UMYU Scientifica, Vol. 3 NO. 3, September 2024, Pp 277 – 283
https://doi.org/10.56919/usci.2433.030
ISSN: 2955 – 1145 (print); 2955 – 1153 (online)
A periodical of the Faculty of Natural and Applied Sciences, UMYU, Katsina

ORIGINAL RESEARCH ARTICLE

Development and Evaluation of a Hybrid Machine Learning-Based Intrusion Detection System Using NSL-KDD Dataset

Mercy Esther Oluwagbenga*, Taiwo Kolajo and Joshua Babatunde Agbogun
Department of Computer Science, Faculty of Science, Federal University Lokoja, Kogi State, Nigeria

ABSTRACT
The growth of IoT devices has resulted in a rise of attack surfaces like firmware, sensitive data in physical interfaces, and default settings. Intrusion Detection Systems (IDSs) in networks are used to alert network administrators to critical issues. Zero-day attack detection is an important topic of research in the field of malicious traffic identification. Current methods rely on Machine Learning (ML) approaches for intrusion detection systems (IDSs); nevertheless, the efficacy of the methodology mechanism is dependent on the feature learning procedure, which is still an unresolved problem. As a consequence, in this article, a Hybrid-Base IDS was implemented, with various metrics used to optimise the network settings. In the simulation, the Network Security Laboratory – Knowledge Discovery Dataset (NSL-KDD) benchmark dataset was employed, as well as measures including accuracy, recall, False Positive Rate (FPR), True Positive Rate (TRP), and other aligned metrics. The paper presents a hybrid intrusion detection system (IDS) combining anomaly detection and signature-based techniques, using machine learning models such as Random Forest and Support Vector Machines. Our model achieved an accuracy of 99.73% and a false positive rate of 0.065, outperforming existing methods of Hadeel et al. (2024) and demonstrating its potential for real-world application. We also ran a comparison study with other current approaches, and the findings show that the suggested IDS scheme is effective in real-world cybersecurity scenarios. For future investigations, it is proposed that the Ensemble approach and real-time implementation be used, which will allow the model to continue operating in real-time circumstances.

ARTICLE HISTORY
Received May 20, 2024
Accepted September 06, 2024
Published September 16, 2024

KEYWORDS
Feature selection, cybersecurity, deep learning, machine learning, intrusion detection system /intrusion protection system, optimization and anomaly detection

INTRODUCTION
An intrusion is a malicious act that aims to compromise the availability, confidentiality, or integrity of any resource or component within the network in order to undermine its security policy (Lirim & Cihan, 2021; Adnan et al., 2020).

In recent times, network security has gained attention due to the rising annual costs associated with data protection. Part of this can be attributed to malicious activity on the part of users who want to limit access to the networks, systems, and services that businesses of all sizes utilise. To address this problem, a range of defence strategies are implemented, including private virtual networks, firewalls, and widespread encryption, in an effort to maintain the security of infrastructures and the confidentiality of online communications. One regularly used technique is intrusion detection (IDS). This allows us to collect information from a variety of known attack types and use it to safeguard the whole infrastructure and identify potential attack sites. This facilitates not just reporting but also continuous growth of the security environment (Jing et al., 2022). One of the most popular techniques for intrusion detection is to utilise an IDS (Intrusion Detection System). This makes it possible to gather information from a variety of recognised attack vectors, which we can subsequently utilise to protect the entire system and locate possible points of attack. This makes it easier to report on security incidents and to keep the security environment evolving (Sung et al., 2022).

Duy et al. (2023) suggested a Deep Learning-based solution for detecting network anomalies. This suggested model multi-classifies normal and intrusive events using a recent dataset dubbed the Coburg Network Intrusion Detection dataset using a dense supervised neural network based on a convolution neural network (CNN). Our result shows that optimising hyperparameters improves performance. The proposed CNN model can detect assaults with an execution time of 12 seconds and an accuracy of 99.13%.

