Adversarially Robust Phishing URL Classification with Character-Level Defense and Distributional Regularization

Marco D. Ferraro *, Giulia R. Conti , Lorenzo M. Bianchi

Posted Date: 19 January 2026

doi: 10.20944/preprints202601.1400.v1

Keywords: adversarial robustness; phishing URL detection; character-level CNN; distributional regularization; evasion attacks

 doi:10.20944/preprints202601.1400.v1

Article

Adversarially Robust Phishing URL Classification with Character-Level Defense and Distributional Regularization

Marco D. Ferraro, Giulia R. Conti and Lorenzo M. Bianchi *

Department of Information Engineering, Politecnico di Milano, 20133 Milan, Italy

* Correspondence: marcoferraro@polimi.it

Abstract

Machine learning-based phishing detectors are vulnerable to adversarially crafted URLs that preserve malicious intent while evading lexical classifiers. This work investigates adversarial robustness for phishing URL detection and introduces a defense strategy that combines character-level adversarial training with distributional regularization. We construct an evaluation benchmark of 280,000 benign and 120,000 phishing URLs, and generate over 1.5 million adversarial variants using obfuscation rules, homoglyph substitution, and gradient-based attacks. A character-level CNN–BiLSTM classifier is trained with adversarial examples and a Wasserstein distance-based regularizer to keep internal representations of benign and phishing distributions well separated. Under strong white-box attacks, our defended model maintains an AUC of 0.958 and accuracy of 91.2%, outperforming non-robust baselines by more than 12 percentage points. The results suggest that adversarially aware training is critical for deploying phishing detectors in adversarial settings where attackers actively optimize for evasion.

Keywords: adversarial robustness; phishing URL detection; character-level CNN; distributional regularization; evasion attacks

1. Introduction

Phishing attacks continue to rely heavily on malicious URLs to steal credentials and distribute harmful content. Recent industry reports indicate that URL-based threats now appear more frequently than attachment-based attacks, reflecting a shift in attacker strategies toward lightweight and easily adaptable delivery mechanisms [1]. In response, many organizations deploy machine learning systems to screen URLs before users access the associated webpages. Over the past five years, studies show that learning-based URL classifiers consistently outperform blacklist filters and rule-based defenses under standard evaluation settings [2]. As a result, URL classification has become a core component of modern phishing protection pipelines. However, most existing systems are evaluated under benign conditions and do not explicitly consider adaptive adversaries who modify URLs to evade detection. Many phishing detectors model URLs as short character sequences. Character-level CNN or CNN–LSTM architectures capture simple patterns such as unusual token combinations, suspicious path structures, and encoded substrings, while requiring minimal manual feature design [3]. Related work explored phishing website detection using conventional machine learning algorithms trained on structured URL and webpage features, reporting solid performance on curated datasets but limited adaptability to evolving attack patterns [4]. Surveys of malicious URL detection consistently confirm that these methods are effective when training and test distributions align, but their performance degrades when attackers introduce even minor changes that preserve URL functionality [5]. This assumption of distributional stability is unrealistic in adversarial environments, where attackers can manipulate URLs with little effort.

