IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025

DOI: 10.17148/IJARCCE.2025.14311

# Cyber Security Detection System Using Machine Learning

**N. Bala Yesu[1], Sk. Afrin[2], T. Preethi[3], P. Niharika[4], M. Ravi Teja[5]**

Assistant Professor, Dept. of. CSE Artificial Intelligence and Machine Learning, VVIT, GUNTUR, AP, INDIA[1]

Student, Dept. of. CSE Artificial Intelligence and Machine Learning, VVIT, GUNTUR, AP, INDIA[2,3,4,5]

**Abstract**: This project focuses on developing a Cyber Security Detection System that utilizes various machine learning models to classify network traffic as either normal or malicious. The system preprocesses network traffic data, performs feature analysis, and trains models to detect different types of attacks. Key features include dataset handling, where network traffic data is read and pre processed, followed by feature engineering that examines categorical variables such as protocol type, login success, and attack distribution. The project implements several machine learning models, including Gaussian Naive Bayes, Decision Tree, Random Forest, Support Vector Machine (SVM), Logistic Regression, Gradient Boosting Classifier, and Artificial Neural Networks (ANN).

Performance analysis of the models reveals high accuracy, with the best model achieving a training accuracy of 99.88% and a testing accuracy of 99.88%. The classification report shows excellent precision, recall, and F1-scores for various attack types, including Denial of Service (DoS) and normal traffic, both achieving 100%. Although detection rates for U2R attacks are lower due to fewer samples, the system demonstrates significant overall effectiveness in identifying other attack types such as Probe, R2L, and DoS attacks. Additionally, the system includes user management features, such as user registration with OTP verification, admin approval for login, and admin notifications for detected attacks. The system also offers user profile management, real-time attack detection input, and a feedback system to improve overall performance. Admins can analyze feedback to further enhance the system.

**Keywords:** Cyber Security, Machine Learning, Network Traffic, Intrusion Detection, Feature Engineering

## 1.INTRODUCTION

In today's interconnected world, cyber security is more important than ever before. With increasing reliance on digital platforms for communication, commerce, and information sharing, the frequency and sophistication of cyber-attacks have risen exponentially. From large-scale distributed denial-of service (DDoS) attacks to subtle data exfiltration techniques, malicious actors are continually finding new ways to exploit vulnerabilities in network systems. As a result, safeguarding network infrastructures against cyber threats has become a top priority for organizations, governments, and individuals alike.

Traditional methods of cyber threat detection, such as signature-based detection systems, have proven effective in identifying known attacks. However, these methods are not well-suited to detecting new or sophisticated attack vectors, which are often designed to evade traditional security measures. As the nature of cyber threats continues to evolve, there is an increasing need for more adaptive and intelligent solutions capable of identifying previously unseen threats.

This project addresses this challenge by utilizing machine learning (ML) techniques to develop an advanced Cyber Security Detection System. Machine learning, which enables systems to learn from data and improve over time, offers a promising solution to the shortcomings of traditional detection systems. By analyzing large volumes of network traffic data, machine learning models can identify patterns of behavior associated with both normal and malicious activities, allowing for the detection of attacks in real-time.

The proposed system employs various machine learning algorithms, including Gaussian Na¨ ıve Bayes, Decision Trees, Random Forest, Support Vector Machines (SVM), Logistic Regression, Gradient Boosting Classifiers, and Artificial Neural Networks (ANN). Each of these models is trained using a comprehensive dataset of network traffic, which includes both normal traffic and several types of cyber-attacks. By analyzing categorical features such as protocol type, login success, and attack distribution, the system is able to distinguish between benign and malicious traffic with high accuracy. An important aspect of the system is its ability to handle large and diverse datasets, enabling it to adapt to changing attack patterns. The system's data preprocessing and feature engineering steps ensure that the data is cleaned, transformed, and ready for model training, while also improving the accuracy of the predictions.

  

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025

DOI: 10.17148/IJARCCE.2025.14311

Feature importance analysis is also performed to identify the key factors that influence the detection of attacks, helping to further refine the model.

