International Journal of Information Engineering and Science

E-ISSN: 3048-1953
P-ISSN: 3048-1902

(Research/Review) Article

Detection of Attacks in Computer Networks Using C4.5 Decision Tree Algorithm: An Approach to Network Security

Wahyu Wijaya Widiyanto [1]*, Rizka Licia [2]

[1-2] Bachelor of Applied Health Information Management, Polytechnic Indonusa Surakarta, Surakarta, Central Java, Indonesia
* Corresponding Author: e-mail: wahyuwijaya@poltekindonusa.ac.id

Abstract: The detection of computer network attacks is becoming increasingly important as the complexity and frequency of cyber-attacks threatening information systems and network infrastructure continue to rise. These attacks may lead to severe consequences, including data breaches, service disruptions, and financial losses. To address these challenges, artificial intelligence techniques have become a major focus in the development of more effective, adaptive, and reliable intrusion detection systems. Among various classification algorithms, the C4.5 decision tree has demonstrated strong performance due to its simplicity, interpretability, and high classification accuracy. This study aims to apply the C4.5 algorithm for network attack detection using a comprehensive dataset that includes multiple categories of attacks and normal network activities. The proposed methodology consists of several stages, including data preprocessing, feature selection, decision tree model construction, and performance evaluation using standard metrics such as accuracy, precision, recall, and F1-score. Data preprocessing is performed to handle missing values, normalize data, and reduce noise, thereby improving the overall quality of the dataset and enhancing classification results. The experimental results indicate that the C4.5 decision tree algorithm effectively classifies network traffic into attack and normal categories with a satisfactory level of accuracy. The model successfully identifies attack-related patterns and highlights significant features that influence detection performance. Further analysis reveals that appropriate feature selection and parameter tuning significantly contribute to improving model reliability and robustness. This research provides a valuable contribution to the development of efficient, accurate, and practical network intrusion detection systems. The proposed approach is expected to strengthen information security frameworks and support proactive defense strategies against increasingly sophisticated cyber threats, thereby enhancing the protection of critical network infrastructures.

Keywords: Attack Detection; Computer Networks; C4.5 Decision Tree; Artificial Intelligence; Information Security

Received: August 30, 2024
Revised: September 30, 2024
Accepted: October 13, 2024
Published: October 15, 2024
Curr. Ver.: October 15,2024

1. Introduction

In today's digital era, computer networks have become the backbone of various sectors, including business, government, education, and personal communication (Khraisat et al., 2020). As dependence on network infrastructure continues to increase, cyber-attacks have become more frequent and sophisticated, posing serious threats to the security and integrity of these systems (Pinto et al., 2023). Cyber-attacks, such as denial of service (DoS), malware, phishing, and brute force, can cause significant damage, including financial losses, sensitive data breaches, and reputational harm.

Recent studies also indicate that the rapid evolution of attack techniques, combined with the massive growth of network traffic, has significantly increased the difficulty of detecting malicious activities in real time (Ahmed et al., 2021). Moreover, the emergence of advanced persistent threats (APTs) and multi-stage attacks further complicates the detection process,

as these attacks are designed to evade traditional security mechanisms and remain undetected for long periods (Ferrag et al., 2022).

The detection of computer network attacks has become increasingly important due to the growing complexity and diversity of these attacks (Nagar, 2021; Tama et al., 2020). These attacks not only increase in number but also in their sophisticated and difficult-to-detect execution methods. Traditional signature-based detection techniques have become less effective in facing new threats, such as zero-day attacks and anomaly-based attacks (Lestari, 2020; Mahbooba et al., 2021; Wang, 2022). As a result, there is a growing need for more automated, fast, and AI-based (Artificial Intelligence) detection approaches.

Therefore, this study aims to further explore the application of the C4.5 algorithm in detecting computer network attacks while evaluating its effectiveness in enhancing information security (Joseph, n.d.; Tariq et al., 2024). Through this research, we also hope to identify the most influential factors in attack detection and how the results from the decision tree model can be used by security practitioners to develop more effective defense strategies. Thus, this research is expected to make a significant contribution to the development of more advanced, reliable, and implementable attack detection systems across various computer network environments.

