WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

ENHANCING THREAT DETECTION ACCURACY THROUGH DECISION TREE CLASSIFIERS

SANTHIYA J M [1], Dr. NANCY JASMINE GOLDENA[2]

[1] Reg.no: 23081205300112026, Department of Computer Application and Research Centre, Sarah Tucker College (Autonomous), Affiliated to Manonmaniam Sundaranar University, Tirunelveli - **627007**, jmsanthiya2001@gamil.com

[2]Associate Professor, Department of Computer Application and Research Centre, Sarah Tucker College (Autonomous), Affiliated to Manonmaniam Sundaranar University, Tirunelveli - **627007**, nancy_lordwin@rediffmail.com, ORCID: 0000-0002-8626-2604

ABSTRACT:

The increasing complexity of cyber threats necessitates robust security mechanisms to protect network infrastructures. This research introduces an Intrusion Detection System (IDS) using a Decision Tree classifier, a popular machine learning algorithm. The goal of the IDS is to detect potential cyber-attacks by analyzing network traffic data and classifying it into different attack types or normal behaviour. The dataset used contains various features like protocol_type, service, and flag, which are essential attributes of network traffic. Since some of these features are categorical (non-numeric), the data preprocessing step involves encoding these categories into numerical values using the LabelEncoder. The target variable in this case is xAttack, which represents the type of network attack or normal behaviour. A Decision Tree classifier is trained on the training data, and predictions are made on the test data. The performance of the model is evaluated. The classification report provides detailed metrics for each attack class. A bar chart is also plotted to visualize the precision, recall, and F1-score for the top five attack categories. The entire system helps in automating the detection of network intrusions, improving security monitoring.

Keywords: Intrusion Detection System, Machine Learning, Decision Tree, Cybersecurity, Network Traffic Analysis, Precision, Recall, Real-Time Processing.

1. Introduction:

In an era of rapidly advancing technology and interconnected systems, the rise of cyber threats poses significant risks to the security of networks and data. An effective Intrusion Detection System (IDS) is vital for identifying and mitigating these threats, ensuring that potential cyberattacks[1] are detected early before they cause severe damage. This project aims to develop a machine learning-based IDS using a Decision Tree classifier[2], one of the most popular algorithms for classification tasks due to its simplicity and interpretability. The core objective of the IDS is to analyze network traffic and classify it as either normal or indicative of various types of cyberattacks. The dataset used for this project contains key features that characterize network traffic, such as protocol type, service, and flag. These features play a crucial role in identifying suspicious behaviour[3]. However, since some of these features are categorical, they must be converted into numerical form to be compatible with machine learning algorithms. The Decision Tree classifier is trained on the training data to learn patterns in the network traffic that indicate whether the traffic is normal or an attack. After training, the model is used to predict the classification of the test data, and its performance is evaluated. These metrics provide a comprehensive evaluation of the model's ability to correctly identify both normal and malicious traffic while minimizing false positives and false negatives. In addition to numerical evaluation, the project includes a bar chart[4] to visually represent the precision, recall, and F1-score for different attack classes. This visual representation helps in understanding the performance of the model across various attack types and highlights areas where the model excels or requires improvement. Finally, the trained model is saved for future use, enabling the IDS to be deployed in real-time environments for continuous monitoring of network traffic[5]. By automating the detection of potential intrusions, this IDS project contributes to enhancing the security of networks, providing a proactive solution for identifying and responding to cyber threats in real time. This system can be integrated into existing cybersecurity infrastructures to improve overall network protection.

2. Literature Review:

Kumar & Rani, 2020: Intrusion Detection Systems (IDS) play a critical role in safeguarding networks against cyber threats. Traditional approaches, such as signature-based detection[6], are highly effective at identifying known attacks by comparing network traffic to predefined patterns. However, they struggle with zero-day threats due to their reliance on constant updates to the signature database[7].

Patel & Soni, 2021: Anomaly-based detection techniques, on the other hand, identify deviations from normal behaviour and are capable of detecting unknown attacks, though they often produce high false-positive rates[8].

Ahmed et al., 2016; Verma & Chandra, 2023; Gamal et al., 2021: The integration of machine learning (ML) techniques in IDS has significantly improved detection capabilities. ML models generalize patterns from historical data, enabling them to identify both known and novel threats. Among these, Decision

Trees[9] and ensemble methods have shown great promise for their simplicity, interpretability, and ability to handle mixed data types. Additionally, advanced techniques like few-shot learning are proving effective in identifying anomalous behaviours in edge networks.

Neto et al., 2023: Recent studies emphasize the importance of real-time processing and adaptive frameworks for modern IDS. For instance, the CICIoT2023 dataset introduces large-scale attack benchmarks for real-time systems, enhancing the evaluation of IDS models

Sundararajan & Garg, 2020; Kaur et al., 2023: Research also highlights the role of deep learning and ensemble methods in boosting detection accuracy and reducing false alarms (Sundararajan & Garg, 2020). Future advancements focus on privacy-preserving frameworks and distributed processing for large-scale attack environments. By leveraging these advancements, IDS systems are evolving into robust tools capable of addressing sophisticated and diverse cyber threats.

