STM JOURNALS

Journal of
Network Security

ISSN: 2395-6739 (Online)
ISSN: 2321-8517 (Print)
Volume 12, Issue 2, 2024
May–August
DOI (Journal): 10.37591/JoNS

https://journals.stmjournals.com/jons

Review JoNS

Network Intrusion Detection System Using Decision Tree

Yash Jadhav[1,*], Harsh Deshpande[2], Ashwini Garole[3], Komal Sawant[4], Aman Patil[5]

Abstract

This paper presents a novel approach to network intrusion detection systems (NIDS) using advanced decision tree algorithms to address critical limitations in existing IDS solutions. Traditional IDSs often struggle with high false positive and negative rates, lack of scalability, and poor interpretability. Our proposed IDS leverages decision trees to enhance detection accuracy, interpretability, and scalability, thereby improving network security. Decision trees are chosen for their adaptive learning capabilities, transparent decision-making processes, and efficiency in real-time threat detection. The system architecture includes key components such as a data collection module for capturing network traffic, a preprocessing module for data cleansing and feature extraction, and a decision tree classifier for classifying traffic into benign and malicious categories. The classifier's performance is rigorously evaluated using metrics like accuracy, precision, recall, and F1-score, demonstrating superior performance with a 100% accuracy rate in model evaluation. The IDS's effectiveness is compared against other machine learning techniques like K-nearest neighbors, logistic regression, and naive Bayes, with decision trees showing the highest accuracy and efficiency. The paper also highlights future directions, including enhanced machine learning integration, behavioral analysis, cloud-based deployment, internet of things security monitoring, and integration with threat hunting and incident response tools. This research underscores the potential of decision tree-based NIDS in providing robust, scalable, and comprehensible intrusion detection, crucial for protecting large-scale, dynamic network environments from diverse cyber threats.

Keywords: Machine learning algorithms, deep learning, classification techniques, decision tree, logistic regression, K-nearest neighbor (KNN), artificial neural network (ANN), supervised learning, anomaly detection, support vector machine, feature selection, data preprocessing, accuracy, precision, real-time intrusion detection

INTRODUCTION

An intrusion detection system (IDS) is a hardware or software application designed to identify any malicious or privacy-infringing activities within a network. Any kind of illegal activity is directly reported to the administrator of the system or it's been sent directly to the central system using a system information and event management (SIEM) system [1]. Any form of secure network should provide proper data privacy, data integrity and also assurance against denial of service (DOS) attack. The primary aim of the intrusion detection is to correctly identify any illegitimate use or exploitation of computer system. IDS is based on the assumption that the system can detect the behavioral difference. An intruder's activities as compared to legitimate user so that their unauthorized actions can be detected [2].

***Author for Correspondence**
Yash Jadhav
E-mail: yashjadhav0412@gmail.com

[1,2,4,5]Student, Computer Science and Engineering (Artificial Intelligence and Machine Learning), Vishwaniketan's iMEET Khalapur, Maharashtra, India
[3]Assistant Professor, Computer Science and Engineering, Vishwaniket's iMEET, Khalapur, Maharashtra, India

Received Date: May 21, 2024
Accepted Date: June 23, 2024
Published Date: July 02, 2024

Citation: Yash Jadhav, Harsh Deshpande, Ashwini Garole, Komal Sawant, Aman Patil. Network Intrusion Detection System Using Decision Tree. Journal of Network Security. 2024; 12(2): 22–33p.

Network intrusion detection systems (NIDS) are strategically positioned at key points within the network to monitor traffic to and from all connected devices.

NIDS performs analysis of all the currently passing traffic present on the entire subnet and matches it with the already known traffic generated by the previous attacks [3].

Once the unmatched behavior is spotted, an alert is directly sent to the network administrator. Also, NIDS can be classified into two types: online and off-line NIDS. Online NIDS deals with the real-time network data while the off-line NIDS looks after the stored data to check and decide whether it is an attack or not. NIDS can be used as a stand-alone network system or can be combined with other technologies to increase the overall detection and prediction rates [4]. One such example is using artificial neural network (ANN)-based IDS, which is capable of handling large volume of data in a very self-organizing structure which in results helps in better recognition of different intrusion patterns. As seen in Figure 1, we can look at the basic components present in the NIDS architecture.

An NIDS identifies malicious traffic on an existing network. Typically, NIDS requires special access to the system network in order to inspect the network, which includes uni-cast traffic as well. NIDS are passive devices that do not interfere with the traffic they monitor. As depicted in Figure 1, a typical NIDS architecture involves the NIDS sniffing the internal interface of the firewall in read-only mode. It then sends alerts to a NIDS management server through a separate network interface that supports read/write operations [5].

