

AI-Enhanced Phishing Detection: Leveraging Advanced Transformer-Based Representations with GAN for Robust Security

Yu-Hung CHEN, Jiann-Liang CHEN, Shih-Ping CHIU

Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan

D10907005@mail.ntust.edu.tw, lchen@gmail.com, spchiu@mail.ntust.edu.tw

Abstract— Phishing is a sophisticated cybercrime technique where attackers trick users into interacting with fraudulent websites, leading to the disclosure of sensitive information. With the increasing frequency and sophistication of phishing attacks targeting both individuals and enterprises, traditional detection methods have become inadequate. Phishing websites often employ content obfuscation and deliberate misspellings to bypass detection systems, making these threats harder to identify. This study proposed an enhanced Transformer-based architecture designed to automatically extract and represent key phishing indicators. The architecture has been improved by modifying the input positional encoding and expanding the analysis beyond the typical textual content of HTML to include comments and attribute data, allowing for a more comprehensive understanding of phishing signals. These indicators are integrated with expert-crafted features to train an enhanced phishing detection model. To further improve the model's generalization to diverse phishing variants, Generative Adversarial Networks (GANs) are employed to generate synthetic phishing sample for more comprehensive training. Extensive evaluations of both model performance and efficiency demonstrate that combining features with extracted HTML content significantly outperforms traditional content-based methods. Moreover, the incorporation of synthetic data into the training process elevated detection accuracy from 87.2% to 93.7%, highlighting the potential of deep learning techniques to bolster phishing detection capabilities in dynamic environments.

Keywords— Phishing Attack, Cybersecurity, Data Imbalance, Generative Adversarial Networks, Transformer, Machine Learning, Deep Learning

I. INTRODUCTION

As digital technologies advance, online interactions grow in scale and complexity, heightening cyber threat risks. The expansion of cloud services and IoT devices increases the attack surface, involving sensitive transactions like online banking and shopping. The transmission of Personal Identifiable Information (PII)—such as account details and passwords—over the Internet risks interception by malicious actors aiming to commit financial crimes or cyber-attacks. [1]-[2]. Phishing remains a prevalent threat in this evolving digital landscape, exploiting both social engineering and technology to unlawfully acquire personal or financial credentials. Phishing attacks often involve deceptive websites that mimic

legitimate entities, such as financial institutions or reputable organizations. Victims are typically tricked into clicking on malicious links that lead to counterfeit sites designed to harvest sensitive information like login credentials and account numbers. Such attacks can extend beyond individual targets, impacting entire organizations by leaking critical information and enabling further internal threats [3].

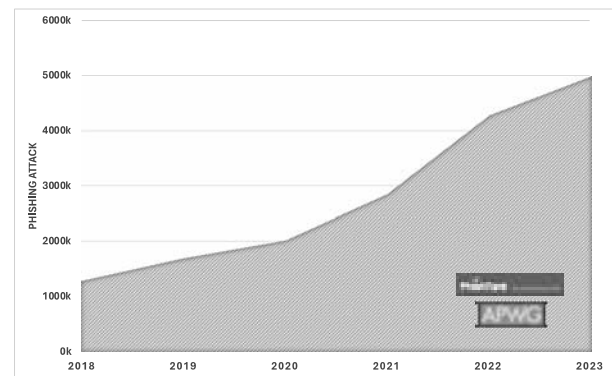


Figure 1. Phishing Attacks from 2018 to 2023

According to analysis by APWG and PhishTank, phishing attacks surged to 5 million incidents in 2023, reflecting a 16.4% increase over 2022 [4], as shown in Figure 1.

Microsoft continues to be the most frequently imitated brand in phishing attacks, with 43.1% of all phishing attempts specifically targeting it. Cybercriminals often exploit the widespread trust and familiarity associated with Microsoft by creating deceptive websites and emails that mimic official communications. The prominence of Microsoft as a target reflects its large global user base, making it an attractive focus for attackers. As shown in Figure 2, Microsoft leads significantly in phishing attempts compared to other major brands.

The rapid expansion of internet usage and online services has intensified the threat of phishing attacks, driving the development of diverse detection strategies. These strategies can be categorized into three primary approaches: blacklisting, visual similarity [5]-[6], and heuristic methods [7]-[10].