Correspondence: Mercy Esther Oluwagbenga. Department of Computer Science, Faculty of Science, Federal University Lokoja, Kogi State, Nigeria. ✉ esther.oluwagbenga@fulokoja.edu.ng. Phone Number: +234 806 959 5521
How to cite: Olugbenga, O. E. M., Kolajo, T., & Babatunde, J. A. (2024). Development and Evaluation of a Hybrid Machine Learning-Based Intrusion Detection System Using NSL-KDD Dataset. *UMYU Scientifica*, 3(3), 277 – 283. https://doi.org/10.56919/usci.2433.030

Ahmad et al. (2022) employed a range of individual and hybrid deep learning classifiers to assess a number of datasets, spanning the years and encompassing both IoT-specific and non-IoT ones. In order to minimise (1) issues related to the quality of the dataset and (2) any bias in the outcomes, the goal was to develop a benchmark that could evaluate numerous classification models across various datasets. Empirical data revealed exciting findings on certain classifiers that took hours to converge yet failed to detect threats. Conversely, some attained the highest levels of accuracy and other performance metrics while quickly reaching a condition of convergence.

Esra et al. (2024) also proposed an IDS defense mechanism to improve the security of IoT networks against DoS attacks using anomaly detection and machine learning (ML). Anomaly detection was used in the IDS proposed to continuously monitor network traffic for deviations from normal profiles.

Hadeel et al. (2024), in their studies, use the CICIoT2023 dataset, which is a two-step data clustering to improve the Intrusion detection system.

Despite numerous advances, existing IDS approaches struggle with high false positive rates and real-time detection. This study introduces a hybrid IDS utilizing anomaly-based detection and Signature-based detection employing rule-based and machine learning algorithms, respectively to enhance detection accuracy and efficiency.

METHODOLOGY

Architectural Design of the System

The hybrid-based IDS system architecture consists of three layers: response and reporting, analysis, and data collecting. The system gathers information from several sources, preprocesses it, and then applies anomaly- and signature-based analysis to it. Combining the findings from the two investigations yields a more thorough and precise evaluation of possible risks compared to the previous studies, which compare the performances of two algorithms. The system creates reports on its performance, sends out alarms, and offers suggestions on how to respond. A hybrid architecture intrusion detection system (IDS) consists of many components working together to provide a comprehensive security solution. Figure 1 gives a detailed explanation of the system architecture of a hybrid-based IDS.