The aim of our project is to use machine learning and its different types of techniques to perform network intrusion detection. For this project, we will be using classification technique based on machine learning [6]. Classification is a supervised machine learning technique where the computer program learns from given input data and then uses this knowledge to categorize information into different forms.

Classification is the process of categorizing given data into various classes. It can be applied to both structured and unstructured data. In machine learning, classification techniques are further divided into different types of models, as illustrated in Figure 2 and listed below.

Key Features and Benefits

- *Adaptive Learning:* The decision tree algorithm facilitates adaptive learning by dynamically adjusting decision boundaries based on evolving network dynamics and threat profiles.
- *Interpretability:* The hierarchical nature of decision trees enables transparent decision-making, facilitating human understanding of detected threats and aiding in response prioritization.

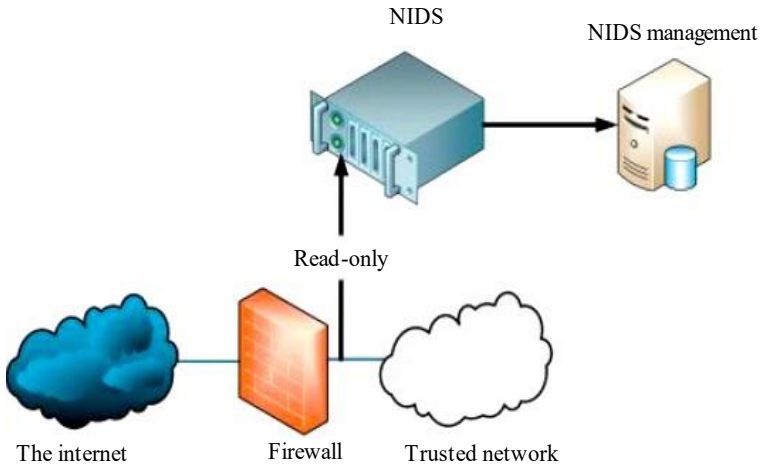

Figure 1. Netwrok intrusion detection system (NIDS) architecture [5].

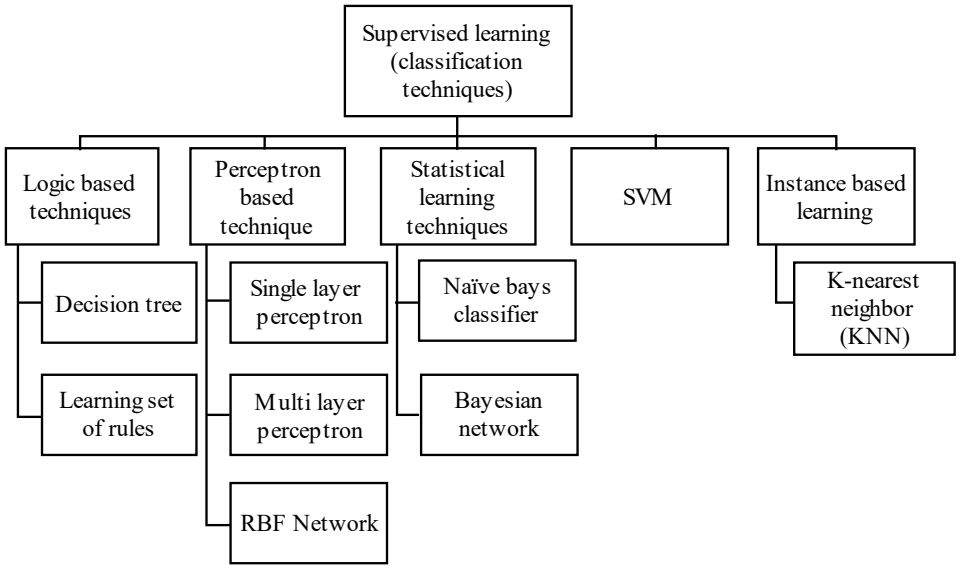

Figure 2. Machine learning and classification techniques.

- *Scalability:* The proposed IDS architecture is designed to scale seamlessly with growing network infrastructures, accommodating large volumes of data traffic without compromising performance.
- *Real-Time Detection:* Leveraging efficient tree traversal algorithms, the system enables real-time detection and response to emerging security threats, minimizing response latency and mitigating potential damages.

RELATED WORKS

Network Intrusion Detection [7]

- The research done by Mukherjee et al. [7], states that intrusion detection is a new approach in the field of security in preexisting computers and other data networks.
- In their research, they concluded that IDSs were based on host-audit-trail and network traffic analysis and their goal was to detect attacks possibly in real time [8].
- A number of prototypes IDSs were built, and their overall concept was quite promising.
- There were number of limitations in prevention-based approach for network security. Some of them are: It was impossible to build a useful system which was precisely secure. Due to some design flaws in the system large number of components cannot be excluded.
- Crypto-based systems was not able to prevent attack against lost or stolen keys, or hacked passwords.