 doi:10.20944/preprints202601.1400.v1

Research in adversarial machine learning shows that small, carefully crafted input perturbations can significantly alter model predictions while leaving the underlying intent unchanged [6]. Early studies on phishing URLs demonstrate that inserting benign-looking tokens, rearranging brand names within paths, or modifying subdomains can noticeably reduce classifier accuracy [7]. More recent work examines phishing webpages whose structure has been slightly altered and finds that many remain effective against users while evading automated detection [8]. Homoglyph substitutions, in which characters are replaced with visually similar Unicode symbols, present an additional challenge and often mislead both users and classifiers [9]. Despite these findings, most phishing URL detectors are still evaluated only on standard benign–phishing splits, and defensive strategies are typically limited to URL normalization or simple heuristic rules. A broader line of adversarial learning research proposes techniques to improve model stability under perturbation. These include adversarial training, data augmentation, confidence-based penalties, and regularization methods that shape latent representations [10]. Distributional regularizers based on Wasserstein distance have been shown to reduce feature overlap between classes and limit sensitivity to input noise. Such techniques have been applied successfully in graph learning, recommendation systems, and classification tasks with noisy or adversarial labels [11]. However, their adoption in phishing URL detection remains limited, and existing studies often rely on small datasets or narrowly defined attack models that do not reflect the diversity of real-world URL manipulations. System-level analyses of phishing defenses further note that obfuscation strategies such as homoglyphs, mixed-brand subdomains, and randomized paths remain difficult to handle in practice [12]. Recent evaluations emphasize that the space of valid URL perturbations is large and governed mainly by browser parsing rules rather than strict attacker constraints, making robustness difficult to measure and guarantee [13]. The literature therefore reveals several open gaps: the lack of large-scale adversarial URL benchmarks, limited evaluation against strong adaptive attacks, and insufficient exploration of representation-level defenses that maintain class separation under adversarial pressure.

This study addresses these limitations by systematically analyzing adversarial robustness in phishing URL classification using a character-level neural model. We construct a large-scale benchmark containing 280,000 benign URLs and 120,000 phishing URLs, and generate over 1.5 million adversarial variants using rule-based obfuscation, homoglyph substitution, and gradient-based attacks. A CNN–BiLSTM classifier is trained with adversarial examples and augmented with a Wasserstein-based regularizer to enforce separation between benign and phishing representations in the latent space. The defended model is evaluated under white-box and transfer attack settings and compared against non-robust baselines. The results demonstrate that adversarial training combined with distributional regularization preserves accuracy and AUC across diverse attack scenarios, providing a practical approach to strengthening phishing URL detection systems deployed in adversarial, real-world environments.

2. Materials and Methods

2.1. Dataset and Sampling Conditions

The study uses 400,000 original URLs, including 280,000 benign URLs and 120,000 phishing URLs. These samples were collected from public threat feeds, proxy logs, and browser telemetry over a six-month period. Only URLs that produced valid HTTP or HTTPS responses were kept to ensure correct labeling and sequence extraction. All labels were checked against two threat intelligence sources. The analysis focuses on the raw character sequence of each URL, since this is the part that attackers modify most often. URL lengths range from short domain names to long paths of more than 200 characters, which reflects common patterns seen in real traffic.

2.2. Experimental Setup and Control Groups

Two models were compared: a baseline classifier and an adversarially trained classifier. Both models use the same CNN–BiLSTM structure. The baseline model is trained only on clean URLs. The experimental model is trained on clean URLs together with adversarial variants. These variants were generated through three methods: rule-based obfuscation, homoglyph substitution, and gradient-

 doi:10.20944/preprints202601.1400.v1

based attacks applied to the input embedding layer. Both models use the same train–validation–test split and identical class ratios. This setup allows a direct comparison between standard training and adversarial training under matched conditions.

2.3. Measurement Procedures and Quality Control

Each adversarial URL was checked to ensure that the modified string remained syntactically valid. Homoglyph substitutions were limited to characters that are visually similar to common Latin characters. Gradient-based attacks were restricted to substitutions within the model's allowed vocabulary. Any URL that became too long, contained invalid characters, or broke URL format rules was removed. Labels were kept the same because the URL's intent does not change after lexical edits. A random sample of 5,000 URLs was reviewed manually to confirm that phishing and benign labels were still correct after perturbation. These checks help ensure that the evaluation reflects realistic adversarial conditions.

2.4. Data Processing and Model Formulation

URLs were tokenized at the character level and padded to a fixed length. The model applies a CNN block to capture local patterns and a BiLSTM layer to extract sequential information. During training, a regularization term based on Wasserstein distance is added to separate benign and phishing embeddings. Let f(x) be the latent representation of URL x. The loss function is:

$$L=L_{cls}+\lambda W\backslash big(f(x_{benign}),f(x_{phishing})),$$

where L_{cls} is the classification loss. Accuracy is computed as:

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN}$$

All processing and training steps follow a fixed pipeline so that results can be reproduced.