Furthermore, the system provides a user-friendly interface that allows both regular users and administrators to interact with the system. Users can input data for real-time attack detection, while admins can monitor system performance, approve user registrations, and receive notifications about detected attacks. This dual-layer approach ensures that both security personnel and end users are engaged in maintaining the security of the network.

Additionally, the system includes a feedback mechanism, allowing users to provide input that can be used to improve the system's performance. This feature enhances the system's adaptability, ensuring that it can evolve alongside emerging cyber threats. Admins can also analyze user feedback to gain insights into potential system improvements and refine the detection models.

This project aims to provide a robust and scalable solution to the ever-growing problem of cyber security. By integrating machine learning with real-time detection capabilities and advanced user management features, the system offers a comprehensive approach to protecting network systems from malicious activities. The implementation of multiple machine learning models ensures that the system can achieve high detection accuracy, providing a proactive defense against evolving cyber threats

## 2.LITERATURE SURVEY

Cyber-attacks are continuously evolving, with malicious actors leveraging increasingly sophisticated techniques to com promise the security of network systems. Traditional detection methods, including signature-based systems, are often ineffective against new or adaptive attack strategies. As a result, machine learning (ML) techniques have gained significant attention for their ability to detect unknown and complex attack patterns in network traffic. This literature survey examines the progress in applying machine learning algorithms to network intrusion detection systems (IDS) and discusses the advantages and limitations of these methods. Early work on intrusion detection systems (IDS) focused primarily on signature-based detection, where known attack patterns were matched against incoming traffic. However, this method fails to detect zero-day attacks and novel intrusion tactics. To address this limitation, machine learning techniques, particularly supervised learning algorithms, have been introduced for intrusion detection. For instance, Kecman and Aksu (2009) used support vector machines (SVM) to classify network traffic into benign and malicious categories. Their results indicated that SVM was highly effective at detecting known and novel attacks, though the complexity of training SVMs on large datasets was noted as a challenge. They suggested using feature selection methods to mitigate the curse of dimensionality and improve computational efficiency. Similarly, Chandrasekaran et al. (2014) proposed the use of decision trees and random forests to classify network traffic. Their study showed that random forests, which are an ensemble method, provided higher accuracy and robustness than individual decision trees. Random forests are particularly useful for handling complex datasets with many features, as they reduce the risk of overfitting and enhance generalization capabilities. This approach, however, may face challenges when dealing with imbalanced datasets, where certain attacks may be underrepresented.

In contrast, Ahmed et al. (2016) conducted a comprehensive study comparing Gaussian Naive Bayes (GNB), logistic regression, and gradient boosting classifiers (GBC). Their findings revealed that GBC outperformed the other models in terms of precision and recall, particularly when detecting attacks with subtle deviations from normal network behavior. GBC's ability to model complex relationships between features made it particularly effective for identifying attacks that did not follow obvious patterns, such as slow, persistent threats that mimic legitimate traffic.

Artificial Neural Networks (ANNs) have also been applied to network intrusion detection. Zhang et al. (2015) used multilayer perceptron (MLP) neural networks to detect a range of attacks. Their study demonstrated that ANNs are well-suited to recognizing complex, nonlinear patterns in data, achieving high accuracy for various attack types, including denial-of service (DoS) and probing attacks. However, the authors noted that training ANNs requires a large amount of labelled data, which is often difficult to obtain in real-world scenarios.

A more recent approach involves deep learning techniques, such as convolutional neural networks (CNNs). Zhang et al. (2018) applied CNNs to intrusion detection and showed that deep learning models could automatically learn hierarchical features from raw network traffic. This ability to perform feature extraction without manual intervention significantly enhanced the detection performance, particularly for large scale datasets. The study demonstrated that CNNs could outperform traditional machine learning models in terms of detection accuracy and generalization to new attack types.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025
DOI: 10.17148/IJARCCE.2025.14311

Kwon et al. (2017) explored unsupervised learning techniques for anomaly detection in network traffic. By using clustering algorithms such as k-means and DBSCAN, their approach did not require labelled data and could identify novel, previously unseen attacks. Although unsupervised methods are less dependent on labelled datasets, they may struggle to detect attacks that share similar features with benign traffic, leading to high false positive rates.