Classification Techniques in Machine Learning: Applications and Issues [9]

- Soofi and Awan [9] took the concept of classification and gave a detailed information about different type of classification technique that are used in machine learning.
- In the research various classification techniques have been in discussed in detail along with their useful applications. An overview of different classification technique with their applicable applications is presented.

Network Traffic Classification Techniques and Comparative Analysis Using Machine Learning Algorithms [10]

- The paper states the theoretical concepts about different machine learning algorithms based on network intrusion detection anomaly.
- Machine learning technique was based on labeled datasets. In this approach, a machine learning classifier is trained with input data for the system. Subsequently, the classifier uses the trained sample predictions to classify unknown classes.

- Network traffic classification was the first step to identify and correctly analyze various types of applications flowing in a network.
- Using this technique helped internet service providers (ISPs) or network operators to manage and upgrade overall performance of the network.
- There were multiple techniques available to classify the internet traffic such as port based, pay load based, and machine learning out of which machine learning was the most popular and useful.

Practical Real-Time Intrusion Detection Using Machine Learning Approaches [11]

- The growing frequency of system attacks is a well-known problem that can jeopardize the availability, confidentiality, and integrity of critical information for both individuals and businesses.
- The study presented here introduces an intrusion detection method that employs a supervised machine learning technique in real time. Our approach is swift, efficient, and applicable to various machine learning techniques.
- They have used various efficient machine learning algorithms to analyze the efficiency of IDS approach. Among the various methods, hypotheses showed that the decision tree approach is capable of surpassing other techniques.
- With the above result, and using the decision tree algorithm, they developed a real-time intrusion detection system that helps in classifying the attack data among the normal online network data.

TECHNIQUES USED

Naive Bayes Classifier

Naive Bayes classifier is a Bayes' theorem–based algorithm, making it the simplest instance of a probabilistic classifier. It is one of the statistical classifications and requires less training data to estimate the number of parameters [12].

Naive Bayes is considered one of the fastest classifiers as it is highly scalable and can easily handle both discrete and continuous data. This algorithm is used to make predictions in real time [13].

Naive Bayes is a type of supervised learning algorithm that uses a maximum likelihood method of parameter estimation.

Naive Bayes classifiers come in various forms, including Bernoulli, Gaussian, and multinomial types [14].

Decision Tree

A decision tree, employed in supervised machine learning for classification and prediction tasks, takes the form of a flowchart and is adept at managing datasets with numerous dimensions (Figure 3).

Root Node: The initial node, symbolizing the complete dataset, diverges into branches guided by decisions.

Internal Nodes (Decision Nodes): Represented by squares, these nodes ask questions about the data based on features.

Leaf Nodes (Terminal Nodes): Represented by triangles, these are the final outcomes, representing classifications or predictions.

Branches: Represent possible paths based on decisions made at internal nodes.

K-Nearest Neighbors

K-nearest neighbors (KNN) is a simple and intuitive classification technique used to categorize new data points based on their similarity to existing data in the training set.

Core Principle: Identify the K closest data points (neighbors) in the training data for a new data point.

Classify the new data point based on the majority vote (for classification) or average value (for regression) of its K neighbors.

Distance Metric: Euclidean distance is a common metric to measure similarity between data points. It computes the Euclidean distance between two points within a space of any number of dimensions (Figure 4).

PROPOSED METHODOLOGY

The proposed NIDS architecture comprises the following key components as shown in Figure 5.

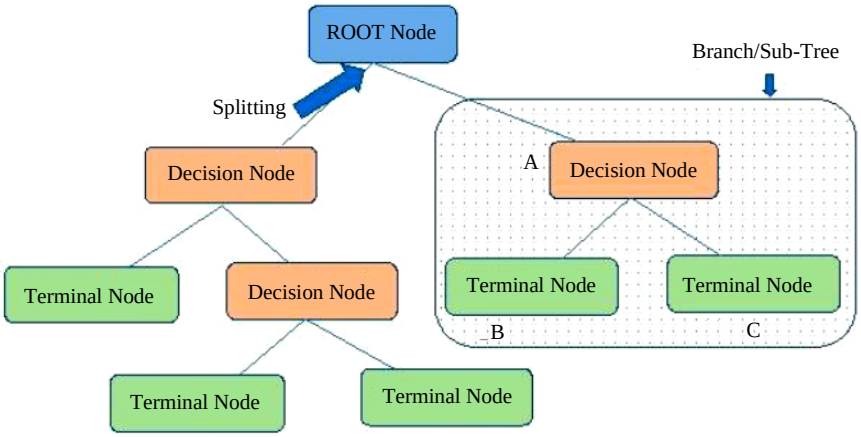

Figure 3. Decision tree.
Note: A is parent node of B and C.