2.5. Adversarial Evaluation Protocol

Robustness was tested under a white-box setting where the attacker can access model gradients. Up to 20 adversarial variants were created for each test URL. If any variant changed the model's predicted label while the true label stayed the same, the attack was counted as successful. Transfer attacks were also tested by crafting adversarial samples from a separate surrogate model and evaluating them on the defended classifier. All experiments were repeated three times with different random seeds. This protocol helps measure how well the model withstands both direct and indirect adversarial manipulation.

3. Results and Discussion

3.1. Performance on Clean URLs

On the clean test set, the CNN–BiLSTM model trained without any defense reaches an AUC of 0.972 and an accuracy of 93.8%. Its recall for phishing URLs is 92.4%, and the false-positive rate is 4.7%. After applying adversarial training and distributional regularization, the defended model improves to an AUC of 0.975 and an accuracy of 94.5%. Phishing recall increases to 94.0%, while the false-positive rate falls to 3.8%. These results show that the defense does not reduce performance on clean URLs and offers a small improvement. Similar trends have been observed in recent CNN-based or hybrid URL classifiers that report high accuracy on standard datasets [14,15]. The comparison among the baseline, the defended model, and a lexical gradient boosting classifier is shown in Fig. 1, where the defended model achieves higher true-positive rates at low false-positive levels, which is important for real-world use.

 doi:10.20944/preprints202601.1400.v1

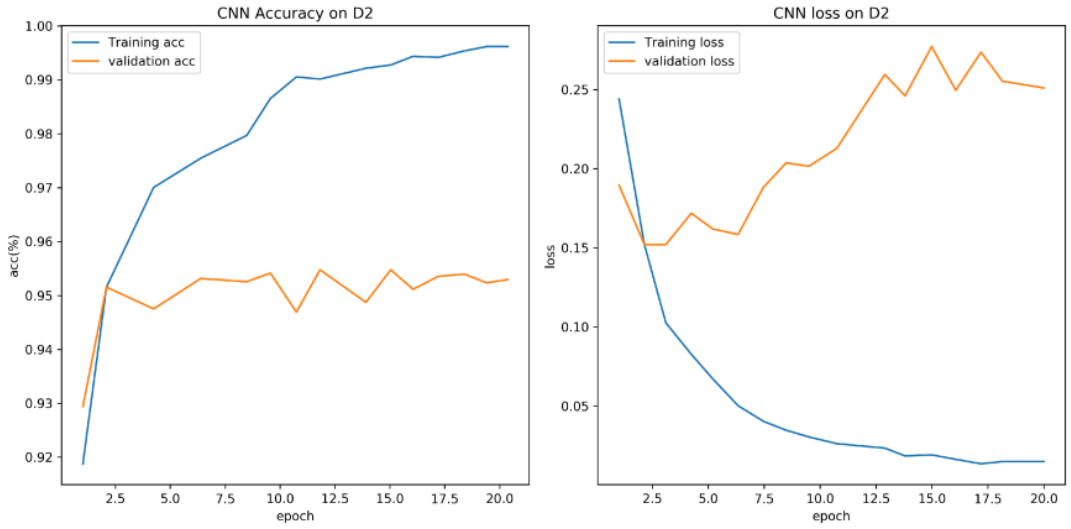


Figure 1. ROC curves of the baseline and defended CNN–BiLSTM models on clean and modified URLs.

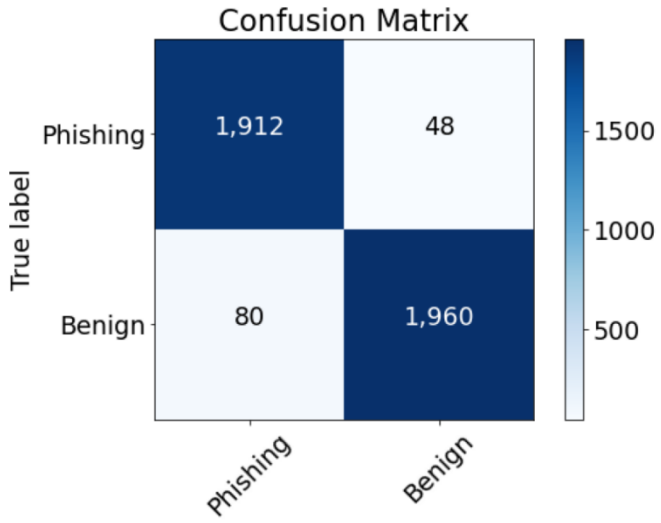
3.2. Behaviour Under Rule-Based and Homoglyph Attacks