Blacklisting is a common method for detecting phishing attacks by comparing URLs to previously reported phishing

links. It relies on a database of approved and unapproved URLs, requiring regular updates to remain effective against phishing threats. Visual-based methods detect phishing by comparing web page appearance, including layout and logos, to legitimate sites. Although effective at identifying well-crafted phishing pages, these approaches are resource-intensive and can be bypassed with slight design changes made by attackers.

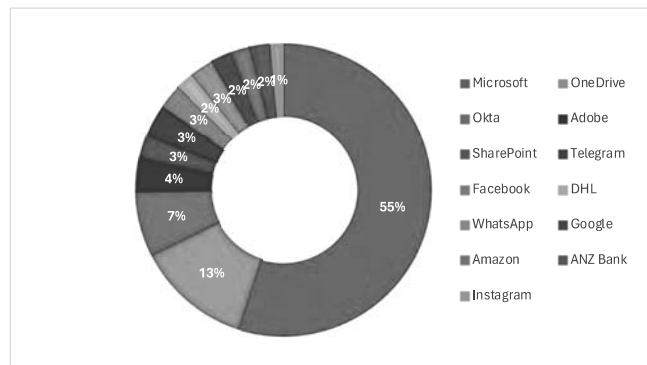


Figure 2. Phishing Attack Statistics By Branding

As artificial intelligence (AI) evolves, its role in cybersecurity, particularly phishing detection, has become vital. Traditional methods often fail to keep up with the rapid sophistication of phishing attacks. In contrast, AI-driven approaches utilize advanced algorithms for more accurate and efficient threat identification. Recent AI advancements have led to the development of models that surpass traditional systems by addressing the dynamic nature of phishing websites. A significant challenge is effectively representing HTML content, as phishing tactics continue to evolve and obfuscate key information. This study further enhanced AI@ntiPhish1.0 PLUS by integrating transformer architecture to better capture features within HTML content. It also employs GANs to generate a broader range of phishing samples, thereby improving feature effectiveness and increasing model diversity. The combined features from these advancements are then integrated with those of AI@ntiPhish1.0 PLUS to train a phishing detection model with enhanced accuracy and stability.

II. RELATED WORK

Phishing has been a persistent cybersecurity issue since the early days of the internet, with attackers continuously evolving tactics to steal sensitive information. To combat these threats, a variety of detection methods have been developed, ranging from traditional blacklisting to more advanced image-based analysis.

Blacklisting relies on maintaining databases of reported phishing URLs. While this approach is integrated into browsers like Google Chrome and Firefox, it is limited by its dependence on user reports, often failing to detect new or unreported phishing attempts, especially zero-day attacks.

Visual-based solution has gained prominence in recent years by analyzing visual elements of web pages. Traditional

image processing techniques like SIFT, SURF, and HOG have been used to extract features, while deep learning models, including CNNs and LSTMs, have further improved detection accuracy. These AI-driven methods excel at identifying patterns and subtle visual cues that may evade conventional techniques, offering a more robust solution.

Niu et al. proposed a computer vision-based framework for phishing identification [5]. Their method first captures screenshots of suspicious webpages and processes them through a dual-path feature aggregation module to extract multi-scale features. The logos are then processed using a recognition network that enhances their feature vectors with dense feature aggregation and self-attention mechanisms. This framework achieves state-of-the-art performance, showing a precision of 97.7% and 95.0% recall on a mixed dataset of phishing and benign webpages. Lindamulage et al. developed a new phishing detection method using VisionGNN, a Graph Neural Network (GNN) architecture [6]. Trained on the VisualPhish dataset with image augmentations and optimization techniques, the VisionGNN model achieved 97% accuracy in identifying phishing sites. This innovative application of VisionGNN highlights its potential for improving cybersecurity and sets the stage for future research in real-world applications.

Heuristic-based solution involves extracting features from both legitimate and phishing sites. These features are then used with AI algorithms to quantify and train the detection model. By learning from these features, the model's ability to recognize phishing attempts is significantly improved, enhancing overall detection accuracy and effectiveness.