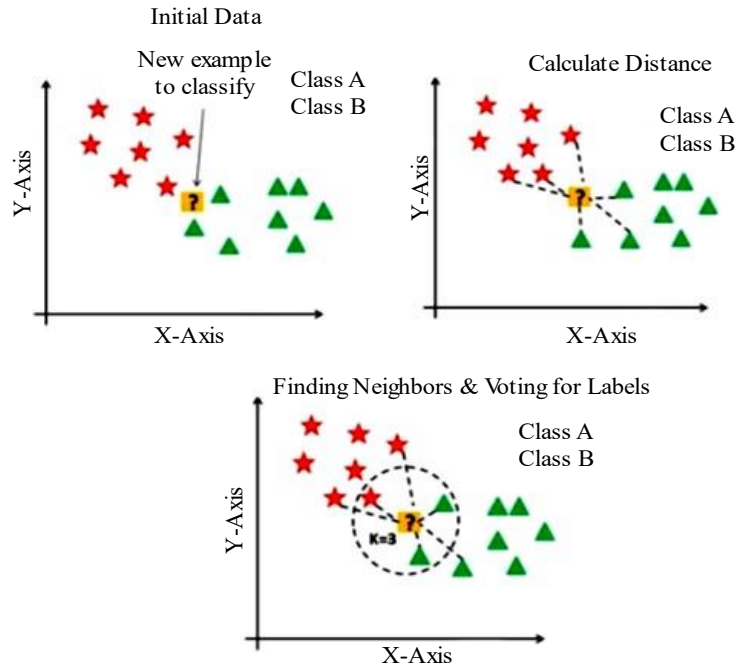

Figure 4. Distance metric.

