

A Hybrid Approach to Phishing URL Detection using Bidirectional LSTM and Evolutionary–Spider Wasp Optimization

Indu Singh*, Ayush Srivastava*, Arjun Vir Vashisht*, Karan*, Md Anas*

*Department of Computer Science and Engineering, Delhi Technological University, New Delhi, India
indusingh@dtu.ac.in, srivastavaayush.1611@gmail.com, arjunvirvashisht@gmail.com,
karan1611177@gmail.com, md02anas02@gmail.com

Abstract—Phishing attacks are a major cybersecurity threat, often relying on misleading URLs to deceive users into sharing confidential information. This research introduces an innovative hybrid approach for detecting phishing URLs, combining deep learning techniques for extracting temporal features with advanced metaheuristic optimization strategies. The proposed method initiates with a robust multi-stage feature selection strategy, which includes mutual information filtering, recursive feature elimination, model-driven importance ranking, and pruning based on feature correlation. This ensures that only the most significant and non-overlapping features are chosen for the subsequent learning process. A Bidirectional Long Short-Term Memory (BiLSTM) network is then applied to extract deep temporal representations from the preprocessed feature vectors. These temporal embeddings are subsequently fed into a Support Vector Machine (SVM) classifier to perform the final prediction. To improve classification accuracy, we introduce a hybrid metaheuristic approach, hEVOSWO, that combines Evolutionary Optimization (EVO) and Spider Wasp Optimization (SWO). This approach simultaneously optimizes feature selection and SVM hyperparameters, enabling efficient exploration and fine-tuning within the search space. Experimental results on the benchmark ISCX-URL2016 dataset demonstrate that our proposed approach achieves an accuracy of 97.31%, a precision of 97.42%, a recall of 97.18%, and an F1-score of 97.30%. The findings confirm the efficiency and robustness of the proposed framework, highlighting its potential for advancing resilient phishing detection systems.

Index Terms—Phishing Detection, Bi-LSTM, Metaheuristic Optimization, Evolutionary Algorithm, Spider-Wasp Algorithm

I. INTRODUCTION

Phishing attacks are still one of the most prevailing and damaging type of cybercrime in today's digitally connected world. These attacks often exploit social engineering techniques [1] and deceptive Uniform Resource Locators (URLs) to lure unsuspecting users into disclosing sensitive information such as login details, financial details, or personal information [2]. According to the Anti-Phishing Working Group (APWG) Q4 2023 report [3], over 5 million phishing attacks were recorded globally in 2023 alone, highlighting the severity of this threat.

The 2021 Cost of Phishing Study by the Ponemon Institute, sponsored by Proofpoint, found that the average annual cost of phishing rose from \$3.8 million in 2015 to \$14.8 million

in 2021 [4]. Though total losses weren't specified, the average cost per incident was about \$136. IBM's 2021 Cost of a Data Breach Report, suggested that the global average breach cost was about \$4.24 million, with breaches involving 50–65 million records averaging \$401 million. While these differ from the \$550 million figure mentioned, they highlight the severe financial impact of large-scale breaches.

Traditional approaches to phishing URL detection have included rule-based systems, blacklists [5], and basic machine learning models [6]. While these methods offer some degree of effectiveness, they often fall short in generalizing to novel or obfuscated attacks. Recently, deep learning models such as CNNs and RNNs [7] have shown promise in automatically extracting relevant patterns from URL strings. However, these models often rely on raw features and require careful tuning to achieve optimal performance.

This paper introduces a hybrid learning model that uses BiLSTM for feature representation and SVM [8] for classification, leveraging BiLSTM's capability to capture sequential URL patterns and SVM's strength in high-margin decision boundaries. The approach is further enhanced by a 4-phase feature selection pipeline and a novel optimization method combining evolutionary Optimization algorithms [9] with Spider Wasp Optimization [10] enabling effective noise reduction, robust parameter tuning, and improved overall detection performance.

The main contributions of this research are mentioned below:

- A multi-phase feature selection strategy combining mutual information, recursive feature elimination, model-based importance, and correlation filtering is used to ensure relevance and avoid multicollinearity.
- A novel hybrid metaheuristic algorithm hEVOSWO integrating Evolutionary Optimization (EVO) and Spider Wasp Optimization (SWO) is proposed to concurrently optimize feature subset selection and SVM hyperparameter tuning.
- BiLSTM-based deep learning architecture is employed to extract sequential and contextual features from URLs, improving the detection accuracy of phishing attempts due to its strong sequence-learning capability.

