OPEN ACCESS
VOL. 6, NO. 2, PP.080-089, MAY 2024

ISSN : 2622-8106 (ONLINE)
DOI: 10.20895/INISTA.V6I2.1389

Journal of Informatics, Information System, Software Engineering and Applications (INISTA)

Analysis of NSL-KDD for the Implementation of Machine Learning in Network Intrusion Detection System

Yuliana [1], Dhoni Hanif Supriyadi [2], Mohammad Reza Fahlevi [*3], Muhamad Rifki Arisagas [4]

[1, *3]Program Studi Teknik Informatika, Universitas Nahdlatul Ulama Indonesia, Jakarta
Jl. Taman Amir Hamzah No.5, Pegangsaan, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10320, Indonesia
[1]yuliana111099@gmail.com
[3]rezafah@unusia.ac.id

[2]Program Studi Teknik Informatika, Universitas Bina Sarana Informatika, Bekasi
Jl. Raya Jatiwaringin No.18, Jaticempaka, Kec. Pd. Gede, Kota Bekasi, Jawa Barat 17411, Indonesia
[2]dhonihanif354@gmail.com

[4]Program Studi Teknik Informatika, Universitas Suryakancana, Cianjur
Jl. Pasirgede Raya, Bojongherang, Kec. Cianjur, Kab. Cianjur, Jawa Barat 43216, Indonesia
[4]arisagasr@gmail.com

Received on 11-11-2023, revised on 22-12-2023, accepted on 25-01-2024

Abstract

In the world of network data communication, anomaly detection is a crucial element in identifying abnormal behavior among the flowing data packets. Research in the field of intrusion detection often focuses on the search and analysis of anomalous patterns and the misuse of communication data. The research methodology in this study adopts CRISP-DM (Cross-Industry Standard Process for Data Mining) as the framework. The primary goal of this research is to conduct a comparative analysis of classification techniques to identify normal and anomaly records within network data. For this purpose, a publicly available standard dataset, NSL-KDD, is used. The NSL-KDD dataset consists of 41 attributes with relevance, and the 42nd attribute is used to identify normal class and four attack classes. The results of the analysis using the NSL-KDD dataset, applying the CRISP-DM methodology and machine learning techniques in the Network Intrusion Detection System, reveal that the Decision Tree model has the highest accuracy, achieving 100% on the training data and 80% on the testing data. These findings are compared with the results of using other models such as Random Forest, Logistic Regression, and K-Nearest Neighbor. This discovery has significant implications for enhancing NIDS's ability to recognize network threats and improve network system security.

Keywords: Anomaly Detection, Network Intrusion Detection System (NIDS), CRISP-DM, Machine Learning

Corresponding Author:
*Mohammad Reza Fahlevi
Program Studi Teknik Informatika, Universitas Nahdlatul Ulama Indonesia
Jl. Taman Amir Hamzah No.5, Pegangsaan, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta, Indonesia
Email: rezafah@unusia.ac.id

I.INTRODUCTION

THE field of network security has become increasingly crucial in the rapidly evolving era of information technology. Network Intrusion Detection Systems (NIDS) play a key role in maintaining the security of computer networks by identifying and mitigating potential threats. It then alerts users or invokes appropriate security measures. Cybersecurity researchers have proposed and used many NIDS systems in the past couple of decades [1]. The two main types of NIDSs that exist are anomaly detection and misuse detection. With the growing complexity and diversity of cyber threats, the integration of machine learning techniques has emerged as a promising approach to enhance the effectiveness of NIDS.

In the digital era, the security of network communication has become highly important. Network Intrusion Detection Systems (NIDS) are designed to monitor network traffic by analyzing data packets to detect signs of unusual behavior or malicious activities [2]. As a result, NIDS are employed to safeguard networks from a wide spectrum of security threats, including but not limited to, unauthorized access attempts, malware intrusions, and denial-of-service attacks [3]. As cyberattacks grow more intelligent, it is becoming increasingly challenging to find advanced cyberattacks in many industries, including industry, national defense, and healthcare. Traditional rule-based approaches, though effective to a certain extent, are increasingly struggling to keep up with the constant evolution of these threats [4]. Machine learning offers the potential to identify subtle patterns and new attack strategies that might be overlooked by traditional methods [5].

