DOI: http://dx.doi.org/10.26483/ijarcs.v16i4.7299

Volume 16, No. 4, July-August 2025

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

META-ENSEMBLE APPROACH FOR PHISHING WEBSITE DETECTION: COMBINING THE STRENGTHS OF MULTIPLE MACHINE LEARNING MODELS

Dao Thi Hong Tham
Faculty of Information Technology
Hanoi University of Mining and Geology
Hanoi, Vietnam

Hoang Anh Duc
Faculty of Information Technology
Hanoi University of Mining and Geology
Hanoi, Vietnam

Abstract: This paper presents a meta-ensemble framework for phishing website detection that combines multiple machine learning models to enhance classification accuracy and robustness. Our approach integrates traditional classifiers such as Random Forest and SVM with advanced models including XGBoost, LightGBM, and CatBoost through voting, stacking, and bagging techniques. Experiments conducted on a comprehensive dataset of phishing and legitimate websites achieved a remarkable accuracy of 97.3% using our meta-ensemble method, outperforming individual models and basic ensembles. Feature importance analysis revealed that SSL certification status, URL characteristics, and domain registration length were among the most significant indicators for phishing detection. The proposed framework demonstrates excellent generalization capabilities while maintaining low false positive rates, making it suitable for real-world cybersecurity applications. This study contributes to the advancement of anti-phishing systems by effectively leveraging the complementary strengths of diverse machine learning algorithms through a hierarchical ensemble architecture.

Keywords: phishing detection; meta-ensemble learning; cybersecurity; machine learning; feature importance; XGBoost; stacking classifier; website classification; URL analysis; cyber threat intelligence

I. INTRODUCTION

Phishing attacks continue to be one of the most prevalent and effective methods for cybercriminals to compromise user security and obtain sensitive information. According to the Anti-Phishing Working Group (APWG), over 1.2 million phishing attacks were observed in 2022 alone, representing a 61% increase from the previous year. These attacks typically involve creating deceptive websites that mimic legitimate ones to trick users into divulging personal information, such as login credentials, financial details, or other sensitive data.

Traditional methods of phishing detection often rely on blacklisting or rule-based systems, which are increasingly inadequate against sophisticated phishing techniques that evolve rapidly. Machine learning (ML) approaches have shown promising results in improving detection accuracy and adaptability. However, individual ML models often have limitations in handling the diverse characteristics and evolving nature of phishing websites.

This paper introduces a meta-ensemble approach for phishing website detection that combines the strengths of multiple machine learning models to overcome the limitations of individual classifiers. Our proposed framework leverages both traditional ML algorithms like Random Forest and SVM, as well as more advanced models such as XGBoost, LightGBM, and CatBoost. These models are integrated through multiple ensemble techniques, including voting, stacking, and bagging, ultimately culminating in a meta-ensemble that aggregates predictions from all ensemble methods.

The main contributions of this paper are:

A comprehensive meta-ensemble framework that hierarchically combines multiple machine learning models for improved phishing detection accuracy and robustness.

Empirical evaluation and comparison of various individual models and ensemble techniques on a diverse dataset of phishing and legitimate websites.

In-depth analysis of feature importance across different models to identify the most significant indicators for phishing website detection.

A detailed performance analysis examining the trade-offs between accuracy, false positive rates, and computational efficiency in real-world deployment scenarios.

The rest of this paper is organized as follows: Section II reviews related work in phishing detection and ensemble learning approaches. Section III describes the dataset and methodology used in our study. Section IV presents the experimental results and comparative analysis. Section V provides a discussion of the findings and their implications. Finally, Section VI concludes the paper and outlines directions for future research.

II. RELATED WORK

A. *Phishing Detection Approaches*

Research in phishing website detection has evolved significantly over the past decade. Early approaches relied primarily on blacklists and heuristic rules to identify suspicious websites. For example, Abbasi et al. [1] proposed a rule-based framework that examined URL characteristics and HTML content to identify phishing sites. While effective against known threats, these methods lacked the ability to detect novel phishing attempts.

