L-013

# Enhancing Network Security Using Automated Threat Hunting and Response System Using a Hybrid Machine Learning Model

Kurra Chaitanya Kumar ‡ Bhed Bahadur Bista ‡ Eiichiro Kodama ‡ Jiahong Wang ‡

## 1. Introduction

In this modern world, technology is growing rapidly every day. Information is also being digitalized in sectors like banking, medical, etc. Cyber threats are becoming more advanced. Network security has been facing more attacks, such as brute-force attacks, Cross-Site Scripting (XSS) and more. Traditional security measures, which are signature-based, often fail to detect modern attacks or unknown attacks like zero-day attacks. So, we need an efficient threat hunting system that can detect threats in the network, not just signature-based, but also new threats. So, we need an automated threat hunting system that detects new threats and monitors network traffic continuously.

Automated threat-hunting and response systems are threat-hunting systems that continuously monitor and detect any threats in the network. They use ML algorithms to detect threats. These systems help security teams by allowing them to focus on high risks. They continuously monitor the network, which can improve overall security, and the efficiency of the security teams is enhanced as these systems reduce labour.

In this paper, we proposed a hybrid model based on a soft voting ensemble method to improve detection efficiency, reduce the false positive rate, and thereby improve the overall performance of the hybrid model. We are using a hybrid model because it can oversee diverse data and also can handle complex tasks by leveraging the strengths of multiple models. This method achieved an overall accuracy of 98.86% and a precision of 98.64%. These improvements make our hybrid model more efficient in the threat hunting process, making it more suitable for diverse real-time applications.

## 2. Related Work

Shan et al. [1] proposed a technique for detecting threats proactively in critical infrastructure using a hybrid machine learning model. The models were trained and tested by using real-time data collected from various open-source websites, which consists of both normal and anomalous data. The anomalous data contains attacks such as man-in-the-middle (MITM) attacks, SWIFT attacks and unauthorized access based on banking transactions [1]. However, their hybrid model still has a few limitations. Firstly, the types of attacks selected are limited, which shows that this model cannot be used in more real-time scenarios. Secondly, the hybrid model has thirty-seven false positives.

Rahman et al. [2] addressed the imbalance among datasets using class weighting but did not use heterogeneous models, which limits the performance.

Singh et al. [3] developed a lightweight hybrid IDS using SVM-RF for SDN environments, but it lacks multi-class handling.

## 3. Methodology

In this paper, we proposed a voting-based hybrid model as shown in Fig. 1. In our research, we focus on reducing the false positives and making it widely applicable. There are various stages of methodology for machine learning-based automated threat hunting and response systems. The phases of methodology include data collection, environment, pre-processing of data, training machine learning models, ensemble techniques, and evaluation of the performance of models.

### 3.1 Data Collection

For this research, we collected real-time open-source datasets from Kaggle [4]. The datasets used comprise a collection of CIC datasets, including CIC-IDS 2017, CIC-DoS 2018, CSE-CIC-IDS 2018, and CIC-DDoS 2019 datasets. These datasets include benign data and malicious data. This anomalous data encompasses various real-time attacks such as DoS attacks, DDoS attacks, brute-force attacks, Cross-Site Scripting (XSS) attacks, and more. 80% of the dataset is used to train, and 20% of the dataset is used to assess the machine learning models' performance.

### 3.2 Experimental Setup

On a Windows 11 PC with an AMD Ryzen 7 5700G 3.80 GHz CPU (Advanced Micro Devices, Santa Clara, CA, USA) with sixteen gigabytes of RAM and a GeForce RTX 3060 GPU (NVIDIA, Santa Clara, CA, USA), the models are trained and assessed using Python. These are conducted in the Anaconda environment utilizing Scikit-learn, TensorFlow. GPU acceleration was enabled through NVIDIA's CUDA toolkit and cuDNN.

### 3.3 Data Pre-Processing

The next step is pre-processing of data to convert the data into a suitable format before training models. The data is cleaned to identify any null values or missing values.

‡ Graduate School of Software and Information Science, Iwate Prefectural University

### 3.4 Machine Learning Models

In this study, models include Random Forest (RF), Neural Network (NN), XGBoost, and hybrid models were trained. Using labelled datasets, the models were trained and tested.