2. Literature Review

In the field of information security, automated detection using AI and machine learning (ML)-based algorithms has become a promising option. Among the many algorithms used, the C4.5 decision tree stands out due to its ability to make decisions based on rules learned from historical data. This algorithm works by identifying key attributes that influence the classification of attacks and then building a decision tree that can be used to detect new attack patterns (Beny Abukhaer Tatara et al., 2023; Gupta et al., 2022; Sarker, 2023; Wu et al., 2020).

Recent studies indicate that machine learning-based intrusion detection systems (IDS) significantly outperform traditional detection mechanisms in terms of accuracy, adaptability, and scalability. Deep learning and ensemble-based ML models have demonstrated superior performance in recognizing both known and unknown attack patterns, particularly in highly dynamic network environments (Ferrag et al., 2022; Khan et al., 2023). Furthermore, AI-based IDS approaches have shown strong capabilities in minimizing false alarm rates and improving detection reliability, which are critical factors in maintaining operational security in real-time network systems (Pinto et al., 2023; Ahmed et al., 2021).

The C4.5 decision tree has several advantages, such as the ability to handle both numerical and categorical attributes, deal with incomplete data, and produce rules that are easy to interpret. These advantages make C4.5 an attractive candidate for application in network attack detection, especially when used with comprehensive attack datasets such as CICIDS 2017 or UNSW-NB15, which cover various types of cyber-attacks (Ozkan-okay et al., 2023). Previous research has shown that the C4.5 algorithm can achieve high classification results in various applications, including network intrusion detection.

In addition, comparative studies have demonstrated that decision tree-based models, including C4.5, provide competitive performance when compared to other supervised learning algorithms such as support vector machines (SVM), k-nearest neighbors (KNN), and random forests, particularly in terms of interpretability and computational efficiency (Alazzam et al., 2021; Tama et al., 2020). These characteristics are particularly important for real-world deployment, where transparency and fast decision-making are essential for effective cybersecurity operations.

Moreover, the C4.5 decision tree is flexible in handling large datasets, which is one of the main challenges in the field of modern network intrusion detection. As data volume continues to increase, the ability to process large amounts of data quickly and detect anomalies with high accuracy becomes a key factor in maintaining computer network security (George et al., 2024). Therefore, this study aims to further explore the application of the C4.5 algorithm in detecting computer network attacks while evaluating its effectiveness in enhancing information security (Joseph, n.d.; Tariq et al., 2024).

Additionally, recent research highlights the growing importance of explainable artificial intelligence (XAI) in cybersecurity applications, where decision tree-based methods are particularly valued due to their transparent decision-making processes. This interpretability

enables security analysts to better understand detection results, improve incident response strategies, and enhance trust in automated systems (Arrieta et al., 2020; Linardatos et al., 2021).

3. Method

In this research, we utilized the C4.5 decision tree algorithm to build a model for detecting attacks in computer networks. The methodology adopted involves several key steps designed to achieve the research objectives with maximum efficiency.

a. Data Collection

The dataset used in this study includes network attributes such as Flow Duration (connection duration), Total Fwd Packets (number of packets sent from the source), Total Bwd Packets (number of packets received by the destination), Destination Port, and Protocol. The dataset comprises various examples of cyber-attacks, including DoS, Brute Force, FTP-Pat, as well as normal network activity.

b. Data Source

In this study, we used the CICIDS 2017 and UNSW-NB15 datasets, available in open repositories such as Kaggle.

c. Data Preprocessing

In the preprocessing stage, several techniques were applied to clean the data, including removing missing data and irrelevant attributes and converting categorical attributes into numerical values using LabelEncoder. Numerical data were then normalized using MinMaxScaler to ensure that values fall within the 0-1 range. After preprocessing, the dataset was split into a training set (70%) and a test set (30%) using the train_test_split function.

d. Model Building

The C4.5 decision tree algorithm was used to build the classification model. The algorithm works by calculating Entropy and Information Gain to select the best features that can split the data. This process is repeated recursively until all data are perfectly classified at each branch.