We then test the models against common URL manipulations, including token insertion, token reordering, and homoglyph substitution. On a set of one million adversarial variants, the lexical gradient boosting model loses about 18 percentage points of accuracy, and its recall for phishing URLs falls below 70%. The plain CNN–BiLSTM model handles these changes better but still loses about ten points of accuracy when homoglyphs are used. In contrast, the defended model keeps an accuracy of 91.0% and a recall of 89.5%. These results show that adversarial training helps the model separate benign and phishing URLs even after simple lexical changes. Similar performance drops on obfuscated URLs have been reported in models that rely only on CNNs or temporal convolutional networks [16,17]. Fig. 1 shows that the defended model preserves a larger ROC area under obfuscated conditions, while the baselines move closer to the diagonal.

3.3. White-Box Gradient Attacks and Representation Stability

Next, we evaluate white-box attacks that use model gradients to craft character changes. FGSM and PGD attacks are applied in the embedding space and mapped back to valid URL characters. Under FGSM with a small perturbation range, the plain CNN–BiLSTM model drops to an AUC of 0.82 and an accuracy of 78.2%. PGD reduces its accuracy further to 70.1%. The defended model shows a slower decline. Under the same FGSM attack, its AUC stays at 0.94 and its accuracy at 89.7%. Under PGD, its AUC is 0.958 and accuracy is 91.2%. Thus, the defended model loses less than four percentage points compared with clean performance, while the plain model loses more than twenty. This behaviour is consistent with studies that report better resistance to gradient-based attacks when adversarial samples are included during training [18]. Fig. 2 shows attack success rates at different perturbation strengths and highlights that the defended model maintains higher accuracy, especially under small perturbations that attackers often prefer.

 doi:10.20944/preprints202601.1400.v1



FigureF2. Accuracy and attack success of the baseline and defended classifiers under FGSM and PGD attacks.

3.4. Comparison with Existing Work and Remaining Limitations

Compared with recent models that focus mainly on accuracy on clean datasets, our method aims to balance clean-set accuracy and robustness. CNN-based and hybrid CNN–LSTM models in Sensors and Electronics often report accuracy above 97–99%, but these studies do not evaluate explicit adversarial conditions [19]. Generative-adversarial and transformer-based detectors include a robustness component, but they are usually trained on smaller datasets or only a limited range of attacks [20,21]. In comparison, our benchmark includes 400,000 original URLs and more than 1.5 million adversarial variants created through both rule-based and gradient-based methods. Remaining errors come mainly from unusually long URLs and rare Unicode sequences, where all tested models show uncertainty, and from benign URLs that share structural patterns with phishing samples. These limitations suggest that character-level defenses should be combined with host-based signals, certificate features, or behavioural cues to further reduce evasion.

4. Conclusions

Our study examined how adversarial changes affect phishing URL classification and evaluated a defense that combines character-level adversarial training with a distribution-based regularizer. The defended model keeps higher accuracy and AUC than the baseline under rule-based edits, homoglyph substitutions, and gradient-driven attacks. These results show that strengthening the training process helps keep the internal features of benign and phishing URLs apart, even when attackers alter the input. The method provides a practical way to improve URL filtering in systems that may face adaptive attacks. However, the model still has difficulty with very long URLs, uncommon Unicode characters, and benign samples that resemble phishing structures. Future work should include host-level or certificate-level information, evaluate larger adversarial datasets, and explore additional ways to reduce errors in these edge cases.