Kawale et al. proposed a phishing detection system using supervised learning [7]. It evaluates several machine learning models—Random Forest, Support Vector Classifier (SVC), and others. Random Forest was selected for its highest accuracy of 96.7%. Asani et al. [8] proposed a novel hybrid method for phishing detection that combines the Gaussian Latent Variable Model with the SVC. The GLVM enhances data understanding and detection efficiency, while the SVC provides robust classification capabilities. The proposed approach was tested on a comprehensive dataset of legitimate and phishing emails, achieving impressive performance metrics: 98.43% accuracy, 99.28% precision, and 99.16% recall. This method effectively reduces FPs and FNs, advancing phishing email detection and improving cybersecurity. Al-Yozbaky et al. [9] introduced a novel model for detecting phishing in Arabic through the application of neural language programming. The model analyzes Arabic email content using three key determinants: a blacklist of common phishing words, their roots, and a list of common phishing sentences. When all three conditions are used together, the model achieves 99% accuracy for legitimate emails and 96% for phishing detection. Utilizing only the blacklist of phishing words results in 99% accuracy for legitimate emails and 94% for phishing emails. This approach significantly enhances phishing email detection and lays the groundwork for further advancements in this field. Chen et al. [10] proposed an enhanced phishing kit detection by

integrating HTML structure embedding with the AI@ntiPhish1.0 learning framework, resulting in a solution known as AI@ntiPhish1.0 PLUS. This novel architecture utilizes deep learning to improve phishing detection by analyzing HTML structures. By combining HTML structural features with those from AI@ntiPhish1.0, the approach addresses the increasing sophistication of phishing attacks and the accessibility of phishing toolkits. Experimental results indicate that this integration boosts detection accuracy from 82% to 87.2%, marking a significant advancement in phishing kit identification and enhancing overall phishing prevention measures. In recent years, the Transformer architecture [11] has garnered significant attention for its ability to generate representations through self-attention mechanisms. The Transformer's attention mechanism allows it to capture intricate dependencies within data, leading to substantial advancements in these applications.

While heuristic analysis can improve phishing detection, its accuracy is often dependent on the volume of data and the quality of the selected features. As the number of phishing website increases, the model can learn a broader range of features, enhancing its detection capability. However, when the training data is insufficient, the model struggles to gather enough information for accurate judgments, leading to higher error rates. Traditional feature development usually relies on manual exploration, and as phishing websites become more sophisticated in avoiding detection, they often employ techniques such as misspelling or obfuscation. For example, certain intentions or keywords present in HTML content might indicate phishing risk, but due to character alterations, such as non-ASCII conversions or replacing "0" with "o", these features become harder to detect. Such variations can significantly affect the model's accuracy, even if the underlying algorithm is robust. To effectively represent the features of HTML content and keep pace with the rapid evolution of phishing websites, more generalized and resilient approaches are needed. Different Transformer architectures have demonstrated unique strengths in learning the representation of phishing HTML content. These models can automatically extract representative features, allowing for improved generalization. This study experimented with various Transformer architectures to evaluate their effectiveness in representing phishing HTML content and integrate the extracted features with the AI@ntiPhish 1.0 PLUS to further enhance detection accuracy and defense capabilities against phishing attacks.

III. PROPOSED ARCHITECTURE

The intelligent learning framework proposed in this study integrates a GAN. It combines feature vectors derived from the HTML representation module, specifically the extracted HTML structure and content features, to enhance phishing website detection. Figure 3 shows the proposed learning architecture.