### 3.5 Ensemble Technique

The hybrid model is ensembled using voting mechanisms. Hard voting and soft voting ensemble methods are implemented to create two different hybrid models. The soft voting mechanism is based on the average predicted probabilities, whereas hard voting is based on the majority predicted probabilities.

### 3.6 Performance Evaluation

The performance of models is evaluated based on accuracy, precision, and recall metrics.

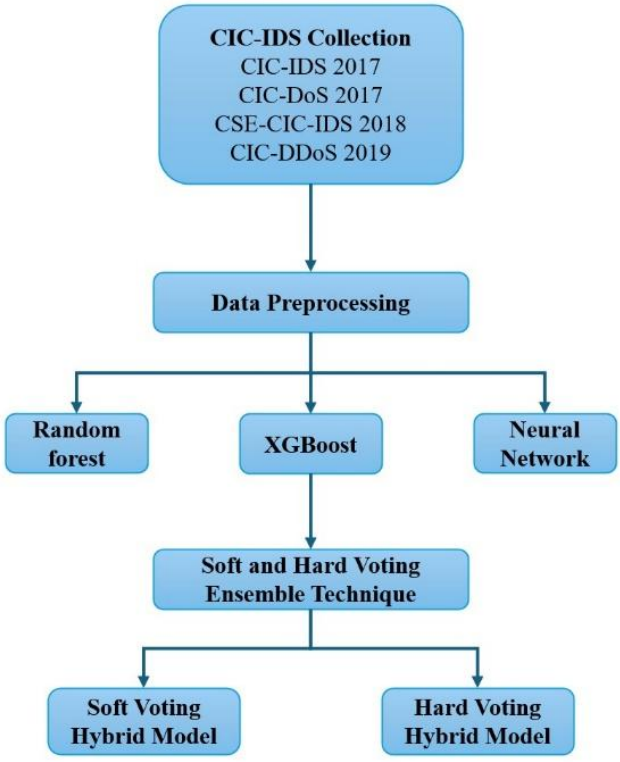


Figure 1. Proposed Hybrid Model Architecture

## 4. Results and Discussion

Various ML models are tested to assess their threat hunting efficiency. The performance of these models is assessed based on evaluation metrics.

Among all the models, the soft voting-based hybrid model gave the best performance with an accuracy of 98.86% and precisely detected distinct types of attacks with 98.64% precision.

The hard voting-based hybrid model also gave a reliable performance with an accuracy of 98.84%, with a precision of 98.53%, slightly lower than the soft voting method.

Then the XGBoost model also gave satisfactory performance with an accuracy of 98.86%, like soft voting method, but failed in detecting individual attacks with 98.45% precision, slightly lower than the hybrid models.

Subsequently, the random forest model demonstrated an accuracy of 98.43% and a precision of 98.01%, while the neural network exhibited the least performance of all, with an accuracy of 97.29% and a precision of 96.44%.

The comparison of these metrics among these models is shown below in Table 1.

Table 1. Comparison of different models' results

| Model | Accuracy | Precision | Recall |
|---|---|---|---|
| Soft Voting | 98.86% | 98.64% | 98.86% |
| Hard Voting | 98.84% | 98.53% | 98.84% |
| XGBoost | 98.86% | 98.45% | 98.86% |
| Random Forest | 98.43% | 98.01% | 98.43% |
| Neural Network | 97.29% | 96.44% | 97.29% |

## 5. Conclusion

Automated threat detection using ML models significantly impacts network security. In this study, we used XGBoost, RF, NN, and hybrid models for the threat hunting process. Although hybrid models achieved promising results, they can be further improved by addressing imbalanced classes and enhancing their performance.

## References

[1] Shan A., Myeong S., "Proactive Threat Hunting in Critical Infrastructure Protection through Hybrid Machine Learning Algorithm Application", Sensors, Vol. 24, No.2 (2024).
[2] Rahman O., Quraishi M. A. G., Lung C. -H., "DDoS Attacks Detection and Mitigated in SDN using Machine Learning", 2019 IEEE World Congress on Services (SERVICES) (2019).
[3] Singh A., Kaur H., Kaur N., "A novel DDoS detection and mitigation technique using hybrid machine learning model and redirect illegitimate traffic in SDN network", Cluster Comput, Vol. 27, No. 2 (2024).
[4] Kaggle, "https://www.kaggle.com".