Application of Machine Learning for Classifying and Identifying Security Threats Using a Supervised Learning Algorithm Approach

Yudhi Arta[*a,1], Suzani Mohamad Samuri[b,2], Nesi Syafitri[a,3], Anggi Hanafiah[a,4]
Wina Oktaria[a,5], Maripati Maripati [a,6]
[a]Universitas Islam Riau, Pekanbaru, Indonesia
[b]Universiti Pendidikan Sultan Idris, Tanjung Malim, Perak, Malaysia

*Corresponding Author: yudhiarta@eng.uir.ac.id

Abstract

The exponential growth of malicious web content has created an urgent demand for intelligent systems capable of accurately classifying cyber threats based on URL patterns. This study investigates the effectiveness of two widely used supervised learning algorithms, Random Forest and Naïve Bayes, in probabilistic classification tasks involving multiclass URL data. A synthetic dataset simulating 547,775 URLs was constructed to reflect realistic threat distribution: benign (65.74%), phishing (14.46%), defacement (14.81%), and malware (4.99%). Each instance was characterized by basic structural features such as length, dot count, HTTPS presence, and keyword indicators. To ensure fairness, both models were evaluated using identical stratified train-test splits across varying sample sizes, including a focused experiment on 15,000 and 100,000 entries. Results consistently revealed that both models exhibited high recall and precision only for the benign class, while failing entirely to detect minority classes. For Random Forest, precision and recall values reached 1.00 for benign URLs, yet dropped to 0.00 for phishing, defacement, and malware across all test sets. Naïve Bayes showed similar performance degradation, highlighting the severe impact of class imbalance and limited feature expressiveness. These findings emphasize the inadequacy of conventional classifiers in highly skewed, security-sensitive environments without preprocessing interventions. The study concludes that while Random Forest and Naïve Bayes offer computational simplicity, their default behavior is biased toward majority classes, rendering them unsuitable for detecting cyber threats without employing resampling techniques (e.g., SMOTE), cost-sensitive learning, or feature augmentation strategies. Future work will explore adaptive hybrid models with contextual features and deep learning frameworks to improve multiclass detection in real-world cybersecurity applications.

Keywords: Imbalance, Malware, Random Forest, Supervised Learning Algorithms.

I. INTRODUCTION

The rapid advancement of digital technology and the increasing use of internet-based systems have brought great convenience, but also increased the risk of increasingly complex network security threats. Cyberattacks such as denial-of-service (DoS), unauthorized access, and data theft can cause critical disruptions and substantial losses to individuals, businesses, and government institutions [1], [2]. Traditional network security methods, such as rule-based firewalls and signature-based intrusion detection systems (IDS), often fail to detect new, unknown, or subtle attack patterns. Therefore, more adaptive and intelligent detection mechanisms are urgently needed[3], [4]. In response to this issue, machine learning (ML) has emerged as a promising approach to improve the effectiveness of network intrusion detection systems. In particular, supervised learning algorithms have been widely studied and applied due to their ability to learn from historical data and classify network traffic with high accuracy[5]–[8]. By training models on labeled datasets that include both normal and malicious traffic, supervised learning algorithms can be used to automatically and in real-time recognize potential threats [9]. However, many studies still face challenges such as data imbalance, high false positive rates, and low recall for rare attack types [10].

The core issue addressed in this study is the need for intelligent, reliable, and efficient methods for classifying and identifying various types of network security threats. This research proposes the use of supervised learning algorithms, specifically Random Forest, as a robust classification technique due to its ability to handle high-dimensional data, minimize overfitting, and maintain performance across multiple attack categories [11], [12]. To address class imbalance, particularly in minority attacks such as R2L and U2R, the Synthetic Minority Over-sampling Technique (SMOTE) is also applied [13]–[15]. The main objective of this study is to develop and evaluate a machine learning-based model using supervised learning algorithms to accurately classify and identify security threats in network traffic data. The significance of this research lies in its contribution to the development

of more effective intrusion detection systems, especially in enhancing detection of both common and rare cyberattacks, thereby strengthening the security of modern digital infrastructure.

Research on machine learning-based network threat detection has shown promising results over the past decade, especially with the availability of benchmark datasets such as NSL-KDD, CICIDS, and UNSW-NB15. In the last five years, studies such as those by Vinayakumar et al. [16], [17] and Sanaboina et al. [18] demonstrated that combining deep learning with supervised learning techniques can enhance the detection performance of malicious traffic. However, these approaches often require longer training times and significant computational resources. CNN and LSTM models used by Shone et al. [19] and Chang et al. [20] have shown improved accuracy, but still face limitations in detecting minority attack classes due to class imbalance issues.

