JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 11 Issue 01
2026
Research Article
https://doi.org/10.56979/1101/2026/1520

A Union-Driven Feature Selection Framework for Robust Phishing URL Classification

Hani Al-Mimi[1], Ali ALALLAWEE[2*], Waheed Javed[3], Raya Basil Alothman[2], Senan A. M. Alhasan[2], and Zaid Fawaz Jarallah[2]

[1]Department of Cybersecurity, Faculty of Science and Information Technology, Al-Zaytooanh University of Jordan, Amman, Jordan.
[2]Department of Computer Science, College of Education for Pure Sceince, University of Mosul, Mosul, Iraq.
[3]Department of Computer Science, Green International University, Lahore, Pakistan.
Corresponding Author: Ali ALALLAWEE. Email: aliabd@uomosul.edu.iq

Received: March 03, 2026 **Accepted:** May 29, 2026

Abstract: Phishing attacks continue to exploit deceptive URL structures to compromise sensitive information and mislead users. This paper proposes a phishing detection framework that integrates a union-based feature selection strategy combining the Whale Optimization Algorithm (WOA) and the Dragonfly Algorithm (DA). The proposed WOA∪DA approach merges complementary feature subsets selected by both optimizers and removes redundancy to construct an informative and consistent feature space. The optimized features are evaluated using Extra Trees (ET) and K-Nearest Neighbor (KNN) classifiers on the ISCX-URL2016 dataset. Experimental results demonstrate that ET achieved 98.70% accuracy, 98.87% precision, 98.48% recall, and 98.68% F1-score, while KNN achieved 97.46% accuracy with competitive precision and recall values. Comparative analysis against related works shows measurable improvements in overall accuracy. The findings confirm that combining ensemble learning with union-based metaheuristic feature selection enhances classification stability, improves generalization, and strengthens phishing detection performance.

Keywords: Phishing detection; ISCX-URL2016; Whale Optimization Algorithm; Dragonfly Algorithm

1. Introduction

The Internet is an integral part of today's society by enabling communication, online transactions, cloud services, and data exchange [1], [2]. It is widely used across sectors such as healthcare [3], [4], [5], finance [6], [7], [8], education [9], [10], and government [11], [12]. The use of online services has grown, making it increasingly important to ensure that these systems remain secure and reliable [13], [14]. With the expansion of online applications, securing these systems from cybersecurity threats, such as phishing attacks, has become a critical requirement for ensuring operational continuity and user trust [15], [16], [17], [18].

Phishing is one of the most prevalent and harmful threats in today's cybersecurity landscape, using social engineering and deceptive artifacts to gain access to credentials and sensitive data [19]. The FBI Internet Crime Complaint Center (IC3) reported that in 2024, more than 190,000 complaints of phishing and spoofing were reported, making it the most common type of cybercrime complaint reported by victims. In 2024, 25% of all cyberattacks detected in companies worldwide were phishing attacks, including those delivered through attachments or links, as shown in Figure 1. This share decreased from 46 percent in 2022 [20], [21]. Figure 2 shows the phishing attack lifecycle, which describes how a threat actor designs, sends, and uses a phishing message and URL to trick a user and gain access to a network. Most traditional phishing detection methods are based on rule-based filtering and signature matching, both of which have been successful in mitigating

known phishing attacks but must be manually updated on a regular basis. These reactive methods do not work well for polymorphic and obfuscated phishing URLs and are less able to adapt to new attack methods, which are often designed to evade static rules [22], [23]. Besides, phishing datasets may also have high-dimensional data or redundant attributes, which can reduce detection accuracy and increase computational complexity when using traditional techniques [24], [25], [26].

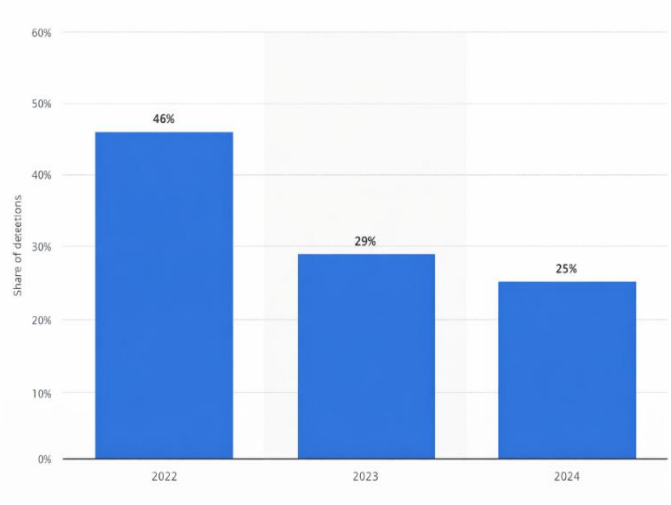

Figure 1. cyberattacks in the companies worldwide

Figure 2. Phishing Attack Lifecycle

Another promising approach for phishing detection is machine learning (ML), which learns discriminative patterns from data and adapts to ever-changing phishing behaviors [27], [28]. In this framework, both Extra Trees (ET) and K-Nearest Neighbors (KNN) classifiers have their own advantages, with ET benefiting from randomized ensemble learning methods that enhance robustness to noise and complex feature interactions,

and KNN benefiting from instance-based classification techniques based on similarity measures. Both of these models constitute a strong foundation for assessing the detection performance of practical phishing data [29], [30].