While the adoption of machine learning in NIDS is promising, it comes with its own set of challenges. Choosing the most suitable machine learning model, understanding the problem effectively, and ensuring methodological alignment with the NIDS context are crucial issues to address. Additionally, a clear understanding of the problem domain, namely the identification of normal network behavior and anomalies, is essential [6]. This study aims to address these challenges and determine the most effective machine learning model for NIDS based on in-depth analysis of the NSL-KDD dataset [7].

The proposed solution involves a comprehensive analysis of various machine learning models using the NSL-KDD dataset. By applying the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, this research aims to compare and evaluate the performance of these models in identifying normal and anomalous network behavior. The goal is to determine the machine learning model that provides the highest accuracy and effectiveness in the context of NIDS, ultimately contributing to the enhancement of network security in the ever-changing cyber landscape.

II.RESEARCH METHOD

This research adopts an empirical experimental approach to evaluate and compare the performance of various machine learning models in the context of Network Intrusion Detection Systems (NIDS). The research design is structured as follows:

A. Data Collection

The NSL-KDD dataset, widely recognized as a reference dataset in the field of network intrusion detection, has been selected for this research. This dataset comprises diverse features related to network traffic and includes both instances of normal and anomalous data. The dataset is divided into training and testing subsets to facilitate the training and evaluation of models.

B. Model Selection

Several leading machine learning models, including Decision Tree, Random Forest, Logistic Regression, and K-Nearest Neighbor, were chosen as candidates for evaluation. Each model was implemented and fine-tuned to optimize its performance within the NIDS framework.

C. Experiment Setup

Experiments were conducted under controlled conditions, ensuring consistency in input data and parameters. Model evaluation metrics, including accuracy, precision, recall, and F1 score, were utilized to assess the performance of these models.

D. Data Analysis

Comprehensive data analysis was carried out to compare the results obtained from each model. This analysis involved a detailed examination of false positives and false negatives, which are crucial in the context of NIDS.

E. Results and Interpretation

Research findings were reported, and the most effective model was identified based on its performance in accurately classifying network data as normal or anomalous. Furthermore, implications and significance of the results were discussed within the context of NIDS and network security.

F. Discussion

Research results were discussed, providing insights into the strengths and weaknesses of various machine learning models. Recommendations for further research and potential areas of improvement in NIDS were explored.

III. RESULTS AND DISCUSSION

A. NSL-KDD Analysis Method

The NSL-KDD analysis method applied in this research is CRISP-DM. It is can be seen in Fig. 1. CRISP-DM is a proven framework for addressing data processing issues in various research contexts [8]. The data mining process following the CRISP-DM guidelines consists of six main stages: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment [9]. However, in this study, the analysis is limited to the existing dataset, without creating a more optimal dataset through specific procedures or methods. Therefore, the research does not extend to the deployment stage.

