Adaptive Phishing URL Classification: A Generative Adversarial Approach

Noah Abdellatif
Department of Management Information Systems
University of Arizona
Tucson, USA
nabdellatif@arizona.edu

Mason Wagner
Department of Management Information Systems
University of Arizona
Tucson, USA
rmw26@arizona.edu

Benjamin M. Ampel
Department of Computer Information Systems
Georgia State University
Atlanta, USA
bampel@gsu.edu

James Lee Hu
Department of Management Information Systems
University of Arizona
Tucson, USA
jameshu@arizona.edu

Zara Ahmad-Post
Department of Management Information Systems
University of Arizona
Tucson, USA
zahmadpost@arizona.edu

Hsinchun Chen
Department of Management Information Systems
University of Arizona
Tucson, USA
hsinchun@arizona.edu

Abstract*—Uniform Resource Locators (URLs) are foundational for navigating the internet. However, URLs remain a critical cybersecurity threat. Billions of URL-based phishing attempts are logged annually. Extant machine and deep learning phishing URL detectors often report accuracy exceeding 95%. However, these models are often trained on outdated datasets that fail to capture current phishing URL patterns. Generative Adversarial Networks (GANs) offer a pathway to improve classifier robustness on unseen data. These networks are predominantly implemented with traditional deep learning generators. We therefore propose a GAN framework that pairs a transformer-based generator with a ModernBERT discriminator and trains them using a reinforcement learning policy-gradient mechanism with a novel multi-signal reward function. The discriminator is pretrained on historical URLs and evaluated on a proprietary dataset of modern phishing URLs to simulate realistic deployment. GAN-based fine-tuning improved the classifier's recall on modern URLs by about 5% on average, indicating that adversarial generation can create generalizable URL classifiers that adapt to evolving attacks.

***Keywords*—Cybersecurity, uniform resource locators, phishing, adversarial generation, deep learning**

I. Introduction

The proliferation of Generative AI (GenAI) and the modern web has reduced the technical barrier for cybercriminals to develop and deploy attack frameworks [1]. Among them, phishing campaigns remain one of the most prolific threats, with phishing-related losses expected to reach around $25 billion in 2026. Phishing is broadly described as the deceptive solicitation of sensitive information by impersonating a trusted source [2]. Uniform Resource Locators (URLs) serve as a primary method for hackers to collect sensitive data in email, SMS, and QR code phishing. There were 3.7 billion recorded phishing incidents over a six-month period in 2024 [3]. Researchers at Proofpoint found that URLs are four times as likely to appear in phishing attacks as file attachments because of their increased likelihood to evade detection by modern email security features [3].

Phishing URLs have grown more sophisticated, shorter, and easier to generate [4]. Attackers now use URL-shortening tools to conceal malicious file paths and ready-to-use phishing kits that enable the creation of HTTPS-certified URLs [5]. Figure 1 compares an older HTTP-based phishing URL with a modern HTTPS URL.

Figure 1. Phishing URLs Examples. Top (A.) shows an older example, while bottom (B.) shows a newer example.

Existing research has developed a range of machine learning (ML) and deep learning (DL) based detection systems, many of which report greater than 95% accuracy in their evaluations [6]. However, these detection models are often trained and evaluated on outdated datasets and may not generalize to modern phishing URL patterns.

Generative Adversarial Networks (GANs) offer a potential solution for keeping phishing URL detection models current. GANs can generate novel and targeted content that mirrors emerging attack patterns [7]. However, their capacity to generate realistic phishing URLs remains unclear, particularly because text is non-differentiable and therefore requires a novel loss function for the GAN to operate effectively.

To address these gaps, we propose a Generative Adversarial Network (GAN)-based framework that leverages a transformer-based generator and a pretrained ModernBERT classifier as the

discriminator. Our GAN is trained using a policy gradient mechanism with a multi-signal reward function that guides generation by evaluating samples for structural validity, lexical deception, and phishing-indicative content. This design incentivizes the model to create both novel and realistic phishing URL patterns. To simulate realistic conditions, the discriminator is pretrained on historical URL data. It is then evaluated against unseen URLs from 2024 onward, both before and after GAN fine-tuning. Model performance on more recent URL data reflects its ability to adapt to emerging attack patterns in modern phishing URLs. Furthermore, to identify the most effective generative objective, we port the defining mechanism of nine well-known text GANs into our framework as reward-function augmentations and compare them under identical conditions. This controlled experiment allows for comparison that their original, architecturally incompatible implementations never permitted. Such comparison informs the adaptation of a prevailing text GAN mechanism into our proposed framework to best address the structural complexities of deceptive phishing URL generation.