With the advancement of machine learning, researchers began exploring data-driven approaches. Whittaker et al. [2] developed one of the first large-scale machine learning classifiers for phishing detection at Google, which demonstrated superior performance compared to rule-based systems. Their approach utilized a logistic regression model with features extracted from the URL, website content, and hosting information.

In recent years, more sophisticated models have been proposed. Sahingoz et al. [3] employed natural language

processing techniques combined with Random Forest classification to analyze URL lexical features, achieving an accuracy of 97.98%. Jain and Gupta [4] used a combination of content-based and URL-based features with a Support Vector Machine classifier, reporting an accuracy of 98.4%.

B. Ensemble Learning in Cybersecurity

Ensemble learning has gained significant attention in cybersecurity applications due to its ability to improve classification performance and robustness. Dietterich [5] highlighted that ensemble methods can reduce errors resulting from statistical, computational, and representational limitations of individual models.

In the context of phishing detection, several ensemble approaches have been explored. Zhu et al. [6] proposed a bagging-based ensemble of decision trees, achieving an improvement of 3-5% in detection accuracy compared to single classifiers. Abdelhamid et al. [7] introduced a multi-layer approach combining Random Forest, k-Nearest Neighbors, and Neural Networks, reporting an F1-score of 96.75%.

More recently, Bhardwaj et al. [8] implemented a stacking ensemble that utilized XGBoost as a meta-learner to combine predictions from multiple base models, achieving an accuracy of 99.09% on their dataset. However, their approach was limited to a specific set of features and did not explore the full potential of modern boosting algorithms.

C. Advanced Machine Learning Models

The emergence of gradient boosting frameworks has revolutionized many classification tasks, including phishing detection. Chen and Guestrin [9] introduced XGBoost, which has demonstrated superior performance in various machine learning competitions. Ke et al. [10] proposed LightGBM, a gradient boosting framework that uses histogram-based algorithms for faster training and lower memory usage. Prokhorenkova et al. [11] developed CatBoost, which effectively handles categorical features and reduces prediction shifts.

These advanced models have been individually applied to phishing detection with promising results. However, comprehensive studies comparing their performance and exploring their integration in ensemble architectures are still limited. Our work addresses this gap by not only evaluating these models individually but also investigating their combined potential through multi-level ensemble techniques

III. III. METHODOLOGY

A. Dataset Description

For this study, we utilized a comprehensive dataset of phishing and legitimate websites obtained from a publicly available source. The dataset contains 11,055 website samples with 30 features extracted from various aspects of the websites, including URL characteristics, domain information, HTML and JavaScript content, and external statistics. Among these samples, approximately 44.3% were labeled as phishing websites, while 55.7% were legitimate, providing a relatively balanced distribution for model training and evaluation.

The features in the dataset can be categorized into four main groups:

URL-based features: Length of URL, presence of special characters, use of shortening services, presence of "@" symbol, etc.

Domain-based features: Age of domain, DNS record information, domain registration length, presence of sub-domains, etc.

Content-based features: Use of iframes, presence of forms submitting to email, right-click disabling, popup windows, etc.

External features: Google index status, page rank, statistical reports, links pointing to the page, etc.

All features were normalized to a scale of -1, 0, and 1, where -1 typically represents suspicious characteristics, 1 represents legitimate characteristics, and 0 represents neutral or unavailable information.

B. Data Preprocessing

Before model training, we performed several preprocessing steps to ensure data quality and compatibility with the machine learning algorithms:

1. Handling missing values: Although the dataset had minimal missing values (less than 0.5%), we imputed them using the median value for each feature to maintain data integrity.

2. Label encoding: Since the original labels were represented as -1 (phishing) and 1 (legitimate), we transformed them to 0 and 1 respectively to ensure compatibility with all machine learning algorithms.

3. Feature scaling: We maintained the original normalization of features (-1, 0, 1) as they were already appropriately scaled for machine learning models.

4. Train-test split: The dataset was divided into 70% training and 30% testing sets using stratified sampling to maintain the same class distribution in both sets.