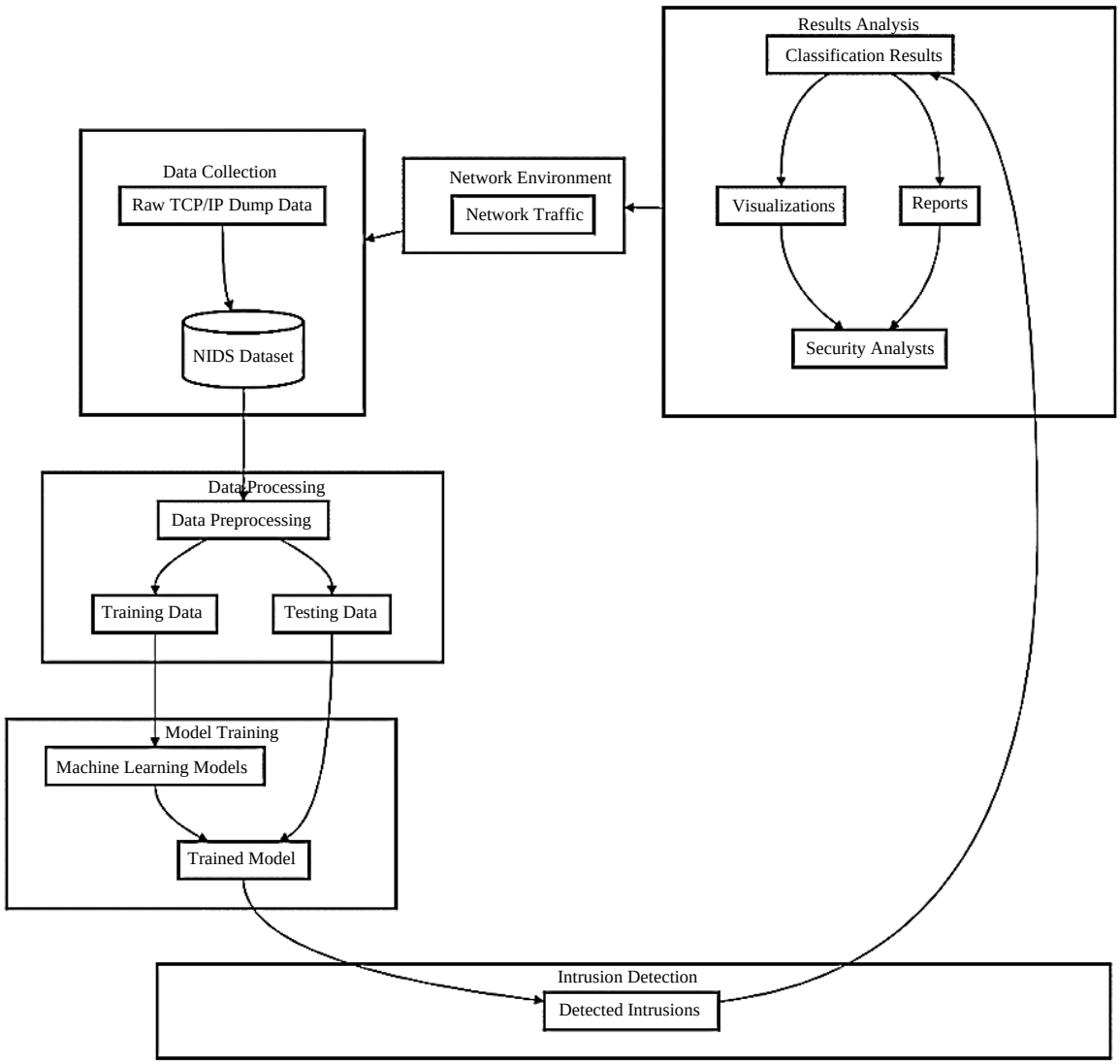

Figure 5. Network intrusion detection system (NIDS) architecture.

Data Collection Module

Responsible for capturing and aggregating network traffic data from various sources, including routers, switches, and firewalls [15]. Utilizes packet sniffing techniques and network monitoring tools to gather comprehensive data sets encompassing both inbound and outbound traffic as shown in Figure 6.

Preprocessing Module

The preprocessing module cleanses and preprocesses raw network data to extract relevant features and eliminate noise.

Decision Tree Classifier

The decision tree classifier utilizes the decision tree algorithm to construct a hierarchical tree structure for classifying network traffic into benign and malicious categories. It implements recursive partitioning to split the feature space into increasingly homogeneous subsets, enabling efficient discrimination of anomalous patterns [16].

Model Evaluation Module

The model evaluation module assesses the performance of the decision tree classifier using metrics such as accuracy, precision, recall, and F1-score as shown in Figure 7.

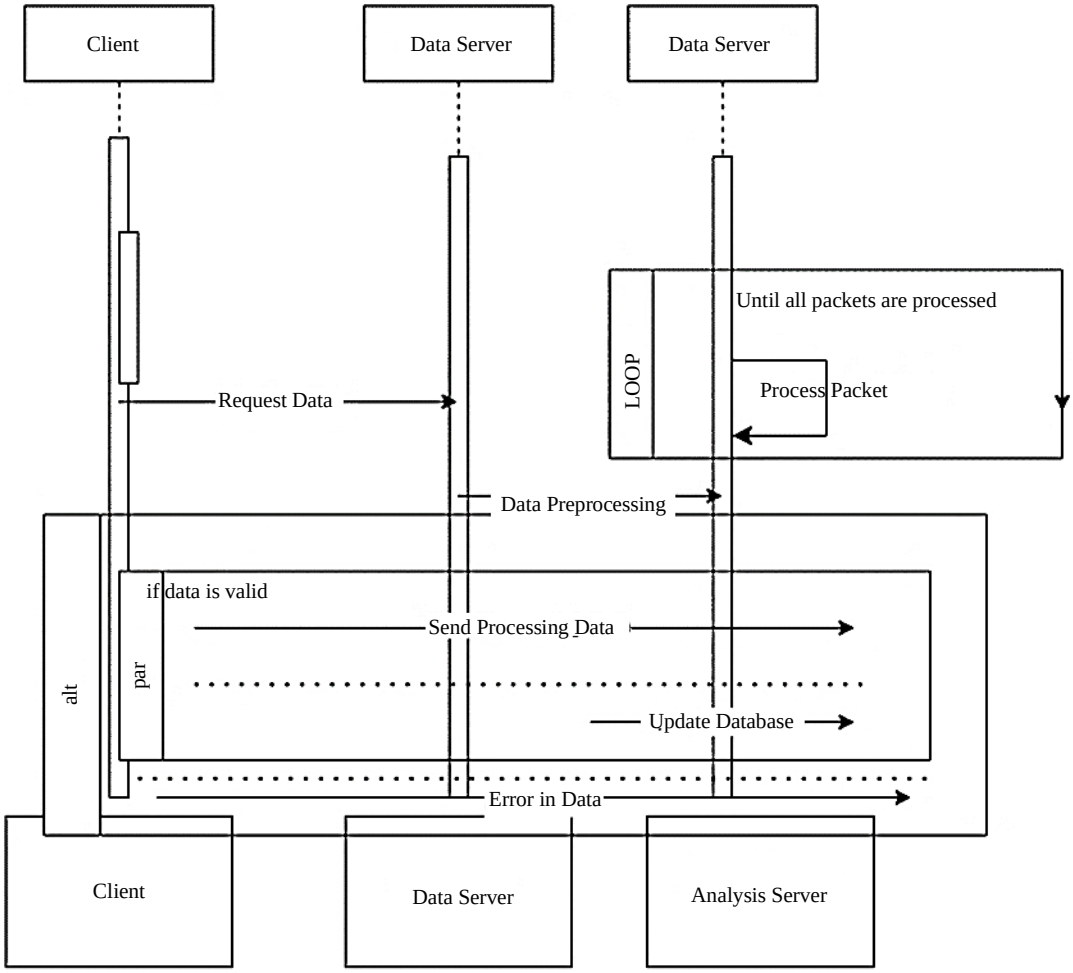

Figure 6. Architecture of network modeling tool.

