



DETECTING PHISHING WEBSITES USING MACHINE LEARNING: A COMPARATIVE STUDY

Mohd Haris Javed Kazi*, Mohammed Yamin Amin Khan, Ahmed Mumtaz Razi,
Saqlain Ghulam Hassan Siddiqui and Omkar Sherkhane

Department of Computer Science,

Pillai College of Arts, Commerce and Science (Autonomous), Panvel, Maharashtra, India

*Corresponding author E-mail: haris26mda@student.mes.ac.in

Received: 30 June 2026

Revised: 02 August 2026

Accepted: 12 September 2026

Published: 29 September 2026

DOI: <https://doi.org/10.5281/zenodo.23294196>

Abstract:

Phishing websites are a major cybersecurity threat that can deceive users into revealing sensitive information. This study investigates the use of machine learning for phishing URL detection using the LegitPhish dataset, which contains URL-based structural and lexical features. Logistic Regression and Random Forest models were trained and evaluated using accuracy, precision, recall, F1-score, and ROC-AUC. The results showed high classification performance for both models, with Logistic Regression achieving 99.7076% accuracy and Random Forest achieving 99.9554% accuracy. Overall, Random Forest performed best, achieving higher accuracy, phishing recall, F1-score, and ROC-AUC than Logistic Regression, while producing fewer classification errors. Feature-importance analysis identified URL characteristics such as path length, token count, HTTPS usage, and numeric characteristics as important features in the classification process. The findings demonstrate that machine-learning models can effectively use URL-based characteristics for phishing URL classification within the evaluated dataset.

Keywords: Phishing Detection, Machine Learning, Random Forest, Logistic Regression, Feature Importance, Legitphish Dataset.

1. Introduction

The rapid growth of the Internet has made websites a must-have part of life. People use websites for banking, shopping, learning and talking to others. This big reliance has also increased the risk of cyberattack. One of the common dangers is phishing. This is done by hackers who replicate websites or links that look real. They try to get people to share details like usernames, passwords and money info. Phishing websites can look exactly like real ones. This makes it hard for people to see the difference on their own. Machine learning helps with this problem. It can learn from examples of phishing and real URLs. It looks for patterns. Finds ways to tell them apart. Things

like how long the URL is, how many dots it has, how many subdomains, if it uses HTTPS, how many numbers and how long the domain is are all clues that can help.

This research uses the LegitPhish dataset to test how well machine learning works for detecting phishing URLs. It compares two methods: Logistic Regression and Random Forest. The results are measured with scores like accuracy, precision, recall and F1-score. The study also looks at the parts of URLs and mistakes the models make. The goal is to show how these methods work and see if using URL features in machine learning is helpful, for catching phishing attacks.

2. Problem statement

Phishing websites are a problem in cybersecurity because they trick people into visiting harmful sites or giving away important information. Machine-learning techniques can help by looking at URL features and deciding if a site is phishing or legitimate. However not all machine-learning methods work the same. It is useful to know which URL features help detect phishing. This study looks at machine-learning methods for classifying phishing URLs and examines which features help these methods make their decisions.

3. Research question

How effectively can machine-learning models distinguish phishing URLs from legitimate URLs, and which URL characteristics contribute most to the classification?

4. Objectives

- To evaluate the ability of machine-learning models to classify phishing and legitimate URLs.
- To compare the performance of Logistic Regression and Random Forest using standard classification metrics.
- To identify the URL characteristics that contribute most to phishing URL classification.

5. Literature review

Phishing detection has been widely studied using machine-learning and other automated techniques. Earlier approaches have included blacklists, heuristic methods, website characteristics, and network-related information. Recent research has increasingly explored machine-learning techniques because they can learn patterns from previously identified phishing and legitimate websites.

Shukla *et al.* (1) investigated phishing attack detection using HTTP header-based information and machine-learning techniques. Their study highlighted the limitations of traditional approaches such as blacklists and demonstrated the potential of using additional website and network characteristics for automated phishing detection.

