Supervised Machine Learning for Scalable and Robust Cybersecurity A Framework for Anomaly Detection and Threat Mitigation

Mehak Eman , Azka Mir , Noor Amin *

Posted Date: 11 November 2025

doi: 10.20944/preprints202511.0761.v1

Keywords: cyber security; machine learning; supervised machine learning; random forest; decision trees; KNN; naive Bayes

 doi:10.20944/preprints202511.0761.v1

Article

Supervised Machine Learning for Scalable and Robust Cybersecurity A Framework for Anomaly Detection and Threat Mitigation

Mehak Eman [1], Azka Mir [1] and Noor Ul Amin [2,*]

[1] Department of Software Engineering, University of Sialkot, Sialkot, Pakistan

[2] School of Computer Science, Taylor's University, Subang Jaya, Malaysia

* Correspondence: nooraminnawab@gmail.com

Abstract

With the growing complexity of cyber threats, there is a critical need for improved solutions for real-time detection and mitigation. This paper explores the use of Naïve Bayes, K-Nearest Neighbors, Decision Trees, and Random Forest algorithms, all implemented using RapidMiner due to its user-friendly interface. Model performance and reliability were enhanced through essential preprocessing steps, including feature selection, normalization, and cross-validation. Google Colab was used for model training and optimization. This study highlights the importance of effective data preparation and algorithm selection in building scalable and robust machine learning models for cybersecurity. Among the evaluated models, Random Forest achieved the highest accuracy (99.04%), followed closely by KNN (98.84%).

Keywords: cyber security; machine learning; supervised machine learning; random forest; decision trees; KNN; naive Bayes

1. Introduction

The rapid growth of cybersecurity threats has driven the need for advanced real-time detection and mitigation techniques. Machine learning is increasingly being used for cyber threat detection and has brought significant advancements in this field. Both supervised and unsupervised techniques are widely applied to address challenges such as cyber threat detection, anomaly detection, network traffic monitoring, and intrusion detection systems (IDS) [1]. Among these, supervised machine learning particularly classification, has become the cornerstone of real-world threat detection and mitigation, as the cyber threat landscape evolves at an alarming pace [20]. Classification algorithms, using labeled datasets, enable the identification of malicious activities, including but not limited to network intrusions, phishing attacks, fraud, and even some forms of ransomware prevention [21].

These methods are robust, capable of detecting anomalies with high accuracy, and can infer potential threats based on identified patterns [2]. Their ability to learn from historical data and generalize to unseen cyber threat behaviors makes them invaluable in responding to the constantly evolving cybersecurity environment [22]. The effectiveness of classification techniques is often evaluated by their detection rates and their ability to minimize false positives, which helps reduce alert fatigue in Security Operations Centers (SOCs) [23].

Random Forest models are particularly valued for their ability to handle high-dimensional data and produce interpretable results [3]. In cybersecurity, where the volume and complexity of information is enormous, this capability is essential. Similarly, SVM, with their use of hyperplanes, can separate complex data patterns and are effective in tasks such as spam filtering and intrusion prevention [4]. On the other hand, security professionals often favor simpler models like Decision Trees due to their interpretability and ease of use [24].

 doi:10.20944/preprints202511.0761.v1

Although the success of supervised classification techniques, their practical feasibility must be critically evaluated. One of the major challenges is data quality machine learning models do not perform well on noisy or inconsistent data [5]. Scalability is another concern; models must be capable of processing large volumes of data to detect cyber threats in real time[9,10]. Additionally, addressing issues such as data privacy, algorithmic fairness, and ethics is essential to build trust in AI-driven cybersecurity solutions [6].

This paper focuses on leveraging supervised classification techniques to enhance cyber threat detection. Using RapidMiner, a no-code platform known for machine learning and data analysis, accurate and efficient models were developed. RapidMiner was chosen for its user-friendly interface and ability to handle large datasets with built-in features. After designing the model in RapidMiner, it was further trained and validated on a cloud-based platform, Google Colab, to ensure its effectiveness.