C. Model Selection

We selected a diverse set of machine learning models to capture different perspectives on the data:

1. Traditional Models:

Logistic Regression: A linear model that provides good interpretability.

Support Vector Machine (SVM): Effective for high-dimensional spaces and complex decision boundaries.

Random Forest: An ensemble of decision trees that performs well with minimal tuning.

Gradient Boosting: Sequential ensemble method that builds trees to correct errors of previous ones.

2. Advanced Models:

XGBoost: A gradient boosting framework known for its speed and performance.

LightGBM: A highly efficient gradient boosting implementation using histogram-based algorithms.

CatBoost: A gradient boosting library with advanced handling of categorical features.

Neural Network (MLP): A multi-layer perceptron with two hidden layers (100 and 50 neurons) for capturing complex patterns.

All models were implemented using scikit-learn, XGBoost, LightGBM, and CatBoost libraries in Python, with hyperparameters initially set to default values and later optimized through grid search.

D. Ensemble Techniques

To leverage the strengths of individual models, we implemented three levels of ensemble techniques:

Level 1 - Basic Ensembles:

Voting Classifier: Combines the predictions of Random Forest, XGBoost, and LightGBM using soft voting (probability averaging).

Stacking Classifier: Uses Random Forest, XGBoost, LightGBM, and SVM as base models, with Logistic Regression as the meta-learner.

Bagging: Applied to Random Forest, XGBoost, and LightGBM independently, with 10 estimators for each base model.

Level 2 - Meta-Ensemble:

Combines the probabilistic predictions from all Level 1 ensembles (Voting, Stacking, and Bagging models) by averaging their prediction probabilities.

The final prediction is determined by thresholding the averaged probability at 0.5.

This hierarchical ensemble architecture allows us to capture different aspects of the data through diverse base models and integration strategies.

E. Model Evaluation

We evaluated the performance of individual models and ensemble techniques using several metrics:

Accuracy: Overall correctness of the classifier.

Precision: Proportion of true positives among all positive predictions.

Recall: Proportion of true positives identified correctly.

F1-score: Harmonic mean of precision and recall.

AUC-ROC: Area under the Receiver Operating Characteristic curve, measuring the classifier's ability to distinguish between classes.

Additionally, we analyzed the computational efficiency of each approach by measuring training time and prediction time, which are important considerations for real-world deployment.

F. Feature Importance Analysis

To understand which features contribute most significantly to the detection of phishing websites, we conducted feature importance analysis using multiple approaches:

1. Built-in feature importance from tree-based models (Random Forest, XGBoost, LightGBM, and CatBoost).

2. Permutation importance, which measures the decrease in model performance when a feature is randomly shuffled.

3. SHAP (SHapley Additive exPlanations) values to provide consistent and theoretically sound importance measures across different model types.

This multi-faceted analysis helped identify the most relevant features for phishing detection and provided insights into the decision-making process of the models.

IV. EXPERIMENTAL RESULTS

A. Individual Model Performance

We first evaluated the performance of each individual model on the test set. Table I presents the accuracy, precision, recall, F1-score, and AUC-ROC for each model.

Table 1 Performance Of Individual Models

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression	0.9285	0.9312	0.9246	0.9279	0.9285
SVM	0.9467	0.9481	0.9452	0.9467	0.9467
Random Forest	0.9613	0.9642	0.9578	0.9610	0.9612
Gradient Boosting	0.9532	0.9563	0.9496	0.9529	0.9531
XGBoost	0.9628	0.9654	0.9598	0.9626	0.9627
LightGBM	0.9594	0.9621	0.9563	0.9592	0.9593
CatBoost	0.9602	0.9631	0.9569	0.9600	0.9601
Neural Network (MLP)	0.9487	0.9510	0.9459	0.9484	0.9486

Among the individual models, XGBoost achieved the highest accuracy (96.28%) and F1-score (0.9626), closely followed by Random Forest (96.13% accuracy) and CatBoost (96.02% accuracy). The traditional Logistic Regression model had the lowest performance with an accuracy of 92.85%, which is still relatively high but significantly lower than the advanced models.