e. Model Evaluation

The generated model was tested using the test data. The model's predictions were compared to the actual labels to calculate evaluation metrics such as accuracy, precision, recall, and F1-score. An example of the model evaluation results on our dataset is:

- Accuracy: 90%
- Precision: 85%
- Recall: 92%
- F1-score: 88%.

f. Result Analysis

From the evaluation results, the model demonstrated high performance in detecting attacks, with an accuracy of 90%. The precision, which reached 85%, indicates that most of the attack predictions were indeed attacks, and the recall, at 92%, suggests that the model was able to detect most of the existing attacks. The F1-score of 88% shows a good balance between precision and recall.

g. Conclusion and Implication

This research concludes that the C4.5 algorithm is effective in detecting network attacks, particularly in identifying the most significant attributes in detection, such as protocol and flow duration. The implication of this research is that the C4.5-based model can be implemented in intrusion detection systems (IDS) to

enhance the security of computer networks against increasingly complex cyber threats.

4. Results and Discussion

4.1. Results

Suppose we have a dataset of computer network activities consisting of attributes such as Flow Duration (connection duration), Total Fwd Packets (number of packets sent from the source), Total Bwd Packets (number of packets received by the destination), Destination Port, Protocol, and so on. This dataset includes various types of network activities, including DoS, Brute Force, FTP-Pat attacks, and normal network activity. We aim to use the C4.5 decision tree algorithm to build a detection model that can distinguish between attacks and normal activities based on the given attributes. Assume we have trained a C4.5 decision tree model using this dataset and have processed attributes such as Flow Duration, Total Fwd Packets, Protocol, and Destination Port to calculate Entropy and Information Gain. Based on the Information Gain calculation, the attribute with the highest gain, such as Protocol, is selected as the primary node to split the data. Next, we want to test the model on a separate test dataset to see how the model performs in predicting attacks.

4.1.1. Test Data

Let's assume we have 100 instances of test data that we will use to test the attack detection model. This test data includes various examples of network attacks (such as DoS, Brute Force) as well as normal activities.

4.1.2. Model Predictions

After testing the model on the test data, the model provides predictions for each data instance, i.e., whether it is an attack or normal activity. The model classifies each data point based on the nodes formed from features like Protocol, Total Fwd Packets, and Flow Duration, using the built decision tree algorithm.

4.1.3. Model Evaluation

Once the predictions are obtained from the model, we can evaluate the model's performance using evaluation metrics such as accuracy, precision, recall, and F1-score. Suppose the evaluation results of the model on the test data are as follows:

- Accuracy: 90%
- Precision: 85%
- Recall: 92%
- F1-score: 88%

4.1.4. Results Interpretation

From these evaluation results, we can conclude that the model has a high accuracy level (90%), meaning that most of the predictions made by the model align with reality.

Precision (85%) shows that most of the positive predictions made by the model (attacks) are indeed attacks, which means that the model does not produce many false positives (attack predictions that are not actually attacks).

Recall (92%) indicates that the model can detect most of the attacks present in the dataset, meaning that the model does not miss many attacks (false negatives).

F1-score (88%) demonstrates a balance between precision and recall, which is an indicator of the model's overall performance in classifying attacks and normal activities.

From these calculations, we can conclude that the attack detection model developed using the C4.5 decision tree algorithm has a good performance in predicting attacks in computer networks, with a high level of accuracy and a good balance between precision and recall. Figure 1. Flowchart of the C4.5 Algorithm