The goal is to develop a scalable and interpretable anomaly detection system by utilizing RapidMiner's robust preprocessing tools and Google Colab's efficient training environment. This study emphasizes the importance of feature selection, model selection, and data preprocessing to improve model accuracy and consistency while reducing false positives[14]. Ultimately, this paper presents a comprehensive framework for developing effective anomaly detection systems using advanced machine learning techniques[25–27].

2. Literature Review

ML has become a crucial tool in enhancing cybersecurity, particularly through supervised, unsupervised, and deep learning techniques for real-time threat detection, anomaly identification, and network intrusion prevention. [7] used supervised learning models for intrusion detection, demonstrating high real-time responsiveness.

[8] achieved an accuracy of 92.5% in high-risk scenarios, highlighting the importance of effective feature selection and robust model design. Unsupervised algorithms like K-Means clustering and hybrid approaches have also shown promise. In the domain of federated learning, researchers [8] employed autoencoders and K-Means clustering to secure IoT systems while managing computational constraints. Farooq and Otaibi [12] demonstrated that clustering algorithms like DBSCAN and One-Class SVM effectively reduce false alarms in large-scale Security Operations Centers (SOCs).

Deep learning is another powerful tool for addressing complex cyber threats. [13] showed that real-time threat detection is possible using CNN and RNN, achieving improved accuracy. [11] proposed a CNN-QRNN model for cyber threat detection in smart cities, achieving high precision on BoT-IoT and TON_IoT datasets. However, their reliance on synthetic data limits real-world applicability.

[15] reported detection accuracies of 91.15% for Random Forest, 88.13% for Decision Trees, 80.82% for SVM, and 68.73% for GNB, although dataset availability remains a challenge. [22] explored neural networks and behavioral modeling to support adaptive classification of emerging threats, aiming to reduce false positives in network traffic analysis.

Supervised, unsupervised, and reinforcement learning have also been applied to fraud detection and ransomware prevention. [18] highlighted the challenges of inconsistent input data and limited model explainability. [19] optimized the Feedzai platform to achieve real-time fraud detection with 83.94% accuracy using Random Forest. Similarly, [16] emphasized the need for ML model scalability in large network environments, proposing a Cyber Threat Detection Graph (CTDG) to enable real-time anomaly analysis.

Machine learning models still face considerable limitations. As noted by [1], ML algorithms struggle with noisy, inconsistent data. This study particularly emphasizes the potential of Random Forest for fraud detection. Yet, real-time processing of massive datasets and integration with extensive network infrastructures remain difficult [17]. These challenges are compounded by the rise of adversarial threats, which can render models ineffective through subtle input manipulation.

 doi:10.20944/preprints202511.0761.v1

Ethical and privacy considerations must be addressed. The development of cybersecurity models should strive to balance innovation with respect for user privacy and algorithmic fairness. Emerging solutions are beginning to address these challenges. For instance, federated learning enables collaborative training without exposing sensitive data [8].

3. Methodology

This study presents a structured framework to evaluate the effectiveness of supervised machine learning techniques in cybersecurity. The dataset used is a standard intrusion detection benchmark obtained from Kaggle, consisting of 42 features and 25,192 instances, with the target labels representing normal and anomalous behavior. Because the dataset includes a wide range of attributes such as network traffic statistics, protocol types, connection duration, and service categories it is well-suited for assessing how effectively machine learning models[28–30] can perform in cybersecurity tasks. The features are well-balanced, comprising both numerical and categorical data, which reflects real-world scenarios where subtle anomalies must be detected within large volumes of data.

The study evaluates the performance of four widely used supervised classification algorithms Naïve Bayes, KNN, Decision Trees, and Random Forest using this benchmark dataset. These models were implemented in RapidMiner, and the algorithm yielding the best results was further trained and optimized. Additionally, the structured framework offers not only an assessment of each algorithm's strengths and weaknesses but also insights into their scalability, interpretability, and applicability for real-time anomaly detection.