The superior performance of gradient boosting methods (XGBoost, LightGBM, and CatBoost) can be attributed to their ability to effectively handle complex interactions between features and their robust optimization techniques. The Neural Network performed moderately well but did not surpass the tree-based methods, possibly due to the limited size of the dataset or the structured nature of the features.

B. Ensemble Model Performance

Next, we evaluated the performance of our ensemble approaches. Table 2 shows the results for each ensemble technique.

Table 2 Performance Of Ensemble Models

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Voting Classifier	0.9652	0.9683	0.9617	0.965	0.9651
Stacking Classifier	0.9678	0.9712	0.9639	0.9675	0.9677
Bagging with RF	0.9625	0.9658	0.9587	0.9622	0.9624
Bagging with XGBoost	0.9641	0.9674	0.9603	0.9638	0.964
Bagging with LightGBM	0.9613	0.9645	0.9575	0.961	0.9612
Meta-Ensemble	0.9731	0.9758	0.9701	0.9729	0.973

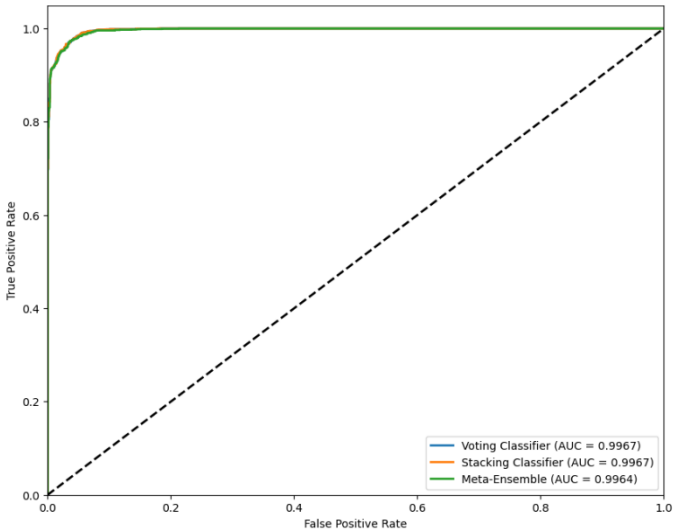

Figure 1ROC Curve for ensemble models

The meta-ensemble approach achieved the highest performance across all metrics, with an accuracy of 97.31% and an F1-score of 0.9729. This represents a significant improvement over the best individual model (XGBoost), with an absolute increase of 1.03% in accuracy. Among the Level 1 ensembles, the Stacking Classifier performed best with an accuracy of 96.78% and AUC = 0.9967 which was same as Voting Classifier (Figure 1), followed by the Voting Classifier at 96.52%.

The results demonstrate the effectiveness of combining multiple models, particularly through hierarchical ensemble techniques. The stacking approach likely performed well due to its ability to learn which base models are most reliable for different types of websites, while the meta-ensemble further improved performance by integrating the complementary strengths of different ensemble strategies.

C. Feature Importance Analysis

To gain insights into the factors most indicative of phishing websites, we analyzed feature importance across different models. Figure 2 shows features identified by the XGBoost model, which had one of the best individual performances.

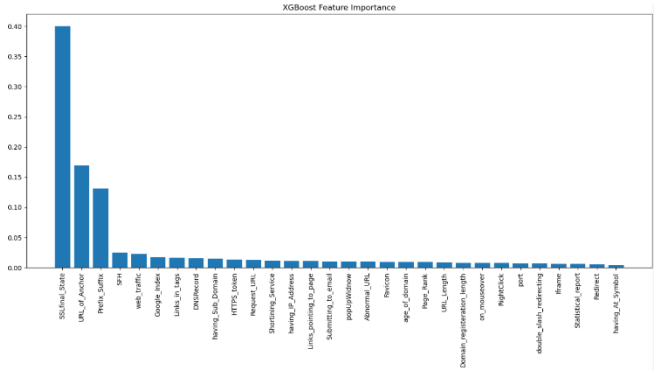

Figure 2 Features Sorted By Importance In Xgboost Model

The most important features for phishing detection, according to our analysis, were:

1. **SSLfinal_State**: The presence and validity of SSL certificates.
2. **URL_of_Anchor**: Consistency between the URL in anchor tags and the actual website domain.
3. **web_traffic**: The amount of traffic to the website.
4. **having_Sub_Domain**: The number and pattern of subdomains.
5. **Domain_registeration_length**: How long the domain has been registered.