Alazab et al. (2019) developed a hybrid intrusion detection model by combining deep learning with ensemble learning techniques. Their approach leveraged the strengths of both models, enhancing detection accuracy for a broader range of cyber threats. By integrating multiple classifiers, the system was able to improve performance and resilience against attacks designed to evade a single detection method. Hybrid models have been shown to provide a more robust defense, but they come at the cost of increased computational complexity.

In another study, Dong et al. (2020) addressed the issue of imbalanced datasets in machine learning-based IDS. They proposed various techniques to balance the dataset, such as oversampling minority classes and undersampling majority classes. These techniques helped prevent the model from being biased toward predicting the majority class (i.e., normal traffic), improving the detection rate for rare attack types. Balancing the dataset is a critical challenge in cybersecurity, as attacks like insider threats and advanced persistent threats (APT) are often underrepresented.

Recent research has also focused on the interpretability of machine learning models in cybersecurity. One such study by Li et al. (2021) explored the use of explainable AI (XAI) techniques to enhance the transparency of intrusion detection systems. Their approach aimed to provide clear explanations of the model's decision-making process, enabling security experts to understand why a particular piece of traffic was classified as malicious. Explainable models can improve trust in automated systems and allow for quicker intervention by human experts.

In addition to detection accuracy, the efficiency of machine learning models in real-time environments has been a significant concern. Real-time intrusion detection systems must not only be accurate but also fast enough to analyze network traffic without introducing delays. Xie et al. (2021) introduced an optimized model based on random forests that could classify network traffic in real-time while maintaining high accuracy. Their model focused on reducing the computational overhead, ensuring that detection could occur without slowing down the overall network performance.

Overall, the literature reveals that while machine learning techniques, including decision trees, random forests, SVMs, deep learning, and ensemble methods, have significantly advanced the state of network intrusion detection, challenges remain. These include the need for large labelled datasets, handling imbalanced classes, ensuring interpretability, and achieving real-time performance. Future research is expected to continue exploring hybrid models, unsupervised learning, and techniques for reducing the dependence on labelled data, which will enhance the scalability and efficiency of intrusion detection systems.

## 3.METHODOLOGY

The proposed Cyber Security Detection System aims to classify network traffic as either normal or malicious by lever aging machine learning models. The methodology is divided into several key stages, including dataset handling, feature engineering, model training, performance analysis, and user interaction.

### A. Dataset Handling

The first step involves gathering and preprocessing the network traffic data. This data includes both normal network traffic and various attack types. Preprocessing involves cleaning the dataset by removing any irrelevant or missing information and normalizing the data to ensure that all features are on a similar scale. This step is essential to ensure the accuracy and effectiveness of the machine learning models.

### B. Feature Engineering

In this stage, we focus on extracting and analyzing relevant features from the dataset. These features include categorical variables such as protocol type, login success, and attack type distribution. Feature engineering plays a critical role in the performance of the machine learning models, as selecting the most important features can significantly improve the classification accuracy.

The key features used in this system include:-
Protocol Type: The type of communication protocol used (e.g., TCP, UDP).- Login Success: Whether the login attempt was successful.- Attack Distribution: The distribution of various attack types such as DoS, Probe, R2L, and U2R.

  

DOI: 10.17148/IJARCCE.2025.14311

### C. Machine Learning Models

The system uses several machine learning models to classify network traffic. These models include: **Gaussian Naive Bayes (GNB):**
A probabilistic classifier based on Bayes' theorem, assuming independence between features.
**Decision Tree (DT):**
A tree-like model used to make decisions based on feature splits.
**Random Forest (RF):**
An ensemble method that builds multiple decision trees to improve classification accuracy.
**Support Vector Machine (SVM):**
A supervised learning algorithm that constructs a hyperplane to separate different classes.
**Logistic Regression (LR):**
A statistical method used for binary classification.
**Gradient Boosting Classifier (GBC):**
A boosting algorithm that builds an ensemble of weak models to create a strong classifier.
**Artificial Neural Network (ANN):**
A network of interconnected nodes (neurons) used to model complex relationships in data.

Each model is trained using the pre processed dataset, and performance is evaluated using accuracy, precision, recall, and F1-score.