Rashid *et al.* (2) examined the generalization of phishing URL detection models using unsupervised domain adaptation. Their study highlighted an important challenge in phishing detection: a model may perform well when training and testing data originate from similar distributions but may face difficulties when detecting newly created phishing domains. This indicates that model evaluation should consider not only classification performance but also the characteristics and limitations of the underlying data.

Ghalechyan *et al.* (3) investigated neural-network-based approaches for phishing URL classification using open-source and production data. Their study evaluated deterministic and probabilistic neural-network models and reported high validation performance in their experimental setting. The research demonstrates the potential of

machine-learning and neural approaches for identifying phishing URLs while also emphasizing the importance of appropriate datasets and validation procedures.

Tamal *et al.* (4) proposed a URL-focused phishing detection approach based on an optimal feature vectorization method. Their dataset contained 247,950 URLs represented using 42 intra-URL features, and 15 supervised machine-learning algorithms were evaluated. Their experiments reported strong performance from Random Forest, demonstrating that structural and lexical characteristics of URLs can provide useful information for phishing classification.

More recently, Potpelwar *et al.* (5) introduced the LegitPhish dataset, which contains 101,219 manually verified phishing and legitimate URLs represented using 17 structural and lexical features. The dataset was developed to support machine-learning research in URL-based phishing detection and provides a large-scale basis for reproducible experiments.

Overall, existing research demonstrates the usefulness of machine learning and URL-based characteristics for phishing detection. However, studies use different datasets, feature sets, algorithms, and evaluation conditions, making direct comparison between approaches difficult. Therefore, this study uses the recently developed LegitPhish dataset to conduct a controlled comparison of Logistic Regression and Random Forest and to examine which URL characteristics contribute to phishing URL classification.

6. Research gap

Existing studies have shown that machine-learning techniques can be used for phishing URL detection. Previous research used different datasets, URL features, algorithms and experimental settings. Because of these differences it is hard to compare the performance of approaches. Some studies looked at algorithms or used very complex models while fewer studies gave a simple controlled comparison of different machine-learning types on a recent large-scale URL dataset. The introduced LegitPhish dataset gives a large collection of manually verified phishing and legitimate URLs. The dataset contains structural and lexical features. Still there is room to study how different conventional machine-learning approaches perform on the LegitPhish dataset. It is also worth finding out which URL characteristics are most useful for classification.

Therefore, this study addresses this gap by conducting a controlled comparison of Logistic Regression and Random Forest on the LegitPhish dataset. In addition to comparing the classification performance, for phishing URL detection the study analyzes URL characteristics and classification errors. This gives an understanding of how phishing URL detection can be improved.

7. Dataset description

This study uses the LegitPhish dataset, a large-scale dataset created for classifying phishing URLs and legitimate URLs. The dataset includes URL-based lexical features that can be used as inputs, for machine-learning models.

The dataset has 101,219 records and 18 columns. These include the URL 17 features related to the URL and one target variable called ClassLabel. The ClassLabel indicates whether a URL is phishing or legitimate. In the dataset 0 means the URL is phishing and 1 means the URL is legitimate.

The 17 features cover aspects of URLs. These include the length of the URL whether it contains an IP address the number of dots if it uses HTTPS, the entropy of the URL the number of tokens the number of subdomains the presence of query parameters characteristics of the top-level domain the length of the path the number of hyphens the number of digits the length of the domain and multiple other numerical or structural properties.

8. Dataset distribution and quality

The dataset includes 63,678 phishing URLs and 37,540 legitimate URLs. One record in the dataset has a missing target label. The feature columns in the dataset do not contain any values. The dataset also holds 346 duplicate records. For the machine-learning experiments the dataset will remove records with a missing target label and duplicate records during preprocessing. This step guarantees that the models train on non-repeated observations from the dataset. The dataset is especially suitable for this study because it offers a collection of labeled URLs along, with standardized URL characteristics. This lets the study compare Logistic Regression and Random Forest using the dataset and experimental conditions. It is important to note that the dataset focuses on URL-level characteristics. Consequently, this study targets phishing URL detection only and does not examine webpage content or website behavior.