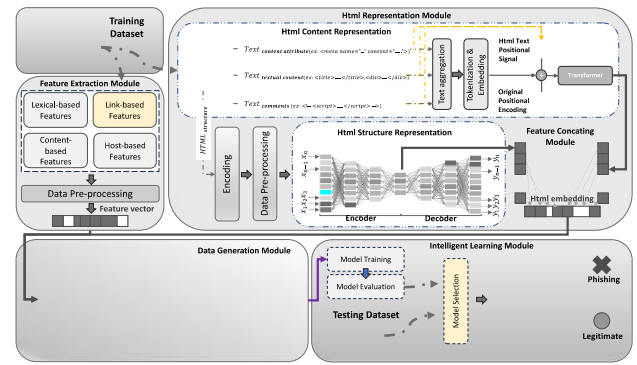


Figure 3. Learning Architecture

A. Data Collection

In this study, a large dataset of phishing and legitimate websites was gathered from various sources, including PhishTank, Alexa, and Kaggle. The collected data was then processed through an HTML representation module and a feature extraction module. The 28 features extracted by the feature extraction module, as listed in Table 1, were integrated with the HTML structure representation learning architecture from framework of AI@ntiPhish1.0 PLUS. A total of 25 million HTML attachments were used to train both the HTML structure representation module and the HTML content representation module. The training data for the phishing detection model included 100,000 phishing sites and 400,000 legitimate sites, collected from 2024/01/01 to 2024/01/31. 3,000 phishing and legitimate sites were collected for testing from 2024/02/01 to 2024/02/10.

B. Feature Extraction & Representation

The training features come from two sources: the feature extraction module and the HTML representation module. The HTML representation module extracts both content features and structure features, as illustrated in Figure 1. Expert features include lexical-based, link-based, content-based, and host-based features. These extracted features are processed through the data processing module, where mathematical methods analyze the importance of each feature, selecting the most effective ones for model training. Based on the architecture analysis of AI@ntiPhish1.0, the most effective features are shown in Table 1.

The HTML representation module is designed to learn both HTML structure and HTML content embedding. This module extracts feature vectors that represent the HTML structure and vectors that describe each intention or description within the HTML content. Finally, these three kinds of features are concatenated and fed into a generation network to produce a variety of phishing samples, ultimately training a more accurate and efficient phishing detection model.

According to analysis by Chen et al., using an LSTM seq-to-seq model for HTML structure representation is more appropriate because of the sequential dependencies between HTML tags in the HTML code. As phishing websites rapidly evolve, bad actors frequently employ evasion techniques to avoid detection, often obfuscating content or using similar