Feature selection plays a pivotal role in boosting the effectiveness of ML models, minimizing overfitting, and, most significantly, increasing computational efficiency [26, 31, 32]. Meta-heuristic optimization algorithms based on natural processes have been found to be effective in feature subset selection in high-dimensional spaces. Some examples of these algorithms are the Numbat Optimization Algorithm, Modified Mongoose Algorithm, Tuatara Optimization Algorithm, Grey Wolf Optimizer, Whale Optimization Algorithm (WOA), and Dragonfly Algorithm (DA) [33, 34, 35]. The Whale Optimization Algorithm (WOA) and Dragonfly Algorithm (DA) are used in this study for the selection of salient features of phishing and non-phishing instances. The proposed framework combines these optimization methods with the classifiers ET and KNN in order to enhance the detection accuracy and generalization ability.

2. Related works

In [29] the researchers have conducted a comparative study of different ML techniques to find phishing websites. A set of 4,898 phishing and 6,157 legitimate websites were used to evaluate 12 classifiers. The models tested were Support Vector Machine, Logistic Regression, Random Forest, Decision Tree, k-Nearest Neighbor, AdaBoost, Neural Networks, Gradient Boosting and XGBoost. The results indicated that ensemble models (particularly XGBoost and Random Forest) performed better. The authors also found that AdaBoost was less prone to overfitting in a low-noise dataset, and was better suited for tasks with interpretability requirements but not for tasks with time sensitivity, since it was slower to train. In [36] the analysis of webpage links is brought to the fore by introducing an ML-based system for phishing web page identification. There are three steps suggested in the approach. Firstly, Python scripts and GNU Wget were used to automatically collect and download the content of web pages including associated HTML resources. Second, features have been extracted from host attributes, vocabulary characteristics and word-level information. Thirdly, Principal Component Analysis (PCA) is employed to reduce the number of features. The developed feature set was used for training and testing classification models. Experimental results indicated that the Support Vector Machine (SVM) performed best with the accuracy of 95.66%. Yi et al. [37] introduced a phishing detection model using the Deep Belief Networks (DBNs) with the combination of original features and features interaction representation. The model was trained with a small set of data for optimization of model parameters and then tested with real IP flow data from an ISP (Internet Service Provider). This experimental environment provided the opportunity to evaluate under real world network traffic. The results showed that the true positive rate was around 90%, the false positive rate was 0.6%, and the accuracy was 89.6%. The study concludes that DBNs can be effectively used to model complex feature interactions and can aid in reliable phishing detection in large-scale network environments. Alqahtani and Abu-Khadrah [38] suggested a phishing detection system that combines several machine learning classifiers such as Random Forest, Support Vector Machine (SVM) and Bagging. The model was tested with a dataset containing 1,353 URLs which are classified as legitimate, phishing and suspicious from the UCI Machine Learning Repository. The Synthetic Minority Oversampling Technique (SMOTE) was used to deal with class imbalance. The correlation-based filtering approach was applied for feature selection to determine the features which are strongly related to the target class. Experimental results revealed that the accuracy of the ensemble model is 92.33%, and precision, recall and F1 score are all higher than 92%, which is better than the individual classifiers.

3. Phishing attack detection system

This section presents the architecture of the proposed phishing detection system. It outlines the data preprocessing procedures, the feature selection process based on WOA and DA optimization techniques, and the classification stage using the chosen machine learning models. The overall workflow of the framework is depicted in Figure 3.