3. Methodology:

3.1 Dataset

The dataset[10] used forms the foundation of the IDS. It contains labelled instances of network traffic, with each record characterized by features that describe the nature of the connection. These include both categorical and numerical attributes[11]:

- *Categorical Features:*
  - protocol type: The protocol used (e.g., TCP, UDP)
  - service: The destination port's associated service (e.g., HTTP, FTP).
  - flag: Connection status, indicating outcomes such as successful connection (SF) or rejection (REJ).
- *Numerical Features:*
  - source bytes and destination bytes: Data volume exchanged between source and destination.
  - duration: The connection duration.
  - attack type: The target variable, indicating normal or specific attack types (e.g., DoS, Probe).

3.2 Data Preprocessing:

The raw dataset undergoes preprocessing[12] to prepare it for training the machine learning model:

1. *Label Encoding:* Categorical features are converted into numerical representations using LabelEncoder, ensuring compatibility with the Decision Tree classifier.
2. *Handling Missing Data:* Missing values, if present, are imputed with statistical measures[13] like mean or median to ensure data integrity.
3. *Data Splitting:* The dataset is partitioned into a training set (70%) for learning and a test set (30%) for evaluation, ensuring a robust performance assessment.

3.3 Feature Selection

This process eliminates redundant and irrelevant attributes, reducing computational complexity while improving model performance. The Decision Tree classifier inherently selects features based on their contribution to minimizing impurity during splits, further reinforcing the focus on significant attributes.

3.4 Model Training

The Decision Tree classifier was chosen for its simplicity, interpretability, and suitability for handling mixed data types. The training phase[13] involves:

- The Decision Tree algorithm iteratively splits the dataset based on feature importance, aiming to maximize information gain or minimize Gini impurity. The resulting tree structure comprises decision rules that classify network traffic into normal or specific attack types.
- The training dataset is used to teach the model the patterns in network traffic indicative of normal behaviour[14] or malicious activity. These patterns are learned through hierarchical splits, creating a model capable of generalizing across unseen data.

3.5 Model Evaluation

The trained model is evaluated on the test dataset using a comprehensive set of metrics:

- *Accuracy:* Assesses the overall correctness of predictions.
- *Precision:* Evaluates the proportion of true positive predictions (e.g., correctly detected attacks) among all positive predictions, highlighting the model's ability to minimize false alarms.
- *Recall:* Measures the model's ability to identify all actual instances of attacks (true positives).
- *F1-Score:* Combines precision and recall into a harmonic mean, providing a balanced measure of performance.

A *classification report* is generated, offering detailed metrics for each attack type, enabling insights into the system's strengths and weaknesses.

3.6 Visualization of Results

Performance metrics, including precision, recall, and F1-score, are visualized using bar charts. These visual summaries highlight the effectiveness of the model across different attack classes, identifying areas requiring improvement, such as underrepresented attack types like User-to-Root (U2R).

4.Results and Analysis :

The results of the Intrusion Detection System (IDS) project highlight the effectiveness of the Decision Tree classifier in detecting and classifying network traffic as either normal or indicative of specific cyber-attacks. The performance was assessed using various metrics, including accuracy, precision, recall, and F1-score, providing a comprehensive evaluation of the model's capabilities. Additionally, visualizations were utilized to analyze performance trends across different attack categories. The classification report (Table 1) and bar chart (Figure 2) provide complementary insights, with the chart offering an intuitive view of the classifier's performance across metrics. The results underscore the need for addressing class imbalance, possibly through techniques like oversampling, synthetic data generation, or the use of ensemble methods to improve recall for rare attack types

Classification Report

This Table 1 illustrates a detailed classification report for each category, including metrics for precision, recall, and F1-score.

1. *Precision:* Measures the proportion of true positive predictions among all positive predictions. The classifier achieved consistently high precision for major categories like normal traffic and DoS attacks, exceeding 99%. This demonstrates the model's ability to minimize false positives effectively.
2. *Recall:* Indicates the proportion of actual attack instances correctly identified. While the recall for frequent categories like normal traffic and DoS was excellent, rare attack types such as U2R showed lower recall, highlighting challenges with dataset imbalance.
3. *F1-Score:* Combines precision and recall into a single metric. High F1-scores for dominant attack types confirm the balanced performance of the classifier for these categories.

Table 1. Classification Report Metrics

	Precision	Recall	F1-score	Support
dos	1.00	1.00	1.00	13825
normal	1.00	1.00	1.00	20083
probe	0.99	0.99	0.99	3540
r2l	0.97	0.98	0.97	327
u2l	0.40	0.59	0.48	17
Accuracy			1.00	37792
Macro avg	0.87	0.91	0.89	37792
Weighted avg	1.00	1.00	1.00	37792

Bar Chart Visualization

The Figure 1 provides a visual representation of the precision, recall, and F1-score across the top attack classes.