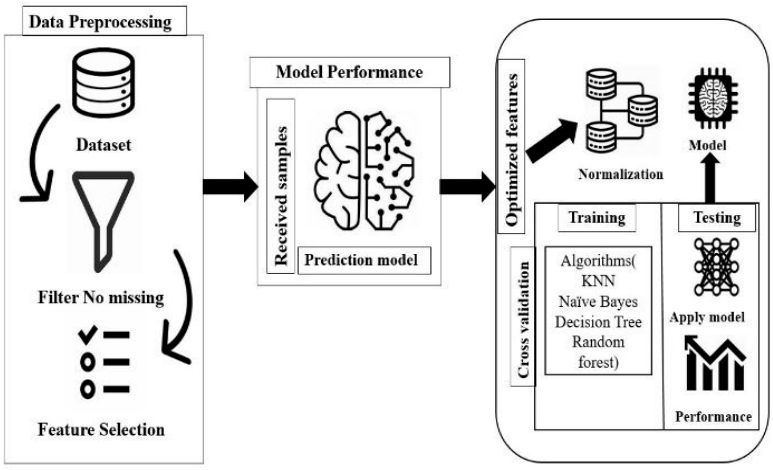

Figure 1. Framework for Anomaly.

In cybersecurity but also provides a clear comparison overview of the individual algorithms. This work adds to the effort of developing robust machine learning models specifically for the particular and constantly evolving domain of cybersecurity threats.

3.1. Data Preprocessing

The first step in model development is preparing the dataset to be suitable for machine learning. Machine learning algorithms are the most important part of today's algorithms, but without proper preprocessing, they can deliver poor or unreliable results even for the most precise algorithms. While the overall process of data preparation includes a series of key steps to improve data quality, improve model accuracy, and reduce computational complexity, there are a suite of nifty sub steps implemented around these key steps which are of fundamental importance to successful data preparation. Moreover, these steps also ensure that the findings from the analysis of the dataset are not derailed by inconsistencies and if not, effective you will have weak results.

 doi:10.20944/preprints202511.0761.v1

3.2. Feature Selection

Irrelevant features were identified and removed to improve model efficiency and accuracy. This step is crucial to reducing the dataset dimensionality, filtering out the noise, and speeding things up. After an analysis of the relevance and impact of these features on the model, these features were excluded. For instance, features which had little variance or correlated closely with other attributes were discarded.

3.3. Normalization

The numerical features were normalized on scale, so values have been normalized between a uniform range. It makes sure that there's no one feature to dominate the model training process and overall performance improves. Normalization also is useful, because it would make distance metric-based algorithms such as KNN to converge faster, because it guarantees that all the features are contributing equally to computing a distance. The models could concentrate on the most informative attributes and prevent overfitting. Besides feature selection also resulted in faster training times and lower memory usage, which are crucial in real world applications with limited resources. This process not only optimized model performance but also increased interpretability by reducing search space to essential predictors of normal and anomalous behavior.

3.4. Cross-Validation

The consistency and reliability of the model are validated using cross-validation. In this process, the training set is split into 10 equal subsets or folds. In each iteration, the model is trained on 9 folds while the remaining 1 fold is used for validation. This process is repeated 10 times, ensuring that each fold is used exactly once as the validation set. Finally, the average performance across all iterations is calculated to provide a more accurate estimation of the model's overall effectiveness. This approach reduces the risk of overfitting and provides a reliable evaluation of model performance across multiple subsets of the data.

The proposed framework begins with essential data preprocessing steps. First, feature selection techniques are applied to identify and eliminate irrelevant or redundant features. This reduces the dimensionality of the dataset and improves model performance by focusing on the most informative variables. Once the relevant features are selected, normalization is performed particularly important when using distance-based algorithms such as KNN. Normalization ensures that all features are on a comparable scale so that no single feature disproportionately influences the model due to its magnitude.

Cross-validation is integrated into the training phase to enhance model evaluation. By dividing the training data into multiple folds, each subset is used as both training and validation data at different points, allowing for a more robust assessment of the model's generalization ability.