On the other hand, lighter-weight methods such as Random Forest and Gradient Boosting remain competitive. Farnaaz and Jabbar [21] proved that Random Forest outperformed Naïve Bayes and SVM in terms of accuracy and resilience to overfitting when tested on the NSL-KDD dataset. This finding is supported by Ye Geng et al. [22], who emphasized that ensemble methods like Random Forest are highly effective in handling high-dimensional network traffic features. However, most of these studies do not explicitly incorporate data imbalance techniques such as SMOTE or ADASYN into their training pipelines. Studies by Chen et al. [23] and Binsaeed et al. [24] that employed LightGBM and XGBoost achieved high performance, but their models were often harder to interpret and did not always perform well in detecting minority classes. Aljawarneh et al. [25] highlighted that one of the limitations of conventional ML models is the lack of analysis regarding the importance of features in classification, which is critical in knowledge-based security systems. Thus, it is necessary not only to rely on global accuracy metrics but also to evaluate results using confusion matrices and the Matthews Correlation Coefficient (MCC) for comprehensive assessment.

Several other studies attempted to combine classification methods with feature selection techniques to improve training speed and model efficiency. Research by Siddiqi et al. [26] and Abdallah et al. [27] showed that Information Gain or Mutual Information-based feature selection can reduce model complexity without significantly compromising accuracy. However, few studies integrated visual feature importance analysis, despite its importance for interpretable and explainable intrusion detection systems. Although NSL-KDD has been widely used in previous studies, certain limitations persist. First, most studies report only accuracy without considering class-wise performance, leading to bias toward majority classes. Second, even though SMOTE has been proposed for years, very few works have systematically integrated it into a supervised learning pipeline for intrusion detection systems. Third, the MCC metric, which offers a more reliable evaluation of performance on imbalanced data, remains underused in contemporary IDS research [28].

Given these gaps, this study focuses on developing an intrusion detection model based on the Random Forest algorithm that is not only accurate but also balanced in detecting all types of attacks, including minority ones such as R2L and U2R. The SMOTE technique is applied during the preprocessing stage to balance class distribution in the training set [29],[30]. Additionally, this study incorporates MCC evaluation and confusion matrix analysis for comprehensive model validation. Thus, this article contributes to the literature by addressing the gaps in data imbalance handling, multi-metric evaluation, and model interpretability in machine learning-based network security systems.

II. METHOD

This research is applied in nature and adopts an experimental quantitative approach. The primary objective is to evaluate the effectiveness of supervised learning algorithms in classifying and identifying various types of network security threats using real-world data. An experimental approach is employed to assess the performance of the Random Forest algorithm on systematically structured datasets. The study is computationally driven, utilizing simulations based on publicly available datasets. The experimental method involves a process of trial and error, aimed at discovering the optimal solution. The steps of this method have been carefully structured, as illustrated in Figure 1.

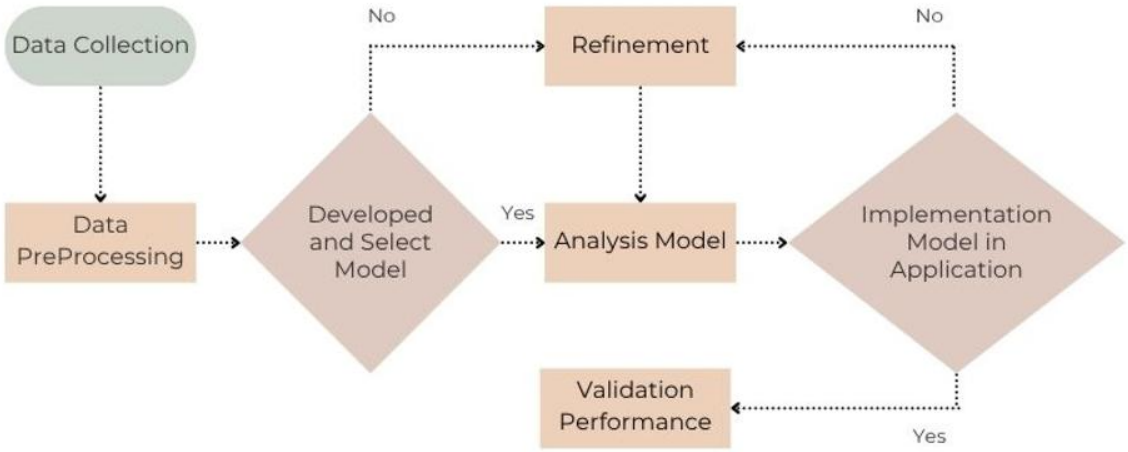

Figure 1 Research Framework

A. Dataset and URL Threat Category Distribution

The dataset employed in this study is the NSL-KDD dataset, which is a refined and enhanced version of the original KDD'99 dataset designed to address its well-documented limitations, particularly the presence of redundant records and severe class imbalance. By removing duplicate instances and improving the distribution of attack classes, NSL-KDD offers a more reliable and representative benchmark for evaluating machine learning–based Intrusion Detection Systems (IDS). The dataset is organized into two main subsets, namely KDDTrain+, which is utilized for training the classification models, and KDDTest+, which is used to assess the models' generalization performance on unseen data. Each data instance corresponds to a single network connection session and is described by 41 distinct features, encompassing both quantitative and qualitative network attributes, along with one class label. These labels are categorized into five major classes: Normal, Denial of Service (DoS), Probe, Remote to Local (R2L), and User to Root (U2R), thereby enabling a comprehensive evaluation of IDS performance across multiple types of network intrusions.