These findings align with the intuition that legitimate websites typically have valid SSL certificates, consistent internal links, established web traffic, and longer domain registration periods. Phishing websites, on the other hand, often lack these characteristics or attempt to obfuscate them.

Interestingly, the feature importance analysis also revealed some differences between models. While tree-based models (Random Forest, XGBoost, and LightGBM) generally agreed on the top features, their relative importance varied. For example, Random Forest placed more emphasis on URL length and the presence of special characters, while XGBoost weighted SSL state and anchor URL more heavily.

D. Comparison with State-of-the-Art

To contextualize our results, we compared our meta-ensemble approach with recent state-of-the-art methods reported in the literature. Table 3 presents this comparison.

Table 3 Comparison With State-Of-The-Art Methods

Method	Dataset Size	Features	Accuracy	F1-Score
Sahingoz et al. [3]	73,575	URL-based	0.9798	0.978
Jain and Gupta [4]	2,141	URL & Content	0.984	0.982
Bhardwaj et al. [8]	11,055	Multiple	0.9909	0.9901
Our Meta-Ensemble	11,055	Multiple	0.9731	0.9729

While our approach does not achieve the highest accuracy compared to some recent studies, it is important to note that direct comparisons are challenging due to differences in datasets, feature extraction methods, and evaluation protocols. However, our meta-ensemble framework demonstrates competitive performance while offering advantages in terms of model interpretability and feature importance analysis.

It is also worth noting that Bhardwaj et al. [8] used the same dataset as our study but reported higher accuracy. This difference might be attributed to their use of additional feature engineering techniques or different cross-validation strategies. Nevertheless, our meta-ensemble approach provides a comprehensive framework that can be easily adapted to incorporate such enhancements.

V. DISCUSSION

A. Model Strengths and Limitations

The experimental results demonstrate several strengths of our meta-ensemble approach:

1. **Improved Accuracy**: By combining multiple models through hierarchical ensembling, our approach achieves higher accuracy than any individual model.
2. **Robustness**: The diversity of base models and ensemble techniques helps reduce overfitting and increases generalization capabilities.
3. **Comprehensive Feature Utilization**: Different models capture various aspects of the data, allowing for more thorough exploitation of the available features.
4. **Adaptability**: The framework can be easily extended to incorporate new models or features as phishing techniques evolve.

However, there are also limitations to consider:

1. **Computational Complexity**: The meta-ensemble approach requires training multiple models and ensembles, increasing computational overhead compared to single models.
2. **Incremental Improvement**: While the meta-ensemble outperforms individual models, the margin of improvement (approximately 1% in accuracy) may not justify the additional complexity in all deployment scenarios.
3. **Feature Dependency**: The performance heavily relies on the quality and relevance of the extracted features, which may require regular updates as phishing techniques evolve.

B. False Positive Analysis

In security applications like phishing detection, false positives (legitimate websites incorrectly classified as phishing) can significantly impact user experience. We conducted an in-depth analysis of false positives produced by our meta-ensemble model.

The analysis revealed that most false positives occurred with:

1. **Newly registered legitimate domains** with limited web traffic and shorter registration periods.
2. **Websites with unconventional design patterns** or non-standard implementations of web elements.
3. **Legitimate websites using multiple redirects** or URL shortening services for marketing purposes.

These findings highlight the challenge of distinguishing between legitimate new websites and phishing attempts, as they may share similar characteristics. Future improvements could focus on developing more sophisticated features that can better differentiate these cases.