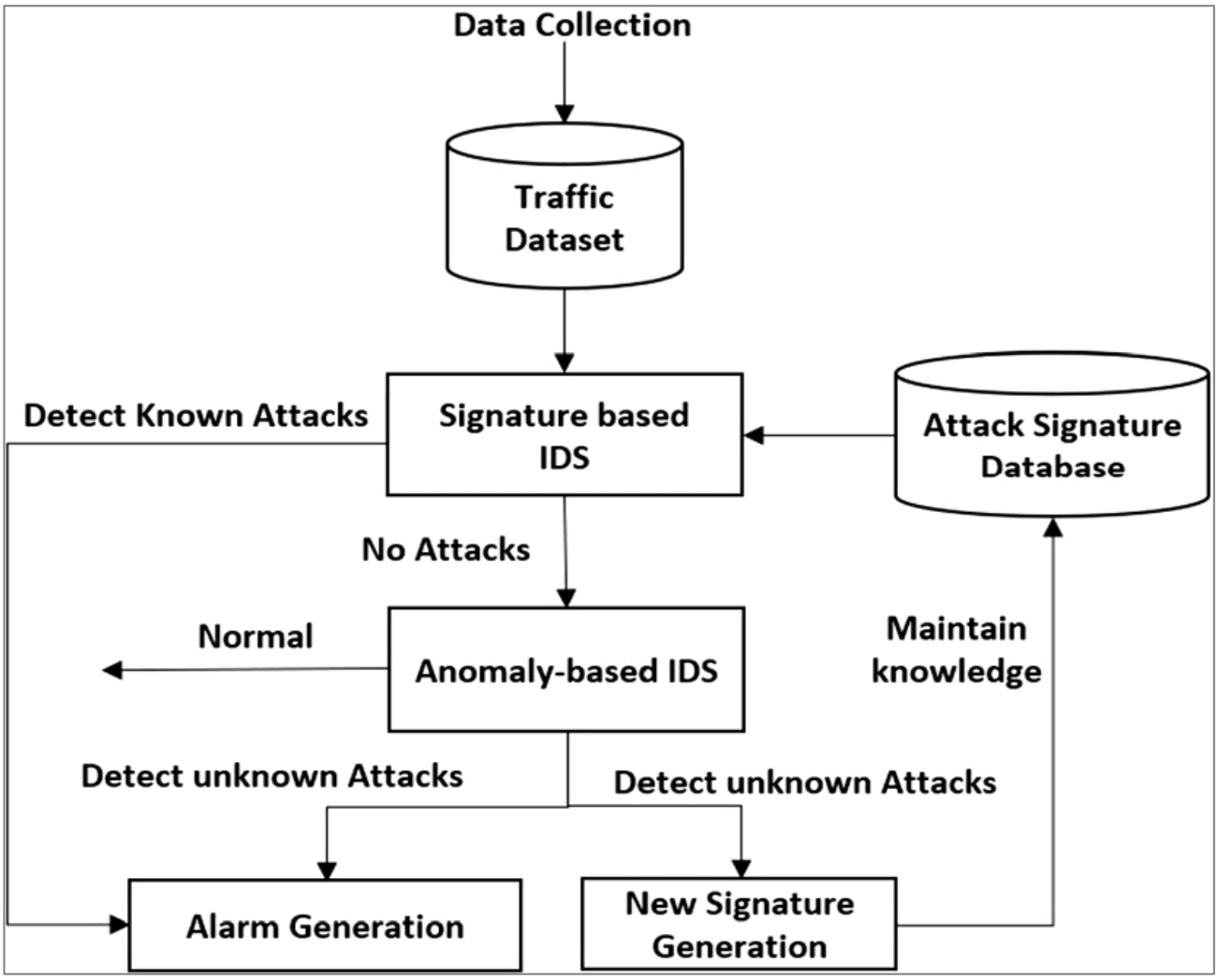

Figure 1: Architectural design of the proposed system

METHODOLOGY

The CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology was used for Intrusion Detection Systems (IDS) in the following way:

Business Understanding: The business objectives and requirements of the IDS were understood in order to determine the kind of attack that the intrusion detection system (IDS) will recognize.

Data Collection

The NSL-KDD dataset, which is a well-used benchmark dataset in the field of network intrusion detection, was acquired as part of the initial phase in the machine learning process. It was created as an upgrade on the original KDD Cup 1999 dataset, which had certain drawbacks but was nevertheless frequently used. Some of the problems of the KDD Cup 1999 dataset, such as duplicate records and an extremely unbalanced class distribution, are addressed in the NSL-KDD dataset. It offers a dataset that is more representative and balanced for assessing how well network intrusion detection technologies operate.

The CRISP-DM methodology stage: The NSL-KDD dataset classifies network connections into different attack categories, including:

- Denial of Service (DoS): Attacks designed to overload network capacity and interfere with regular operations.
- User to Root (U2R): An effort by unauthorised users to take over a system's root privileges.
- Remote to Local (R2L): Unauthorised users trying to connect remotely to a system and obtain local access.
- Probing: The act of an unauthorised user looking for information regarding possible security holes on a network.

A common benchmark for assessing the effectiveness of network intrusion detection systems is the NSL-KDD dataset.

Figure 2 illustrates the successful application of the Cross-Industry Standard Process for Data Mining (CRISP-DM) approach to the creation and implementation of intrusion detection systems (IDS). The following application of the CRISP-DM approach was made:

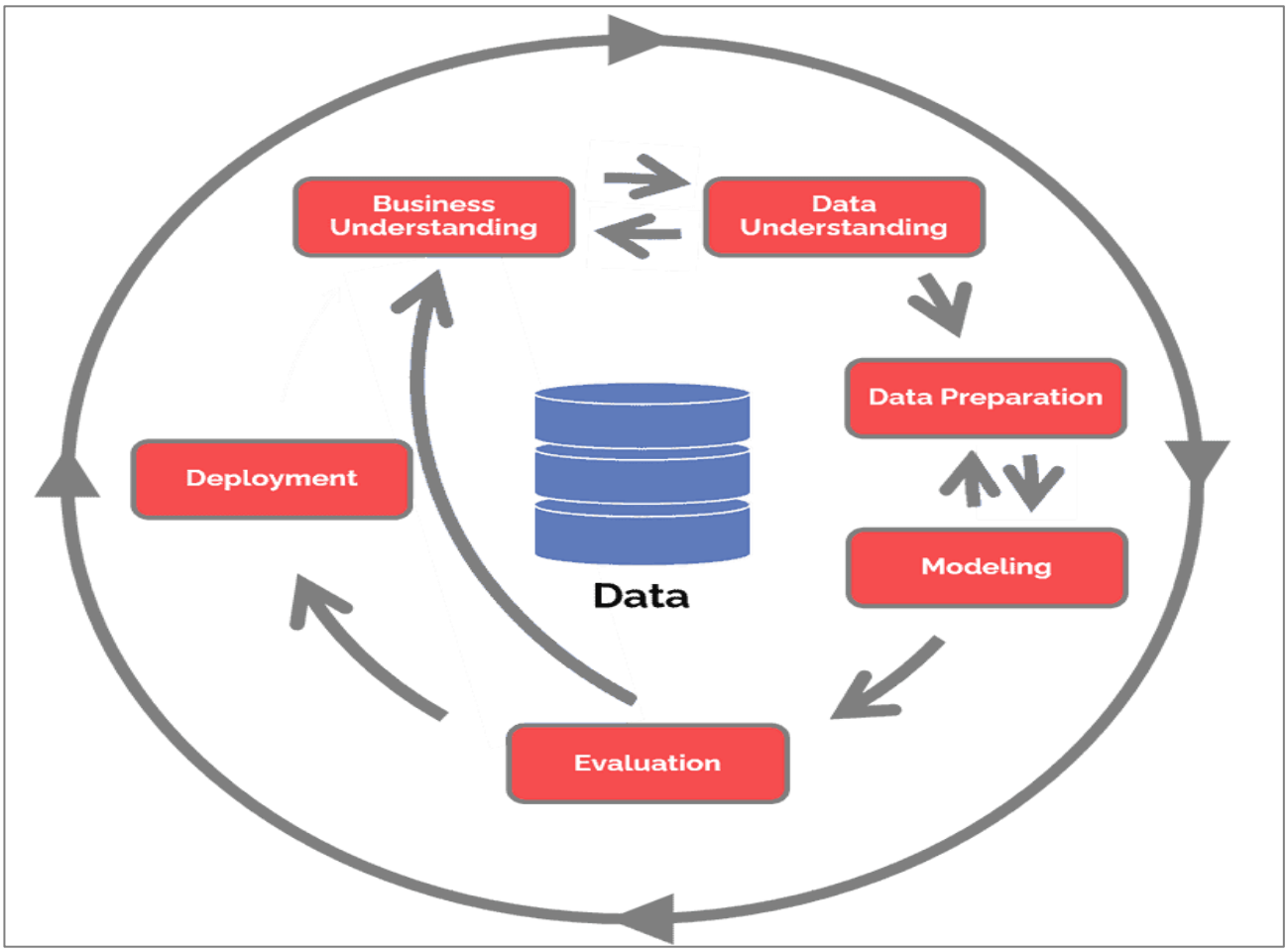

Figure 2: The CRISP-DM methodology stages

Data Preprocessing

The next phase in building a hybrid-based intrusion detection system (IDS) is data preprocessing, which is crucial. The primary objective of data preparation is to convert the raw data into a format that can be easily entered into the hybrid base IDS. These procedures are designed to clean, convert, and normalise data in order to maximise the effectiveness of the hybrid base IDS.

Choosing the model

The machine learning models were selected because they are highly useful for improving the accuracy and adaptability of intrusion detection systems. Also, we selected Random Forest due to its robustness in handling imbalanced data and its ability to provide feature importance metrics, which are crucial for optimizing our IDS compared to the previous studies which Decision Tree. They can also be used to detect unknown and known attacks, learn from past data, and adapt to new attack patterns.

Data Splitting: The data frame was shuffled and initially was split into training and testing, as shown in Table 1.