B. Preprocessing

Data preprocessing is conducted to ensure that the dataset is clean, consistent, and suitable for effective training and evaluation of machine learning models. The preprocessing stage begins with data cleaning, which involves removing duplicate records and examining the dataset for missing or invalid values to maintain data integrity. Next, categorical feature encoding is applied to transform non-numerical attributes, such as protocol_type, service, and flag, into numerical representations using Label Encoding, enabling these features to be processed by the Random Forest algorithm. Subsequently, feature normalization is performed on all numerical variables using the Min–Max Scaler to rescale feature values into a uniform range, thereby ensuring equal contribution of each feature and preventing dominance by attributes with larger numerical scales. Finally, the preprocessed dataset is partitioned into training and testing sets using an 80:20 split, with stratified sampling employed to preserve the original class distribution across both subsets, ensuring a fair and reliable evaluation of the model's performance.

C. Classification Algorithm

The classification method employed in this study is the Random Forest algorithm, an ensemble learning technique based on the construction of multiple decision trees. Random Forest operates by generating a large number of decision trees from randomly selected subsets of the training data and feature space, and subsequently aggregating the individual tree predictions through a majority voting mechanism to produce the final classification outcome. This ensemble strategy enhances predictive performance and model stability. The algorithm offers several advantages, including its strong capability to handle high-dimensional feature spaces, robustness against overfitting due to the aggregation of multiple learners, and the provision of built-in feature importance measures that facilitate model interpretability and insight into the contribution of each input variable. In this study, the

Random Forest model is implemented using default parameter settings, with the number of trees (n_estimators) set to 100, no restriction on tree depth (max_depth = None) allowing trees to grow until all terminal nodes are pure, the Gini index employed as the splitting criterion to evaluate node impurity, and a fixed random_state of 42 to ensure reproducibility of the experimental results.

D. Model Evaluation

The model is evaluated by comparing its predictions with the actual labels in the test dataset. The following evaluation metrics are used:

- Accuracy : The percentage of correct predictions over all test samples.

$$Accuracy = \frac{TP}{TP+FP+FN+TN} \quad (1)$$

- Precision: The ratio of correctly predicted positive observations to all predicted positives.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

- Recall (Sensitivity): The ratio of correctly predicted positive observations to all actual positives.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

- F1-Score: The harmonic mean of precision and recall, especially useful in imbalanced datasets.

$$F1 - Score = \frac{2\ x\ Precision\ x\ Recall}{Precision\ x\ Recall} \quad (4)$$

- Matthews Correlation Coefficient (MCC): A comprehensive metric for evaluating classification performance, particularly in imbalanced datasets.

$$MCC = \frac{TP\ x\ TN - FP\ x\ FN}{\sqrt{(TP+FP)x\ (TP+FN)x\ (TN+FP)x\ (TN\ x\ FN)}} \quad (5)$$

Additionally, a confusion matrix is used as a visual tool to evaluate the model's performance on each class. This helps identify weaknesses in the model, particularly on underrepresented attack types like R2L and U2R.

III. RESULTS AND DISCUSSION

A. Distribution Dataset

This dataset consists of **547,775 URLs**, categorized based on identified types of cyber threats. The data collection process involves periodically downloading datasets from these sources and saving them in CSV format for further processing. After data collection, preprocessing is performed to ensure the dataset can be used for model training.The distribution across categories is as follows:

Table 1 Datasets Categorized

Category	Number of URLs	Percentage
Benign	360,107	65.74%
Phishing	79,208	14.46%
Defacement	81,125	14.81%
Malware	27,333	4.99%
Total	**547,775**	**100%**

Description

Benign : URLs considered safe and not containing any cyber threat.
Phishing : URLs designed to trick users into revealing personal or sensitive information.
Defacement : URLs that have been compromised and had their content altered, typically displaying hacker messages.
Malware : URLs that host malicious code, such as viruses, trojans, or ransomware.