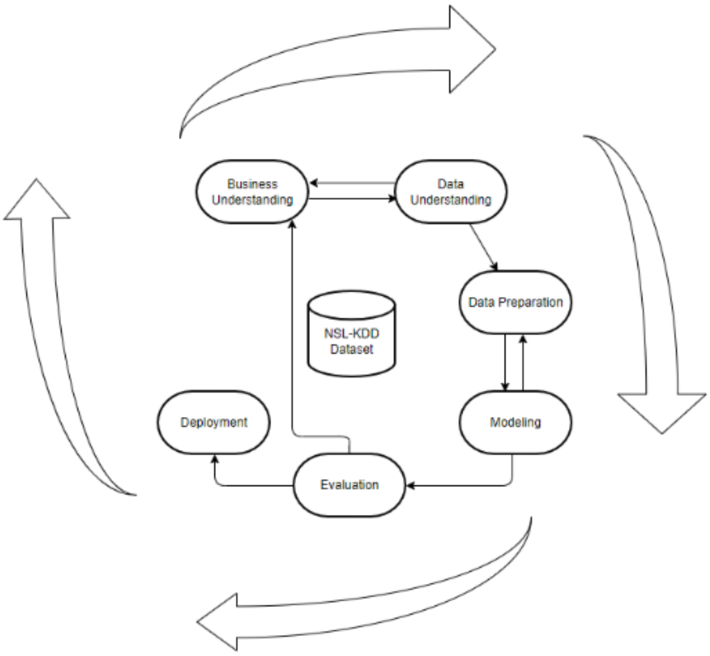


Fig. 1. CRISP-DM

The main objective and motivation of this research initiative are to conduct a comparative analysis of classification techniques for identifying normal and anomalous records in network-related data. Therefore, a standard dataset that is publicly available and accessible on NSL-KDD [10] is utilized, involving the comparison of four machine learning algorithms. The process of identifying normal and anomalous records includes the application of four classification algorithms to the processed NSL-KDD dataset. These algorithms are the Random Forest Classifier, Decision Tree Classifier, Logistic Regression, and K-Nearest Neighbor (KNN). All mentioned classifiers will be implemented on the dataset using a Python machine learning library [11].

1. Business Understanding

In the business understanding phase, the initial step is to define the research objectives, expected benefits, and relevant limitations. The main goal of this research is to investigate data classification methods applicable in analyzing NIDS log data. The study aims to comprehend and evaluate the performance of classification algorithms such as the Random Forest Classifier, Decision Tree Classifier, Logistic Regression, and K-Nearest Neighbor [12] in classifying NIDS log data. Additionally, the research aims to measure the accuracy levels of these models and identify various types of attacks present in NIDS log data.

a) The use of the NSL-KDD dataset as an initial step in analyzing the performance of algorithms in the NIDS.
b) The machine learning methods employed include the Random Forest Classifier, Decision Tree Classifier, Logistic Regression, and KNN algorithms.
c) The evaluation measurement for comparing data mining algorithms involves crucial metrics such as precision, recall, F1-score, and support.
d) In this study, the analysis is limited to the existing dataset without creating a more optimal dataset through specific procedures or methods, and it does not extend to the deployment stage.

2. Data Understanding

This dataset consists of a number of features extracted from data communication recordings [13]. NSL-KDD originates from a selected set of data from the KDD CUP 99 dataset, and its usage has several advantages:

a) There is no excessive addition of records in the training set, avoiding bias in classification results by the classifier.
b) The test data does not contain duplicates, resulting in a better reduction rate.
c) The number of records taken from various levels of groups is inversely proportional to the representation of records in the original KDD dataset.

This dataset consists of 41 available attributes, with the 42nd attribute describing the normal class and four different attack classes. There are two types of data in the dataset: training data, used to train machine learning algorithms, and testing data, used to evaluate the accuracy of the trained algorithm experiments [14]. Several successfully categorized attributes in the dataset can be found in Table I.

TABLE I. NSL-KDD ATTRIBUTES

No	Attribute Name	Data Type
1	*Duration*	*continuous*
2	*Protocol_type*	*symbolic*
3	*service*	*symbolic*
4	*flag*	*symbolic*
5	*src_bytes*	*continuous*
6	*dst_bytes*	*continuous*
7	*land*	*continuous*
8	*wrong_fragment*	*continuous*
9	*urgent*	*continuous*
10	*hot*	*continuous*
11	*num_failed_logins*	*continuous*
12	*logged_in*	*symbolic*
13	*num_compromised*	*continuous*
14	*root_shell*	*continuous*
15	*su_attempted*	*continuous*
16	*num_root*	*continuous*
17	*num_file_creations*	*continuous*
18	*num_shells*	*continuous*
19	*num_access_files*	*continuous*

No	Attribute Name	Data Type
20	*num_outbound_cmds*	*continuous*
21	*is_host_login*	*continuous*
22	*is_guest_login*	*continuous*
23	*count*	*continuous*
24	*srv_count*	*continuous*
25	*serror_rate*	*continuous*
26	*srv_serror_rate*	*continuous*
27	*rerror_rate*	*continuous*
28	*srv_rerror_rate*	*continuous*
29	*same_srv_rate*	*continuous*
30	*diff_srv_rate*	*continuous*
31	*srv_diff_host_rate*	*continuous*
32	*dst_host_count*	*continuous*
33	*dst_host_srv_count*	*continuous*
34	*dst_host_same_srv_rate*	*continuous*
35	*dst_host_diff_srv_rate*	*continuous*
36	*dst_host_same_src_port_rate*	*continuous*
37	*dst_host_srv_diff_host_rate*	*continuous*
38	*dst_host_serror_rate*	*continuous*
39	*dst_host_srv_serror_rate*	*continuous*
40	*dst_host_rerror_rate*	*continuous*
41	*dst_host_srv_rerror_rate*	*continuous*
42	*class*	*symbolic*