II. Literature Review

A. Phishing URL Detection

Phishing URL detection studies mostly focus on the automated detection of malicious URLs [8]. Early approaches relied predominantly on blacklisting and heuristic-based methods, which could not quickly adapt to evolving URLs without significant manual labor [6].

The field then began implementing ML and DL models, which advanced detection capabilities by enabling classifiers to learn features automatically rather than relying on a static rule set. Early systems leaned on an ensemble of ML models, making it one of the first systems capable of identifying AI-generated, shortened phishing URLs [9]. These feature-based methods performed significantly better than prior approaches on contemporary datasets. However, their reliance on manually engineered features limited adaptability to new phishing URLs.

The limitations of manual feature engineering motivated a shift toward DL-based detection, in which models learn representations directly from raw URL data. Literature explored modular design, combining CNN and LSTM architectures to enable rapid reconfiguration to new phishing URLs [10]. Research then adapted the pre-trained DistilBERT model to achieve generalizable phishing detection [11]. More recent research has adapted Large Language Model (LLM)-based one-shot learning frameworks for state-of-the-art phishing URL detection [12]. However, these frameworks were often trained and evaluated on dated public datasets. This necessitates fresh URL data.

B. Phishing URL Generation

Phishing URL generation refers to the automated creation of malicious URL strings designed to evade detection systems and deceive end users [13]. Rather than relying solely on real-world phishing data, generative approaches aim to produce novel URL samples that can drive improvements in the training process by exposing classifiers to patterns not present in historical data. Phishing URL generation research predominantly uses DL-based GAN architectures.

GAN-based architectures can improve URL classification performance by continually evading ML-based detection [7]. For example, a Wasserstein GAN (WGAN) framework generated adversarial samples that improved training across ML, ensemble, and DL sequence detectors (LSTM, RNN, and CNN). Research found that, among DL models, an LSTM generator paired with a CNN discriminator was the best combination for detection performance [15]. A later comparative study evaluated GAN, DCGAN, WGAN, and SeqGAN architectures for URL generation and classification using an LSTM model [13]. Across this literature, the generators are almost always LSTMs.

C. Malicious LLM Generation

Large Language Models (LLMs) are a class of transformer-based generative models pretrained on vast corpora of text data, providing them with state-of-the-art text generation capabilities [16]. LLMs' large embedding spaces and extensive pretraining enable them to produce high quality and contextually appropriate text that varies stylistically. This has attracted cybersecurity interest in LLMs as tools for generating malicious content, which informs their potential as adversarial URL generators.

Attackers first attempted to bypass LLMs through prompt engineering. Early work showed that Do Anything Now (DAN) prompting, reverse psychology, and prompt injection elicit malicious outputs from GPT-4, including phishing emails [17] and targeted spear-phishing attacks [18]. A multi-LLM pipeline then generated convincing phishing websites across GPT, Claude, and Bard without manual prompting [19], later refined by an output-optimization algorithm pairing beam search with Top-K selection [20]. Every thread, however, targets free-form content (e.g., emails, SMS, websites). URLs are a harder problem, as they must satisfy strict formatting rules while remaining lexically deceptive, which prompt engineering alone cannot guarantee. Whether an LLM can produce valid, deceptive phishing URL strings remains largely unexplored.