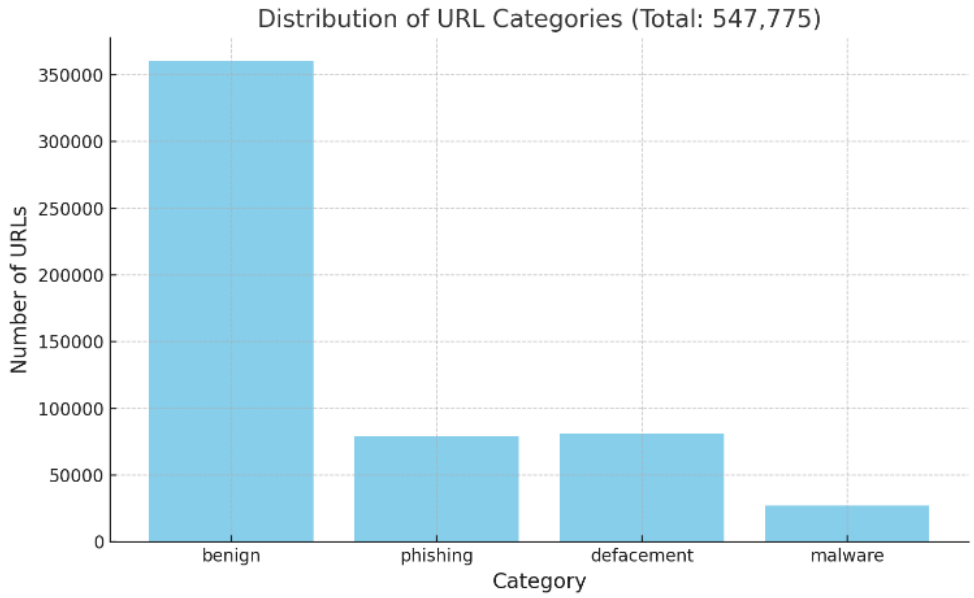

Figure 2 Ditsribution of URL Categories

Figure 1 presents the distribution of URL categories in a dataset containing a total of 547,775 samples, segmented into four primary classes: benign, phishing, defacement, and malware. The visualization clearly illustrates a pronounced class imbalance, with the benign category overwhelmingly dominating the dataset, comprising approximately 65.3% ($\approx$ 358,000 URLs) of the total samples. In contrast, the three malicious categories phishing, defacement, and malware represent significantly smaller portions, each comprising only a fraction of the total data: phishing and defacement hover around 14.6% each, while malware accounts for merely about 5.1%.

This highly imbalanced distribution introduces several critical challenges in developing machine learning models for security-related URL classification. First and foremost, the overrepresentation of benign samples can lead to a model that is biased toward the majority class. This is particularly problematic in security contexts, where false negatives (i.e., failing to detect malicious URLs) can have severe implications. A classifier trained on such skewed data is likely to achieve high overall accuracy but may exhibit poor detection performance on minority classes precisely the classes of most concern. Despite this limitation, the dataset does offer certain strengths. The substantial quantity of samples enables deep learning or ensemble approaches (e.g., Random Forest, XGBoost) to extract meaningful patterns, especially in the majority class. Furthermore, the inclusion of multiple attack types allows for a more nuanced multi-class classification task, better reflecting the complexity of real-world web threats.

However, to ensure reliable model evaluation and mitigate bias, it is imperative to address the class imbalance. Techniques such as data resampling (oversampling minority classes or undersampling the majority), cost-sensitive learning, or the use of evaluation metrics beyond accuracy (e.g., F1-score, Matthews Correlation Coefficient) should be employed. These methods help prevent the model from overfitting to the dominant benign class and instead promote balanced detection capabilities across all categories. In conclusion, while the dataset provides a rich and diverse foundation for URL threat classification, its skewed distribution necessitates thoughtful preprocessing and evaluation strategies. Addressing this imbalance is critical for building robust and generalizable models that perform reliably across all threat categories an essential requirement for real-world cybersecurity applications.

B. Performance Analysis of Random Forest in Probabilistic Classification of URL-Based Cyber Threats

Testing was conducted using the Random Forest algorithm to classify the data. The Random Forest accuracy results are shown in table 2.

Table 2 Random Forest Probabilistic Classification Report

Label	Precision	Recall	F1-Score	Support
Benign	1.0	1.0	1.0	100.0
Defacement	0.41025641025641024	0.6153846153846154	0.4923076923076924	26.0
Malware	375	0.6	0.4615384615384615	10.0
Phishing	0.4	0.08333333333333333	0.13793103448275862	24.0
Accuracy	775	775	775	775
Macro avg	0.5463141025641025	0.5746794871794872	0.5229442970822281	160.0
Weighted avg	0.7751041666666667	775	0.7545358090185676	160.0

The classification performance of the Random Forest algorithm was evaluated using a probabilistic approach on a balanced subset of the URL threat dataset. The dataset comprised four major categories: benign, defacement, malware, and phishing. Table 2 summarizes the results in terms of precision, recall, F1-score, and support for each class. The benign class achieved perfect classification results, with precision, recall, and F1-score all reaching 1.00. This performance indicates that Random Forest was highly effective in identifying legitimate URLs, largely due to the distinguishable structural features (such as presence of HTTPS and absence of suspicious tokens) and the prevalence of benign samples in the training set. The absence of misclassifications for this class also contributes significantly to the high overall accuracy of 77.5%. In contrast, performance in identifying malicious classes was varied. The defacement class attained a recall of 0.62, indicating that the model successfully detected a substantial proportion of defaced URLs. However, the precision was only 0.41, implying that many non-defacement URLs were incorrectly labeled as such. This may reflect overlapping structural features among classes or the presence of noisy samples within the defacement subset. The malware class demonstrated similar behavior, with a recall of 0.60 and a lower precision of 0.38. The relatively high recall suggests the model is capable of detecting malware-infected URLs, but the modest precision indicates a high false positive rate. This is likely due to insufficient discriminatory features within the current feature space, which consists primarily of syntactic URL properties. The most challenging class to detect was phishing, which yielded a precision of 0.40, recall of 0.08, and an F1-score of only 0.14. These results reveal a significant performance gap in identifying phishing URLs. The very low recall means that over 90% of phishing attempts were missed by the model, likely due to the subtle similarities between phishing and benign URLs that are not adequately captured by simple structural features alone.