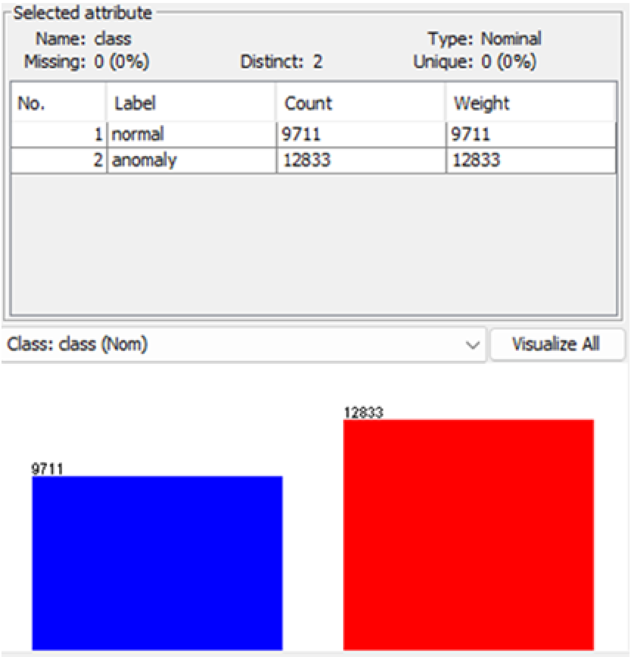
3. *Data Preparation*

Training and testing data from NSL-KDD, utilized in this study, are categorized into five attack categories, encompassing network traffic data gathered from various points in the network or relevant data sources [2]. These are displayed in the Table II

TABLE II. DATASET DESCRIPTION

Class Type	*Instances in KDDTrain*	*Instances in KDDTest*
Normal	67343	9711
Dos	45927	745
Probe	11656	2421
U2R	52	200
R2L	995	2754
Total	125973	22544

From the results of the research, labels were obtained in the form of classifications that categorize whether a data traffic contains anomalies or is normal, and this was done using Weka. See in Fig. 2.



Selected attribute

Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	normal	9711	9711
2	anomaly	12833	12833

Class: class (Nom)

Fig. 2. KDDTest and Weka

4. *Modeling*

In creating the model, it is necessary to first divide the data into training and testing or validation data [15]. This dataset has been tested with several models including K-Nearest Neighbors, Logistic Regression, Decision Tree, and Random Forest. For the K-Nearest Neighbors model, it was tested with various nearest neighbors, and the evaluation of the model in Fig. 3.