D. GANs for Text Generation

GANs are a class of generative models that employ an adversarial learning architecture in which a generator and discriminator are trained simultaneously in a minimax optimization framework [21]. Originally developed for image creation, GANs have since been extended to text generation tasks, where they have demonstrated promising results in producing realistic and diverse discrete sequences. However, text generation introduces a fundamental challenge absent in image generation, namely the non-differentiability of discrete token selection. Because sampled tokens carry no gradients, the discriminator's signal cannot flow back to the generator, and this single obstacle defines the design space for every technique that follows [22][23]. Addressing this challenge is critical for enabling effective GAN-based fine-tuning in the URL generation setting, where the generator must produce character-level sequences that are both structurally valid and lexically deceptive [14]. Figure 2 presents a diagram illustrating the non-differentiability problem with discrete token selection. To inform our methodology, we review seminal optimization techniques for discrete sequence generation in GANs.

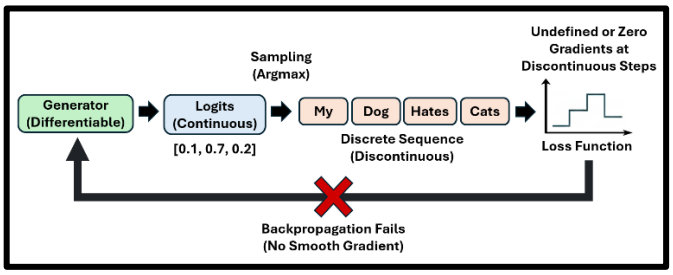

Figure 2. Non-Differentiability Problem with Discrete Token Generation

Gumbel-Softmax is a technique that approximates discrete token selection with a differentiable soft probability distribution, enabling standard backpropagation through the generator without the need for reinforcement learning [24]. By reparametrizing the sampling process using Gumbel noise, the technique allows gradients to flow directly through token selection during training. This technique underpins several text-generation GANs, including RelGAN [25]. While this approach offers low variance and full differentiability, it introduces approximation bias that grows as the temperature parameter is annealed toward a truly discrete distribution. Furthermore, the requirement that the generator and discriminator share compatible embedding spaces limits architectural flexibility, making Gumbel-Softmax less suitable for settings where the generator and discriminator differ in architecture or embedding dimensionality.

REINFORCE and PPO are both policy gradient methods that address the non-differentiability problem by framing text generation as a reinforcement learning problem, where the generator is treated as a policy that receives reward signals from the discriminator [23][26]. Rather than approximating the sampling process, both methods operate directly over discrete tokens, computing parameter updates through the expected reward gradient, and impose no constraints on the relationship between the generator and discriminator embedding spaces. While REINFORCE offers low bias and broad architectural compatibility, it is susceptible to high variance in gradient estimates, particularly in settings with sparse or delayed reward signals, which can result in slow convergence and training instability [20]. PPO addresses these limitations by introducing a clipped surrogate objective that constrains the magnitude of policy updates at each training step [27], achieving more stable training while retaining the architectural flexibility inherent to policy gradient methods. These properties have made PPO the prevailing method for fine-tuning LLMs.

Of these three techniques, REINFORCE and PPO are better suited for URL sequence generation tasks than Gumbel-Softmax, as policy gradient methods enable training over discrete tokens without architectural constraints or approximation bias. This flexibility is particularly important given that LLMs occupy a substantially larger embedding space than most prevailing transformer-based classifiers [28].

III. Research Gaps and Questions

Based on our literature review, we identified the following research gaps. First, most current URL detection models rely on data that is many years old and may not be representative of modern phishing techniques. Second, prior research has not tested the ability of LLM generated URLs in evading ML and DL based detectors. Third, LLMs ability to produce realistic phishing URLs remains underexplored. Lastly, the non-differentiability of discrete token selection during text generation poses a challenge to backpropagation in GANs. To address these gaps, we pose the following research questions:

- How is the performance of extant detection models affected when evaluated on current phishing URL data?
- How can LLMs be adapted for malicious URL generation to enhance the training of modern detectors?
- How can we leverage LLMs to generate lexically deceptive phishing URLs?
- How can a GAN-based training loop be used to fine-tune state-of-the-art LLM architectures for adversarial URL generation?

IV. Research Design

This study proposes a novel GAN framework that leverages a transformer-based generator to create more robust phishing URL classifiers. Our research method comprises of four main components (Figure 3): Data Collection, Preprocessing, Framework, and Evaluations.