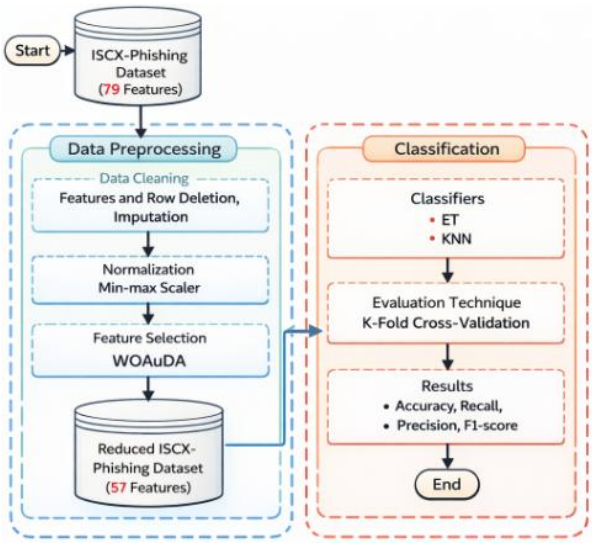

Figure 3. The proposed Phishing detection system

3.1. Phishing dataset

The ISCX-URL2016-phishing (ISCX-Phishing) dataset [39], [40] is a commonly used benchmark for phishing website detection research and was developed by the Canadian Institute for Cybersecurity. It consists of 14,478 labeled URLs, including both phishing and legitimate instances, with each URL described by 79 numerical features. These features capture lexical, domain-based, and URL structural characteristics such as 1) urlLen, 2) Querylength, 3) domainlength, 4) pathLength, 5) subDirLen, 6) fileNameLen, 7) this.fileExtLen, 8) ArgLen, 9) domain_token_count, 10) path_token_count, 11) avgdomaintokenlen, 12) longdomaintokenlen, 13) avgpathtokenlen, 14) tld, 15) NumberofDotsinURL, 16) ISIpAddressInDomainName, 17) executable, 18) isPortEighty, 19) charcompvowels, 20) charcompace, 21) ldl_url, 22) ldl_domain, 23) ldl_path, 24) ldl_filename, 25) ldl_getArg, 26) dld_url, 27) dld_domain, 28) dld_path, 29) dld_filename, 30) dld_getArg, 31) pathurlRatio, 32) ArgUrlRatio, 33) argDomanRatio, 34) domainUrlRatio, 35) pathDomainRatio, 36) argPathRatio, 37) URL_DigitCount, 38) host_DigitCount, 39) Directory_DigitCount, 40) File_name_DigitCount, 41) Extension_DigitCount, 42) Query_DigitCount, 43) URL_Letter_Count, 44) host_letter_count, 45) Directory_LetterCount, 46) Filename_LetterCount, 47) Extension_LetterCount, 48) Query_LetterCount, 49) LongestPathTokenLength, 50) Domain_LongestWordLength, 51) Path_LongestWordLength, 52) sub-Directory_LongestWordLength, 53) Arguments_LongestWordLength, 54) URL_sensitiveWord, 55) URLQueries_variable, 56) spcharUrl, 57) delimeter_Domain, 58) delimeter_path, 59) delimeter_Count, 60) NumberRate_URL, 61) NumberRate_Domain, 62) NumberRate_DirectoryName, 63) NumberRate_FileName, 64) NumberRate_Extension, 65) NumberRate_AfterPath, 66) SymbolCount_URL, 67) SymbolCount_Domain, 68) SymbolCount_Directoryname, 69) SymbolCount_FileName, 70) SymbolCount_Extension, 71) SymbolCount_Afterpath, 72) Entropy_URL, 73) Entropy_Domain, 74) Entropy_DirectoryName, 75) Entropy_Filename, 76) Entropy_Extension, 77) Entropy_Afterpath, 78) CharacterContinuityRate, and 79) LongestVariableValue. Some attributes contain noisy or infinite values and exhibit different scales, requiring data cleaning and normalization prior to model training. The dataset reflects real-world phishing behaviors, particularly URL obfuscation techniques, and supports the evaluation of supervised learning and feature selection methods for early phishing detection [41].

3.2. ISCX-Phishing dataset preparation

Prior to the development of the models, the ISCX-URL2016 dataset was preprocessed cautiously to make the data valid and robust. The first steps consisted of loading the dataset and checking the feature types, ranges, and statistical distributions. This check showed that there were invalid values that would have an adverse impact on learning behavior. Namely, there were missing values in 13) avgpathtokenlen, 36) argPathRatio, 62) NumberRateDirectoryName, 63) NumberRateFileName, 64) NumberRateExtension, 65) NumberRateAfterPath, 74) EntropyDirectoryName, 75) EntropyFilename, and 76) EntropyExtension.

Furthermore, there were infinite values in 77) EntropyAfterpath. All the affected features were eliminated in order to ensure consistency and prevent distortions due to artificial replacement. The dataset was left with 69 valid features after this cleaning process [18], [19].

The next preprocessing step after data cleaning was normalization to prepare the data to be analyzed through machine learning. Data cleaning is intended to eliminate invalid features that had missing or infinite data to guarantee consistency and reliability. After obtaining a clean set of features, Min-Max scaling was used to normalize the rest of the numeric features to a fixed range and to overcome scale discrepancies between attributes, otherwise biasing distance-based and tree-based learning methods [18], [42] . For instance, raw feature vectors such as {48, 80, 5.0833335, 0.025, 54}, {17, 62, 8.166667, 0.032258064, 26}, {54, 3.6923077, 68, 39, 0.5147059}, {44, 69, 4.1666665, 0.028985508, 50}, and {72, 7.6, 93, 0.021505376, 61} were transformed using Min-Max normalization into scaled representations including {0.157894737, 0.138263666, 0.154322353, 0.027443609, 0.154121864}, {0.055921053, 0.080385852, 0.248061343, 0.035411108, 0.053763441}, {0.132616487, 0.112032619, 0.099678457, 0.128289474, 0.565015502}, {0.144736842, 0.102893891, 0.126453996, 0.031818678, 0.139784946}, and {0.218637993, 0.230833627, 0.180064309, 0.023607405, 0.200657895}. This step ensured that all features contributed proportionally during model training.

3.3. Propose union feature selection

Feature selection reduces data dimensionality by eliminating redundant and irrelevant attributes, thereby improving model accuracy, generalization capability, and computational efficiency [43], [44]. This study proposes a union-based feature selection approach that combines features selected by two metaheuristic algorithms, Whale Optimization Algorithm (WOA) and Dragonfly Algorithm (DA), denoted as WOA ∪ DA. The union operation integrates complementary feature subsets identified through different search behaviors, thereby reducing optimizer-dependent bias and lowering the risk of excluding informative features. Since WOA and DA employ distinct exploration mechanisms, their combination enhances feature diversity and robustness. The rationale for integrating WOA and DA within the proposed union-based framework is summarized in Table I.

Table 1. Justification for Combining WOA and DA in Feature Selection

#	Criterion	WOA	DA	Rationale
1	Search Behavior	Focuses on global search using encircling and spiral updates	Focuses on local search using swarm interactions	Combining global and local search improves coverage of the feature space
2	Convergence Style	Converges quickly toward the best solution	Explores multiple regions before convergence	Reduces the risk of premature convergence and selection bias
3	Feature Selection Tendency	Selects compact and high-impact features	Selects more diverse feature subsets	The union captures both strong and diverse features
4	Fitness Compatibility	Uses accuracy and feature size in evaluation	Uses the same fitness criteria	Ensures consistent and fair combination of selected features

5	Complementary Strengths	Captures global URL characteristics	Captures local and structural URL patterns	Improves robustness by covering different phishing characteristics

The ISCX-URL2016 phishing data was initially subjected to the WOA algorithm, and a subset of 43 useful features was chosen. Simultaneously, the DA algorithm was applied to the same dataset, and it determined 46 features. In order to leverage the merits of both optimization methods, a union-based feature selection method, referred to as WOA∪DA, was used. The result of this integration was a total of 57 features that were obtained by eliminating duplicate features to obtain a non-redundant and consistent subset. The resulting unified feature set was used to train and evaluated on the phishing detection model. The process of integration is presented in Figure 4, and the results of the feature subsets of WOA, DA, and their combination are summarized in Table 2.