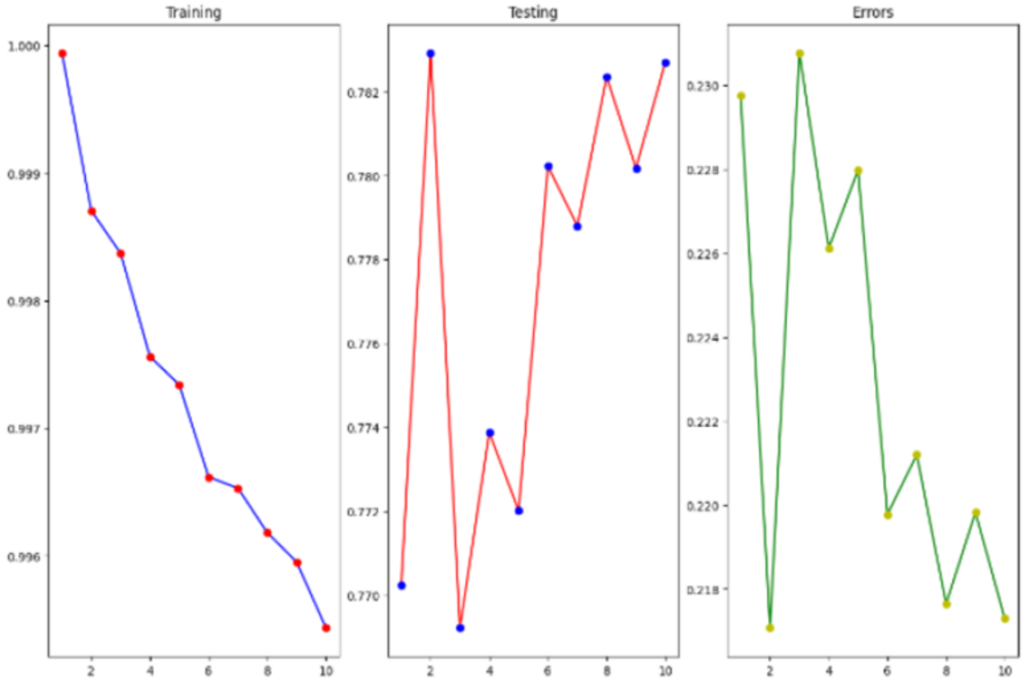


Fig. 3. Evaluation of the K-Nearest Neighbor Model

It can be concluded that increasing the number of nearest neighbors will lead to a decrease in accuracy in training, while accuracy in testing will increase. Knowing this, testing multiple parameters using cross-validation was conducted to find the best parameters suitable for building this model, which, in this case, involved using 10 nearest neighbors.

Furthermore, other models such as Logistic Regression, Decision Tree, and Random Forest were also tested. Cross-validation was applied to these models as well to find the best parameters suitable for building the model. In creating the Logistic Regression model, the coefficients or relationships between independent variables and the dependent variable were obtained in Fig. 4.

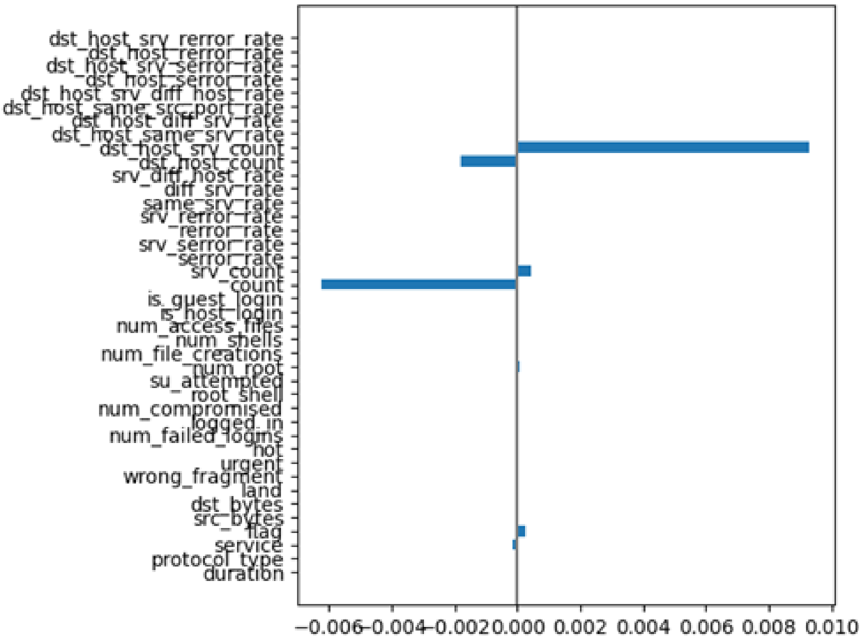


Fig. 4. Coefficients of the Logistic Regression Model

It can be said that there are many features that do not have a significant relationship with the dependent variable, so there are plans to try removing some of these features. Next, when testing the Decision Tree model, some features considered important by the model were obtained in Fig. 5.