A. Data Collection

Legitimate URLs were collected from two sources totaling 210,536 instances. The Dephides dataset contributed 177,996 legitimate URLs spanning 2018 to 2024, while the LegitPhish dataset contributed an additional 37,540 URLs collected in 2025. Both sources collected legitimate URLs from publicly available tools such as the Google Search API and Common Crawl, providing access to popular domain data [29][30],.

Phishing URLs were also collected from two sources, totaling 182,457 instances. The Dephides dataset contributed 138,873 phishing URLs from 2018, while our anonymous industry partner contributed 43,584 phishing URLs spanning 2024 to 2025. Partner data captures recent phishing URL data, enabling performance evaluation on modern phishing patterns unseen during model training. A subset of 5,000 legitimate URLs from the Dephides dataset serves as real samples during GAN training. In total, our combined dataset comprises 392,993 URLs spanning 2018 to 2025 across three sources.

B. Preprocessing

Prior to training, all collected URL data underwent a standardized preprocessing pipeline to ensure consistency across sources. Raw URL strings were ingested and organized using the pandas data manipulation library, with records sorted and validated to confirm structural integrity across each data source. Duplicate entries were identified and removed to prevent data leakage between training and evaluation splits. The target label column was one-hot encoded, with phishing URLs assigned a value of 1 and legitimate URLs assigned a value of 0, producing a binary classification target compatible with all types of classifiers. Following preprocessing, the finalized dataset of 392,993 URL entries was partitioned into training and evaluation splits, with models trained on historical URL data

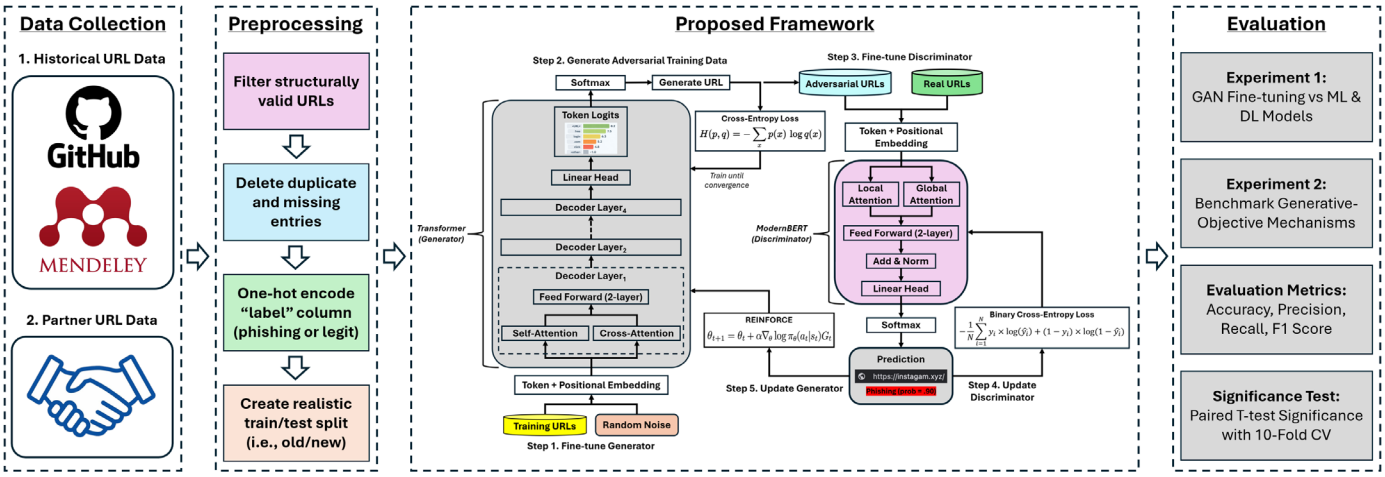

Figure 3. Research Design

from 2018 and performance evaluated on proprietary phishing URLs from 2024 onward to simulate realistic conditions.

C. Proposed Framework

As seen in Figure 3, our proposed framework comprises a character-level autoregressive transformer generator and a pretrained ModernBERT discriminator. GAN fine-tuning of the discriminator uses policy gradients guided by a multi-signal reward function. REINFORCE was selected as the policy gradient mechanism due to its low bias, broad architectural compatibility, and ability to operate directly over discrete tokens without imposing constraints on model embedding spaces [23]. Our framework follows a five-step procedure to improve classifier robustness to modern phishing patterns. Steps 1 and 2 generate unique, realistic phishing URLs to serve as adversarial samples for fine-tuning. Steps 3 through 5 fine-tune the pretrained ModernBERT classifier as well as the transformer generator in a GAN-based adversarial training loop.