words in the HTML content. This makes it crucial to detect phishing intentions within the text, even when words are misspelled or partially obfuscated. Content representation goes beyond the visible text on a webpage. It also includes hidden elements, such as comments in the HTML code or attributes within tags, where phishing kits often insert common phishing intentions. Therefore, when training the HTML content transformer, we extract not only the standard textual content but also two commonly overlooked locations, as shown in Figure 4.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="description" content="Secure website for your personal banking"> ----- Text content attribute
  <title>Microsoft login</title>
</head>
<body>
  <form action="/submit-login" method="POST">
    <label for="username">Username</label> ----- Text textual content
    <input type="text" id="username" name="username"><br>
    <label for="password">Password</label> ----- Text textual content
    <input type="password" id="password" name="password"><br>
    <input type="submit" value="Login" onclick="window.location.href='http://.....'">
  </form>
  <p>This is a secure platform, please login to continue your session</p>
  <p>Click<a href="http://.....">here</a> for a special offer</p>
  <!-- Phishingkit xxx: Phishing intention may be hidden in this comment --> ----- Text comments
  <!-- Please login to receive your prize now! -->
</body>
</html>
```

Figure 4. Text extracting from content attribute, textual content and comments

Since importance of intentions can vary based on their location within the HTML, it's vital for the transformer model to account for this in training. Although transformer typically uses positional encoding, it is based on word order rather than feature extraction location. To accommodate this, this study enhanced the transformer architecture, incorporating location-specific information into the training process, as detailed in Figure 3.

TABLE 1. PHISHING FEATURES

Feature Type	Symbol	Feature
Lexical-based	F1	is special words
	F2	get title feature
Link-based	F3	is http connection
	F4	is redirect
	F5	is weird serial
	F6	get time
	F7	url linkin num
Content-based	F8	get kbytes
	F9	is login form
	F10	is frame
	F11	is meta redirect
	F12	is meta base64 redirect
	F13	same extern domain script rate
	F14	same external domain link rate
	F15	same external domain img rate
	F16	external a tag same domain
	F17	null a tag
	F18	script block rate
	F19	style block rate
Host-based	F20	dots
	F21	is ip address
	F22	url traffic rank
	F23	ipv4 numbers

	F24	Ipv6 numbers
	F25	organization
	F26	is with whois
	F27	is alias
	F28	get day age

C. Data Generation

Vectors combines features extracted by both the HTML representation module and the feature extraction module. It then forwards the combined vector to the data generation module for generating new data. In practice, legitimate websites typically provide a much larger dataset compared to phishing sites. Previously, AI@ntiPhish 1.0 PLUS addressed the issue of data imbalance by employing SMOTE, which generates synthetic data using a linear interpolation between two points and random values weighted between 0 and 1. This linear method restricts the range of generated data. To enhance data generation, this study introduces GAN to replace SMOTE, offering a more advanced approach for data generation.

The concept of GAN [12] is to generate data through a competitive process between two neural networks, iterating to match the original data distribution. This unsupervised approach involves training a generative model (G) to replicate data and a discriminative model (D) to differentiate real from generated data. The training continues until D can no longer reliably distinguish between the two, typically reaching an accuracy of 50%, signaling that G and D are in equilibrium. The objective function is shown in Equation (1).

$$\begin{aligned}
 \min_G \max_D V(D, G) &= E_{x \sim p_{data}(x)} [\log D(x)] \\
 &+ E_{z \sim p_z(z)} [\log (1 - D(G(z)))] \quad (1)
 \end{aligned}$$

D. Learning Algorithm

After data generation, all data and features are fed into the intelligent learning module for training. This module trains and optimizes the phishing detection model based on the input data. According to the analysis results of AI@ntiPhish1.0 PLUS, the XGBoost model demonstrates higher accuracy and stability in phishing detection compared to other algorithms. To more effectively compare the differences before and after, this study used this algorithm for subsequent learning and model optimization. Additionally, data generated using GAN instead of SMOTE has significantly improved the accuracy of the detection model compared to SMOTE. In this study, all data (including generated data) and features are trained using XGBoost, and the advantages of the HTML content embedding extracted from the transformer are evaluated in the context of phishing detection.