3.5. Algorithms

Four classification algorithms were used to see which performs the best for this dataset. An anomalous and normal behavior prediction was then made with naive bayes, an efficient probabilistic model, with an accuracy of 89.41%. A distance-based algorithm KNN was used which showed 98.84% accuracy to detect patterns in the data and was demonstrated as having a strong ability to detect patterns in the data. An accuracy of 96.47% was captured from decision trees which are known to be simple. Finally, it was discovered that Random Forest, an ensemble technique combining multiple decision trees, produced a very robust model and had an accuracy of 99.04%.

 doi:10.20944/preprints202511.0761.v1

4. Results

4.1. Performance Evaluation

The performance of each model is evaluated based on its accuracy, precision and recall. Random Forest demonstrated the best because it could work with high dimensional data and generalize well to unseen data. Feature normalization helped the KNN algorithm perform exceptionally well too. While Naïve Bayes was efficient, it showed some limitations due to dataset's complexity, and Decision Trees maintained a balance between accuracy and interpretability. The results are shown in Table 1 from lowest to highest accurate model.

Table 1. Performance Metrics.

Algorithm	Accuracy (%)	Precision	Recall
Naive bayes	89.41	89.37	89.51
Decision trees	96.47	97.26	96.85
K nearest neighbors	98.84	99.01	99.00
Random Forest	99.04	99.09	98.98

4.2. Model Training

After evaluating the efficacy of these algorithms on RapidMiner and creating a design of the model, the model was tested and trained using a cloud-based platform "Google Colab". The model is coded in python performing very well even on the unseen data. The dataset was first normalized using a python script incorporating one-hot encoding for non- numerical data, split into training and testing sets with 70:30 trained and evaluated using the testing sets and gave an accuracy of 99.7%. The performance metrics of Random Forest are plotted as a graph using pyplot a Python library. Figure 2 shows the results showing the best performance of random forest algorithm.

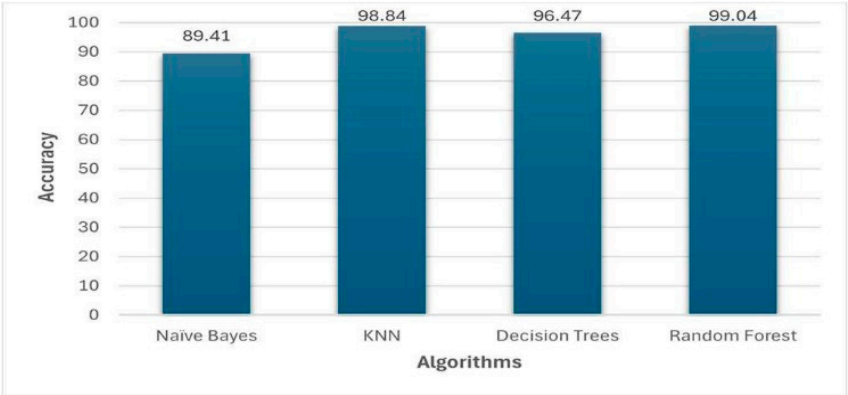

Figure 2. Performance Comparison of Four Algorithms.

RapidMiner was used because it has a user-friendly interface, and full capabilities for data preprocessing, modeling and evaluation. By utilizing built-in inspection tools on the platform for cross validation and feature selection, the workflow was simplified and allowed for testing alternative algorithms and subsequently their performance optimization. This also provided some further ability to visualize data transformations and model outputs as we implement. Additionally, Google Colab was leveraged because of its cloud-based platform using Python and machine learning libraries like scikit-learn to train and optimize model designed using RapidMiner. The study showed

 doi:10.20944/preprints202511.0761.v1

the effect of preprocessing, feature selection and algorithm choice in building working machine learning models for cybersecurity. Disentangling the features present in the data and normalizing the data helped the model's baseline and better identify patterns in the data. This leads to the observation of the effectiveness of ensemble methods at tackling complex and high dimensional datasets. ratios, performed cross-validation using Random Forest as it performed the best in RapidMiner.