9. Proposed methodology

The proposed methodology of this study consists of several stages, beginning with dataset preprocessing and exploratory analysis and followed by machine-learning model development and evaluation. The overall workflow is designed to compare Logistic Regression and Random Forest for phishing URL classification using the same dataset and evaluation conditions.

9.1 Research workflow

The proposed workflow is:

LegitPhish Dataset → Data Preprocessing → Exploratory Data Analysis → Feature Preparation → Train-Test Split → Model Training → Model Evaluation → Feature Importance → Error Analysis

Each stage contributes to understanding how effectively URL-based characteristics can be used to distinguish phishing URLs from legitimate URLs.

9.2 Data preprocessing

The data set will be looked at first to find any values, any repeated records, the types of data and how the target variable is spread out. The record that has a ClassLabel will be taken out because it can't be used as a labeled example for training. Repeated records will also be taken out to make sure that the same observations don't affect how the model is tested.

The URL column will not be used directly as text input. Instead the 17 features that were created for the URL that're, in the data set will be used as the input features. The ClassLabel column will be used as the target variable, where 0 means phishing and 1 means an URL.

9.3 Exploratory data analysis

Exploratory Data Analysis (EDA) will be performed to understand the characteristics of the dataset before model training. The class distribution will be examined to find out how many phishing URLs there are compared to URLs. The distributions of selected features will also be analyzed. Correlation analysis will be used to look at relationships, between URL characteristics and to spot features that may be related. These observations will give a picture of the dataset before we apply machine-learning models.

9.4 Feature preparation

The 17 URL related characteristics will be used as input variables for the machine-learning models. These features represent lexical properties of URLs. The URL length is one of the features. The domain characteristics are another.

Subdomains are also part of the features. Digits are included. Query parameters are there. HTTPS usage is considered. Other URL properties are also part of the list.

For Logistic Regression the feature scaling will be considered. The numerical features may have ranges. For Random Forest the feature scaling is not required. The model is based on decision tree splits. The model does not need feature scaling. The Random Forest model works without feature scaling. The Logistic Regression model needs feature scaling. The feature scaling is important for Logistic Regression. The feature scaling is not important, for Random Forest.

9.5 Train-test split

The cleaned dataset will be divided into training and testing sets. The training data will be used to develop the machine-learning models, while the testing data will be kept separate for evaluating their performance on previously unseen records.

A stratified split will be used so that the relative distribution of phishing and legitimate URLs is maintained between the training and testing sets.

9.6 Machine-learning models

Two supervised machine-learning algorithms will be evaluated in this study.

9.6.1 Logistic regression

Logistic Regression will be used as a relatively simple and interpretable classification model. It estimates the probability that a URL belongs to a particular class based on the provided URL characteristics.

The model provides a useful baseline for determining how well a conventional linear classification approach can distinguish phishing URLs from legitimate URLs.

9.6.2 Random forest

Random Forest will be used as a tree-based ensemble classification model. It combines multiple decision trees to make classification decisions and can capture nonlinear relationships between URL characteristics.

Random Forest will also be used to examine feature importance, allowing the study to identify which URL characteristics contribute most to the model's classification decisions.

9.7 Model evaluation

The trained models will be evaluated using the testing dataset. The primary evaluation metrics will be:

- Accuracy – the proportion of correctly classified URLs.
- Precision – the proportion of URLs predicted as a particular class that actually belong to that class.
- Recall – the proportion of actual URLs belonging to a class that are correctly identified.
- F1-score – the harmonic mean of precision and recall.
- Confusion Matrix – shows the numbers of correctly and incorrectly classified instances for each class.

ROC-AUC may also be calculated to provide an additional measure of the models' ability to distinguish between the two classes.

9.8 Feature importance and error analysis

Once we have evaluated the model, we will check for feature importance, particularly for the Random Forest model. Then, the URL features with the highest impact will be extracted and discussed within the framework of phishing URL classification.