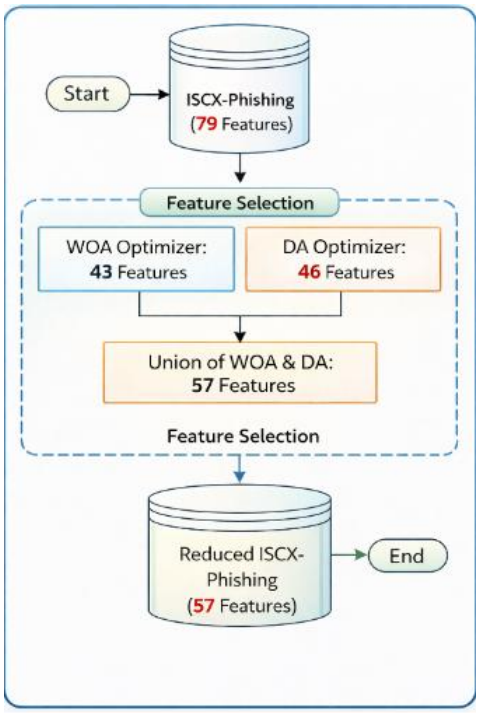

Figure 4. Feature selection union process

Table 2. The selected features by HHO, BA, HHO∩BA, and HHO∪BA methods

Method	Feature set
WOA (43 features)	1) urlLen, 2) Querylength, 14) Tld, 15) NumberofDotsinURL, 16) ISIpAddressInDomainName, 17) executable, 18) isPortEighty, 19) charcompvowels, 20) charcompace, 21) ldl_url, 22) ldl_domain, 24) ldl_filename, 25) ldl_getArg, 26) dld_url, 27) dld_domain, 28) dld_path, 30) dld_getArg, 32) ArgUrlRatio, 33) argDomanRatio, 34) domainUrlRatio, 38) host_DigitCount, 41) Extension_DigitCount, 43) URL_Letter_Count, 44) host_letter_count, 45)

	Directory_LetterCount, 46) Filename_LetterCount, 47) Extension_LetterCount, 49) LongestPathTokenLength, 50) Domain_LongestWordLength, 51) Path_LongestWordLength, 53) Arguments_LongestWordLength, 55) URLQueries_variable, 56) spcharUrl, 57) delimeter_Domain, 60) NumberRate_URL, 61) NumberRate_Domain, 67) SymbolCount_Domain, 69) SymbolCount_FileName, 70) SymbolCount_Extension, 72) Entropy_URL, 73) Entropy_Domain, 78) CharacterContinuityRate, 79) LongestVariableValue.
DA (46 features)	1) urlLen, 3) domainlength, 4) pathLength, 7) fileNameLen, 11) avgdomaintokenlen, 14) tld, 16) ISIpAddressInDomainName, 17) executable, 18) isPortEighty, 19) charcompvowels, 22) ldl_domain, 23) ldl_path, 24) ldl_filename, 26) dld_url, 27) dld_domain, 28) dld_path, 29) dld_filename, 30) dld_getArg, 31) pathurlRatio, 32) ArgUrlRatio, 33) argDomanRatio, 34) domainUrlRatio, 35) pathDomainRatio, 36) argPathRatio, 37) URL_DigitCount, 38) host_DigitCount, 39) Directory_DigitCount, 41) Extension_DigitCount, 42) Query_DigitCount, 44) host_letter_count, 45) Directory_LetterCount, 46) Filename_LetterCount, 50) Domain_LongestWordLength, 51) Path_LongestWordLength, 52) sub-Directory_LongestWordLength, 53) Arguments_LongestWordLength, 54) URL_sensitiveWord, 56) spcharUrl, 60) NumberRate_URL, 66) SymbolCount_URL, 67) SymbolCount_Domain, 69) SymbolCount_FileName, 70) SymbolCount_Extension, 72) Entropy_URL, 73) Entropy_Domain, 79) LongestVariableValue.
WOA∪DA (57 features)	1) urlLen, 2) Querylength, 3) domainlength, 4) pathLength, 7) fileNameLen, 11) avgdomaintokenlen, 14) Tld, 15) NumberofDotsinURL, 16) ISIpAddressInDomainName, 17) executable, 18) isPortEighty, 19) charcompvowels, 20)

	charcompace, 21) ldl_url, 22) ldl_domain, 23) ldl_path, 24) ldl_filename, 25) ldl_getArg, 26) dld_url, 27) dld_domain, 28) dld_path, 29) dld_filename, 30) dld_getArg, 32) ArgUrlRatio, 33) argDomanRatio, 34) domainUrlRatio, 35) pathDomainRatio, 36) argPathRatio, 37) URL_DigitCount, 38) host_DigitCount, 39) Directory_DigitCount, 41) Extension_DigitCount, 42) Query_DigitCount, 43) URL_Letter_Count, 44) host_letter_count, 45) Directory_LetterCount, 46) Filename_LetterCount, 47) Extension_LetterCount, 49) LongestPathTokenLength, 50) Domain_LongestWordLength, 51) Path_LongestWordLength, 52) sub-Directory_LongestWordLength, 53) Arguments_LongestWordLength, 54) URL_sensitiveWord, 55) URLQueries_variable, 56) spcharUrl, 57) delimeter_Domain, 60) NumberRate_URL, 61) NumberRate_Domain, 66) SymbolCount_URL, 67) SymbolCount_Domain, 69) SymbolCount_FileName, 70) SymbolCount_Extension, 72) Entropy_URL, 73) Entropy_Domain, 78) CharacterContinuityRate, 79) LongestVariableValue.