5. Conclusion

This study highlighted the importance of careful preprocessing and algorithm selection in building robust machine learning models for. By using Random Forest, the models achieved high accuracy of 99.7%. The results here show promise for scalability and reliability within supervised classification techniques added to real time anomaly detection. Additionally, feature selection techniques integration was key to boosting model performance, showing that a clear attribution can drastically change its outcome. This methodology can be further improved in future work by extending this to other feature selection methods and applying testing on additional algorithms and addressing the real time implementation challenges to strengthen the current defenses present in the cybersecurity space.

References

1. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "IntruDTree: A machine learning-based cybersecurity intrusion detection model," Symmetry, vol. 12, no. 5, May 2020, doi: 10.3390/SYM12050754.
2. N. Aboueata, S. Alrasbi, A. Erbad, A. Kassler, and D. Bhamare, "Supervised machine learning techniques for efficient network intrusion detection," in Proc. Int. Conf. on Computer Communications and Networks (ICCCN), IEEE, Jul. 2019, doi: 10.1109/ICCCN.2019.8847179.
3. M. T. Abdelaziz et al., "Enhancing network threat detection with Random Forest-based NIDS and permutation feature importance," J. Network and Systems Management, vol. 33, no. 1, Mar. 2025, doi: 10.1007/s10922-024-09874-0.
4. [Author Unknown], "Optimized intrusion detection model for identifying known and innovative cyber attacks using support vector machine (SVM) algorithms," Introduction, 2024.
5. K. Shaukat, S. Luo, V. Varadharajan, I. A. Hameed, and M. Xu, "A survey on machine learning techniques for cybersecurity in the last decade," IEEE Access, vol. 8, pp. 222310–222354, 2020, doi: 10.1109/ACCESS.2020.3041951.
6. A. Atadoga, E. O. Sodiya, U. J. Umoga, and O. O. Amoo, "A comprehensive review of machine learning's role in enhancing network security and threat detection," World J. Adv. Res. Rev., vol. 21, no. 2, pp. 877–886, Feb. 2024, doi: 10.30574/wjarr.2024.21.2.0501.
7. A. Khanza, F. D. Yulian, N. Khairunnisa, and N. A. Yusuf, "Evaluating the effectiveness of machine learning in cyber threat detection," J. Comput. Sci. Technol. Appl. (CORISINTA), vol. 1, no. 2, pp. 172–179, 2024. Available: https://journal.corisinta.org/index.php/corisinta/index
8. F. Alrowais, S. Althahabi, S. S. Alotaibi, A. Mohamed, M. A. Hamza, and R. Marzouk, "Automated machine learning-enabled cybersecurity threat detection in Internet of Things environments," Comput. Syst. Sci. Eng., vol. 45, no. 1, pp. 687–700, 2023, doi: 10.32604/csse.2023.030188.
9. H. M. Farooq and N. M. Otaibi, "Optimal machine learning algorithms for cyber threat detection," in Proc. UKSim-AMSS 20th Int. Conf. on Modelling and Simulation, IEEE, Dec. 2018, pp. 32–37, doi: 10.1109/UKSim.2018.00018.
10. T. R. Bammidi, "Enhanced cybersecurity: AI models for instant threat detection," unpublished.
11. N. Al-Taleb and N. A. Saqib, "Towards a hybrid machine learning model for intelligent cyber threat identification in smart city environments," Appl. Sci., vol. 12, no. 4, Feb. 2022, doi: 10.3390/app12041863.
12. E. Kocyigit, M. Korkmaz, O. K. Sahingoz, and B. Diri, "Real-time content-based cyber threat detection with machine learning," in Proc. 2021, pp. 1394–1403, doi: 10.1007/978-3-030-71187-0_129.
13. F. Bouchama and M. Kamal, "Cyber threat detection through machine learning-based behavioral modeling of network traffic patterns," 2021.