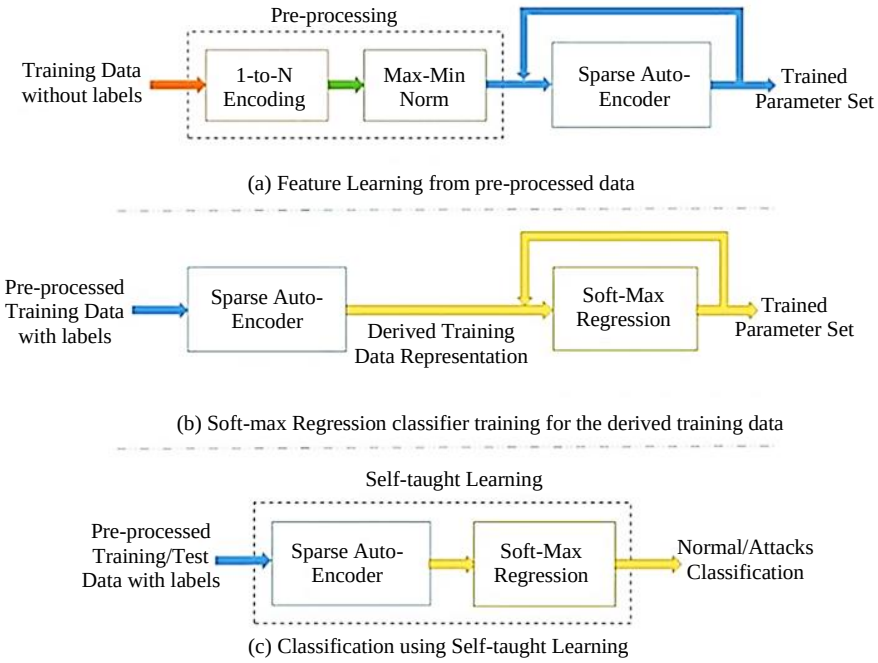

Figure 7. Steps involved in network intrusion detection system (NIDS) implementation.

Feature Selection

Feature selection is the process where we can automatically or manually select features from the dataset that can contribute the most for our prediction variable or output in which we are interested in [17]. We have done feature selection using random forest classifier. We imported the Random Forest Classifier library from Scikit [18].

Random forest is used in conjunction with feature selection because it is a tree-based strategy which ranks and selected the features based on the purity of the node, which in our case is Source.

Recursive Feature Elimination

Recursive feature elimination (RFE) is a feature selection method that fits into a model and is used to eliminate all the non-essential features that are present or until the specified number of features are reached in the dataset [19].

IMPLEMENTATION AND RESULTS

Model evaluation constitutes a crucial aspect of the machine learning model development process. Finding the optimal model that accurately represents our dataset is crucial for predicting its future performance effectively [20]. For our dataset, we have selected four classification techniques.

Decision Tree

Based on the results produced using decision tree in model evaluation, the model showed an accuracy of 100% and with cross validation mean score of 0.99. All the accuracy metrics such as accuracy, precision, recall, and F-1 score were highest among all other classification techniques used in the procedure (Figure 8).

K-Nearest Neighbors Classifier

Performing KNN showed result with model accuracy of 99.37% along with cross validation mean score of 0.99. The results were also high in terms of anomaly precision and normal network recall (Figure 9).

Logistic Regression

Results of logistic regression were achieved with model accuracy of 95.49% and cross validation mean score of 0.95 (Figure 10 and Table 1).

```
============================== Decision Tree Classifier Model Evaluation ==============================

Cross Validation Mean Score:
 0.9956333771928133

Model Accuracy:
 1.0

Confusion matrix:
 [[8245    0]
 [   0 9389]]

Classification report:
              precision    recall  f1-score   support

     anomaly       1.00      1.00      1.00      8245
      normal       1.00      1.00      1.00      9389

    accuracy                           1.00     17634
   macro avg       1.00      1.00      1.00     17634
weighted avg       1.00      1.00      1.00     17634
```

Figure 8. Decision tree classifier model.


```
============================== KNeighborsClassifier Model Evaluation ==============================

Cross Validation Mean Score:
 0.9914370153431007

Model Accuracy:
 0.9937620505840989

Confusion matrix:
 [[8168   77]
 [  33 9356]]

Classification report:
              precision    recall  f1-score   support

     anomaly       1.00      0.99      0.99      8245
      normal       0.99      1.00      0.99      9389

    accuracy                           0.99     17634
   macro avg       0.99      0.99      0.99     17634
weighted avg       0.99      0.99      0.99     17634
```