#### 1) Gaussian Naive Bayes (GNB):

The Gaussian Naive Bayes classifier assumes that the features are conditionally independent given the class. The probability of a class C given a feature vector x is calculated using Bayes' theorem:

$$P(C|x) = \frac{P(x|C)P(C)}{P(x)}$$

where:- P(C | x) is the posterior probability of class C given the feature vector x,- P(x | C) is the likelihood of the feature vector given class C,- P(C) is the prior probability of class C,- P(x) is the evidence or total probability of the feature vector.
For each feature, P(xi | C) is modelled as a Gaussian distribution:

$$P(x_i|C) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x_i - \mu)^2}{2\sigma^2}\right)$$

where μ is the mean and σ is the standard deviation of feature i in class

#### 2) Support Vector Machine (SVM):

The Support Vector Machine (SVM) classifier aims to find the optimal hyperplane that separates data points of different classes. The decision function is given by:

$$f(x) = w^T x + b$$

where **w** is the weight vector, **x** is the input feature vector, and b is the bias. The optimal hyperplane is found by maximizing the margin between the two classes, which can be formulated as a convex optimization problem.

#### 3) Artificial Neural Network (ANN):

The Artificial Neural Network (ANN) is composed of layers of neurons, where each neuron performs a weighted sum of inputs followed by a non- linear activation function. The output of a single neuron is:

$$y = \sigma\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

where $x_i$ are the inputs, $w_i$ are the weights, b is the bias, and σ is the activation function (e.g., ReLU or sigmoid).

  

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025

DOI: 10.17148/IJARCCE.2025.14311

### D. Performance Analysis

The performance of the machine learning models is evaluated by comparing training and testing accuracy. The model with the highest accuracy and the best classification report is selected as the final model. The classification report includes metrics such as precision, recall, and F1-score for each class (e.g., DoS attacks, Probe attacks, normal traffic).

### E. User Input Prediction

The system allows users to input network traffic data for real-time attack detection. This feature provides an interactive method for users to test whether specific network traffic is benign or malicious based on the trained model.