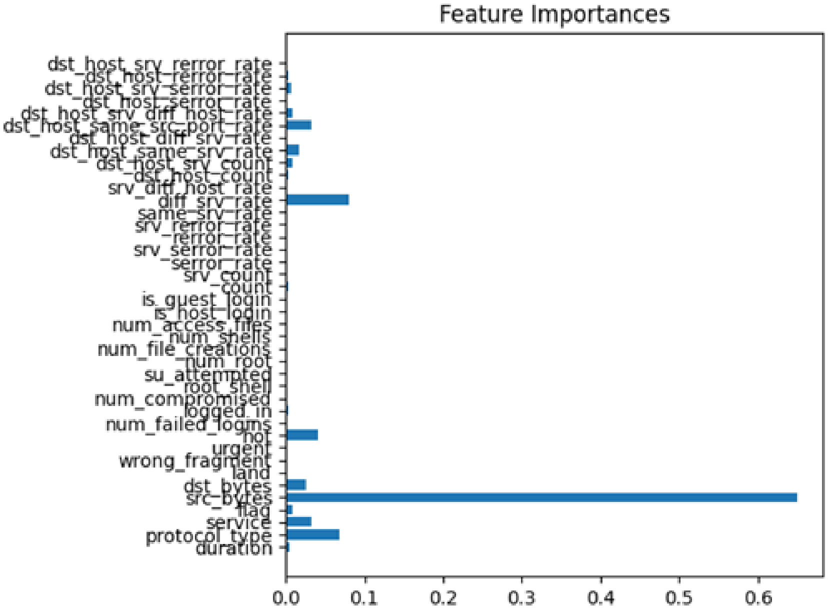


Fig. 5. Features of the Decision Tree Model

Here, it is also mentioned that some features are not considered important in model building. Knowing this, the next step is to test the Random Forest model to see some features considered important. Several features considered important by the model were obtained in Fig. 6.

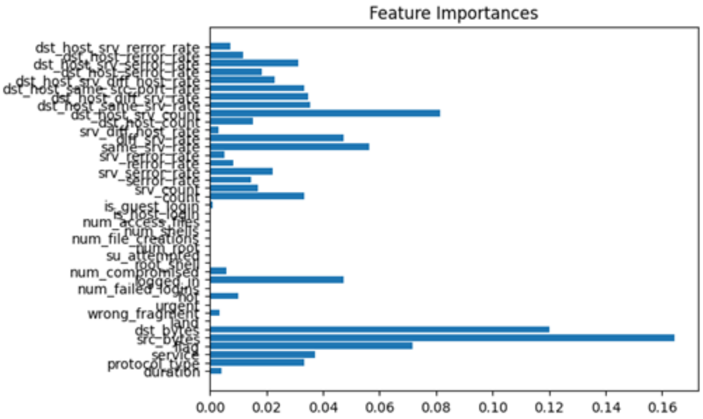


Fig. 6. Features of the Random Forest Model

After examining the above graph, it can be concluded that there are some features that are still considered important by the model, and there are some features that are truly not considered important by the model.

5. *Evaluation*

From the creation of the tested models with this dataset, several pieces of information related to each model were obtained. Next, evaluating which model is most suitable for this dataset by comparing the accuracy among models is crucial. See the Fig. 7.

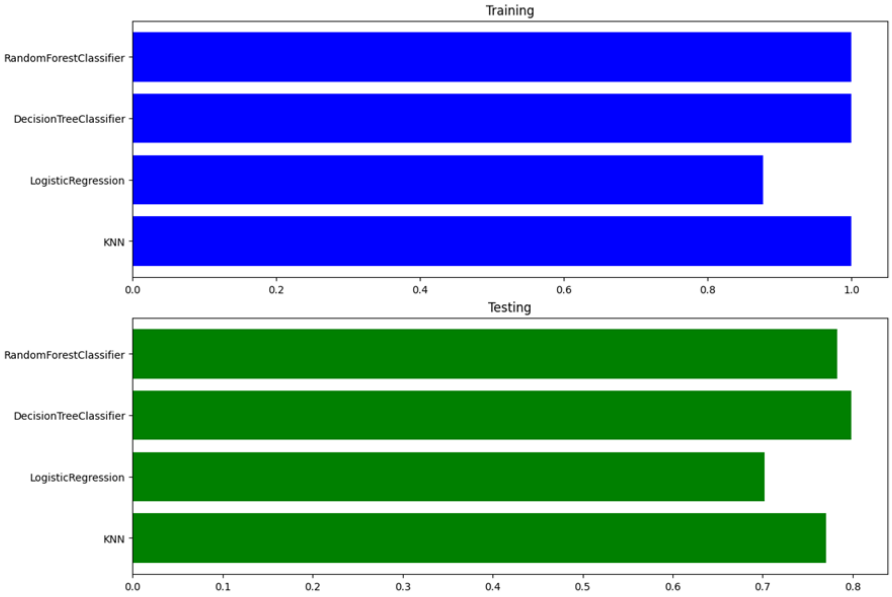
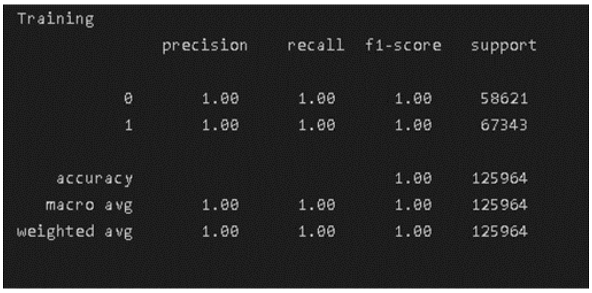