In Step 1, pretraining uses the standard cross-entropy loss objective seen below:

$$H(p,q) = -\sum_x p(x) \log q(x), \qquad (1)$$

where x is an individual character, $p(x)$ is the true probability distribution of the next character in a sequence, $q(x)$ is the generator's predicted probability distribution for the next character in a sequence, and $H(p,q)$ is the cross-entropy of the distribution q relative to distribution p. The transformer generator learns to minimize the cross-entropy between the two distributions, effectively learning next token prediction for generating character sequences that model realistic phishing URLs.

Step 2 invokes the pretrained transformer to generate adversarial URL data by autoregressively sampling character tokens from its learned distribution. These URLs are passed to the discriminator, at each round of fine-tuning. In Step 3, the discriminator takes these URLs as input in batches along with real benign samples from the designated subset of training data.

The discriminator's parameters are updated using binary cross-entropy loss in step 4:

$$-\frac{1}{N}\sum_{i=1}^{N} y_i \times \log(\hat{y}_i) + (1 - y_i) \times \log(1 - \hat{y}_i), \qquad (2)$$

where N is the total number of samples in a batch, y_i is the ground-truth label for the i-th URL, and $\hat{y}_i$ is the predicted probability that i belongs to the phishing class. By minimizing this loss function during fine-tuning, the discriminator adjusts its weights to account for patterns within the adversarial generated samples.

In Step 5, the generator is optimized through discriminator feedback using the REINFORCE policy gradient update rule:

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta \log \pi_\theta(a_t s_t) G_t, \qquad (3)$$

where θ_t is the model's current parameters, α is the learning rate, $\pi_\theta(a_t s_t)$ is the policy that determines the probability of taking an action a_t given the current state s_t, and G_t is the accumulated reward signal assigned to a URL sample at each token generation step t. To guide these updates to adversarial generation, we designed the multi-signal reward function summarized in Table I.

TABLE I. MULTI-SIGNAL REWARD FUNCTION

Signal	Weight	Measurement
Phishing	0.25	Probability that URL is phishing
Lookalike	0.20	Levenshtein distance from popular domains
Strategy	0.20	Advanced phishing techniques (e.g., subdomain squatting, free TLDs, raw IPs)
Validity	0.15	Measurement of URLs structural validity
Differentiation	0.12	Edit distance from known benign URLs
Content	0.08	Presence of phishing-hint tokens

D. Evaluation

To assess the effectiveness of the proposed framework, we conduct two experiments. Models are evaluated using a proprietary dataset comprised of phishing URL data from 2024 onward to demonstrate the effectiveness of adapting to modern attack patterns. Evaluation across all experiments prioritizes recall as the primary performance metric, as undetected phishing URLs pose a greater risk than mislabeled legitimate

URLs. Accuracy, precision, and F1 score are still reported to provide a comprehensive assessment of classifier performance.

V. Results and Discussion

A. Experiment 1: GAN Fine-tuning Against ML and DL Models

To evaluate the impact of GAN-based fine-tuning on modern phishing URL detection, we benchmark a range of ML and DL classifiers trained on historical URL data against a GAN fine-tuned version of the best-performing benchmarked model. Results are summarized in Table II.

TABLE II. Experiment 1 Results

Model	Accuracy	Precision	Recall	F1 Score
Gaussian SVM	47.67% ***	99.84% ***	2.79% ***	5.42% ***
Logistic Regression	62.10% ***	**99.96%** ***	29.59% ***	45.66% ***
Decision Tree	66.38% ***	89.14% ***	42.75% ***	57.78% ***
Random Forest	62.50% ***	80.91% ***	39.70% ***	53.27% ***
K-Nearest Neighbor	65.64% ***	98.81% ***	36.60% ***	53.42% ***
LSTM	85.18% ***	96.03% ***	75.60% ***	84.60% ***
BERT	84.73% ***	95.89% ***	74.84% ***	84.07% ***
RoBERTa	89.03%	98.49% ***	80.87% ***	88.81% ***
ModernBERT + LSTM GAN	88.39% **	89.16% ***	**89.28%** ***	89.22%**
ModernBERT	**89.26%** *	96.57% ***	82.99% ***	89.27% *
ModernBERT + Transformer GAN	88.98%	91.38%	87.80%	**89.56%**

Note: * indicates statistically significant difference from best model at $p < 0.05$, ** at $p < 0.01$, and *** at $p < 0.001$.