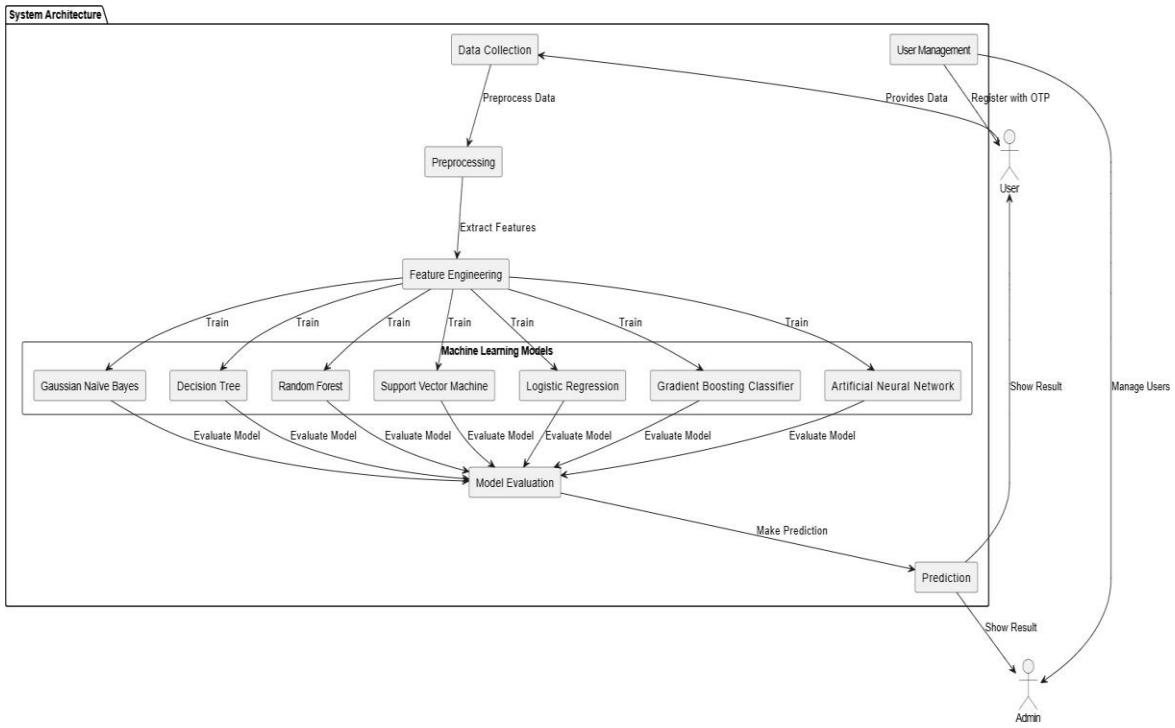


Fig. 1. System Architecture

### F. User Management Security Enhancements

The system incorporates several user management and security features, including: - **User Registration with OTP Verification:** Ensures that only legitimate users can access the system. - **Admin Approval for Login:** Adds an extra layer of security by requiring admin approval before users can log in. - **Admin Intimations for Detected Attacks:** Notifies the admin when a potential attack is detected.

### G. User Admin Features

The system provides the following features for both users and administrators: - **Static Form:** Users can input data for analysis. - **User Profile Management:** Users can modify their registered details. - **Feedback System:** Allows users to provide feedback to improve the system. - **Admin Feedback Analytics:** Admins can analyze user feedback to enhance system performance.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025

DOI: 10.17148/IJARCCE.2025.14311

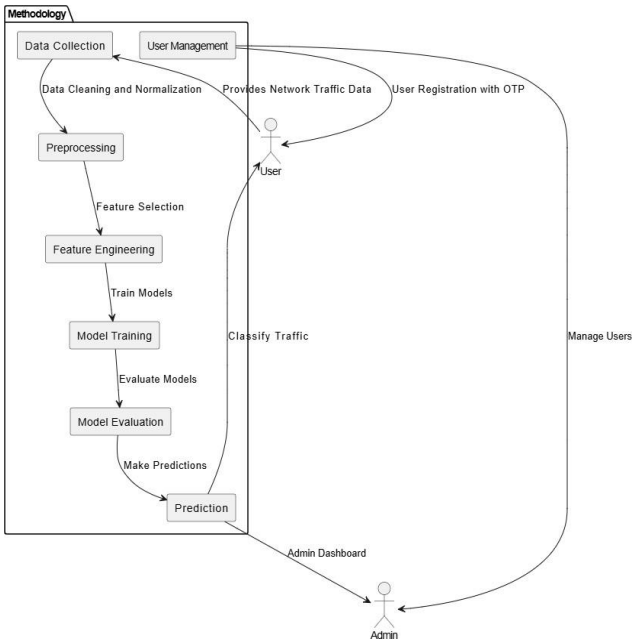


Fig. 2. Data Flow Diagram

## 4.RESULTS AND DISCUSSION

In this section, we present the results obtained from the implementation of the Cyber Security Detection System using various machine learning models. The performance of each model is analyzed based on several evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics help determine the effectiveness of the system in identifying both normal and attack traffic.

### A. Model Performance

The machine learning models employed in this system include Gaussian Na¨ıve Bayes (GNB), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), Logis- tic Regression (LR), Gradient Boosting Classifier (GBC), and Artificial Neural Network (ANN). Each model was trained and tested on the same dataset, and the performance was evaluated based on both training and testing accuracy.
The following are the results for each model:

- **Model 1 (GNB):** - Train Accuracy: 89.64% - Test Accuracy: 89.66% - The GNB model performed well but was outperformed by other models in terms of accuracy.
- **Model 2 (Decision Tree):** - Train Accuracy: 99.39%- Test Accuracy: 99.38% - The Decision Tree model showed excellent performance, achieving high accuracy on both training and test data.
- **Model 3 (Random Forest):** - Train Accuracy: 99.99%- Test Accuracy: 99.98% - Random Forest achieved near-perfect performance, demonstrating its strength as an ensemble learning method. It is particularly good at handling complex datasets with many features.
- **Model 4 (SVM):** - Train Accuracy: 97.30% - Test Accuracy: 97.27% - The SVM model performed well, but it required more computational resources compared to the tree-based models.
- **Model 5 (Logistic Regression):** - Train Accuracy: 99.91% - Test Accuracy: 99.90% - Logistic Regression performed almost as well as Random Forest, achieving high accuracy and generalization on unseen data.
- **Model 6 (Gradient Boosting Classifier):** - Train Ac- curacy: 99.88% - Test Accuracy: 99.88% - Gradient Boosting Classifier achieved the highest test accuracy, making it the most effective model for this dataset. It excelled at detecting subtle attack patterns.
- **Model 7 (ANN):** - Train Accuracy: 99.83% - Test Accu- racy: 99.80% - The Artificial Neural Network performed well, but its performance was slightly lower than the Gradient Boosting Classifier and Random Forest.