1. *Strengths:*
   - Normal traffic and DoS attacks exhibit minimal variation across precision, recall, and F1-score, reflecting the model's strong capability to handle well-represented categories in the dataset.
   - Probe attacks also demonstrate consistent performance, though slightly lower than normal traffic and DoS.
2. *Areas for Improvement:*
   - For rare attack types like U2R, the recall is noticeably lower, indicating that the classifier struggles to detect these infrequent categories. This limitation is primarily due to the class imbalance in the dataset, where certain attack types have significantly fewer samples compared to others.

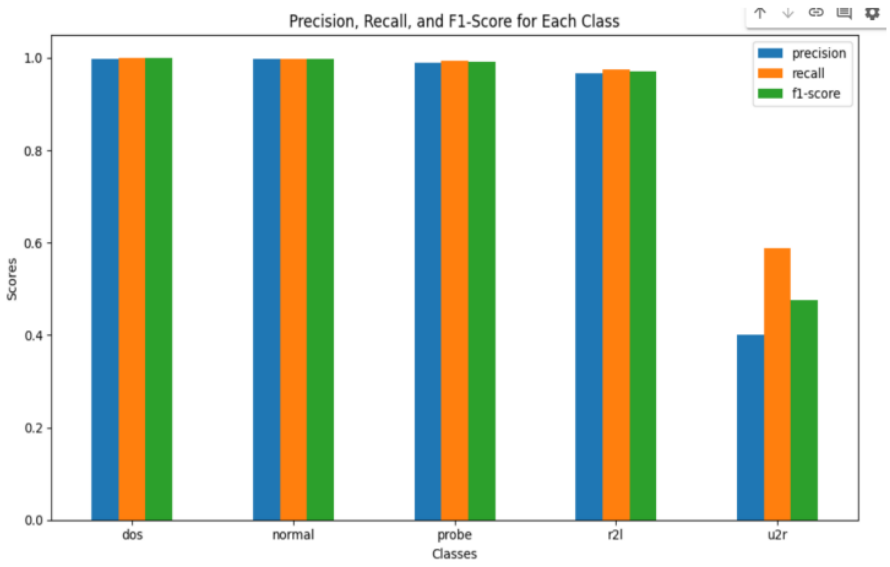

Figure 1. Bar Chart Visualization

5. Future Enhancements:

- Implement advanced machine learning techniques like Random Forests or Gradient Boosting to enhance accuracy and robustness. Deep learning models such as CNNs or RNNs can also be explored for detecting complex attack patterns.
- Incorporate real-time data processing through stream analytics or edge computing to improve responsiveness and enable immediate detection of threats.
- Address class imbalance using techniques such as oversampling, undersampling, or synthetic data generation like SMOTE to improve detection rates for rare attack types.
- Integrate behavioural analysis to identify subtle deviations from normal activity, enhancing the system's ability to detect evolving threats.
- Develop automated response mechanisms to isolate compromised systems or block malicious traffic, reducing manual intervention and response time.

6. Conclusion:

This research presents the development of an Intrusion Detection System (IDS) using a Decision Tree classifier, highlighting the role of machine learning in modern cybersecurity. The system effectively analyzes network traffic and classifies it as normal or one of several attack types, achieving an impressive accuracy of 99.69%. By preprocessing data, selecting relevant features, and employing robust evaluation metrics such as precision, recall, and F1-score, the IDS demonstrates high reliability in detecting common attack types. Despite its success, the system faces challenges with rare attack categories due to class imbalance, underscoring the need for future enhancements like advanced ensemble methods, synthetic data generation, and behavioral analysis. The integration of real-time detection and automated response mechanisms could further strengthen its applicability.Overall, this IDS offers a scalable and adaptive solution for safeguarding networks against evolving cyber threats, providing a solid foundation for future research in intrusion detection.

7.REFERENCES:

[1] Patil, N.V., Krishna, C.R., & Kumar, K. (2022). "SSK-DDoS: Distributed Stream Processing Framework Based Classification System for DDoS Attacks." Clust. Comput., 25, 1355–1372.
[2] Gamal, M., Abbas, H.M., Moustafa, N., Sitnikova, E., & Sadek, R.A. (2021). "Few-Shot Learning for Discovering Anomalous Behaviors in Edge Networks." Comput. Mater. Contin., 69, 1823–1837.
[3] Kaur, J., Agrawal, A., & Khan, R.A. (2023). "P2ADF: A Privacy-Preserving Attack Detection Framework in Fog-IoT Environment." Int. J. Inf. Secur., 22, 749–762.
[4] Verma, R., & Chandra, S. (2023). "RepuTE: A Soft Voting Ensemble Learning Framework for Reputation-Based Attack Detection in Fog-IoT Milieu." Eng. Appl. Artif. Intell., 118, 105670.
[5] Neto, E.C.P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A.A. (2023). "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment." Sensors, 23, 5941.
[6] Alshamrani, A. (2021). "An Overview of Machine Learning Techniques in Intrusion Detection Systems."
[7] Kumar, R., & Rani, S. (2020). "A Comparative Study of Machine Learning Algorithms for Intrusion Detection Systems."
[8] Sundararajan, V., & Garg, S. (2020). "Enhancing Intrusion Detection Systems Using Deep Learning."
[9] Patel, R., & Soni, S. (2021). "A Survey on the Application of Machine Learning in Intrusion Detection Systems."