Also, we will analyse the errors made, by looking at wrongly classified records. This will help to find patterns on URLs that are hard to be classified by the models and give additional insight into the limitations of URL-based phishing detection.

Finally, we will compare the performance of Logistic Regression and Random Forest with the same testing data and evaluation metrics. The final comparison will be based on the experimental results and not on any assumption of which model will do better.

10. Results

The performance of Logistic Regression and Random Forest was evaluated on the cleaned LegitPhish data set. After preprocessing, the dataset consisted of 100,872 records and 16 URL-based features. Stratified sampling was applied to divide the data into 80% training data and 20% test data. The final test set contained 20,175 records; 12,699 were phishing URLs and 7,476 were legitimate URLs.

10.1 Model performance

The final models were evaluated using accuracy, precision, recall, F1-score, and ROC-AUC. The reported results were obtained from the untouched test set after model selection and hyperparameter tuning.

Model	Accuracy (%)	Phishing Precision (%)	Phishing Recall (%)	Phishing F1 (%)	ROC-AUC (%)
Logistic Regression	99.7076	99.8895	99.6456	99.7674	99.9842
Random Forest	99.9554	99.9370	99.9921	99.9646	99.9997

Both models achieved high classification performance on the test dataset. Logistic Regression achieved an accuracy of 99.7076%, while Random Forest achieved 99.9554%. For the phishing class, Logistic Regression obtained a precision of 99.8895%, recall of 99.6456%, and F1-score of 99.7674%. Random Forest obtained a phishing precision of 99.9370%, recall of 99.9921%, and F1-score of 99.9646%.

The ROC-AUC values were also very high for both models, with 0.999842 for Logistic Regression and 0.999997 for Random Forest.

10.2 Cross-validation results

To examine the consistency of model performance, 5-fold stratified cross-validation was performed using the training data only. Logistic Regression achieved a mean accuracy of 0.996567 and a mean phishing F1-score of 0.997270. Random Forest achieved a mean accuracy of 0.999331 and a mean phishing F1-score of 0.999469.

The standard deviation of phishing F1-score across the five folds was 0.000307 for Logistic Regression and 0.000160 for Random Forest, indicating relatively small variation across the folds. The final test set was not used during cross-validation or hyperparameter tuning.

10.3 Confusion matrix analysis

The confusion matrices provide a detailed view of the classification errors.

For Logistic Regression, the confusion matrix was:

	Predicted Phishing	Predicted Legitimate
Actual Phishing	12,654	45
Actual Legitimate	14	7,462

The model correctly classified 12,654 phishing URLs and 7,462 legitimate URLs. It incorrectly classified 45 phishing URLs as legitimate and 14 legitimate URLs as phishing.

For Random Forest, the confusion matrix was:

	Predicted Phishing	Predicted Legitimate
Actual Phishing	12,698	1
Actual Legitimate	8	7,468

Random Forest correctly classified 12,698 phishing URLs and 7,468 legitimate URLs. Only one phishing URL was classified as legitimate, while eight legitimate URLs were classified as phishing.

10.4 ROC-AUC analysis

The ROC-AUC results were 0.999842 for Logistic Regression and 0.999997 for Random Forest. These values indicate that both models produced strong separation between the phishing and legitimate classes on the test dataset.

The ROC curve analysis also showed that Random Forest achieved a slightly higher AUC than Logistic Regression in the final evaluation.

10.5 Feature importance

Feature importance was analyzed using the Random Forest model to identify the URL characteristics that contributed most to its classification decisions.

Rank	Feature	Importance
1	path_length	0.403048
2	token_count	0.146255
3	https_flag	0.117216
4	percentage_numeric_chars	0.104899
5	number_of_digits	0.101588
6	dot_count	0.042315
7	subdomain_count	0.025764
8	url_length	0.018526

The highest feature-importance score was obtained by path_length, with an importance value of 0.403048. This was followed by token_count, https_flag, percentage_numeric_chars, and number_of_digits.