3.4. KNN and ET classifiers

The attack detection model used the ET and KNN classifiers. The ET classifier is an ensemble learning algorithm that builds several random decision trees and integrates the results to enhance classification performance and minimize overfitting. Its capability to process high-dimensional data and interactions among features is the reason why it is applicable in detecting phishing. The KNN classifier, on the other hand, is a distance-based supervised learning algorithm that classifies data based on the nearest training examples in the feature space. KNN is not complex and works well when similar phishing patterns are clustered. Hence, ET and KNN are suitable classifiers for detecting phishing attacks. Table III indicates the key features of the ET and KNN classifiers in terms of their application in phishing attack detection [30], [42], [45].

Table 3. Main aspects of the ET and KNN classifiers

Aspect	Extra Trees (ET)	K-Nearest Neighbor (KNN)
Learning Approach	Ensemble learning (Randomized trees)	Instance-based supervised learning
Working Mechanism	Builds multiple randomized decision trees and averages their outputs	Classifies samples based on the majority label of nearest neighbors

Suitability for Phishing Detection	Handles high-dimensional data and complex feature interactions	Effective when similar phishing patterns cluster in feature space
Computational Complexity	Moderate to high during training	Low training cost but higher prediction cost
Interpretability	Moderate due to tree structure	Low, decisions depend on neighbor distances
Scalability	Suitable for medium to large datasets	Less efficient for very large datasets
Accuracy in Phishing Detection	Generally strong due to ensemble diversity	Depends on choice of k and distance metric
Main Default Hyperparameters	n_estimators = 100, criterion = gini, max_depth = None	n_neighbors = 5, metric = minkowski, weights = uniform

4. Performance evaluation

Four standard classification metrics are used to evaluate the proposed model: accuracy, precision, recall, and F1-score. These measures are calculated based on the confusion matrix, which comprises true positive (TP), true negative (TN), false positive (FP), and false negative (FN). All metrics offer different perspectives on model performance. Table IV gives the definitions and equations of the evaluation metrics in detail [46–51]. The comparative analysis of ET and KNN is shown in Figures 5–8 in terms of four performance measures. The accuracy results are given in Figure 5, where both classifiers performed quite well, with accuracies of 98.70% for ET and 97.46% for KNN, representing a difference of 1.24%. Precision is presented in Figure 6, where ET achieves slightly better performance than KNN (98.87% vs. 98.71%). As can be seen in Figure 7, the recall gap of 2.38% is greater, as ET achieved 98.48% compared to KNN, which achieved 96.10%. Figure 8 shows the F1-score, in which ET outperformed KNN by 1.29% (98.68% vs. 97.39%). Although both classifiers gave good and stable results, ET consistently gave better results across all metrics. The improvements are credited to the proposed WOA∪DA feature selection process based on the union of various feature sets extracted from the filters, which removes redundant features and retains complementary features. The optimized feature space enhances the discriminative power, stability, and robustness of the phishing detection model.

Table 4. Evaluation Metrics

Metric	Definition	Purpose	Strengths	Equation
Accuracy	Ratio of correctly predicted instances to total instances	Measures overall classification performance	Easy to interpret and provides a general performance view	(TP + TN) / (TP + TN + FP + FN)
Precision	Ratio of correctly predicted positive instances to all predicted positives	Evaluates correctness of positive predictions	Reduces false positives in phishing detection	TP / (TP + FP)

Recall	Ratio of correctly predicted positive instances to all actual positives	Measures ability to detect phishing instances	Reduces false negatives in security applications	TP / (TP + FN)
F1-Score	Harmonic mean of precision and recall	Balances precision and recall	Suitable for imbalanced phishing datasets	2 × (Precision × Recall) / (Precision + Recall)