Fig. 7. Evaluation of Machine Learning Models

It can be concluded that the most suitable model is the Decision Tree, with an accuracy of 100% on the training data and 80% on the testing data. The evaluation results are shown in the Fig. 8 for Training Data and Fig. 9 for Testing Data.

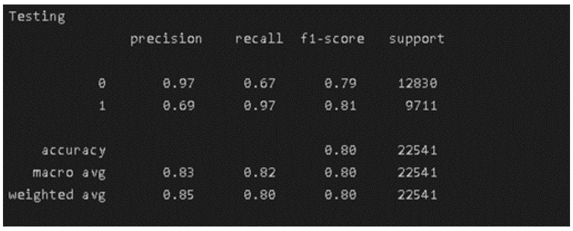


```
Training
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     58621
           1       1.00      1.00      1.00     67343

    accuracy                           1.00    125964
   macro avg       1.00      1.00      1.00    125964
weighted avg       1.00      1.00      1.00    125964
```

Fig. 8. Evaluation of the Decision Tree Model (Training Data)



```
Testing
              precision    recall  f1-score   support

           0       0.97      0.67      0.79     12830
           1       0.69      0.97      0.81      9711

    accuracy                           0.80     22541
   macro avg       0.83      0.82      0.80     22541
weighted avg       0.85      0.80      0.80     22541
```

Fig. 9. Evaluation of the Decision Tree Model (Testing Data)

The data mining process following the CRISP-DM guidelines consists of six main stages: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment [9]. However, in this study, the analysis was limited to the existing dataset, avoiding additional steps or methods to create a more optimal dataset, so the deployment stage was not pursued.

In the Business Understanding phase, the research objectives, benefits, and limitations were determined. Subsequently, in the Data Understanding phase, the dataset contains various features extracted from communication data. NSL-KDD is a data collection consisting of selected records from the KDD CUP 99 dataset. The dataset includes 41 diverse attributes, with the 42nd attribute indicating the "normal" category and four attack categories. To improve data correlation, one feature containing only the same data was removed.

Next, in the Data Preparation phase, training and testing data from NSL-KDD were categorized into five attack categories, as explained in the previous research [2]. From the research results, labels were obtained as classifications categorizing whether data traffic contains anomalies or is normal. With more anomalous data than normal data, it can be said that this data is imbalanced.

Then, in the model-building phase, we tested this dataset with four machine learning models: K-Nearest Neighbors, Logistic Regression, Decision Tree, and Random Forest. We also selected the best parameters using cross-validation. After testing several models, evaluating these models in the model evaluation phase, we found that the best model for this dataset is Decision Tree with an accuracy of 100% on the training data and 80% on the testing data.

IV.CONCLUSION

Overall, our analysis of the NSL-KDD dataset, using the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology and applying machine learning to the Network Intrusion Detection System (NIDS), has provided valuable insights. These findings indicate that the Decision Tree model, with an outstanding accuracy of 100% on the training data and 80% on the testing data, emerges as the most suitable choice among the tested models, outperforming alternatives such as Random Forest, Logistic Regression, and K-Nearest Neighbor.

These results underscore the potential of machine learning, particularly the Decision Tree algorithm, in enhancing the effectiveness of NIDS in identifying network anomalies and intrusions. It is evident that further research and in-depth analysis are crucial to improving accuracy and reducing error rates. Subsequent studies can explore additional refinements and advanced techniques to strengthen the security infrastructure of network systems, thereby mitigating the evolving threats in the ever-dynamic cyber landscape.

ACKNOWLEDGMENT

The authors would like to express deep gratitude to the institutions and individuals who have supported and contributed to the completion of this research. Sincere thanks are extended to our supervisors and mentors for their guidance and valuable insights throughout the research journey. We also acknowledge the research community that shared the NSL-KDD dataset, which served as the foundation for our study. Not to be forgotten, we are immensely grateful to our colleagues and friends for their encouragement and support. Without their support, this research would not have come to fruition.