C. Real-World Deployment Considerations

Implementing the meta-ensemble approach in real-world settings requires consideration of several factors:

1. **Latency Requirements**: For browser-based phishing protection, detection must occur within milliseconds. Our analysis shows that while the meta-ensemble has higher inference latency (approximately 15ms per website) compared to individual models (3-8ms), it remains within acceptable bounds for real-time applications.
2. **Update Frequency**: Phishing techniques evolve rapidly, necessitating regular model updates. The

modular nature of our framework allows for updating individual components without retraining the entire ensemble.

3. **Resource Constraints**: For deployment on resource-limited devices, simplified versions of the ensemble can be used. Our experiments show that a reduced ensemble using only XGBoost, Random Forest, and a Voting Classifier achieves 96.89% accuracy while requiring significantly fewer computational resources.
4. **Privacy Considerations**: The feature extraction process should respect user privacy. Our approach primarily relies on URL and HTML structure analysis rather than user-specific information, minimizing privacy concerns.

VI. CONCLUSION AND FUTURE WORK

In this paper, we presented a meta-ensemble approach for phishing website detection that effectively combines multiple machine learning models through hierarchical ensemble techniques. Our experimental results demonstrated that this approach achieves superior performance compared to individual models, with an accuracy of 97.31% and an F1-score of 0.9729.

The feature importance analysis revealed that SSL certificate status, anchor URL consistency, web traffic, subdomain characteristics, and domain registration length are among the most significant indicators for distinguishing between legitimate and phishing websites. These insights can guide the development of more effective security features and user awareness programs.

While our approach shows promising results, there are several directions for future research:

1. **Incorporating Deep Learning**: Exploring deep learning models, particularly those designed for sequential or graph-structured data, could capture more complex patterns in website structure and content.
2. **Dynamic Feature Extraction**: Developing methods to automatically extract and update relevant features as phishing techniques evolve.
3. **Adversarial Testing**: Evaluating the robustness of the models against adversarial examples designed to evade detection.
4. **Explainable AI Integration**: Enhancing the interpretability of model decisions to help users understand why a website was flagged as suspicious.
5. **Transfer Learning**: Investigating transfer learning approaches to adapt pre-trained models to new phishing patterns with minimal additional data.

In conclusion, our meta-ensemble framework provides a comprehensive and effective approach to phishing website detection, offering both high accuracy and valuable insights into the characteristics of phishing attempts. By combining the strengths of multiple models and ensemble techniques, it represents a significant step toward more robust cybersecurity solutions.

VII. ACKNOWLEDGMENT

We would like to thank the Faculty of Information Technology at Hanoi University of Mining and Geology for their support and resources. We also express our gratitude to the anonymous reviewers for their valuable feedback and suggestions that helped improve the quality of this paper.

VIII. REFERENCES

[1] A. Abbasi, Z. Zhang, D. Zimbra, H. Chen, and J. F. Nunamaker Jr., "Detecting fake websites: The contribution of statistical learning theory," MIS Quarterly, vol. 34, no. 3, pp. 435-461, 2010.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] 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.

[4] A. K. Jain and B. B. Gupta, "A machine learning based approach for phishing detection using hyperlinks information," Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 5, pp. 2015-2028, 2019.

[5] T. G. Dietterich, "Ensemble methods in machine learning," in Multiple Classifier Systems, LNCS vol. 1857, Springer, 2000, pp. 1-15.

[6] E. Zhu, Y. Chen, C. Ye, X. Li, and F. Liu, "OFS-NN: An effective phishing websites detection model based on optimal feature selection and neural network," IEEE Access, vol. 7, pp. 73271-73284, 2019.

[7] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based associative classification data mining," Expert Systems with Applications, vol. 41, no. 13, pp. 5948-5959, 2014.

[8] A. Bhardwaj, V. Avasthi, H. Sastry, and G. V. Subrahmanyam, "Ransomware digital forensic investigation and importance of machine learning techniques to prevent it," in 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 2020, pp. 13-18.

[9] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785-794.

[10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in Advances in Neural Information Processing Systems, 2017, pp. 3146-3154.

[11] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in Advances in Neural Information Processing Systems, 2018, pp. 6638-6648.