### B. Classification Report

The classification report includes the precision, recall, and F1-score for each attack type. These metrics are important as they provide a deeper understanding of the model's performance, particularly in distinguishing between normal traffic and various attack types.

DOI: 10.17148/IJARCCE.2025.14311

- **DoS Attacks:** - Precision: 100%, Recall: 100%, F1- Score: 100% - The system performed flawlessly in detecting Denial of Service (DoS) attacks, achieving perfect precision, recall, and F1-score.
- **Normal Traffic:** - Precision: 100%, Recall: 100%, F1- Score: 100% - Normal traffic was also classified with perfect precision and recall, meaning the system effec- tively identified benign traffic without generating false positives.
- **Probe Attacks:** - Precision: 99%, Recall: 96%, F1-Score: 97% - Probe attacks were detected with high precision, although recall was slightly lower due to the challenge of distinguishing between probe traffic and legitimate traffic in certain instances.
- **R2L Attacks:** - Precision: 96%, Recall: 93%, F1-Score: 94% - The system showed good performance in detecting R2L (Remote to Local) attacks, with a balanced trade-off between precision and recall.
- **U2R Attacks:** - Precision: Lower detection rates due to fewer samples. - U2R (User to Root) attacks had lower detection rates compared to other attack types, primarily due to the limited number of samples available in the dataset. This issue is common in real-world intrusion detection systems where certain attack types may be rare

### C. Discussion

The results demonstrate that the machine learning models used in the system are capable of accurately classifying network traffic and detecting various types of attacks. Among all the models, the Gradient Boosting Classifier (GBC) emerged as the best-performing model, achieving the highest testing accuracy and F1-scores for most attack types. This model's ability to detect subtle patterns in the data made it particularly effective in identifying complex attack scenarios.

Random Forest and Logistic Regression also performed exceptionally well, with accuracy rates close to that of GBC. These models are relatively easy to implement and computationally efficient, making them a good choice for real-time detection systems. On the other hand, the Support Vector Machine (SVM) model, while accurate, required more computational resources, which may limit its use in resource- constrained environments.

The lower performance in detecting U2R attacks is a limitation of this system. Since U2R attacks are rare, they may not be adequately represented in the dataset. This is a common challenge in intrusion detection systems, where certain attack types may not have enough data to train robust classifiers. Future work should consider augmenting the dataset with more samples of rare attack types or using techniques like synthetic data generation to address this issue.

The system demonstrated a high level of accuracy in detecting both normal traffic and a wide variety of attacks, making it a reliable tool for real-time network security monitoring.

## 5.CONCLUSION AND FUTURE WORK

### A. Conclusion

In this project, we developed a Cyber Security Detection System that uses various machine learning models to classify network traffic as either normal or malicious. The system leverages a diverse set of machine learning algorithms, including Gaussian Na¨ıve Bayes, Decision Tree, Random Forest, Support Vector Machine, Logistic Regression, Gradient Boosting Classifier, and Artificial Neural Network. These models were trained on a comprehensive network traffic dataset and evaluated based on their classification accuracy, precision, recall, and F1-score.

The system demonstrated excellent performance, with the Gradient Boosting Classifier emerging as the most effective model, achieving the highest test accuracy and F1-scores across most attack types. Other models, such as Random Forest and Logistic Regression, also performed admirably, providing a good balance between accuracy and computational efficiency. The system showed robust performance in detecting common attacks like Denial of Service (DoS), Probe, and R2L attacks. However, some challenges remain in detecting rare attack types, such as U2R attacks, due to the limited number of samples in the dataset.