E. Evaluation Standard

The performance of the proposed model is assessed using the following evaluation metrics: accuracy, precision, recall,

and F1-score. The formulas for calculating these metrics are detailed in Table 2.

TABLE 2. EVALUATION METRICS

Symbol	Formula
Precision (Pr)	$TP/(FP+TP)$
Recall (Rc)	$TP/(FN+TP)$
F1 score (F1)	$(2*Precision*Recall)/(Precision+Recall)$
Accuracy (ACC)	$(TP+TN)/(TP+FN+FP+TN)$

IV. EXPERIMENTAL RESULTS

In this experiment, Ten-fold cross-validation is conducted to determine the optimal number of estimators for the XGBoost algorithm, selecting the one that provides the best balance of performance and stability.

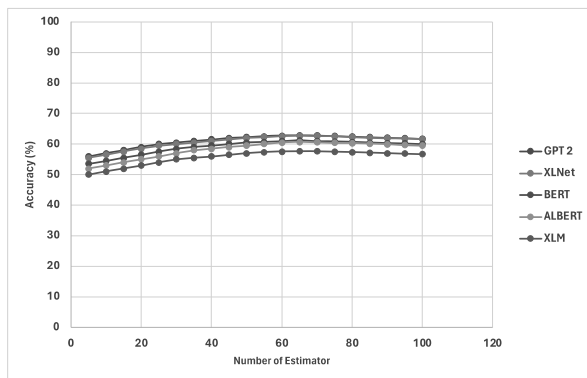


Figure 5. Accuracy of XGBoost with different transformer framework

Before integrating transformer models and GAN into the learning architecture, it is crucial to assess how various transformers affect HTML content representation and identify the best transformer for phishing detection. The initial experiment tested 5 transformers: BERT, ALBERT, GPT-2, XLNet, and XLM—using original HTML content to extract features for XGBoost training and subsequent performance evaluation.

During the experiment, all transformers uniformly returned 256-dimensional feature vectors to represent HTML content. This study also assessed the impact of different numbers of estimators, adjusting the count from 5 to 100 in increments of 5, and calculated the accuracy of the model for each scenario. Figure 5 illustrates the detection rates achieved by the XGBoost algorithm under various estimator counts and transformer architectures. This analysis results show that the model achieves optimal detection performance when the number of estimators is set to 65, regardless of the transformer used. Notably, the HTML content representation vectors generated by GPT-2 proved to be the most effective for phishing detection training, achieving a detection accuracy of 62.9%. This accuracy is only 0.002% lower than that achieved using XLNet. Figure 6 shows the detailed performance evaluation of the XGBoost model with different transformers with 65 estimators.

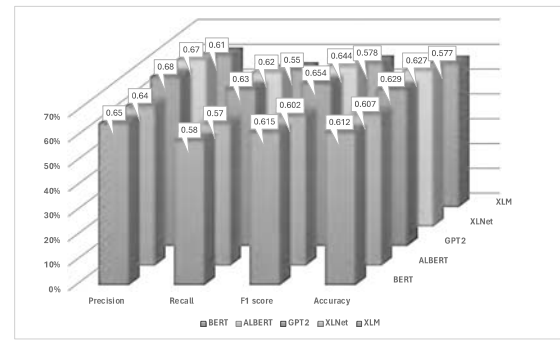


Figure 6. Metrics of XGBoost with different transformer framework (Estimator = 65)

Next, this study evaluated the performance of the model incorporating both GAN and transformer across 6 scenarios, with features including the original 28 features and HTML structure embeddings: XGBoost trained without HTML content features, XGBoost trained with HTML content features, XGBoost trained without HTML content features combined with SMOTE, XGBoost trained with HTML content features combined with SMOTE, XGBoost trained without HTML content features combined with GAN, and XGBoost trained with HTML content features combined with GAN. The experimental design involving these 6 scenarios not only reveals whether combining features extracted by transformers with existing features leads to further improvements but also examines the impact of data generated by GAN compared to SMOTE on the model. Additionally, the analysis can determine whether these two generation methods contribute to enhancements in detection accuracy and generalization of the model.

Figure 7 shows the effects of incorporating HTML content embeddings versus not using them. It also reveals how data generated by SMOTE and GANs can influence the training of detection models, highlighting which approach better improves the model's accuracy and robustness.

According to the experimental result, it demonstrates that integrating HTML content embeddings and data generation techniques significantly enhances the performance of the XGBoost model in detecting phishing threats. When comparing models trained with and without HTML content embeddings, those with embeddings show a substantial increase in accuracy, from 79.4% to 85.6%. This improvement underscores the effectiveness of embedding HTML content in enhancing the model's phishing detection capabilities, providing better feature representation and improving precision and recall. The introduction of data generation methods further improves model performance. XGBoost combined with SMOTE achieves an accuracy of 87.2%, while models trained with GAN show even more impressive results, pushing accuracy to 89.5%. The most significant gains are observed when both HTML content embeddings and GAN are applied together, yielding a precision of 0.95, recall of 0.92, F1 score of 0.935, and accuracy of 93.7%, which is 0.027 higher than when using SMOTE. These results highlight that the combination of HTML embeddings and advanced data

generation techniques—particularly GAN—creates a powerful phishing detection system. The embeddings increase detection accuracy, while the use of GAN provides the largest performance boost, optimizing both precision and recall.