 doi:10.20944/preprints202511.0761.v1

14. S. S. Balantrapu, "Exploring machine learning techniques for cyber threat detection," Int. J. Sustain. Dev. Through AI, ML and IoT, [Online]. Available: https://ijsdai.com/index.php/IJSDAI/index
15. M. R. Labu and M. F. Ahammed, "Next-generation cyber threat detection and mitigation strategies: A focus on artificial intelligence and machine learning," 2024, doi: 10.32996/jcsts.
16. D. Mahendiran, "Cyber threat detection using AI," Int. J. Multidiscip. Res. Explorer (IJMRE), vol. 4, 2024, doi: 10.0410/IJMRE.2024973591.
17. C. Ravikumar et al., "Exploring machine learning algorithms for robust cyber threat detection and classification: A comprehensive evaluation," in Proc. Int. Conf. on Intelligent Systems for Cybersecurity (ISCS), IEEE, 2024, doi: 10.1109/ISCS61804.2024.10581226.
18. Wayne, "AI and machine learning in cyber threat detection," Corrosion Management, [Online]. Available: https://corrosion-management.com/
19. Almulhim, M., Islam, N., & Zaman, N. (2019). A lightweight and secure authentication scheme for IoT based e-health applications. International Journal of Computer Science and Network Security, 19(1), 107-120.
20. Zaman, N., Low, T. J., & Alghamdi, T. (2014, February). Energy efficient routing protocol for wireless sensor network. In 16th international conference on advanced communication technology (pp. 808-814). IEEE.
21. Azeem, M., Ullah, A., Ashraf, H., Jhanjhi, N. Z., Humayun, M., Aljahdali, S., & Tabbakh, T. A. (2021). Fog-oriented secure and lightweight data aggregation in iomt. IEEE Access, 9, 111072-111082.
22. Ahmed, Q. W., Garg, S., Rai, A., Ramachandran, M., Jhanjhi, N. Z., Masud, M., & Baz, M. (2022). Ai-based resource allocation techniques in wireless sensor internet of things networks in energy efficiency with data optimization. Electronics, 11(13), 2071.
23. Khan, N. A., Jhanjhi, N. Z., Brohi, S. N., Almazroi, A. A., & Almazroi, A. A. (2022). A secure communication protocol for unmanned aerial vehicles. CMC-Computers Materials & Continua, 70(1), 601-618.
24. Muzafar, S., & Jhanjhi, N. Z. (2020). Success stories of ICT implementation in Saudi Arabia. In Employing Recent Technologies for Improved Digital Governance (pp. 151-163). IGI Global Scientific Publishing.
25. Jabeen, T., Jabeen, I., Ashraf, H., Jhanjhi, N. Z., Yassine, A., & Hossain, M. S. (2023). An intelligent healthcare system using IoT in wireless sensor network. Sensors, 23(11), 5055.
26. Shah, I. A., Jhanjhi, N. Z., & Laraib, A. (2023). Cybersecurity and blockchain usage in contemporary business. In Handbook of Research on Cybersecurity Issues and Challenges for Business and FinTech Applications (pp. 49-64). IGI Global.
27. Hanif, M., Ashraf, H., Jalil, Z., Jhanjhi, N. Z., Humayun, M., Saeed, S., & Almuhaideb, A. M. (2022). AI-based wormhole attack detection techniques in wireless sensor networks. Electronics, 11(15), 2324.
28. Shah, I. A., Jhanjhi, N. Z., Amsaad, F., & Razaque, A. (2022). The role of cutting-edge technologies in industry 4.0. In Cyber Security Applications for Industry 4.0 (pp. 97-109). Chapman and Hall/CRC.
29. Humayun, M., Almufareh, M. F., & Jhanjhi, N. Z. (2022). Autonomous traffic system for emergency vehicles. Electronics, 11(4), 510.
30. Muzammal, S. M., Murugesan, R. K., Jhanjhi, N. Z., & Jung, L. T. (2020, October). SMTrust: Proposing trust-based secure routing protocol for RPL attacks for IoT applications. In 2020 International Conference on Computational Intelligence (ICCI) (pp. 305-310). IEEE.