References

1. Saka, T., Vaniea, K., & Kökciyan, N. (2025). SoK: Grouping Spam and Phishing Email Threats for Smarter Security. IEEE Access.
2. Kailas, S., & Roopalakshmi, R. (2025). 'Think Before You Click'-Malicious URL Detection in Cybersecurity: A Systematic Review and Research Roadmap. IEEE Access.
3. Alshingiti, Z., Alaqel, R., Al-Muhtadi, J., Haq, Q. E. U., Saleem, K., & Faheem, M. H. (2023). A deep learning-based phishing detection system using CNN, LSTM, and LSTM-CNN. Electronics, 12(1), 232.
4. Bai, W. (2020, August). Phishing website detection based on machine learning algorithm. In 2020 International Conference on Computing and Data Science (CDS) (pp. 293-298). ieee.

 doi:10.20944/preprints202601.1400.v1

5. Sahoo, D., Liu, C., & Hoi, S. C. (2017). Malicious URL detection using machine learning: A survey. arXiv preprint arXiv:1701.07179.
6. Luo, D., Gu, J., Qin, F., Wang, G., & Yao, L. (2020, October). E-seed: Shape-changing interfaces that self drill. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (pp. 45-57).
7. Su, X. Vision Recognition and Positioning Optimization of Industrial Robots Based on Deep Learning.
8. Bharati, R., Bharti, J., Dehalwar, V., & Kishore, J. (2025). Design of an Iterative Cross-Modal and Context-Aware Deep Analytical Framework for Hate Speech and Fake Post Detection on Social Media Sets.
9. Feng, H. (2024, October). Design of Intelligent Charging System for Portable Electronic Devices Based on Internet of Things (IoT). In 2024 5th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE) (pp. 568-571). IEEE.
10. Bipasha, S. (2025). Literature Survey of Image Forgery Detection Using Machine Learning.
11. Chen, H., Ning, P., Li, J., & Mao, Y. (2025). Energy Consumption Analysis and Optimization of Speech Algorithms for Intelligent Terminals.
12. Christensen, H., Amato, N., Yanco, H., Mataric, M., Choset, H., Drobnis, A., ... & Sukhatme, G. (2021). A roadmap for us robotics–from internet to robotics 2020 edition. Foundations and Trends® in Robotics, 8(4), 307-424.
13. Hu, W. (2025, September). Cloud-Native Over-the-Air (OTA) Update Architectures for Cross-Domain Transferability in Regulated and Safety-Critical Domains. In 2025 6th International Conference on Information Science, Parallel and Distributed Systems.
14. Reynolds, J., Bates, A., & Bailey, M. (2022, September). Equivocal urls: Understanding the fragmented space of url parser implementations. In European Symposium on Research in Computer Security (pp. 166-185). Cham: Springer Nature Switzerland.
15. Tan, L., Liu, X., Liu, D., Liu, S., Wu, W., & Jiang, H. (2024, December). An Improved Dung Beetle Optimizer for Random Forest Optimization. In 2024 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC) (pp. 1192-1196). IEEE.
16. Prakash, C. D., & Karam, L. J. (2021). It GAN do better: GAN-based detection of objects on images with varying quality. IEEE Transactions on Image Processing, 30, 9220-9230.
17. Wang, Y., & Sayil, S. (2024, July). Soft Error Evaluation and Mitigation in Gate Diffusion Input Circuits. In 2024 IEEE 6th International Conference on Power, Intelligent Computing and Systems (ICPICS) (pp. 121-128). IEEE.
18. Shahriar, S. (2025). Linguistic Deception Detection–Models, Domains, Behaviors, Stylistic Patterns to Large Language Models (LLMs) (Doctoral dissertation).
19. Yang, M., Wu, J., Tong, L., & Shi, J. (2025). Design of Advertisement Creative Optimization and Performance Enhancement System Based on Multimodal Deep Learning.
20. Moghaddam, P. S., Vaziri, A., Khatami, S. S., Hernando-Gallego, F., & Martín, D. (2025). Generative Adversarial and Transformer Network Synergy for Robust Intrusion Detection in IoT Environments. Future Internet, 17(6), 258.
21. Wu, C., & Chen, H. (2025). Research on system service convergence architecture for AR/VR system.