Section II reviews studies done by researchers on detecting phishing URL, Section III elaborates the methodology we have proposed. Section IV shows our experimental results. At last, Section V deals with the conclusions and proposes further exploration for future research.

II. LITERATURE REVIEW

Sohaib et al. (2023) [11] conducted a systematic literature review of phishing website detection techniques, analyzing 80 studies across five years. Their work categorizes detection approaches into list-based, visual similarity, heuristic, and machine learning techniques, identifying Random Forest as the most used algorithm.

Sahingoz et al. (2019) [12] addressed the weaknesses of blacklist-based methods and emphasized ML's capability in detecting previously unseen URLs. Their analysis covered several classification models and discussed important challenges like feature extraction and adversarial evasion.

Chen and Omote (2022) [13] proposed a modular three-step framework—segmentation, embedding, and machine learning—to detect malicious URLs. Their approach emphasizes context-aware embeddings, demonstrating improved detection accuracy over traditional methods.

Rani et al. (2024) [14] proposed a machine learning-based phishing detection system that leverages URL features to enhance detection accuracy achieving a notable accuracy rate of 96.7%. This study underscores the potential of combining URL feature analysis with machine learning techniques to improve phishing detection mechanisms.

Gopal et al. (2023) [15] introduced an autoencoder-based deep neural network (DNN) architecture aimed at identifying and mitigating phishing URL attacks within IoT environments. Their model operates at the network layer, effectively detecting and blocking phishing websites by leveraging the capabilities of autoencoders combined with DNNs. This approach addresses the unique security challenges posed by IoT devices, offering a robust solution for enhancing cybersecurity in IoT networks.

A.K. Putri et al. (2024) [16] have developed a phishing detection model using the Random Forest algorithm, achieving high performance with an accuracy of 93.7% and an F1-score of 94.4%. The model was trained on a dataset of over 11,000 URLs containing 32 phishing-relevant attributes. Compared to SVM and KNN, Random Forest showed superior adaptability and precision in classifying malicious URLs. This work highlights the algorithm's robustness in identifying phishing threats through data-driven classification techniques.

Jishnu K S et al. (2023) [17] have proposed a deep learning-based phishing URL detection model that leverages RoBERTa for contextual feature extraction and LSTM for sequential classification. Their approach effectively captures both the structural and semantic patterns in URLs. Using a balanced dataset having 300,000 URLs, the model achieved a high accuracy of 97.14%, outperforming conventional machine learning

methods depicting the strength of combining transformer-based embeddings with recurrent networks.

III. PROPOSED METHODOLOGY

This section outlines the proposed model, as depicted in Figure 1, designed to detect phishing URLs by leveraging deep sequential learning and hybrid metaheuristic optimization. The proposed system integrates data preprocessing, a hybrid feature selection pipeline, deep feature extraction [18] using BiLSTM [19], and a novel hybrid optimization technique hEVOSWO based on Evolutionary Optimization (EVO) and Spider Wasp Optimization (SWO). BiLSTM is employed for its strong ability to capture sequential and contextual patterns within URLs, while hEVOSWO is adopted to achieve efficient search-space exploration and more reliable parameter tuning. The final SVM classifier benefits from both fine-tuned hyperparameters and a reduced, highly discriminative feature set, resulting in improved classification stability and accuracy.

A. Data Preprocessing

The raw phishing dataset is first subjected to meticulous preprocessing. Infinite values are replaced with NaNs to maintain numerical integrity, and any feature with more than 30% missing values is discarded entirely. The remaining rows containing NaNs are also removed to ensure no missing values are left in the final training matrix. The target labels are encoded into binary classes (legitimate vs. phishing), and numerical features are standardized using a z-score normalization approach via StandardScaler.

B. Multi-phase Voting Feature Selection Framework

We propose a multi-phase feature selection pipeline that unifies filter, wrapper, and embedded strategies to obtain a compact yet informative feature subset. As mentioned in Algorithm 1, in the initial stage, mutual information between individual features and the target class is calculated, and only those exceeding the 70th percentile are retained. The second stage uses Recursive Feature Elimination (RFE) [21] with a lightweight Random Forest estimator to recursively eliminate weak features based on predictive performance. Next, a model-based selection is conducted using SelectFromModel coupled with a deeper Random Forest ($n=50$ estimators), which ranks features based on importance scores learned during training. Finally, a correlation-based pruning stage removes redundant features with inter-feature correlation coefficients exceeding 0.8, retaining the more informative one using mutual information scores as tie-breakers. The retained features are those selected by at least three of the four strategies, using a majority voting mechanism. In rare cases where the selection becomes too sparse, a fallback mechanism selects the top 20% features based on mutual information.