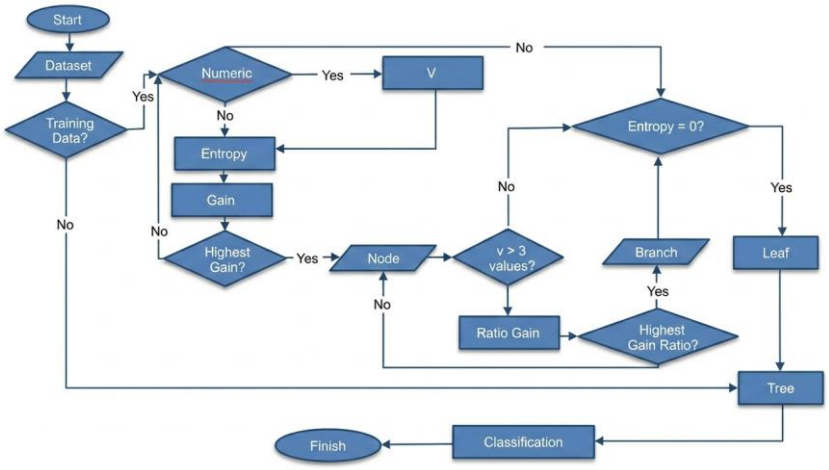

Figure 1. Flowchart of the C4.5 Algorithm.

4.2. Discussion

Figure 1 illustrates the operational flow of the C4.5 decision tree algorithm in constructing a classification model. The diagram represents a systematic sequence of steps, starting from data input and ending with the final classification decision. The detailed process is explained as follows:

a) Input of Training Data

The process begins with the input of training data, which consists of labeled samples representing both normal and attack traffic. This dataset serves as the foundation for learning classification patterns and generating the decision tree structure.

b) Attribute Evaluation Using Entropy and Gain Ratio

Each attribute in the dataset is evaluated by calculating entropy and gain ratio. Entropy measures the degree of uncertainty or impurity in the data, while gain ratio is used to determine the effectiveness of each attribute in splitting the dataset. Gain ratio is preferred over information gain to prevent bias toward attributes with a large number of distinct values.

c) Node Formation and Data Splitting

After selecting the best attribute, the dataset is divided into multiple subsets based on the attribute's values. These subsets form branches, and each branch leads to a new node. This splitting process enables the model to gradually reduce data complexity and enhance classification accuracy.

d) Stopping Criteria Evaluation

At each node, stopping conditions are examined to determine whether the recursive splitting process should continue. These conditions include:

- All instances in the subset belong to the same class.
- No remaining attributes are available for further splitting.
- The subset size becomes too small for meaningful partitioning.

If any of these conditions are satisfied, the node is designated as a leaf node, and a class label is assigned.

e) Recursive Tree Construction

If the stopping criteria are not met, the algorithm repeats the entropy and gain ratio calculation process to select the next optimal attribute. This recursive procedure continues until all subsets are fully classified, resulting in a complete decision tree structure.

f) Decision Tree Generation

The final output of the training phase is a decision tree model, which consists of nodes, branches, and leaf nodes representing classification rules. These rules are typically expressed in the form of if–then statements, allowing clear interpretation of decision logic.

g) Classification of New Data

In the final stage, the constructed decision tree is used to classify unseen data. Each new instance is traversed through the tree starting from the root node and following the appropriate branches until a leaf node is reached, where the predicted class label is obtained.

4.2.1. Dataset Used

The dataset used in this study consists of eight data instances, as presented in Table 1. Each instance is characterized by six attributes, namely Flow Duration, Total Forward Packets, Total Backward Packets, Destination Port, Protocol, and Label. The Label attribute represents the classification of network traffic, which includes one normal class and several types of attack classes, such as DoS, FTP-Patator, Brute-Force, DNS, SSH-Patator, and Web Attack.

Table 1. Dataset

Flow Duration	Total Fwd Packets	Total Bwd Packets	Destination Port	Protocol	Label
123456	10	8	80	TCP	Normal
987654	15	5	443	TCP	DoS
543210	12	6	21	TCP	FTP-Pat
234567	9	9	80	TCP	Normal
345678	8	12	443	UDP	Brute-Force
123789	16	4	53	UDP	DNS
765432	18	2	22	TCP	SSH-Pat
678901	20	10	8080	TCP	Web-Att

At the initial stage, all data instances are placed at the root node (Node 1). Each attribute is then examined to determine whether it is numerical or nominal. Flow Duration, Total Forward Packets, and Total Backward Packets are categorized as numerical attributes, while Protocol and Destination Port are considered nominal attributes.