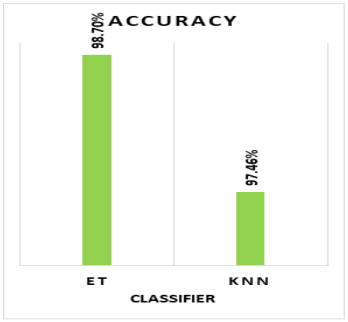

Figure 5. Phishing Detection Accuracy

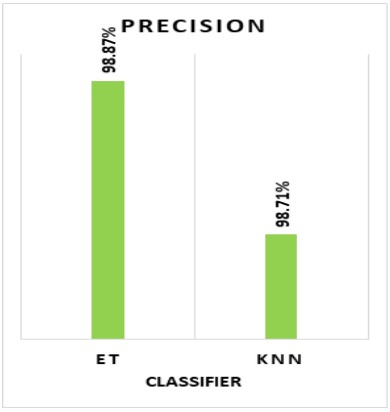

Figure 6. Phishing Detection Precision

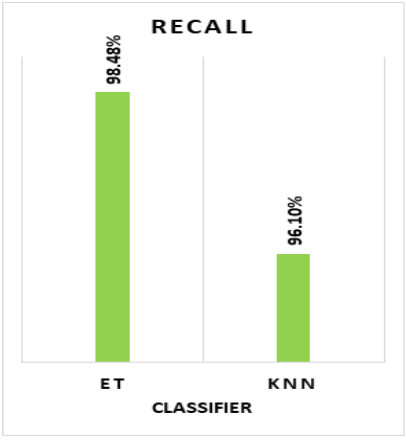

Figure 7. Phishing Detection Recall

Figure 8. Phishing Detection F1-score

Figure 9 presents the accuracy comparison of the proposed ET and KNN classifiers using the WOA∪DA feature selection method against previous studies. The proposed phishing detection model based on ET and WOA∪DA achieved the highest accuracy of 98.70%, followed by KNN with 97.46%. The ET-based model outperformed RW2 (95.66%) by 3.04%, RW3 (89.60%) by 9.10%, and RW4 (92.33%) by 6.37%. Similarly, KNN surpassed RW2 by 1.80%, RW3 by 7.86%, and RW4 by 5.13%. These results demonstrate that integrating ET with the WOA∪DA union-based feature selection method provides valuable improvements over existing approaches. The findings confirm that optimizing the feature space enhances classifier effectiveness and strengthens phishing detection performance.

Figure 9. Comparative Accuracy Analysis of the Proposed Framework and Related Works

5. Conclusion

This study presented a phishing detection framework that integrates ET and KNN classifiers with a WOA∪DA union-based feature selection strategy. The proposed method combines complementary feature subsets generated by WOA and DA while eliminating redundant attributes to form a refined feature space. Experimental evaluation showed that both classifiers achieved high performance, with ET outperforming KNN by achieving 98.70% accuracy, 98.87% precision, 98.48% recall, and 98.68% F1-score, compared to 97.46% accuracy for KNN. The consistent improvement across evaluation metrics demonstrates the effectiveness of the proposed union-based feature optimization approach. Comparative analysis confirmed competitive results relative to existing methods. The findings highlight the value of integrating complementary metaheuristic algorithms to enhance feature relevance, improve model generalization, and strengthen phishing detection capability in supervised learning environments.

References

1. A. S. Hyassat, I. Kraidia, M. M. Abualhaj, I. Qaddara, A. Latif, and A. Dalhoum, "Leveraging Cellular Automata for Strip-Cut Document Reconstruction: A Path Travelling Salesman Problem Formulation," *J. of Soft Computing and Data Mining*, vol. 7, no. 2, pp. 45–57, 2026, doi: 10.30880/jscdm.2026.07.02.004.
2. M. M. Abualhaj, S. Al-Khatib, M. Al-Zyoud, M. O. Hiari, A. Al-Allawee, and M. A. Alsharaiah, "A Customized Machine Learning Model for Improving Malware Detection," *International Journal of Computer Network and Information Security*, vol. 18, no. 1, pp. 1–17, Feb. 2026, doi: 10.5815/ijcnis.2026.01.01.
3. A. A. Abu-Shareha, H. Qutaishat, and A. Al-Khayat, "A framework for diabetes detection using machine learning and data preprocessing," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1654–1667, 2024.
4. Nadeem, M. W., Goh, H. G., Hussain, M., Liew, S. Y., Andonovic, I., & Khan, M. A. (2022). Deep learning for diabetic retinopathy analysis: a review, research challenges, and future directions. *Sensors*, *22*(18), 6780.
5. Zubair, M., Hussain, M., Al-Bashrawi, M. A., Bendechache, M., & Owais, M. (2025). A comprehensive review of techniques, algorithms, advancements, challenges, and clinical applications of multi-modal medical image fusion for improved diagnosis. Computer Methods and Programs in Biomedicine, 109014.
6. F. Rahaman Chowdhury, M. Masum Alam Nahid, F. Chowdhury, M. Masum, U. Cali, and M. Sadek Ferdous, "Self-Sovereign Identity Empowered Automated Teller Machines," IEEE Access, vol. 13, pp. 203444–203480, 2025, doi: 10.1109/ACCESS.2025.3638625.
7. I. Cho, J. Ho Kwak, and B. Gyou Lee, "Mobile Banking Usage Through Biometric Authentication: Effects of Smartphone Attributes and Privacy Consent Index," IEEE Access, vol. 13, pp. 202919–202937, 2025, doi: 10.1109/ACCESS.2025.3631849.
8. Zubair, M., Owais, M., Hassan, T., Bendechache, M., Hussain, M., Hussain, I., & Werghi, N. (2025). An interpretable framework for gastric cancer classification using multi-channel attention mechanisms and transfer learning approach on histopathology images. Scientific Reports, 15(1), 13087.
9. J. Fu, L. Luo, Z. Zhong, and J. Qin, "AHP-Guided Stacked Ensemble Modeling for Student Engagement Level Prediction in Online Education," IEEE Access, vol. 13, pp. 114230–114240, 2025, doi: 10.1109/ACCESS.2025.3575529.
10. A. Birk, "In-Presence, Take-Home, and Online Education: The State of Equipment Use and Practices in Robotics Labs and Other Hands-On Activities," IEEE Access, vol. 13, pp. 171984–171997, 2025, doi: 10.1109/ACCESS.2025.3614055.
11. M. Alfadhli, M. Kucukvar, N. C. Onat, S. Al-Maadeed, and A. Abdessadok, "Government Digital Transformation: A Tailor-Made Digital Maturity Assessment Framework," IEEE Access, vol. 13, pp. 71120–71132, 2025, doi: 10.1109/ACCESS.2025.3560999.
12. F. T. Alorini and A. A. Alkhalifah, "Exploring Artificial Intelligence Adoption for E-Government: A Systematic Literature Review," IEEE Access, vol. 13, pp. 175686–175703, 2025, doi: 10.1109/ACCESS.2025.3618778.
13. Vasan, D., Akram, J., Hammoudeh, M., & Ahmed, A. F. (2025). An advanced ensemble framework for defending against obfuscated Windows, Android, and IoT malware. Applied Soft Computing, 173, 112908.
14. M. M. Abualhaj et al., "A firewall model for attack detection using machine learning and metaheuristic feature selection algorithms," Bulletin of Electrical Engineering and Informatics, vol. 15, no. 2, pp. 1265–1282, Apr. 2026, doi: 10.11591/eei.v15i2.9887.
15. Q. Shambour, M. Al-Zyoud, and O. Almomani, "Quantum-Inspired Hybrid Metaheuristic Feature Selection with SHAP for Optimized and Explainable Spam Detection," Symmetry (Basel)., vol. 17, no. 10, Oct. 2025, doi: 10.3390/sym17101716.
16. O. Almomani, A. M. Arabiat, A.-A. Hind, and E. Alsariera, "Hybridization of Deep Learning Models for Multiclass Attack Detection in Wireless Sensor Networks," Journal of Communications, vol. 21, no. 1, 2026.
17. O. Almomani, A. Alsaaidah, M. M. Abualhaj, M. A. Almaiah, A. Almomani, and S. Memon, "URL Spam Detection Using Machine Learning Classifiers," in 2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA), IEEE, 2025, pp. 1–6.
18. M. M. Abualhaj et al., "A Robust IDS System for Intelligent Phishing Website Detection," International Journal of Electrical and Electronic Engineering and Telecommunications, vol. 15, no. 2, pp. 116–127, Mar. 2026, doi: 10.18178/ijeetc.15.2.116-127.

