". *Computer Science, Cryptography and Security*, https://doi.org/10.48550/arXiv.2411.16751.

[13] R. Chataut, P. K. Gyawali and Y. Usman. "Can AI Keep You Safe? A Study of Large Language Models for Phishing Detection". *2024 IEEE 14th Annual Computing and Communication Workshop and Conference* (CCWC), Las Vegas, NV, USA, 2024, pp. 0548-0554, https://10.1109/CCWC60891.2024.10427626.

[14] F. S. Alsubaei, A. A. Almazroi and N. Ayub. "Enhancing Phishing Detection: A Novel Hybrid Deep Learning Framework for Cybercrime Forensics". *IEEE Access*, vol. 12, pp. 8373-8389, 2024, https://10.1109/ACCESS.2024.3351946.

[15] R. Goenka, M. Chawla, N. Tiwari. "A comprehensive survey of phishing: mediums, intended targets,

attack and defence techniques and a novel taxonomy". *Int. J. Inf. Secur.* 23, 819–848 (2024). https://doi.org/10.1007/s10207-023-00768-x.

[16] M. Somesha, A.R. Pais. "DeepEPhishNet: a deep learning framework for email phishing detection using word embedding algorithms". *Sādhanā* 49, 212 (2024), https://doi.org/10.1007/s12046-024-02538-4.

[17] A. Bezerra, I. Pereira, M.Â. Rebelo. "A case study on phishing detection with a machine learning net". *Int J Data Sci Anal* (2024). https://doi.org/10.1007/s41060-024-00579-w.

[18] N. Altwaijry, I. Al-Turaiki, R. Alotaibi, F. Alakeel. "Advancing Phishing Email Detection: A Comparative Study of Deep Learning Models". *Sensors* 2024, 24, 2077. https://doi.org/10.3390/s24072077.

[19] S. R. Bauskar, C. R. Madhavaram, E. P. Galla, J. R. Sunkara, H. K. Gollangi. "AI-Driven Phishing Email Detection: Leveraging Big Data Analytics for Enhanced Cybersecurity". *Library Progress International*, 2024, 44(3), 7211-7224.

[20] Email Spam or Not (Classification), version 1, 2024. Internet: https://www.kaggle.com/datasets/devildyno/email-spam-or-not-classification [Apr. 28, 2025]

[21] SMS Spam Detection Dataset, version 1, 2024. Internet: https://www.kaggle.com/datasets/vishakhdapat/sms-spam-detection-dataset [Apr. 28, 2025]

[22] SMS Spam Collection Dataset, version 1, 2016. URL: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset/data [Apr. 28, 2025]

[23] Spam SMS Classification Using NLP, version 1, 2024. Internet: https://www.kaggle.com/datasets/mariumfaheem666/spam-sms-classification-using-nlp [Apr. 28, 2025]