REFERENCES

[1] M. Al Lail, A. Garcia, and S. Olivo, “Machine Learning for Network Intrusion Detection—A Comparative Study,” Future Internet, vol. 15, no. 7, Jul. 2023, doi: 10.3390/fi15070243.

[2] R. Rama Devi and M. Abualkibash, “Intrusion Detection System Classification Using Different Machine Learning Algorithms on KDD-99 and NSL-KDD Datasets - A Review Paper,” *International Journal of*

Computer Science and Information Technology, vol. 11, no. 03, pp. 65–80, Jun. 2019, doi: 10.5121/ijcsit.2019.11306.

[3] O. Kayode-Ajala, “Anomaly Detection in Network Intrusion Detection Systems Using Machine Learning and Dimensionality Reduction,” *SSRAML SageScience*, vol. 4, no. 1, pp. 12–26, Apr. 2021.

[4] M. Esmaeili, S. H. Goki, B. H. K. Masjidi, M. Sameh, H. Gharagozlou, and A. S. Mohammed, “ML-DDoSnet: IoT Intrusion Detection Based on Denial-of-Service Attacks Using Machine Learning Methods and NSL-KDD,” *Wirel Commun Mob Comput*, vol. 2022, 2022, doi: 10.1155/2022/8481452.

[5] P. Maniriho, L. J. Mahoro, E. Niyigaba, Z. Bizimana, and T. Ahmad, “Detecting intrusions in computer network traffic with machine learning approaches,” *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 3, pp. 433–445, 2020, doi: 10.22266/IJIES2020.0630.39.

[6] A. Devarakonda, N. Sharma, P. Saha, and S. Ramya, “Network intrusion detection: A comparative study of four classifiers using the NSL-KDD and KDD’99 datasets,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jan. 2022. doi: 10.1088/1742-6596/2161/1/012043.

[7] F. Masoodi, A. M. Bamhdi, and T. A. Teli, “Machine Learning for Classification analysis of Intrusion Detection on NSL-KDD Dataset,” *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 10, pp. 2286–2293, 2021.

[8] F. Ariadi, “Analisa Perbandingan Algoritma DT C.45 dan Naïve Bayes Dalam Prediksi Penerimaan Kredit Motor Article History ABSTRAK,” *Jurnal Riset Inovasi Bidang Informatika dan Pendidikan Informatika (KERNEL)*, vol. 1, no. 1, Jun. 2020.

[9] S. Huber, H. Wiemer, D. Schneider, and S. Ihlenfeldt, “DMME: Data mining methodology for engineering applications - A holistic extension to the CRISP-DM model,” in *Procedia CIRP*, Elsevier B.V., 2019, pp. 403–408. doi: 10.1016/j.procir.2019.02.106.

[10] “NSL KDD-Dataset.” Accessed: Jan. 20, 2023. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html

[11] U. Ahmad, S. Naseer, and H. Asim, “Analysis of Classification Techniques for Intrusion Detection,” in *International Conference on Innovative Computing (ICIC)*, Lahore, 2019.

[12] R. N. Wibowo, P. Sukarno, and E. M. Jadied, “Pendeteksian Serangan DoS Menggunakan Multiclassfier Pada NSL-KDD Dataset,” *e-Proceeding of Engineering*, vol. 5, no. 3, pp. 7885–7893, Dec. 2018.

[13] “KDD CUP 1999 Data.” Accessed: Jan. 19, 2023. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[14] A. Agung Riyadi, F. Amsury, I. Saputra, and T. Pattiasina, “Comparative Analysis of The K-Nearest Neighbor Algorithm on Various Intrusion Detection Datasets,” *JURNAL RISET INFORMATIKA*, vol. 4, no. 1, Dec. 2021, doi: https://doi.org/10.34288/jri.v4i1.341.

[15] R. A. R. Mahmood, A. H. Abdi, and M. Hussin, “Performance evaluation of intrusion detection system using selected features and machine learning classifiers,” *Baghdad Science Journal*, vol. 18, pp. 884–898, Jun. 2021, doi: 10.21123/bsj.2021.18.2(Suppl.).0884.