For numerical attributes, entropy and information gain are computed using a threshold-based splitting approach. In contrast, nominal attributes are evaluated based on their categorical values. This distinction is essential to ensure appropriate splitting strategies for different types of attributes.

a) Entropy Calculation at the Root Node

Entropy for Label :

Total number of data point = 8

Label distribution :

Normal = 2

DoS = 1

FTP-Patator = 1

Brute-Force = 1

DNS = 1

SSH-Patator = 1

Web Attack = 1

This distribution forms the basis for calculating the entropy at the root node.

Entropy is calculated using Eq. (1):

$$H(S) = -\sum_{i=1}^{K} p_i log_2(p_i) \quad (1)$$

Where p_i denotes the probability of occurrence of class i, and k represents the total number of classes. Based on the class distribution, the entropy of the root node is computed using Eq. (2):

$$H(S) = -\left(\frac{2}{8}log_2\frac{2}{8} + \frac{1}{8}log_2\frac{1}{8} + \cdots + \frac{1}{8}log_2\frac{1}{8}\right) \quad (2)$$

Which results in an entropy value of **2.75**, indicating a relatively high degree of class impurity in the dataset.

b) Entropy for Numerical Feature (Flow Duration)

For numerical attributes like Flow Duration, we split based on a threshold value v.

Suppose we choose v = 345000 as the threshold.

`Flow Duration ≤ 345000: 5 data points.`

Flow Duration > 345000: 3 data points.

Calculate the entropy for each subset and the total information gain for this feature.

After selecting the optimal splitting attribute, the dataset is partitioned into several branches. For each resulting branch, the same process of entropy and information gain calculation is performed recursively until all data instances in a node belong to the same class or no further meaningful splits can be achieved. This recursive procedure results in a structured decision tree model capable of effectively classifying network traffic and detecting intrusion attempts.

5. Conclusion

In this study, we successfully applied the C4.5 decision tree algorithm for attack detection in computer networks. By utilizing a dataset that includes various types of attacks such as DoS, Brute Force, FTP-Pat, as well as normal network activities, we managed to build an effective and reliable attack detection model. The evaluation results indicate that the attack detection model developed using the C4.5 algorithm is able to classify network activities with satisfactory accuracy, as well as a good balance between precision and recall. This model has proven to be capable of recognizing attack patterns present in the dataset and effectively distinguishing them from normal network activities.

Moreover, we also successfully identified the most influential attributes in detecting attacks, such as protocol, flow duration, and the number of forward packets. These insights are crucial for network security practitioners in developing more effective and responsive detection strategies against existing threats. In conclusion, this research provides a significant contribution to the development of more advanced and reliable attack detection systems in the context of computer networks. By leveraging the power of the C4.5 decision tree algorithm, we hope this study can enhance the security of computer networks and help protect information infrastructure from increasingly complex cyber threats.

Author Contributions: W.W.W. and R.L. contributed to the conceptualization of the study. W.W.W. was responsible for methodology design, software development, investigation, and data curation. Validation and formal analysis were carried out by W.W.W. and R.L. The original draft of the manuscript was prepared by W.W.W., while review and editing were conducted by R.L. Visualization was completed by W.W.W. Supervision, project administration, and funding acquisition were managed by R.L. All authors have read and agreed to the published version of the manuscript.

Funding: Please add: This research received no external funding

Data Availability Statement: The data supporting the findings of this study are available from the corresponding author upon reasonable request. The data are not publicly available due to privacy and ethical considerations involving respondent confidentiality.

Acknowledgments: The authors would like to express their gratitude to the educators and students who participated in this study for their time and valuable responses. Appreciation is also extended to the institutional partners who supported the data collection process and the implementation of the EvaloExam application. The authors acknowledge the use of digital tools and AI-assisted writing support for language refinement and manuscript editing, while ensuring that all intellectual content, analysis, and interpretations remain the responsibility of the authors.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

Ahmed, M., Mahmood, A. N., & Hu, J. (2021). A survey of network anomaly detection techniques. *IEEE Access, 9*, 103598–103620. https://doi.org/10.1109/ACCESS.2021.3058004