Table 1: Splitting of data into train and test

Datasets	Proportion (%)	Number of Samples
Train	80	100777
Test	20	25195

Training the model

At this point, a model that combines the anomaly and signature systems was created, utilising statistical analysis and machine learning techniques to find trends and anomalies suggestive of infiltration, but in the previous studies, anomaly detection systems were created to develop a defense mechanism to improve security.

Evaluating the Model

A range of criteria, including accuracy, precision, recall, and the receiver operating characteristic (ROC) curve, were taken into account to assess each model's performance. These measurements shed light on the model's overall performance and its capacity to accurately categorise network connections. The area under the curve (AUC) gives an overview of the model's performance, and the ROC curve allows one to visualise the trade-off between a true positive rate and a false positive rate.

Hyperparameters tuning and optimization

The model was regularly assessed using the proper metrics (accuracy, precision, recall, and F1-score) for hyperparameter adjustment and optimisation. This is essential to the IDS model's precise and reliable operation. Based on performance outcomes, hyperparameter optimisation was also used.

Predictions and Deployment

The algorithms detect suspicious activity or possible assaults in real time and use network traffic patterns to anticipate abnormalities. The detection model may be implemented for real-time intrusion detection in the production environment after it has been assessed and verified. The IDS will be closely watched after deployment to make sure it keeps functioning properly. This includes retraining the model with fresh data, updating it often, and adjusting to novel attack types.

RESULTS AND DISCUSSION

This section presents the outcomes of the model's implementation and assessment. Using the NSL-KDD dataset, a total of six (6) machine learning models were trained and evaluated. The decision tree, random forest, logistic regression, k-nearest neighbours, naïve bayes, and support vector machines are the machine learning models. Table 2 displays these models' respective performances.

Table 2: The performance of the machine learning models on NSL-KDD dataset

Model	Training Accuracy (%)	Test Accuracy (%)	Precision (Training) (%)	Precision (Test) (%)	Recall (Training) (%)	Recall (Test) (%)
LR	88.58	88.42	85.21	85.25	91.26	91.04
k-Nearest N	99.05	98.94	99.23	99.06	98.73	98.67
Naive Bayes	91.80	91.61	92.63	92.53	89.48	89.30
SVM	97.01	96.94	96.86	96.65	96.70	96.82
Tree	99.99	99.89	100.00	99.86	99.99	99.91
Random Forest	99.99	99.90	99.99	99.96	99.99	99.83

KEY: LR: Logistic Regression; k-Nearest N: k-Nearest Neighbors; SVM: Support Vector Machines

The accuracy, precision, recall, and area under the ROC curve of each model are highlighted in these findings. On both the training and test datasets, the models display encouraging accuracy levels, demonstrating their potency in identifying abnormalities and categorising network connections.

The Random Forest model performed better than the other models, obtaining the best accuracy, precision, recall, and AUC, according to the data in this work. However, in the previous studies of Hadeel et al. (2024), other metrics, such as training and testing times, showed that the Decision Tree was superior. This demonstrates how well it can properly categorise network connections as harmful or legitimate. The models' success may be ascribed to the proper algorithm selection, efficient preprocessing methods, and PCA feature engineering. The models benefited from the thorough examination and performance comparison made possible by the rigorous evaluation methodology. Given that the random forest yields the greatest results, we generated a confusion matrix to illustrate the random forest's performance, as seen in Figure 3.

One well-liked ensemble learning method that is well-known for its effectiveness and interpretability is the random forest. During training, many decision trees are constructed, and the average of each tree's predictions is the final prediction. Figure 4 displays the quantifiable attributes or feature relevance of the NSL-KDD dataset using random forest.