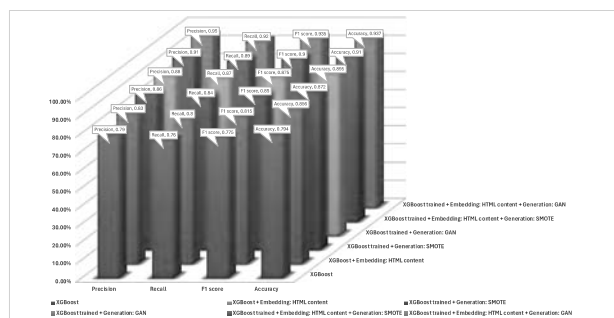


Figure 7. Accuracy of XGBoost with html content representation vectors and different transformer framework (Estimator = 65)

V. CONCLUSIONS

This study proposed an enhanced Transformer-based architecture combined with advanced data generation techniques, marking a significant advancement in phishing detection. By enhancing the model's capacity to capture both textual and hidden signals in HTML content, such as comments and attributes, this approach improves detection accuracy from 79.4% to 85.6%. The integration of GAN-generated synthetic data further boosts performance, achieving a detection accuracy of 93.7%, surpassing SMOTE by 2.7%. This study's success highlights the effectiveness of combining HTML content embeddings with GAN for detecting sophisticated phishing attacks. Future research will explore transforming HTML content into graph neural networks and applying hierarchical Transformers to improve performance and stability, aiming to enhance detection efficiency for large-scale attacks and analyze complex network relationships.

REFERENCES

- [1] R. Zieni, L. Massari and M. C. Calzarossa, "Phishing or Not Phishing? A Survey on the Detection of Phishing Websites," in IEEE Access, vol. 11, pp. 18499-18519, Feb, 2023.
- [2] S. Baki and R. M. Verma, "Sixteen Years of Phishing User Studies: What Have We Learned?," in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 2, pp. 1200-1212, 1 March-April 2023.
- [3] Gunjan and R. Prasad, "Phishing Email Detection Using Machine Learning: A Critical Review," 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), Greater Noida, India, pp. 1176-1180, 2024.
- [4] APWG, PHISHING ACTIVITY TRENDS REPORTS, <https://apwg.org/trendsreports/>
- [5] T. Niu and B. Wu, "Visual-based Phishing Website Recognition," 2024 IEEE 6th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, pp. 992-997, 2024.
- [6] J. M. Lindamulage, M. L. Y. S.P.J, P. I.S.S. and J. Krishara, "Vision GNN Based Phishing Website Detection," 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICES), Chennai, India, pp. 1-7, 2023.
- [7] M. Kawale, B. Maru, S. Dagu, M. Varghese and V. Gupta, "Machine Learning based Phishing Website Detection," 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, pp. 833-837, 2024.

- [8] E. O. Asani, V. O. Adedayo-Ajayi, A. E. Tunbosun, N. Enumah and D. R. Aremu, "Detection of Phishing Emails using Support Vector Classifier and Gaussian Latent Variable Model," 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG), Omu-Aran, Nigeria, pp. 1-5, 2024.
- [9] R. S. Al-Yozbaky and M. Alanezi, "Detection and Analyzing Phishing Emails Using NLP Techniques," 2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Istanbul, Turkiye, pp. 1-6, 2023.
- [10] Y. H. Chen and J. L. Chen, "Intelligent Learning Architecture with Hybrid Features for Phishing Detection," 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), Jeju Island, Korea, pp. 225-230, 2021.
- [11] A. Berroukham, K. Housni and M. Lahraichi, "Vision Transformers: A Review of Architecture, Applications, and Future Directions," 2023 7th IEEE Congress on Information Science and Technology (CiSt), Morocco, pp. 205-210, 2023.
- [12] A. Popuri and J. Miller, "Generative Adversarial Networks in Image Generation and Recognition," 2023 International Conference on Computational Science and Computational Intelligence (CSCI), USA, pp. 1294-1297, 2023.



Yu-Hung Chen received his B.S. degree in Electrical Engineering from National Kaohsiung University of Applied Sciences, Taiwan, in 2016. Since September 2016, he has studied for an M.S. degree in Electrical Engineering at National Taiwan University of Science and Technology. His main research interests include Software- Defined Network, Cyber Security, Network Security, Artificial Intelligence and Deep Learning.



JIANN-LIANG CHEN (Senior Member, IEEE) was born in Taiwan in December 1963. He received a Ph.D. in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 1989. Since August 1997, he has been with the Department of Computer Science and Information Engineering, National Dong Hwa University, where he is a Professor and the Vice Dean of the Science and Engineering College. He joins the Department of Electrical Engineering, National Taiwan University of Science and Technology, as a Distinguished Professor. His current research interests include cellular mobility management, cyber security, personal communication systems, and the Internet of Things (IoT).



Shih-Ping Chiu was born in Taipei, Taiwan, in 1984. She received the B.S. degree in 2007. She is a research assistant in electrical engineering from the National Taiwan University of Science and Technology, Taipei. Her main research interests include data analysis and the Internet of Things (IoT).