C. BiLSTM-based feature extraction

Following feature selection, the retained feature vectors are reshaped into a sequential format to exploit temporal relationships using a stacked Bidirectional Long Short-Term Memory

Algorithm 1 Hybrid Multi-Stage Feature Selection Algorithm**Require:**

$X \in R^{n \times d}$: Input feature matrix with n samples and d features
 $y \in \{0, 1\}^n$: Binary target labels (0 = legitimate, 1 = phishing)
 $MI \in R^d$: Mutual information scores vector
 $\mathcal{F}_{MI}, \mathcal{F}_{RFE}, \mathcal{F}_{SFM}, \mathcal{F}_{Corr} \subseteq \{1, \dots, d\}$: Feature subsets from each stage
 $\mathcal{C} \in R^{d \times d}$: Absolute Pearson correlation matrix
 τ_{MI} : 70th percentile MI threshold (scalar)

Ensure:

$\mathcal{F}^* \subseteq \{1, \dots, d\}$: Final set of selected feature indices
1: Compute MI scores $MI = \text{mutual_info_classif}(X, y)$
2: Let $\mathcal{F}_{MI} \leftarrow \{j \mid MI_j \geq \text{70th percentile of } MI\}$
3: Apply Recursive Feature Elimination (RFE) with Random Forest
4: Let $\mathcal{F}_{RFE} \leftarrow$ top 40% features selected by RFE
5: Train Random Forest and apply SelectFromModel (SFM) with median threshold
6: Let $\mathcal{F}_{SFM} \leftarrow$ features with importance $\geq$ median
7: Compute absolute Pearson correlation matrix $\mathcal{C} \in R^{d \times d}$
8: Initialize $\mathcal{F}_{Corr} \leftarrow$ all features
9: **for all** i, j such that $|\mathcal{C}_{ij}| > 0.8$ and $i < j$ **do**
10: Remove the feature with lower mutual information from $\mathcal{F}_{Corr}$
11: **end for**
12: Count inclusion of each feature across $\{\mathcal{F}_{MI}, \mathcal{F}_{RFE}, \mathcal{F}_{SFM}, \mathcal{F}_{Corr}\}$
13: Let $\mathcal{F}^* \leftarrow$ {features included in at least 3 of 4 sets}
14: **if** $|\mathcal{F}^*| < 5$ **then**
15: $\mathcal{F}^* \leftarrow$ top 20% features from MI
16: **end if**
17: **return** $\mathcal{F}^*$

(BiLSTM) network. This framework is initialized with 160 BiLSTM units in the first layer, followed by Layer Normalization and a dropout of 0.25 to mitigate overfitting. A second BiLSTM layer with 80 units and additional normalization follows. The sequence output is then passed through a dense fully connected layer with 256 ReLU-activated neurons and a dropout of 0.3 to further regularize the learning. A sigmoid-activated output layer performs binary classification. The network is trained with binary cross-entropy loss, optimized using Adam (learning rate 0.0004) and gradient clipping. Early stopping with a patience of five epochs halts training when validation performance degrades.

D. Novel Hybrid Evolutionary-Spider Wasp Optimization (hEVOSWO)

The core innovation of hEVOSWO lies in a novel hybrid metaheuristic optimization algorithm that combines the exploratory strength of EVO with the fine-tuning capabilities of SWO. The objective is to jointly select the most relevant BiLSTM feature embeddings and the optimal SVM hyperpa-

Algorithm 2 Hybrid Evolutionary-Spider Wasp optimization (hEVOSWO) Algorithm**Require:**

$X \in R^{n \times d}$: Feature matrix (n samples $\times$ d features)
 $y \in \{0, 1\}^n$: Binary labels (0=legitimate, 1=phishing)
 P : Population size, T : Max iterations
 $\alpha \in (0, 1]$: Learning rate, $\delta \sim \mathcal{N}(0, \sigma^2)$: Noise

Ensure:

$f^* \in \{0, 1\}^d$: Optimal feature mask
 (C^*, γ^*) : Optimal SVM parameters ($C \in [0.1, 50]$, $\gamma \in [10^{-4}, 1]$)
1: **Initialize:** Population $\mathcal{F} \leftarrow \{f_1, \dots, f_P\}$, $f_i \in \{0, 1\}^d$ (random binary masks)
2: Parameters $\mathcal{S} \leftarrow \{(C_i, \gamma_i)\}$ with $C_i \sim \mathcal{U}(0.1, 50)$, $\gamma_i \sim \mathcal{U}(10^{-4}, 1)$
3: $(f^*, C^*, \gamma^*, \text{bestScore}) \leftarrow (\mathbf{0}_d, 0, 0, -\infty)$
4: **for** $t \leftarrow 1$ to T **do**
5: **for** $i \leftarrow 1$ to P **do**
6: **if** $\sum f_i = 0$ **then** $f_i[\text{rand}(1, d)] \leftarrow 1$
7: **end if**
8: Train SVM(C_i, γ_i) on $X[:, f_i = 1]$, evaluate s_i (F1-score)
9: **if** $s_i > \text{bestScore}$ **then** Update $(f^*, C^*, \gamma^*, \text{bestScore})$
10: **end if**
11: **end for**
12: $e \leftarrow \arg \max s_i$ $\triangleright$ Elite solution index
13: **for** $i \in \{1..P\} \setminus e$ **do**
14: **for** $j \leftarrow 1$ to d **do**
15: **if** $f_i[j] \neq f_e[j]$ and $\text{rand}() < (1 - t/T)$ **then**
16: $f_i[j] \leftarrow f_e[j]$ $\triangleright$ SWO feature inheritance
17: **end if**
18: **end for**
19: $C_i \leftarrow \text{clip}(C_i + \alpha(C_e - C_i) + \delta, 0.1, 50)$
20: $\gamma_i \leftarrow \text{clip}(\gamma_i + \alpha(\gamma_e - \gamma_i) + \delta, 10^{-4}, 1)$
21: **end for**
22: **return** (f^*, C^*, γ^*)

rameters [22] : the penalty parameter C and the RBF kernel [23] parameter γ . Each candidate solution in the population represents a binary mask over the BiLSTM features and a pair of real-valued hyperparameters (C, γ) , where $C \in [0.1, 50]$ and $\gamma \in [10^{-4}, 1]$.

The optimization process is governed by three key mathematical principles:

$$f_i^{(t+1)}[j] = \begin{cases} f_e^{(t)}[j] & \text{with probability } 1 - \frac{t}{T} \\ f_i^{(t)}[j] & \text{otherwise} \end{cases} \quad (1)$$

$$f_i^{(t+1)} = f_i^{(t)} \oplus \text{Mutation}(p_m), \quad p_m = 0.1 \times \left(1 - \frac{t}{T}\right) \quad (2)$$

$$\delta^{(t)} = \mathcal{N}(0, \sigma_t^2), \quad \sigma_t = \sigma_0 \times e^{-\lambda t} \quad (3)$$

where $\sigma_0 = 0.2$ is the initial noise magnitude and $\lambda = 0.01$ controls the decay rate.

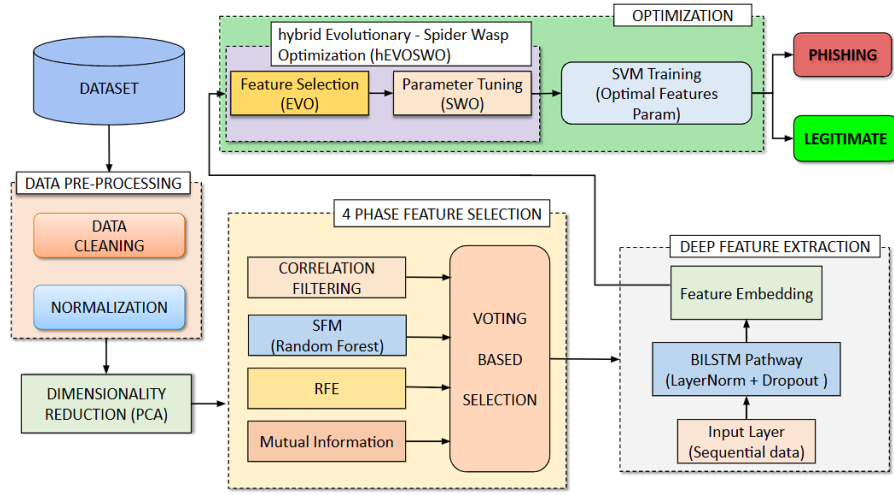


Fig. 1. Model Architecture

The population is initialized with binary vectors where each bit denotes whether a feature is included or not, and random values are assigned to C and γ . During each of the 100 iterations, a dynamic sample of up to 1000 training examples is selected to evaluate the fitness of each individual via F1-score. If a binary mask ends up with no active features, a random feature is activated to preserve model viability. After each iteration, the elite individual is retained based on the best fitness, and other individuals undergo adaptive updates. EVO introduces diversity through Eq. 2 using traditional genetic-style perturbations. Meanwhile, SWO imposes localized refinements via Eq. 1, probabilistically updating feature masks towards the elite. The complete parameter update rules are:

$$C_i^{(t+1)} = \text{clip}(C_i^{(t)} + \alpha(C_e^{(t)} - C_i^{(t)}) + \delta^{(t)}, 0.1, 50) \quad (4)$$

$$\gamma_i^{(t+1)} = \text{clip}(\gamma_i^{(t)} + \alpha(\gamma_e^{(t)} - \gamma_i^{(t)}) + \delta^{(t)}, 10^{-4}, 1) \quad (5)$$

Here, α is a learning coefficient and $\delta^{(t)}$ is the adaptive Gaussian noise from Eq. 3 that ensures stochastic search behavior. This hybridization enables the optimizer to simultaneously navigate large-scale solution spaces through EVO's global exploration and refine around promising regions via SWO's local optimization capabilities.

E. Classifier Integration

The best feature mask and hyperparameters identified by the hybrid optimizer are used to retrain a final SVM model with RBF kernel on the full training set. This optimized classifier is then evaluated on a stratified holdout test set comprising 30% of the dataset. The training history of the BiLSTM model, along with visual tools such as the confusion matrix, ROC curve, and precision-recall curve, are utilized to analyze learning behavior and validate model generalizability.

IV. EXPERIMENTATION AND RESULTS

A. Dataset

The dataset used in this study is derived from a labeled collection of URLs compiled from the **ISCX-URL2016** bench-

mark [20]. It comprises a total of 11,055 samples, with each sample represented by a wide range of numerically encoded lexical and structural features extracted from the URLs [14]. The target label, `URL_Type_obf_Type`, identifies each URL as either phishing or legitimate. The dataset captures diverse URL characteristics, including length-based metrics, special character counts, digit ratios, entropy, and more and offers a balanced yet challenging environment for phishing detection research, with over 5,500 phishing URLs and 5,500 legitimate URLs. For model development and evaluation, the dataset was partitioned using a 70%–30% train–test split. Its rich feature representation supports the development and benchmarking of advanced machine learning and deep learning models for early phishing detection using only URL-level data.

B. Performance Evaluation

In order to evaluate the effectiveness of the proposed framework, a series of experiments were conducted on the benchmark **ISCX-URL2016** dataset. The model was assessed based on multiple standard performance metrics, including accuracy, precision, recall, F1-score, ROC AUC, and average precision (PR AUC). Additionally, visual analyses such as the confusion matrix, ROC curve, and precision-recall curve were employed to gain deeper insights into the classifier's predictive behavior.

1) Performance Metrics: The final Support Vector Machine (SVM) classifier was trained using the feature subset and hyperparameters optimized by the **hEVOSWO** algorithm. Evaluation on the hidden test data yielded the following results:

TABLE I
COMPARISON OF PROPOSED APPROACH WITH EXISTING METHODS

Author	Methodology	Accuracy	Recall	Precision	F1 Score
Jishnu. K. S et al. (2023) [17]	RoBERTa + LSTM	0.9714	0.9711	0.9716	0.9714
Ujah-Ogbuagu et al. (2023) [24]	CNN + LSTM	0.9680	0.9700	0.9600	0.9700
S.B. Gopal et al. (2023) [15]	Auto-Encoder Based Model	0.8608	0.8468	0.9095	0.8760
A. K. Putri et al. (2024) [16]	RF Model	0.9370	0.9320	0.9560	0.9440
M. Irsan et al. (2024) [25]	KNN Model	0.9500	0.9500	0.9500	0.9500
Our Proposed Approach	BiLSTM with hEVOSWO	0.9731	0.9718	0.9742	0.9730