19. M. M. Abualhaj, S. N. Al-Khatib, A. A. Abu-Shareha, A. Hyassat, and M. S. Daoud, "Smart Firewall for Phishing Detection Powered by Bio-Inspired Algorithms," Journal of Advances in Information Technology, vol. 16, no. 11, pp. 1529–1539, 2025, doi: 10.12720/jait.16.11.1529-1539.
20. F. B. of Investigation, "Internet Crime Report 2024," Federal Bureau of Investigation, 2024. [Online]. Available: https://www.ic3.gov/AnnualReport/Reports/2024_IC3Report.pdf
21. S. R. Department, "Global share of phishing attacks 2022-2024," Statista, 2026. [Online]. Available: https://www.statista.com/statistics/1620648/global-phishing-attacks/
22. M. Ramaiah, V. Chandrasekaran, V. Chand, A. Vasudevan, and S. Ibrahim, "Enhanced phishing detection: an ensemble stacking model with DT-RFECV and SMOTE," Appl. Math. (Irvine)., vol. 18, no. 6, pp. 1481–1493, 2024.
23. O. Almomani, A. Alsaaidah, A. A. Abu-Shareha, A. Alzaqebah, M. A. Almaiah, and Q. Shambour, "Enhance url defacement attack detection using particle swarm optimization and machine learning," Journal of Computational and Cognitive Engineering, 2025.
24. M. M. Abualhaj, S. Al-Khatib, A. Alalousi, M. O. Hiari, and M. Al Zyoud, "Phishing Attack Detection Using Whale Optimization Algorithm-Based Feature Selection," in 2025 5th International Conference on Emerging Smart Technologies and Applications (eSmarTA), IEEE, 2025, pp. 1–6.
25. Vinod, D., & Prasad, M. (2026). Enhancing Network Security: A Novel Intrusion Detection System Utilizing Dual-Optimization Techniques for Feature Selection and Classification. Computer Networks, 112021.
26. Yadav, D. K., Raj, A., Rajlakshmi, Kumar, N., & Kumari, R. (2025). Enhancing Email Security: A Real-Time Machine Learning-Based Spam Detection System. Internet Technology Letters, 8(5), e618.
27. A. Alsaaidah, O. Almomani, A. A. Abu-Shareha, M. M. Abualhaj, and A. Achuthan, "ARP spoofing attack detection model in IoT network using machine learning: Complexity vs. accuracy," Journal of Applied Data Sciences, vol. 5, no. 4, pp. 1850–1860, 2024.
28. O. Almomani, A. Alsaaidah, M. A. Almaiah, A. Alzaqebah, M. M. Abualhaj, and W. J. Alzyadat, "Evaluating Machine Learning Classifiers for Detecting Distributed Denial of Service Attacks," in 2025 12th International Conference on Information Technology (ICIT), 2025, pp. 134–140. doi: 10.1109/ICIT64950.2025.11049130.
29. G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification," in OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer, 2003, pp. 986–996.
30. A. S. Hyassat, R. E. Abu Zayed, E. A. Al Khateeb, A. Shalaldeh, M. M. Abdelhamied, and I. Qaddara, "Unveiling Cyber Threats: An In-Depth Study on Data Mining Techniques for Exploit Attack Detection," Engineering Proceedings, vol. 104, no. 1, p. 28, 2025.
31. Barbosa, M. T., Barros, E. B., Mota, V. F., Leite Filho, D. M., Sampaio, L. N., Kuehne, B. T., ... & Peixoto, M. L. (2025). Focca: Fog–cloud continuum architecture for data imputation and load balancing in smart grids. Computer Networks, 258, 111031.
32. M. M. Abualhaj, S. N. Al-Khatib, M. Al-Zyoud, I. Qaddara, M. O. Hiari, and S. M. A. Aldossary, "Enhanced Network Communication Security Through Hybrid Dragonfly-Bat Feature Selection for Intrusion Detection," Journal of Communications, vol. 20, no. 5, 2025.
33. Wasif, M. S., Miah, M. P., Hossain, M. S., Alenazi, M. J., & Atiquzzaman, M. (2025). CNN-ViT synergy: An efficient Android malware detection approach through deep learning. Computers and Electrical Engineering, 123, 110039.
34. Zhang, S., Su, H., Liu, H., & Yang, W. (2025). MPDroid: A multimodal pre-training Android malware detection method with static and dynamic features. Computers & Security, 150, 104262.
35. M. M. Abualhaj, S. N. Al-Khatib, M. A. Alsharaiah, and M. O. Hiari, "Performance Comparison of Whale and Harris Hawks Optimizers with Network Intrusion Prevention Systems," Journal of Applied Data Sciences, vol. 5, no. 4, pp. 1530–1538, 2024.
36. J. Rashid, T. Mahmood, M. W. Nisar, and T. Nazir, "Phishing Detection Using Machine Learning Technique," in Proceedings - 2020 1st International Conference of Smart Systems and Emerging Technologies, SMART-TECH 2020, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, pp. 43–46. doi: 10.1109/SMART-TECH49988.2020.00026.
37. P. Yi, Y. Guan, F. Zou, Y. Yao, W. Wang, and T. Zhu, "Web phishing detection using a deep learning framework," Wirel. Commun. Mob. Comput., vol. 2018, 2018, doi: 10.1155/2018/4678746.