Traditional ML classifiers performed poorly on modern phishing URLs, reflecting the limited generalization of handcrafted features. Gaussian SVM reached just 2.79% recall, with Logistic Regression and Random Forest at 29.59% and 39.70% respectively. High precision with low F1 shows these models overclassify test data as benign, which is consistent with our review that ML approaches are ill-equipped to handle the sophistication of modern phishing URL construction.

DL models performed substantially better, validating learned representations over handcrafted features. LSTM and BERT reached 75.60% and 74.84% recall and RoBERTa 80.87%, while ModernBERT was strongest among non-GAN models at 82.99%, consistent with its state-of-the-art pretraining.

GAN fine-tuning with our framework improved ModernBERT recall from 82.99% to 87.80%. An LSTM generator reached slightly higher recall (89.28%) but dropped F1 below the base ModernBERT model, signaling an overall performance decrease. This suggests that transformer-based generation improves classification without sacrificing precision on legitimate URLs, while exposing the discriminator to adversarial samples meaningfully improves generalization to modern phishing patterns—supporting the central premise of our framework.

B. Experiment 2: Ablating Generative-Objective Mechanisms

Having established that GAN fine-tuning improves detection (Experiment 1), we next isolate which generative-objective mechanism most benefits adversarial URL generation. Rather than re-implementing nine distinct GAN architectures, each with its own generator, discriminator, and training scheme, we hold our framework fixed and implement only the defining mechanism each contributes, re-expressed as an augmentation to our base reward function (Table I). This includes diversity objectives (RelGAN, StyleGAN), Wasserstein and least-squares stabilization (WGAN-GP, LSGAN), auxiliary-classifier conditioning (ACGAN, Conditional SeqGAN, Conditional MaliGAN), and memory- and masking-based objectives (LeakGAN, MaskGAN). This standardization enables a direct comparison their original, architecturally incompatible implementations never permitted. Results are summarized in Table III.

TABLE III. Experiment 2 Results

Model	Accuracy	Precision	Recall	F1 Score
Conditional MaliGAN	88.42% ***	89.47% ***	**88.96%** ***	89.21% ***
WGAN-GP	88.80% ***	90.71% ***	88.23% ***	89.45% *
Conditional SeqGAN	88.79% **	90.58% ***	88.36% ***	89.46%
LeakGAN	88.83% ***	90.93% ***	88.03% ***	89.46% **
MaskGAN	88.85% ***	91.00% ***	88.00% ***	89.47% **
ACGAN	88.86% ***	90.98% ***	88.02% ***	89.48% **
LSGAN	88.88% **	91.04% ***	88.01% ***	89.50%
StyleGAN	88.95%	**91.43%**	87.69% **	89.52%
RelGAN	**88.98%**	91.38%	87.80%	**89.56%**

Note: * indicates statistically significant difference from best model at $p < 0.05$, ** at $p < 0.01$, and *** at $p < 0.001$.

Performance was stable across all nine mechanisms (F1 89.21%–89.56%; recall 87.69%–88.96%), indicating that the proposed framework is the primary driver of performance. Among mechanisms, RelGAN's diversity-promoting signal yielded the marginal best F1 (89.56%) by discouraging repetitive URL features [25]. We adopt this configuration as our final model, reported as the ModernBERT + Transformer GAN row in Table II. This diversity signal exposes the discriminator to broader phishing patterns, improving generalization to unseen attacks. Conditional MaliGAN yielded the highest recall (88.96%) yet was the only mechanism to show an overall decrease relative to the base ModernBERT classifier, suggesting its objective may introduce instability.