Alazzam, H., Sharieh, A., & Sabri, K. E. (2021). A novel hybrid intrusion detection system based on decision tree and support vector machine. *Journal of Information Security and Applications, 58*, 102715. https://doi.org/10.1016/j.jisa.2020.102715

Arrieta, A. B., et al. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges. *Information Fusion, 58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Beny Abukhaer Tatara, B., Abdurachman, B., Mustofa, D. L., & Yacobus, D. (2023). The potential of cyber attacks in Indonesia's digital economy transformation. *NUANSA: Jurnal Penelitian Ilmu Sosial Dan Keagamaan Islam*, 20(1), 19–37. https://doi.org/10.19105/nuansa.v20i1.7362

Ferrag, M. A., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2022). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Future Generation Computer Systems, 128*, 260–281. https://doi.org/10.1016/j.future.2021.09.012

George, A. S., Baskar, T., & Srikaanth, P. B. (2024). Cyber threats to critical infrastructure: Assessing vulnerabilities across key sectors. *Partners Universal International Innovative Journal*, 2(1), 51–75. https://doi.org/10.5281/zenodo.10639463

Gupta, C., Johri, I., Srinivasan, K., Hu, Y., & Qaisar, S. M. (2022). A systematic review on machine learning and deep learning. *Progress in Biophysics and Molecular Biology*, June. https://doi.org/10.1016/j.pbiomolbio.2022.07.004

Joseph, S. (n.d.). Computers' effect on society: Exposing their manifold benefits.

Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J., & Alazab, A. (2020). Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine. *Electronics (Switzerland)*, 9(1). https://doi.org/10.3390/electronics9010173

Khan, M. A., Karim, M. M., & Kim, Y. (2023). A scalable and hybrid intrusion detection system based on deep learning. *IEEE Access, 11*, 24678–24692. https://doi.org/10.1109/ACCESS.2023.3245617

Lestari, A. (2020). Increasing accuracy of C4.5 algorithm using information gain ratio and AdaBoost for classification of chronic kidney disease. *Journal of Soft Computing Exploration*, 1(1), 32–38. https://doi.org/10.52465/joscex.v1i1.6

Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2021). Explainable AI: A review of machine learning interpretability methods. *Entropy, 23*(1), 18. https://doi.org/10.3390/e23010018

Mahbooba, B., Timilsina, M., Sahal, R., & Serrano, M. (2021). Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model. *Complexity*, 2021. https://doi.org/10.1155/2021/6634811

Nagar, U. (2021). A study on feature analysis and ensemble-based intrusion detection scheme using CICIDS-2017 dataset.

Ozkan-Okay, M., Yilmaz, A. A., Akin, E., Aslan, A., & Aktug, S. S. (2023). A comprehensive review of cyber security vulnerabilities. *Electronics*, 12(1333).

Pinto, A., Herrera, L.-C., Donoso, Y., & Gutierrez, J. A. (2023). Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure. *Sensors, 23*(5), 2415. https://doi.org/10.3390/s23052415

Sarker, I. H. (2023). Machine learning for intelligent data analysis and automation in cybersecurity: Current and future prospects. *Annals of Data Science*, 10(6), 1473–1498. https://doi.org/10.1007/s40745-022-00444-2

Tama, B. A., Nkenyereye, L., Islam, S. M. R., & Kwak, K. S. (2020). An enhanced anomaly detection in web traffic using a stack of classifier ensemble. *IEEE Access*, 8, 24120–24134. https://doi.org/10.1109/ACCESS.2020.2969428

Tariq, R., Casillas-Muñoz, F. A., Hassan, S. T., & Ramírez-Montoya, M. S. (2024). Synergy of Internet of Things and education: Cyber-physical systems contributing towards remote laboratories, improved learning, and school management. *Journal of Social Studies Education Research*, 15(2 Special issue), 305–352.

Wang, J. (2022). Application of C4.5 decision tree algorithm for evaluating the college music education. *Mobile Information Systems*, 2022. https://doi.org/10.1155/2022/7442352

Wu, Y., Wei, D., & Feng, J. (2020). Network attacks detection methods based on deep learning techniques: A survey. *Security and Communication Networks*, 2020. https://doi.org/10.1155/2020/8872923.