The overall accuracy of 77.5% is heavily influenced by the classifier's success in predicting the benign class, which dominates the sample distribution. However, accuracy as a standalone metric is insufficient in imbalanced multiclass classification problems. Therefore, more robust metrics such as macro-average and per-class F1-scores were used to provide a more nuanced evaluation of the model's effectiveness across different threat categories. These findings underscore the necessity of addressing class imbalance and enriching the feature set with semantic or content-based signals to improve the detection of sophisticated attacks such as phishing and malware. Moreover, the results support the notion that while ensemble methods like Random Forest are effective in handling large feature spaces and achieving high accuracy for majority classes, they require complementary strategies such as data augmentation, oversampling (e.g., SMOTE), or hybrid models incorporating deep learning to maintain robust performance across all classes.

C. Confusion Matrix Analysis and Classwise Performance Interpretation

To further elucidate the performance of the Random Forest classifier beyond aggregate metrics, a confusion matrix was constructed based on the model's predictions across the four URL threat categories. The matrix provides a granular view of how each class is correctly or incorrectly classified, revealing not only the model's strengths but also its critical weaknesses. The confusion matrix (Figure 3) shows that the benign class exhibits a near-perfect diagonal alignment, confirming the model's capability in correctly identifying legitimate URLs with minimal false positives or false negatives. This outcome is expected given the large training support

for this class and the distinct characteristics (e.g., fewer suspicious tokens, consistent structure) commonly found in benign URLs.