Figure 4 presents token-level attribution maps for two adversarial URLs from the RelGAN fine-tuned transformer; colors reflect the discriminator's weight from benign (dark blue) to phishing (dark red).

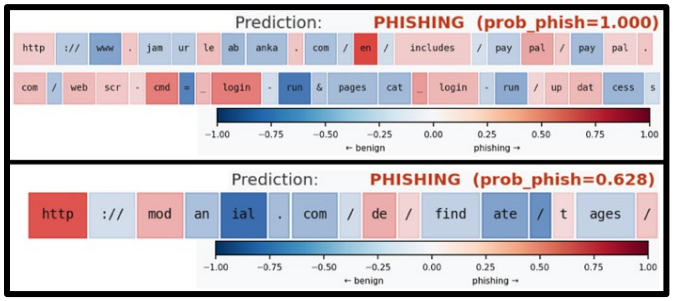

Figure 4. Token-level Attribution Maps of Phishing URL Samples

In the top example, the URL is classified as phishing with 100% probability, with phishing-weighted tokens on HTTP, multiple Top-Level Domains (TLDs), and payment- or credential-related terms (e.g., "pay," "login"). In the bottom

example, it is again identified at 62.8% probability: a small token count and common TLD ("com") reduce phishing-indicative density, yet the HTTP prefix and atypical domain structure remain dominant. These maps confirm the reward function's central role in robust phishing-feature identification.

VI. Conclusion and Future Work

This study proposed a GAN-based adversarial framework with a multi-signal reward function to improve phishing URL detection against modern attacks. GAN fine-tuning raised ModernBERT recall on modern phishing URLs by approximately 5%. A controlled ablation of nine generative-objective mechanisms showed the framework is robust to the choice of mechanism (F1 within 0.35 points across all nine), with a diversity-promoting signal yielding the marginal best. Transformer-based adversarial generation thus increases robustness to evolving phishing threats while reducing dependence on current URL data.

Acknowledgment

This work was supported in part by the National Science Foundation under Grant No. DGE-2336109.

References

[1] Y. Bengio et al., "Managing extreme AI risks amid rapid progress," Science, vol. 384, no. 6698, pp. 842–845, 2024.

[2] K. Stouffer et al., "Guide to Operational Technology (OT) Security," National Institute of Standards and Technology, NIST SP 800-82r3, 2023. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r3.pdf

[3] Proofpoint, "The human factor 2025 vol. 2: URL phishing," 2025. [Online]. Available: https://www.proofpoint.com/us/resources/threat-reports/human-factor-url-phishing

[4] D. M. Divakaran and A. Oest, "Phishing detection leveraging machine learning and deep learning: A review," IEEE Security & Privacy, vol. 20, no. 5, pp. 86–95, 2022.

[5] M. Sameen, K. Han, and S. O. Hwang, "PhishHaven—An efficient real-time AI phishing URLs detection system," IEEE Access, vol. 8, pp. 83425–83443, 2020.

[6] S. Asiri, Y. Xiao, S. Alzahrani, S. Li, and T. Li, "A survey of intelligent detection designs of HTML URL phishing attacks," IEEE Access, vol. 11, pp. 6421–6443, 2023.

[7] T. N. Bac, P. T. Duy, and V.-H. Pham, "PWDGAN: Generating adversarial malicious URL examples for deceiving black-box phishing website detector using GANs," in Proc. 2021 IEEE Int. Conf. Machine Learning and Applied Network Technologies (ICMLANT), 2021, pp. 1–4.

[8] F. Sadique, R. Kaul, S. Badsha, and S. Sengupta, "An automated framework for real-time phishing URL detection," in Proc. 2020 10th Annu. Computing and Communication Workshop and Conf. (CCWC), 2020, pp. 335–341.

[9] K. Haan, "Top website statistics today," Forbes, 2025. [Online]. Available: https://www.forbes.com/advisor/business/software/website-statistics/

[10] J. Lee, F. Tang, P. Ye, F. Abbasi, P. Hay, and D. M. Divakaran, "D-Fence: A flexible, efficient, and comprehensive phishing email detection system," in Proc. 2021 IEEE Eur. Symp. Security and Privacy (EuroS&P), 2021, pp. 578–597.