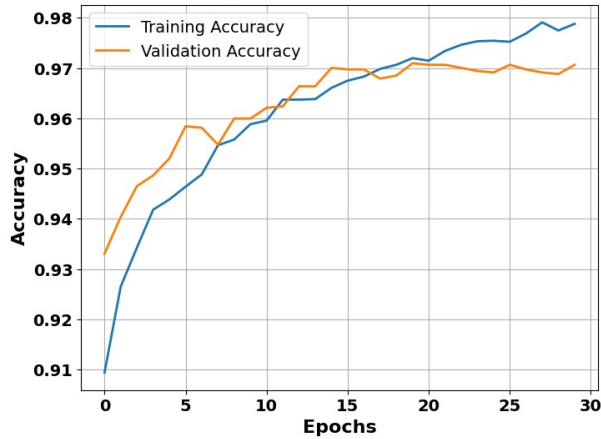


Fig. 2. Training And Validation Accuracy

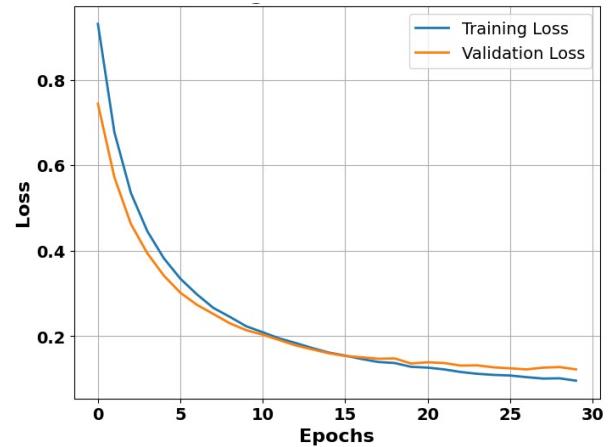


Fig. 3. Training And Validation Loss

TABLE II
PERFORMANCE METRICS OF THE OPTIMIZED SVM CLASSIFIER

Metric	Value
Accuracy	0.9731
Precision	0.9742
Recall	0.9718
F1 Score	0.9730
ROC AUC	0.9867
Average Precision (PR AUC)	0.9772

TABLE III
TRAINING AND VALIDATION ACCURACY LOG (EVERY 5 EPOCHS)

Epoch	Training Accuracy	Validation Accuracy
1	0.8873	0.9330
6	0.9447	0.9584
11	0.9601	0.9621
16	0.9694	0.9697
21	0.9726	0.9707
26	0.9754	0.9707
30	0.9796	0.9707

Figures 2–4 illustrate the training and validation behavior and evaluation metrics from a single hold-out split (70%-30% training-testing), representing results from individual folds.

The training and validation accuracy curves, shown in Figure 2, indicate stable convergence, with validation accuracy closely following training accuracy throughout the epochs. This suggests that the model generalizes well to unseen data and does not suffer from significant overfitting.

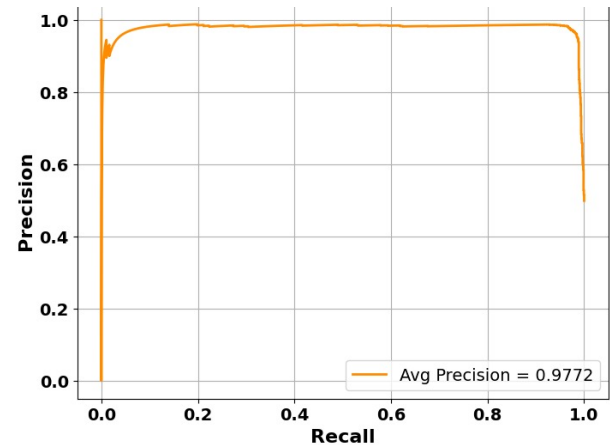


Fig. 4. Precision Recall Curve

The Training And Validation loss curve shown in Figure 3, illustrates a steady decline in loss, followed by stabilization, further validating the effectiveness of the early stopping mechanism employed during training. The Precision-Recall (PR) curve shown in Figure 4 maintains high precision across a broad range of recall values, suggesting the classifier performs robustly even when class imbalance is present.

The Receiver Operating Characteristic (ROC) curve, displayed in Figure 5, demonstrates excellent performance with an AUC value close to 1. The sharply rising curve towards the top-left corner indicates a high true positive rate and low