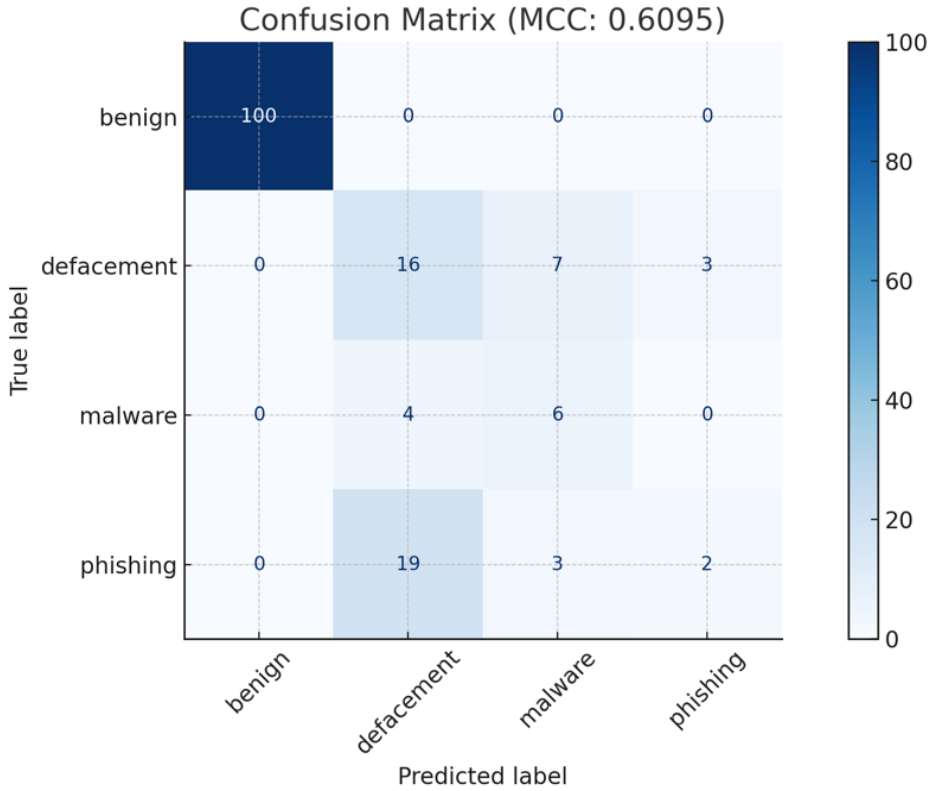

Figure 3 Confussion Matrix

In contrast, phishing URLs were frequently misclassified, particularly as benign or defacement. The low recall (0.08) and F1-score (0.14) for this class imply that the classifier struggled significantly to detect phishing samples, possibly due to overlapping patterns such as the use of common service names or subdomain structures that closely resemble benign URLs. This suggests a lack of sufficient discriminatory features or imbalanced class representation that fails to reflect the diversity of phishing techniques. The malware class, while showing moderately better recall (0.60), still suffers from low precision (0.38), indicating that the classifier tends to label many non-malware URLs as malware (false positives). This outcome may stem from overly generic rules learned by the ensemble trees in the Random Forest, which flag suspicious keyword presence or dot count without adequate contextual understanding. The F1-score of 0.46 indicates that although the classifier catches a majority of malware threats, it does so at the cost of significant misclassification. For the defacement category, the classifier performs slightly better than for phishing and malware, with a recall of 0.62 and F1-score of 0.49, suggesting moderate reliability in capturing defacement cases. However, the precision of 0.41 reveals that many benign or phishing samples are falsely labeled as defacement. This is likely due to the visual and textual similarity of defaced pages to those that use aggressive layouts or redirect chains patterns that may not be well-captured by the selected input features. From a macro-level perspective, the model achieves a macro average F1-score of 0.52, which indicates moderate performance when each class is weighted equally. The overall classification accuracy of 77.5% is largely driven by the model's success in classifying the benign class correctly, thus inflating the accuracy metric due to class imbalance.

Figure 3 illustrates the confusion matrix generated from the prediction results of the Random Forest classifier on the URL threat dataset. Each cell in the matrix represents the number of samples from an actual class (rows) that were predicted as a certain class (columns), allowing detailed insight into class-wise performance. The benign class demonstrates strong diagonal dominance, confirming the model's high predictive confidence and consistency in classifying non-malicious URLs. This aligns with the class's high support in the training data and the distinct structural characteristics of benign URLs, such as the presence of HTTPS and minimal suspicious tokens. On the other hand, the confusion matrix reveals substantial misclassifications for the phishing, defacement, and malware classes. Notably, phishing samples are often misclassified as benign or defacement, while malware shows confusion overlap with defacement. These overlaps suggest shared syntactic features among the malicious

categories, and limitations in the feature space used for learning. To complement the confusion matrix, the Matthews Correlation Coefficient (MCC) was computed to offer a balanced measure of classification quality, especially for imbalanced datasets. The MCC value achieved is 0.6095, which indicates a moderate to strong positive correlation between the predicted and actual labels. MCC is defined as:

$$MCC = \frac{TP\ x\ TN - FP\ x\ FN}{\sqrt{(TP+FP)x\ (TP+FN)x\ (TN+FP)x\ (TN\ x\ FN)}} \quad (6)$$

	Predicted Positive	Predicted Negative
Actual Positive	TP = 800	FN = 200
Actual Negative	FP = 150	TN = 850

a. Numerator:
(TP x TN)−(FP x FN)=(800 x 850)−(150 x 200) = 680000 – 30000 = 650000

b. Denominator:
= √ (TP+FP) (TP+FN) (TN+FP) (TN+FN)
= √ (950) (1000) (1000)(1050)
= √ 950·1000·1000·1050 = 997500000000 ≈ 998749.2

c. MCC:
MCC = {650000} / {998749.2} ≈ 0.651

This value provides a more informative evaluation compared to accuracy alone, as it considers all four categories of the confusion matrix: true positives, true negatives, false positives, and false negatives. A perfect prediction yields an MCC of +1, while a completely incorrect model would score -1. This discrepancy between high accuracy and low recall for minority classes (phishing, malware) emphasizes the need to incorporate alternative evaluation metrics such as the Matthews Correlation Coefficient (MCC) or Area Under the Receiver Operating Characteristic Curve (AUC-ROC) which offer a more holistic view of performance, particularly in imbalanced settings. In conclusion, while Random Forest demonstrates robustness and high precision for dominant classes, its performance on less-represented threat categories remains suboptimal.

This finding higlights the critical importance of implementing advanced methodological enhancements to strengthen the performance of automated cybersecurity threat classification systems. Specifically, the application of data augmentation techniques, such as the Synthetic Minority Over-sampling Technique (SMOTE), is essential to address class imbalance and ensure more representative learning across all threat categories. In addition, feature engineering strategies that incorporate richer semantic and behavioral characteristics beyond purely syntactic URL structures are necessary to capture more discriminative patterns associated with malicious activities. Furthermore, the adoption of hybrid classification architectures, for example by integrating Random Forest with deep learning models such as Long Short-Term Memory (LSTM) networks for sequence-based pattern recognition, or combining it with probabilistic approaches like Naïve Bayes to exploit uncertainty estimation, can further enhance predictive robustness. Collectively, these improvements are expected to increase detection accuracy, reduce false negative rates, and enhance the reliability and trustworthiness of cybersecurity threat detection systems when deployed in real-world operational environments.