38. H. Alqahtani and A. Abu-Khadrah, "Enhance the accuracy of malicious uniform resource locator detection based on effective machine learning approach," Bulletin of Electrical Engineering and Informatics, vol. 13, no. 6, pp. 4422–4429, Dec. 2024, doi: 10.11591/eei.v13i6.7797.
39. M. Mamun, M. Rathore, A. Lashkari, N. Stakhanova, and A. A. Ghorbani, "Detecting Malicious URLs Using Lexical Analysis," in International Conference on Network and System Security, 2016, in Lecture Notes in Computer Science, vol. 9955. 2016. doi: 10.1007/978-3-319-46298-1.
40. M. M. Abualhaj et al., "Intelligent Malware Detection through Bio-Inspired Optimization and Gradient Boosting," Journal of Advances in Information Technology, vol. 17, no. 2, pp. 239–250, 2026, doi: 10.12720/jait.17.2.239-250.
41. F. Rizk, R. Rizk, D. Rizk, P. Rizk, and C. H. Henry Chu, "KAN-MID: A Kolmogorov-Arnold Networks-Based Framework for Malicious URL and Intrusion Detection in IoT Systems," IEEE Access, vol. 13, pp. 160855–160873, 2025, doi: 10.1109/ACCESS.2025.3605171.
42. M. M. Abualhaj et al., "Spam Detection Boosted by Firefly-Based Feature Selection and Optimized Classifiers," International Journal of Advances in Soft Computing & Its Applications, vol. 17, no. 3, 2025.
43. Hussain, M., Al-Haiqi, A., Zaidan, A. A., Zaidan, B. B., Kiah, M. M., Anuar, N. B., & Abdulnabi, M. (2016). The rise of keyloggers on smartphones: A survey and insight into motion-based tap inference attacks. Pervasive and Mobile Computing, 25, 1-25.
44. M. M. Abualhaj, M. Al-Zyoud, A. Alsaaidah, A. Abu-Shareha, and S. Al-Khatib, "Enhancing Malware Detection through Self-Union Feature Selection Using Firefly Algorithm with Random Forest Classification.," International Journal of Intelligent Engineering & Systems, vol. 17, no. 4, 2024.
45. Lin, L., Zhong, Q., Qiu, J., & Liang, Z. (2025). E-GRACL: an IoT intrusion detection system based on graph neural networks: L. Lin et al. The Journal of Supercomputing, 81(1), 42.
46. M. Abualhaj et al., "Comparative Analysis of LSTM-Based Variant Models for Detecting Attacks in IoT Networks," Journal of Computing & Biomedical Informatics, vol. 10, no. 01, Dec. 2025, doi: 10.56979/1001/2025/1169.
47. P. Sadatian Moghaddam, M. Mahmoudi, N. Serrano, F. Hernando-Gallego, and D. Martín, "Unified Anomaly Detection in IoT and Cyber-Physical Networks Using Evo-Transformer-LSTM: Validation on Four CIC Benchmarks," Dec. 09, 2025. doi: 10.20944/preprints202512.0763.v1.
48. M. Hiari, Y. Alraba'nah, and I. Qaddara, "A deep learning-based intrusion detection system using refined LSTM for DoS attack detection," Engineering, Technology & Applied Science Research, vol. 15, no. 4, pp. 25627–25633, 2025.
49. Y. Alraba'nah, S. Al-Sharaeh, and G. Al Hindi, "Enhancing Intrusion Detection Using Hybrid Long Short-Term Memory and XGBoost," Journal of Soft Computing and Data Mining, vol. 6, no. 1, pp. 247–261, 2025.
50. Farfoura, M.E., Batyha, R.M., Ibrahim, A.Z. et al. Feature resonance classifier: a novel neural network for efficient network traffic classification. J Cloud Comp 15, 61 (2026). https://doi.org/10.1186/s13677-026-00897-3.
51. Kharabsheh, M., AlZu'bi, S., Alsarhan, A., Mughaith, A. A., Aljawabreh, N., & Alabdullatif, M. (2025). A Comprehensive Machine Learning Approach for Email and URL Threat Detection: Using Feature Importance Analysis. International Journal of Advances in Soft Computing & Its Applications, 17(2).