Figure 9. Performing K-nearest neighbors (KNN).

```
============================== LogisticRegression Model Evaluation ==============================

Cross Validation Mean Score:
 0.9537821405741347

Model Accuracy:
 0.9549166383123512

Confusion matrix:
 [[7763  482]
 [ 313 9076]]

Classification report:
              precision    recall  f1-score   support

     anomaly       0.96      0.94      0.95      8245
      normal       0.95      0.97      0.96      9389

    accuracy                           0.95     17634
   macro avg       0.96      0.95      0.95     17634
weighted avg       0.96      0.95      0.95     17634
```

Figure 10. Logistic regression model.

Table 1. Model accuracy of logistic regression.

	Model Validation			
	Accuracy	***Precision***		***Recall***
Decision tree	99.56	99.49%	100%	99%
K-nearest neighbors	99.14	99.16%	99%	99%
Logistic regression	95.37	95.54%	95%	97%
Naïve Bayes	90.71	90.67%	88%	95%

Naive Bayes

Naive Bayes results had the least model accuracy and cross validation mean score. Model accuracy of 90.71% and cross validation mean score of 0.90 were obtained (Figure 11 and Table 2).


```
============================== Naive Baye Classifier Model Evaluation ==============================

Cross Validation Mean Score:
 0.9071666840303904

Model Accuracy:
 0.9071679709651809

Confusion matrix:
 [[7000 1245]
 [ 392 8997]]

Classification report:
              precision    recall  f1-score   support

     anomaly       0.95      0.85      0.90      8245
      normal       0.88      0.96      0.92      9389

    accuracy                           0.91     17634
   macro avg       0.91      0.90      0.91     17634
weighted avg       0.91      0.91      0.91     17634
```

Figure 11. Naïve Bayes classifier model.

Table 2. Naive Bayes model accuracy.

Model Evaluation			
Accuracy	***Precision***	***Recall***	***Cross Validation Score***
Decision tree	100%	100%	100%
K-nearest neighbors classifier	99.37%	99%	100%
Logistic regression	95.49%	95%	97%
Naive Bayes	90.71%	88%	96%

CONCLUSION

In this article, we have presented a practical and efficient NIDS using classification and deep learning technique, which can be implemented on other machine learning algorithms and on existing systems.

Intrusion detection is a realistic and functional solution in offering a particular definition of defense in our large and current networks (possible uncertainty), computers, and networking programs. Intrusion monitoring systems are based on host audit-trail and network track analysis and are designed to detect threats, preferably in real time.

Various popular classification and deep learning techniques for machine learning have been discussed with their basic working mechanisms, strengths, and weaknesses. Potential implementations and problems related to their possible approaches have also been highlighted. Classification methods tend to be strong in modeling interactions.

- In the research study, supervised machine learning systems and rules were applied to intrusion detection data sets to forecast the probability of attack in the network context and the performance of the learning methods were evaluated on the basis of their predictive precision and ease of learning.
- A modern intelligent NIDS has been developed using a two-stage (anomaly-misuse) classification and deep learning methodology, as it is checked. Decision tree, logistic regression, KNN, and naive Bayes were employed for classifying potential traffic disruptions, while deep learning utilized an artificial neural network (ANN) to detect and classify attacks if they manifest. The complete 42-dimensional characteristics of the training data collection have been used in the experiment.
- In both algorithms (stage-1 and stage-2), separate functions and parameters are evaluated. The results of the evaluation show that the high accuracy rate is 100% with a recall rate of 0.10 at

stage 1 of the evaluation and the accuracy rate of the decision tree is 99.5% with a recall rate of 0.99 at stage 2 of the validation.

- All the results of the various models show that decision tree algorithm results in high accuracy compared to other logistic regression, KNN and naive Bayes machine learning classifiers. Like the neural network–based approach, our approach has a higher detection efficiency, less time-consuming and a low-cost factor. It does, though, produce a few more false positives.

Future Scope

Enhanced Machine Learning Integration

Investigate cutting-edge machine learning methodologies like deep learning and reinforcement learning to enhance the IDS's ability to detect threats with greater accuracy and adaptability to evolving cybersecurity challenges. This could involve developing models capable of learning from real-time network data and dynamically adjusting detection algorithms based on detected anomalies.

Behavioral Analysis and User Profiling

Incorporate behavioral analysis capabilities into the IDS to profile user behavior and identify deviations from normal patterns. By understanding typical user activities and network behavior, the IDS can better distinguish between legitimate and malicious activities, reducing false positives and improving overall detection efficiency.