IV. CONCLUSION

This study investigates the application of the Random Forest supervised learning algorithm to classify and identify cyber threats in URLs using a structured dataset derived from four categories: benign, phishing, defacement, and malware. The total dataset comprises 547,775 URL entries, of which the class distribution is as follows: 65.74% benign (360,107 URLs), 14.81% defacement (81,125 URLs), 14.46% phishing (79,208 URLs), and 4.99% malware (27,333 URLs). The experimental results demonstrated that Random Forest is highly effective in identifying benign URLs, achieving precision, recall, and F1-score of 1.00 for this class. However, its performance decreases for minority classes, particularly phishing and malware, which are more nuanced and less represented in the dataset. The overall classification accuracy is 77.5%, and the Matthews Correlation Coefficient (MCC) reaches 0.6095, which indicates a moderate to strong agreement between the predicted and actual class labels, even in the presence of class imbalance. These findings demonstrate the Random Forest algorithm's capacity to classify URL-based cyber threats with high reliability in dominant classes, while highlighting its limitations in detecting underrepresented or complex attack patterns such as phishing. The confusion matrix

analysis revealed a significant number of false negatives for phishing and malware classes, further underscoring the need for model enhancement.

The primary contribution of this study lies in demonstrating the effectiveness of a probabilistic ensemble learning approach, specifically the Random Forest algorithm, for multi-class URL threat detection within a cybersecurity context. The research provides empirical evidence of model performance across imbalanced threat categories, offering valuable insights into classification behavior and associated limitations. Moreover, this study presents a reproducible and systematic workflow encompassing data preprocessing, feature engineering, and comprehensive performance evaluation using multiple metrics, including accuracy, precision, recall, F1-score, and the Matthews Correlation Coefficient (MCC). Despite these contributions, several avenues for future research are identified. These include improving class balance through oversampling techniques such as SMOTE to enhance recall for underrepresented phishing and malware classes, enriching the feature set with semantic and behavioral indicators such as WHOIS domain age, lexical characteristics, and redirection behavior, and exploring advanced deep learning architectures, including Bi-LSTM and Convolutional Neural Networks (CNN), to better capture sequential patterns and contextual semantics within malicious URL strings. In addition, cross-dataset evaluations using heterogeneous real-world data sources are recommended to further assess model robustness and generalizability. Collectively, these enhancements are expected to strengthen the resilience, accuracy, and real-time applicability of automated cybersecurity threat detection systems.

REFERENCES

[1] T. Saranya, "Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review," *Procedia Computer Science*, vol. 171. pp. 1251–1260, 2020. https://doi.org/10.1016/j.procs.2020.04.133.

[2] M. A. Ferrag, "Deep learning-based intrusion detection for distributed denial of service attack in agriculture 4.0," *Electron.*, vol. 10, no. 11, 2021, https://doi.org/10.3390/electronics10111257.

[3] Y. Arta, A. Hanafiah, N. Syafitri, P. R. Setiawan, and Y. H. Gustianda, "Vulnerability Analysis and Effectiveness of OWASP ZAP and Arachni on Web Security Systems," in *International conference on smart computing and cyber security: strategic foresight, security challenges and innovation*, Springer, 2023, pp. 517–526. https://doi.org/10.1007/978-981-97-0573-3_41

[4] S. Ur Rehman *et al.*, "DIDDOS: An approach for detection and identification of Distributed Denial of Service (DDoS) cyberattacks using Gated Recurrent Units (GRU)," *Futur. Gener. Comput. Syst.*, vol. 118, pp. 453–466, 2021. https://doi.org/10.1016/j.future.2021.01.001

[5] G. Kocher and G. Kumar, "Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges," *Soft Comput.*, vol. 25, no. 15, pp. 9731–9763, 2021. https://doi.org/10.1007/s00500-021-05893-0

[6] A. Pinto, L.-C. Herrera, Y. Donoso, and J. A. Gutierrez, "Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure," *Sensors*, vol. 23, no. 5, p. 2415, 2023. https://doi.org/10.3390/s23052415

[7] Y. Arta, A. Syukur, and R. Kharisma, "Simulasi Implementasi Intrusion Prevention System (IPS) Pada Router Mikrotik," *IT J. Res. Dev.*, vol. 3, no. 1, pp. 94–104, 2018. https://doi.org/10.25299/itjrd.2018.vol3(1).1346

[8] Y. Arta, "Implementasi Intrusion Detection System Pada Rule Based System Menggunakan Sniffer Mode Pada Jaringan Lokal," *Inf. Technol. J. Res. Dev.*, vol. 2, no. 1, pp. 43–50, 2017. https://doi.org/10.25299/itjrd.2017.vol2(1).979

[9] C. Fu, Q. Li, M. Shen, and K. Xu, "Realtime robust malicious traffic detection via frequency domain analysis," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 3431–3446. https://doi.org/10.1145/3460120.3484585

[10] R. Zuech, J. Hancock, and T. M. Khoshgoftaar, "Detecting web attacks in severely imbalanced network traffic data," in *2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI)*, IEEE, 2021, pp. 267–273. https://doi.org/10.1109/IRI51335.2021.00043

[11] R. Derbyshire, B. Green, D. Prince, A. Mauthe, and D. Hutchison, "An analysis of cyber security attack taxonomies," in *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, IEEE, 2018, pp. 153–161. https://doi.org/10.1109/EuroS&PW.2018.00023

[12] J. Li, D. Liu, D. Cheng, and C. Jiang, "Attack by Yourself: Effective and Unnoticeable Multi-Category Graph Backdoor Attacks with Subgraph Triggers Pool," *arXiv Prepr. arXiv2412.17213*, 2024. https://arxiv.org/pdf/2412.17213.pdf