These values indicate the relative contribution of the features to the Random Forest's classification decisions. They should not be interpreted as evidence that a particular URL characteristic directly causes a URL to be phishing.

10.6 Error analysis

Error analysis was performed on the final predictions to examine the misclassified records. Logistic Regression produced 59 total errors, consisting of 45 phishing false negatives and 14 legitimate false positives. Random Forest produced 9 total errors, consisting of 1 phishing false negative and 8 legitimate false positives.

The error patterns showed that some phishing URLs had feature characteristics that were closer to the patterns observed in legitimate URLs, resulting in false-negative predictions. Similarly, some legitimate URLs contained structural characteristics that caused them to be classified as phishing.

For Logistic Regression, the average path length of phishing false negatives was 10.49, while legitimate false positives had an average path length of 22.21. For Random Forest, the single phishing false negative had a path length of 1, while the legitimate false positives had an average path length of 11.75.

Overall, the error analysis shows that both models made relatively few classification errors on the test dataset, with Random Forest producing fewer errors than Logistic Regression.

11. Discussion

The experimental results show that both Logistic Regression and Random Forest achieved high performance in differentiating phishing and legitimate URLs on the LegitPhish dataset. Logistic Regression has achieved an accuracy of 99.7076% and Random Forest has achieved an accuracy of 99.9554%. The results obtained by the two models indicates that the URL based characteristics in the dataset contain useful information to distinguish between the two classes (6).

Results also show a difference between the two approaches. Logistic Regression is a linear classifier whereas Random Forest can capture nonlinear relations and interactions among features (7). The final test evaluation Random Forest resulted in less classification errors than Logistic Regression. However, these results are for the performance on evaluated dataset and should not be considered as a promise for the same performance on all future or unseen phishing URLs (8).

The analysis of the confusion matrix gives additional insights into the classification performance. The Logistic Regression model misclassified 45 phishing URLs as legitimate and 14 legitimate URLs as phishing. In contrast, Random Forest misclassified only one phishing URL as legitimate URL and 8 legitimate URLs as phishing. This is particularly important as false positive predictions correspond to phishing URLs classified as legitimate and the number of false negatives obtained for Random Forest is lower (9).

Furthermore, the feature-importance analysis provides an insight into the characteristics used by the Random Forest model. The most important feature was `path_length` followed by `token_count`, `https_flag`, `percentage_numeric_chars` and `number_of_digits`. This means that these URL features were important for the model to determine the classification. However, feature importance shows the contribution of a feature in the trained model, and does not reveal a direct causal link between a certain characteristic and the occurrence of phishing activity (10).

The error analysis also reveals that the misclassified URLs may have features overlapping the phishing and legitimate classes. Some phishing URLs, which had similar feature structures to legitimate URLs, were predicted as false-negative. Similarly, some legitimate URLs showed characteristics that made the models classify them as phishing. This demonstrates the difficulty of classification when structural properties of URLs from the two classes are similar (11).

The cross-validation results matched the test results. Both models kept results in each validation fold. Cross-validation results show that the good results are not just due, to one training-validation split. Nevertheless the evaluation still depends on the features. Spread of the LegitPhish dataset.

Overall the findings show that conventional machine-learning models can use engineered URL characteristics well for phishing URL classification in the dataset we looked at. When we examine the results it becomes clear that machine-learning models vary in how they handle URL characteristics for phishing URL classification. The comparison also shows that different model structures can create error patterns and the error patterns of each model show that not all mistakes are the same. The performance of each machine-learning model depends on how it treats URL characteristics during phishing URL classification so the performance of each machine-learning model is not the same. This points to the need to look at both performance and each classification error, not just

accuracy and looking at classification errors helps us understand where the models fail. Accuracy alone does not tell the story, about the performance of machine-learning models (12).

12. Limitations

Although the proposed models achieved classification performance on the LegitPhish dataset the study has several limitations.

First the study focuses on URL-based features than complete webpage content. The models use lexical characteristics extracted from URLs and therefore do not analyze webpage HTML, visual appearance, scripts, page content or other behavioral characteristics that may provide additional information, for phishing detection.