Cloud-Based Deployment and Scalability

Explore options for deploying the IDS in cloud environments, leveraging scalable infrastructure and elastic computing resources to handle fluctuating network loads effectively. This capability would allow organizations to deploy and oversee IDS systems with greater flexibility, adapting to expansion and changing network environments.

Internet of Things Security Monitoring

Extend the IDS's capabilities to include monitoring and securing internet of things (IoT) devices and networks. With the proliferation of IoT devices in enterprise environments, there is a growing need to detect and mitigate threats targeting these devices, such as IoT botnets and device exploits.

Integration with Threat Hunting and Incident Response Tool

Integrate the IDS with threat hunting platforms and incident response tools to streamline the investigation and remediation process. By correlating IDS alerts with threat intelligence data and historical incident data, security teams can gain deeper insights into security incidents and respond more effectively to security breaches.

REFERENCES

1. Ojha Y. A Beginners Guide to Machine Learning — Data Preprocessing. [Online]. Medium.com. February 17, 2019. Available at https://medium.com/@yogeshojha/data-preprocessing-75485c7188c4
2. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015; 521 (7553): pp. 436–444.
3. Machine Learning Glossary—ML Glossary documentation. [Online]. Readthedocs.io. 2022. Available at https://ml-cheatsheet.readthedocs.io/en/latest/
4. Dong B, Wang X. Comparison deep learning method to traditional methods using for network intrusion detection. In: 2016 8th IEEE International Conference on Communication Software and Networks (ICCSN), Beijing, China, June 4–6, 2016. pp. 581–585.
5. Conrad E, Misenar S, Feldman J. Domain 7: security operations. In: Eleventh Hour CISSP®: Study Guide. Third Edition. Rockland, MA, USA: Syngress; 2017. pp. 145–183.
6. Brownlee J. What is deep learning? MachineLearningMastery.com. [Online]. 2019. Available at https://machinelearningmastery.com/what-is-deep-learning/
7. Mukherjee B, Heberlein LT, Levitt KN. Network intrusion detection. IEEE Netw. 1994; 8 (3): 26–41.

8. Pedamkar P. Machine Learning Algorithms. [Online]. EDUCBA.com. 2019. Available athttps://www.educba.com/machine-learning-algorithms/
9. Soofi AA, Awan A. Classification techniques in machine learning: applications and issues. J Basic Appl Sci. 2017; 13 (1): 459–465.
10. Shafiq M, Yu X, Laghari AA, Yao L, Karn NK, Abdessamia F. Network traffic classification techniques and comparative analysis using machine learning algorithms. In: 2016 2nd IEEE International Conference on Computer and Communications (ICCC), Chengdu, China, October 14–17, 2016. pp. 2451–2455.
11. Sangkatsanee P, Wattanapongsakorn N, Charnsripinyo C. Practical real-time intrusion detection using machine learning approaches. Computer Commun. 2011; 34 (18): 2227–2235.
12. Farnaaz N, Jabbar MA. Random forest modeling for network intrusion detection system. Procedia Computer Sci. 2016; 89: 213–217.
13. Taulli T. Deep Learning: What You Need to Know. [Online]. Forbes. October 5, 2023. Available at https://www.forbes.com/sites/tomtaulli/2020/03/27/deep-learning-what-you-need-to-know/
14. Ganapathy S, Yogesh P, Kannan A. An intelligent intrusion detection system for mobile ad-hoc networks using classification techniques. In: Advances in Power Electronics and Instrumentation Engineering: Second International Conference, PEIE 2011, Nagpur, Maharashtra, India, April 21–22, 2011. Berlin, Germany: Springer; 2011. pp. 117–122.
15. Niyaz Q, Sun W, Javaid AY, Alam M. A deep learning approach for network intrusion detection system. In: Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONET-ICS), New York, NY, USA, December 3–5, 2015. pp. 21–26.
16. Ahmim A, Maglaras L, Ferrag MA, Derdour M, Janicke H. A novel hierarchical intrusion detection system based on decision tree and rules-based models. In: 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini, Greece, May 29–31, 2019. pp. 228–233.
17. Li W. Using genetic algorithm for network intrusion detection. In: Proceedings of the United States Department of Energy Cyber Security Group 2004 Training Conference, Kansas City, KS, USA, May 2004. pp. 1–8.
18. Harrison O. Machine learning basics with the k-nearest neighbors algorithm. [Online]. Towards Data Science. September 11, 2018. Available at https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761
19. Hussain J, Lalmuanawma S, Chhakchhuak L. A two-stage hybrid classification technique for network intrusion detection system. Int J Comput Intell Syst. 2016; 9 (5): 863–875.
20. Sanjeevi M. Chapter 4: Decision Trees Algorithms. [Online]. Deep Math Machine learning.ai - Medium. October 6, 2017. Available at https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1