[11] A. Aljofey, S. A. Bello, J. Lu, and C. Xu, "BERT-PhishFinder: A robust model for accurate phishing URL detection with optimized DistilBERT," IEEE Transactions on Dependable and Secure Computing, vol. 22, no. 4, pp. 4315–4329, 2025.

[12] F. Rashid, N. Ranaweera, B. Doyle, and S. Seneviratne, "LLMs are one-shot URL classifiers and explainers," Computer Networks, vol. 258, art. no. 111004, 2025.

[13] T. D. Pham, P. T. T. Tran, and V. C. Ta, "Evaluation of GAN-based models for phishing URL classifiers," International Journal of Computer Network and Information Security, vol. 15, no. 2, pp. 1–14, 2023.

[14] P. T. Duy, V. Q. Minh, B. T. H. Dang, N. D. H. Son, N. H. Quyen, and V.-H. Pham, "A study on adversarial sample resistance and defense mechanism for multimodal learning-based phishing website detection," IEEE Access, vol. 12, pp. 137805–137824, 2024.

[15] S. Al-Ahmadi, A. Alotaibi, and O. Alsaleh, "PDGAN: Phishing detection with generative adversarial networks," IEEE Access, vol. 10, pp. 42459–42468, 2022.

[16] C. Peng et al., "Generative large language models are all-purpose text analytics engines: Text-to-text learning is all your need," Journal of the American Medical Informatics Association, vol. 31, no. 9, pp. 1892–1903, 2024.

[17] M. Gupta, C. Akiri, K. Aryal, E. Parker, and L. Praharaj, "From ChatGPT to ThreatGPT: Impact of generative AI in cybersecurity and privacy," IEEE Access, vol. 11, pp. 80218–80245, 2023.

[18] Q. Qi, Y. Luo, Y. Xu, W. Guo, and Y. Fang, "SpearBot: Leveraging large language models in a generative-critique framework for spear-phishing email generation," Information Fusion, vol. 122, art. no. 103176, 2025.

[19] S. S. Roy, P. Thota, K. V. Naragam, and S. Nilizadeh, "From chatbots to phishbots?: Phishing scam generation in commercial large language models," in Proc. 2024 IEEE Symp. Security and Privacy (SP), 2024, pp. 36–54.

[20] J. Fairbanks and E. Serra, "Generating phishing attacks and novel detection algorithms in the era of large language models," in Proc. 2024 IEEE Int. Conf. Big Data (BigData), 2024, pp. 2314–2319.

[21] I. J. Goodfellow et al., "Generative adversarial nets," in Advances in Neural Information Processing Systems, vol. 27, 2014.

[22] I. J. Goodfellow, Y. Bengio, and A. Courville, "Differentiable generator networks," in Deep Learning. Cambridge, MA, USA: MIT Press, 2016, p. 691.

[23] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in Proc. AAAI Conf. Artificial Intelligence, vol. 31, no. 1, 2017.

[24] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel-Softmax," in Proc. Int. Conf. Learning Representations (ICLR), 2017.

[25] W. Nie, N. Narodytska, and A. Patel, "RelGAN: Relational generative adversarial networks for text generation," in Proc. Int. Conf. Learning Representations (ICLR), 2018.

[26] T. Wang, L. Liu, H. Zhang, L. Zhang, and X. Chen, "Joint character-level convolutional and generative adversarial networks for text classification," Complexity, vol. 2020, no. 1, art. no. 8516216, 2020.

[27] Q. Wu, L. Li, and Z. Yu, "TextGAIL: Generative adversarial imitation learning for text generation," in Proc. AAAI Conf. Artificial Intelligence, vol. 35, no. 16, 2021, pp. 14067–14075.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186.

[29] O. K. Sahingoz, E. Buber, and E. Kugu, "DEPHIDES: Deep learning-based phishing detection system," IEEE Access, vol. 12, pp. 8052–8070, 2024.

[30] R. S. Potpelwar, U. V. Kulkarni, and J. M. Waghmare, "LegitPhish: A large-scale annotated dataset for URL-based phishing detection," Data in Brief, vol. 60, art. no. 111972, 2025