Second the evaluation relies on the LegitPhish dataset and its specific mix of phishing and legitimate URLs. I think the high performance achieved in this study may therefore rely partly on the traits of the LegitPhish dataset and may not directly show how well the method works on all real-world phishing URLs or on phishing campaigns.

Finally phishing techniques and URL structures can change over time. The features that help classify phishing techniques and URL structures in the dataset may become less useful when attackers change their methods. Therefore, we can conclude regular checks, with independently gathered datasets would be helpful to see if the models still work well.

13. Conclusion

This study looked at Logistic Regression and Random Forest to detect phishing URLs using the LegitPhish dataset. Both models showed good results with Random Forest having 99.9554% accuracy and Logistic Regression having 99.7076% accuracy.

The findings show that features based on URLs can give information to tell the difference between phishing URLs and legitimate URLs. The analysis of the features and the errors also gave understanding of the traits that are connected to the models decisions. Overall the study shows that traditional machine-learning methods can be useful, for detecting phishing URLs.

14. Future work

Future work can expand on this study by testing the models using phishing URL datasets that were collected separately. It would also be helpful to include features from webpages, such, as HTML structure, JavaScript behavior and content details to strengthen detection beyond just looking at URLs. Different machine-learning and deep-learning methods can be tested to see how they compare. Eventually the approach developed here could be built into a real-time phishing detection system to help protect users.

References

1. Shukla, S., Misra, M., & Varshney, G. (2024). HTTP header based phishing attack detection using machine learning. *Transactions on Emerging Telecommunications Technologies*, 35(1), e4872. <https://doi.org/10.1002/ett.4872>
2. Rashid, F., Doyle, B., Han, S. C., & Seneviratne, S. (2024). Phishing URL detection generalisation using unsupervised domain adaptation. *Computer Networks*, 245, 110398. <https://doi.org/10.1016/j.comnet.2024.110398>
3. Ghalechyan, H., Israyelyan, E., Arakelyan, A., Hovhannisyan, G., & Davtyan, A. (2024). Phishing URL detection with neural networks: An empirical study. *Scientific Reports*, 14, 25134. <https://doi.org/10.1038/s41598-024-74725-6>

4. Tamal, M. A., Islam, M. K., Bhuiyan, T., & Sattar, A. (2024). Dataset of suspicious phishing URL detection. *Frontiers in Computer Science*, 6, 1308634. <https://doi.org/10.3389/fcomp.2024.1308634>
5. Potpelwar, R. S., Kulkarni, U. V., & Waghmare, J. M. (2025). LegitPhish: A large-scale annotated dataset for URL-based phishing detection. *Data in Brief*, 63, 111972. <https://doi.org/10.1016/j.dib.2025.111972>
6. Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, 117, 345–357. <https://doi.org/10.1016/j.eswa.2018.09.029>
7. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
8. Al-Sarem, M., Saeed, F., Al-Mekhlafi, Z. G., Mohammed, B. A., & Al-Hadhrami, T. (2021). An optimized stacking ensemble model for phishing website detection. *Electronics*, 10(11), 1285. <https://doi.org/10.3390/electronics10111285>
9. Rao, R. S., & Pais, A. R. (2019). Detection of phishing websites using an efficient feature selection approach. *Neural Computing and Applications*, 31(11), 7751–7773. <https://doi.org/10.1007/s00521-018-3604-x>
10. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS 2017)* (Vol. 30, pp. 4765–4774). Curran Associates, Inc.
11. Hannousse, A., & Yahiouche, S. (2021). Towards benchmark datasets for machine learning based website phishing detection: An experimental study. *Engineering Applications of Artificial Intelligence*, 104, 104347. <https://doi.org/10.1016/j.engappai.2021.104347>
12. Althobaiti, O. S. (2022). A review of machine learning performance metrics and their applications in cyber security. *International Journal of Computer Science and Network Security*, 22(4), 185–194. <https://doi.org/10.22937/IJCSNS.2022.22.4.25>