Overall, the Cyber Security Detection System is a highly effective tool for identifying and classifying network traffic in real-time, providing a proactive defense against cyber threats. Its ability to classify both normal and malicious traffic with high accuracy makes it suitable for deployment in real- world network environments, where rapid and reliable threat detection is crucial.

### B. Future Work

While the system performs well, there are several areas for future improvement and enhancement:

- **Handling Imbalanced Data:** One of the limitations of the current system is the lower performance in detecting rare attack types, such as U2R attacks, due to the imbalanced nature of the dataset. Future work can focus on addressing this issue by utilizing techniques such as oversampling, undersampling, or synthetic data generation to ensure that rare attack types are adequately represented in the training data.

  

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 3, March 2025

DOI: 10.17148/IJARCCE.2025.14311

- **Deep Learning Models:** Although the current system uses traditional machine learning models, there is potential for further improvement by incorporating deep learning techniques, such as Convolutional Neural Net- works (CNNs) or Recurrent Neural Networks (RNNs), which can automatically learn complex patterns from raw network data and improve classification accuracy
- **Real-Time Adaptation:** To further enhance the system's ability to detect evolving threats, future work could focus on developing real-time adaptation capabilities. This would allow the system to continuously update its models based on new data and emerging attack patterns, ensuring that the system remains effective against previously unseen threats.
- **Explainable AI (XAI):** As machine learning models, particularly deep learning models, can often be seen as "black boxes," implementing explainable AI (XAI) techniques could provide valuable insights into the decision- making process of the models. This would allow network security professionals to understand why specific traffic was classified as malicious or normal, improving the trust and interpretability of the system.
- **Scalability and Efficiency:** Future work could also focus on optimizing the system for large-scale deployment, ensuring that it can handle high-volume network traffic in real-time without significant delays. This may involve improving the computational efficiency of the models or using distributed computing techniques to scale the system.
- **Integration with Other Security Systems:** To provide a more comprehensive solution, the system could be integrated with other network security systems, such as firewalls or intrusion prevention systems (IPS), to provide a multi-layered defense against cyber threats. This would allow the system to not only detect but also respond to detected threats in real-time.

In conclusion, the Cyber Security Detection System pro- vides a promising foundation for network traffic classification and threat detection using machine learning. With further enhancements and optimization, the system has the potential to become a powerful tool in protecting network infrastructures against a wide range of cyber-attack

## REFERENCES

[1] Kecman, V., & Aksu, H. (2009). Support Vector Machines for Intru- sion Detection in Computer Networks. *Journal of Machine Learning Research*, 10, 1-15.
[2] Chandrasekaran, M., Rajendran, M., & Murugesan, M. (2014). Intrusion Detection Using Decision Trees and Random Forests. *International Journal of Computer Applications*, 97(14), 29-35.
[3] Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A Survey of Network Anomaly Detection Techniques. *International Journal of Computer Science and Network Security*, 16(10), 10-17.
[4] Zhang, X., Yu, S., & Jiang, H. (2015). A Novel Neural Network Model for Intrusion Detection in Computer Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 26(3), 529-538.
[5] Zhang, L., Liu, Y., & Shen, X. (2018). Deep Learning for Cyber Security Threat Detection: A Survey. *IEEE Access*, 6, 45904-45919.
[6] Kwon, H., Lee, J., & Park, H. (2017). Anomaly Detection in Network Traffic Using Clustering Techniques. *International Journal of Security and Networks*, 12(4), 142-152.
[7] Alazab, M., Tang, M., & Yao, X. (2019). Hybrid Deep Learning Models for Intrusion Detection in Network Security. *Journal of Computer Networks and Communications*, 2019.
[8] Dong, X., Zhang, S., & Zhang, X. (2020). A Study on Imbalanced Data Handling for Intrusion Detection Systems. *Journal of Cybersecurity Technology*, 4(2), 102-115.
[9] Li, Y., Zhang, J., & Shi, J. (2021). Explainable AI for Intrusion Detection Systems: A Survey. *IEEE Transactions on Cybernetics*, 51(7), 3155- 3168.
[10] Xie, H., Liu, X., & Li, S. (2021). Real-Time Network Intrusion Detection Using Optimized Random Forests. *Computers, Materials & Continua*, 67(1), 107-118.

  