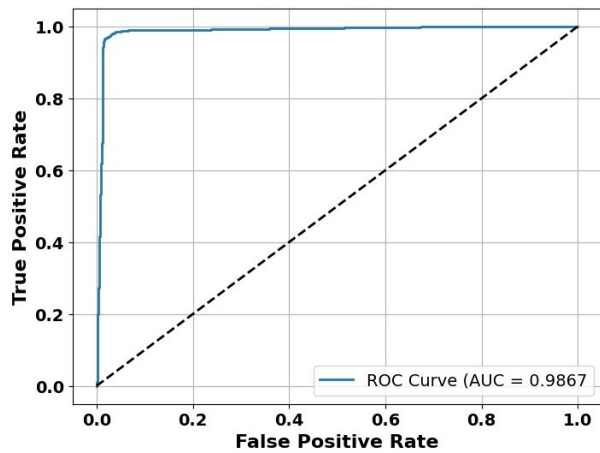


Fig. 5. ROC Curve

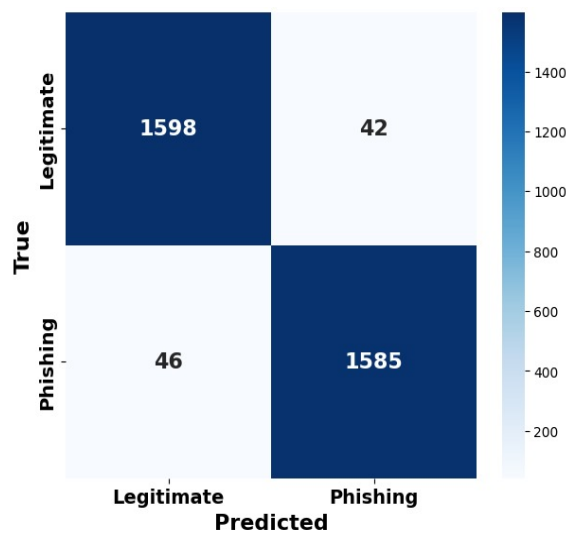


Fig. 6. Confusion Matrix

false positive rate.

The confusion matrix shown in Figure 6 demonstrates that the model is able to distinguish between phishing and legitimate URLs with high accuracy. Most predictions fall on the diagonal, indicating a low false positive and false negative rate.

These results underscore the efficacy of the proposed ensemble feature selection and hyperparameter optimization strategy in enhancing phishing detection performance based solely on URL-level features.

V. CONCLUSION AND FUTURE WORK

This paper presented a hybrid framework for phishing URL detection that combines deep temporal feature learning with metaheuristic optimization. The approach integrates a four-stage feature selection process combining mutual information, recursive feature elimination, model-based importance, and correlation filtering, a BiLSTM network for deep embedding extraction, and a SVM classifier optimized through a

custom Evolutionary-Spider Wasp Optimization hEVOSWO algorithm. The hybrid optimization mechanism effectively performs both feature subset refinement and SVM hyperparameter tuning, ensuring that the classifier operates on a compact, highly discriminative feature space. Experimental results on the ISCX-URL2016 dataset demonstrate outstanding performance across all evaluation metrics, achieving 97.31% accuracy, 97.42% precision, and 97.18% recall, with an F1-score of 97.30%. The model's strong discrimination capability is evidenced by a ROC AUC of 0.9867, while its superior ranking performance is reflected in the average precision (PR AUC) of 0.9772. These results indicate robust generalization and computational efficiency, making the approach particularly suitable for practical deployment in real-world cybersecurity applications. Future work may include real-time deployment as a browser extension or API, support for multilingual or obfuscated URLs, integration with transformer-based encoders, and evaluation under adversarial phishing scenarios to assess robustness.

REFERENCES

- [1] Arachchilage NAG, Love S. Security awareness of computer users: A phishing threat avoidance perspective. *Computers in Human Behavior* 2014; 38: 304–312.
- [2] M. Sanchez-Paniagua, E. F. Fern ´andez, E. Alegre, W. Al-Nabki, and ´ V. Gonzalez-Castro, "Phishing url detection: A real-case scenario through ´ login urls," *IEEE Access*, vol. 10, pp. 42 949–42 960, 2022.
- [3] Aron, G. (2024) Phishing activity trends report, 4th quarter 2023, Anti Phishing Working Group (APWG). Available at: https://docs.apwg.org/reports/apwg_trends_report_q4_2023.pdf
- [4] Sulli, B. (2021) Sullivan Privacy Report, Ponemon. Available at: <https://ponemonsullivanreport.com/2021/08>.
- [5] J. Ma, L. K. Saul, S. Savage, S. Savage, and G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious urls," *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, p. 1245–1254, 2009.
- [6] M. Aljabri et al., "Detecting Malicious URLs Using Machine Learning Techniques: Review and Research Directions," in *IEEE Access*, vol. 10, pp. 121395–121417, 2022, doi: 10.1109/ACCESS.2022.3222307.
- [7] Y. Wang, W. Zhu, H. Xu, Z. Qin, K. Ren, and W. Ma, "A large-scale pretrained deep model for phishing url detection," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [8] Q. Wang, "Support Vector Machine Algorithm in Machine Learning," *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, Dalian, China, 2022, pp. 750–756, doi: 10.1109/ICAICA54878.2022.9844516.
- [9] Bartz-Beielstein, Thomas Branke, Juergen Mehnen, Jorn Mersmann, Olaf. (2014). *Evolutionary Algorithms*. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 4. 10.1002/widm.1124.
- [10] Abdel-Basset, M., Mohamed, R., Jameel, M. et al. Spider wasp optimizer: a novel meta-heuristic optimization algorithm. *Artif Intell Rev* 56, 11675–11738 (2023). <https://doi.org/10.1007/s10462-023-10446-y>
- [11] M. Sohaib, N. Anwar, M. Shoaib, A. Ali, M. A. Baig, and F. Iqbal, "Phishing website detection using machine learning: A systematic literature review," *Arabian Journal for Science and Engineering*, vol. 49, no. 2, pp. 1343–1360, 2023.
- [12] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.
- [13] Q. Chen and K. Omote, "A Three-Step Framework for Detecting Malicious URLs," *2022 International Symposium on Networks, Computers and Communications (ISNCC)*, Shenzhen, China, 2022, pp. 1–6, doi: 10.1109/ISNCC55209.2022.9851734.