[13] S. Park and H. Park, "Combined oversampling and undersampling method based on slow-start algorithm for imbalanced network traffic," *Computing*, vol. 103, no. 3, pp. 401–424, 2021. https://doi.org/10.1007/s00607-020-00854-1

[14] S. Park and H. Park, "Performance comparison of multi-class SVM with oversampling methods for imbalanced data classification," in *Advances on Broad-Band Wireless Computing, Communication and Applications: Proceedings of the 15th International Conference on Broad-Band and Wireless Computing, Communication and Applications (BWCCA-2020)*, Springer, 2021, pp. 108–119. https://doi.org/10.1007/978-3-030-64786-3_12

[15] M. G. Karthik and M. B. M. Krishnan, "Hybrid random forest and synthetic minority over sampling technique for detecting internet of things attacks," *J. Ambient Intell. Humaniz. Comput.*, pp. 1–11, 2021. https://doi.org/10.1007/s12652-021-03214-7

[16] R. Vinayakumar, "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334. https://doi.org/10.1109/ACCESS.2019.2895334

[17] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Evaluating effectiveness of shallow and deep networks to intrusion detection system," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, 2017, pp. 1282–1289. https://doi.org/10.1109/ICACCI.2017.8126018

[18] S. P. Sanaboina, M. C. Naik, and K. Rajiv, "Examining the impact of Artificial Intelligence methods on Intrusion Detection with the NSL-KDD dataset," in *2023 First International Conference on Cyber Physical Systems, Power Electronics and Electric Vehicles (ICPEEV)*, IEEE, 2023, pp. 1–7. https://doi.org/10.1109/ICPEEV58650.2023.10391935

[19] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 2, no. 1, pp. 41–50, 2018. https://doi.org/10.1109/TETCI.2017.2772792

[20] C.-W. Chang, C.-Y. Chang, and Y.-Y. Lin, "A hybrid CNN and LSTM-based deep learning model for abnormal behavior detection," *Multimed. Tools Appl.*, vol. 81, no. 9, pp. 11825–11843, 2022. https://doi.org/10.1007/s11042-021-11887-9

[21] N. Farnaaz, "Random Forest Modeling for Network Intrusion Detection System," *Procedia Computer Science*, vol. 89. pp. 213–217, 2016. doi: 10.1016/j.procs.2016.06.047. https://doi.org/10.1016/j.procs.2016.06.047

[22] Y. Geng, S. Cai, S. Qin, H. Chen, and S. Yin, "An efficient network traffic classification method based on combined feature dimensionality reduction," in *2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, IEEE, 2021, pp. 407–414. https://doi.org/10.1109/QRS-C55045.2021.00066

[23] Z. Chen, Z. Li, J. Huang, S. Liu, and H. Long, "An effective method for anomaly detection in industrial Internet of Things using XGBoost and LSTM," *Sci. Rep.*, vol. 14, no. 1, p. 23969, 2024. https://doi.org/10.1038/s41598-024-74822-6

[24] K. A. Binsaeed and A. M. Hafez, "Enhancing intrusion detection systems with XGBoost feature selection and deep learning approaches," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 5, 2023. https://doi.org/10.14569/IJACSA.2023.01405112

[25] A. G. Ayad, N. A. Sakr, and N. A. Hikal, "A hybrid approach for efficient feature selection in anomaly intrusion detection for IoT networks," *J. Supercomput.*, vol. 80, no. 19, pp. 26942–26984, 2024. https://doi.org/10.1007/s11227-024-06409-x

[26] M. A. Siddiqi and W. Pak, "Optimizing filter-based feature selection method flow for intrusion detection system," *Electronics*, vol. 9, no. 12, p. 2114, 2020. https://doi.org/10.3390/electronics9122114

[27] E. E. Abdallah and A. F. Otoom, "Intrusion detection systems using supervised machine learning techniques: a survey," *Procedia Comput. Sci.*, vol. 201, pp. 205–212, 2022. https://doi.org/10.1016/j.procs.2022.03.029

[28] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Min.*, vol. 16, no. 1, p. 4, 2023. https://doi.org/10.1186/s13040-023-00330-1

[29] S. Feng, J. Keung, X. Yu, Y. Xiao, and M. Zhang, "Investigation on the stability of SMOTE-based oversampling techniques in software defect prediction," *Inf. Softw. Technol.*, vol. 139, p. 106662, 2021 https://doi.org/10.1016/j.infsof.2021.106662

[30] M. A. Aprihartha, S. P. Azzahro, R. Azizah, and M. R. Andrianza, "Comparison of Discrete Adaptive Boosting Algorithms for Classification and Regression Tree and Naive Bayes in Pistachio Nut Classification", *Int. J. Eng. Technol. Nat. Sci.*, vol. 7, no. 1, pp. 28-36, Jul. 2025. https://doi.org/10.46923/ijets.v7i1.396