- [14] R. S. Rani, N. Silpa, G. N. Satish, N. Amrutha and G. N. Reddy, "Optimizing Phishing Detection: Leveraging URL Features with Machine Learning," 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2024, pp. 2094-2099, doi: 10.1109/ICACCS60874.2024.10717158.
- [15] S B, Gopal Poongodi, C. Nanthiya, D. Kirubakaran, T. Kulavishnusan-avan, B. Logeshwar, D.. (2023). Autoencoder-Based Architecture for Identification and Mitigating Phishing URL Attack in IoT Using DNN. Journal of The Institution of Engineers (India): Series B. 104. 10.1007/s40031-023-00934-8.
- [16] A. K. Putri, J. Wiratama, S. A. Sanjaya, S. F. Wijaya, M. E. Johan and A. Faza, "Web URLs Phishing Detection Model with Random Forest Algorithm," 2024 5th International Conference on Big Data Analytics and Practices (IBDAP), Bangkok, Thailand, 2024, pp. 1-5, doi: 10.1109/IBDAP62940.2024.10689685.
- [17] J. K. S and A. B, "Phishing URL detection by leveraging RoBERTa for feature extraction and LSTM for classification," 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2023, pp. 972-977, doi: 10.1109/ICAISS58487.2023.10250684.
- [18] K. P. Shroff and H. H. Maheta, "A comparative study of various feature selection techniques in high-dimensional data set to improve classification accuracy," 2015 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2015, pp. 1-6, doi: 10.1109/ICCCI.2015.7218098.
- [19] Schuster, Mike Paliwal, Kuldip. (1997). Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing., 45. 2673 - 2681. 10.1109/78.650093.
- [20] C. I. for Cybersecurity (CIC), "Url dataset (iscx-url2016)," 2016
- [21] X. Zeng, Y. -W. Chen and C. Tao, "Feature Selection Using Recursive Feature Elimination for Handwritten Digit Recognition," 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kyoto, Japan, 2009, pp. 1205-1208, doi: 10.1109/IIH-MSP.2009.145.
- [22] S. Simon, N. Kolyada, C. Akiki, M. Potthast, B. Stein and N. Siegmund, "Exploring Hyperparameter Usage and Tuning in Machine Learning Research," 2023 IEEE/ACM 2nd International Conference on AI Engineering – Software Engineering for AI (CAIN), Melbourne, Australia, 2023, pp. 68-79, doi: 10.1109/CAIN58948.2023.00016.
- [23] S. Han, Cao Qubo and Han Meng, "Parameter selection in SVM with RBF kernel function," World Automation Congress 2012, Puerto Vallarta, Mexico, 2012, pp. 1-4.
- [24] Ujah-Ogbuagu, B.C., Akande, O.N. Ogbuju, E. A hybrid deep learning technique for spoofing website URL detection in real-time applications. Journal of Electrical Systems and Inf Technol 11, 7 (2024). <https://doi.org/10.1186/s43067-023-00128-8>
- [25] M. Irsan, F. Febriana, H. H. Nuha and H. R. Putra Sailellah, "Phishing Detection on URL Data Using K-Nearest Neighbors Method," 2024 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Sakhr, Bahrain, 2024, pp. 792-797, doi: 10.1109/3ict64318.2024.10824630.