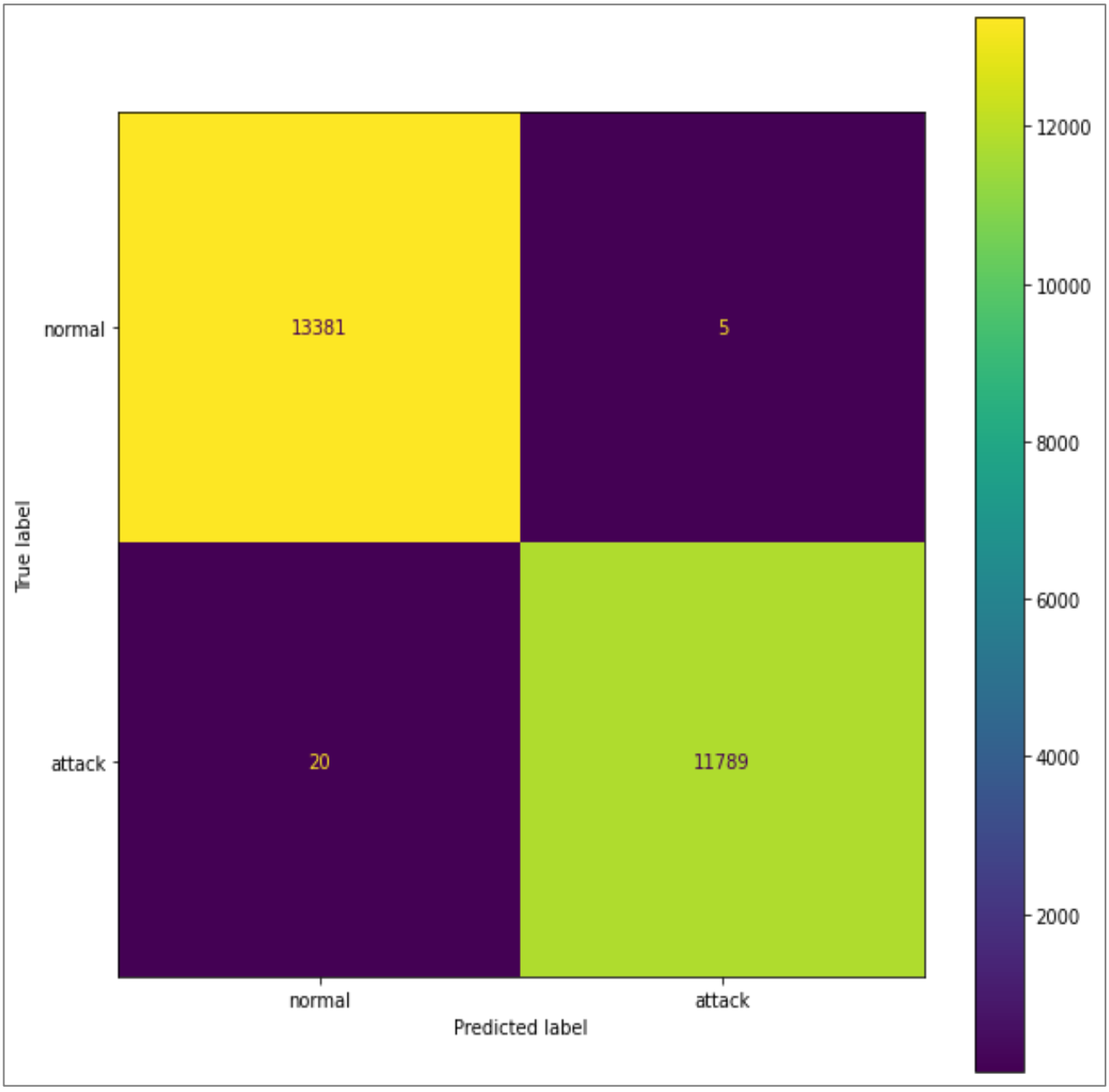

Figure 3: Confusion matrix: Random Forest

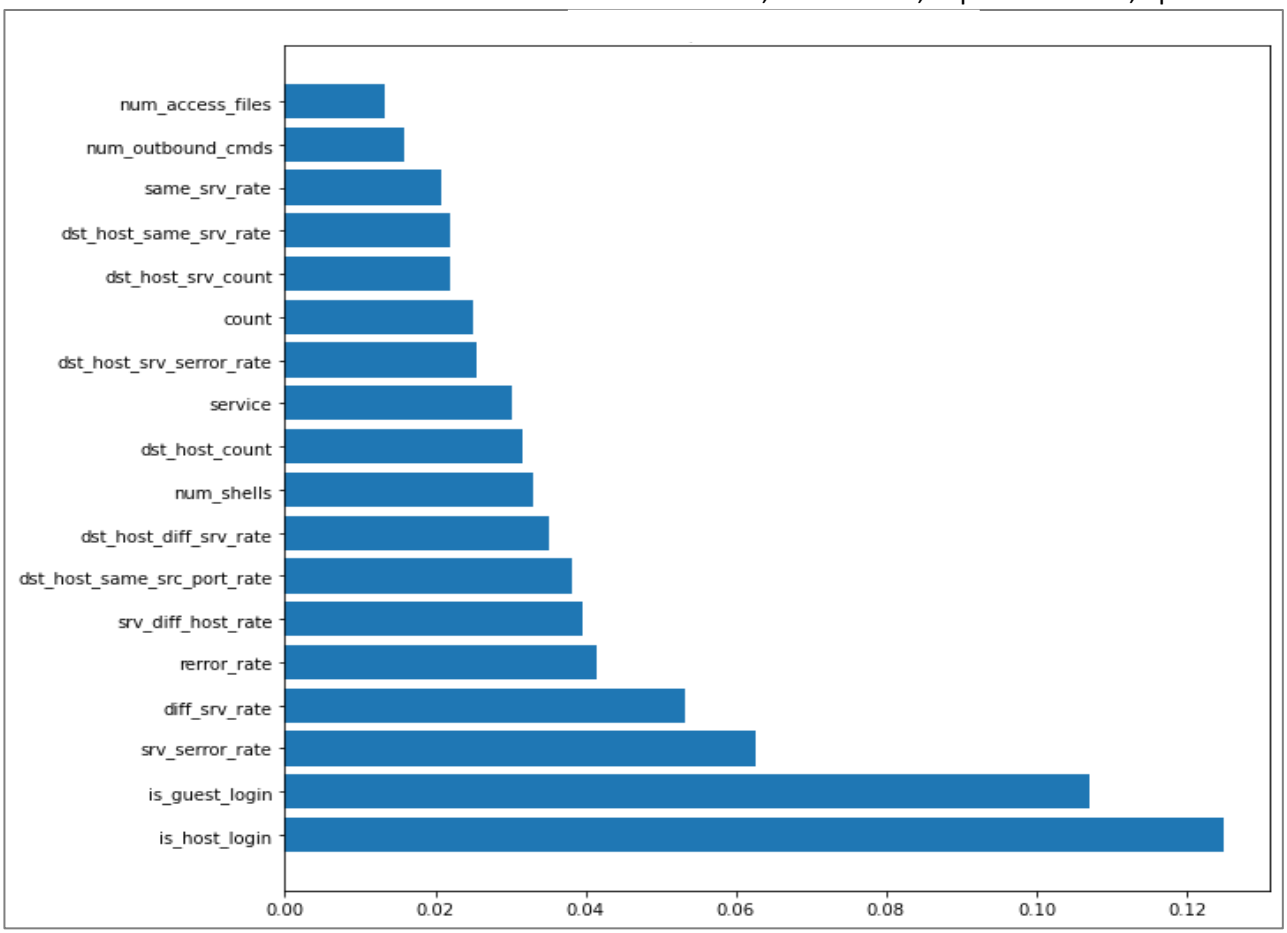

Figure 4: Feature importance of NSL-KDD dataset with random forest

System Testing

To make sure the system was reliable and functioning, it was put through extensive testing. To ensure that the models generated reliable forecasts and operated well on various data subsets, system tests were carried out. The system performed satisfactorily and yielded trustworthy network intrusion detection findings, which shows a better performance compared to the previous work in which the system had problems with accuracy, false alarms, and execution time. The table below (Table 3) shows Testing and validation results from the system.

Table 3: Testing and validation results from the Intrusion detection model.

	Logistic Regression (%)	K Neighbors Classifier (%)	Gaussian Naive Bayes (%)	Support Vector Machines (%)	Decision Tree (%)	Random Forest (%)
Accuracy	0.884	0.989	0.916	0.969	0.998	0.999
Precision	0.853	0.991	0.925	0.967	0.998	0.999

Scalability

A model combining the strengths of rule-based for known attacks and machine learning for unknown attacks offers a robust approach to intrusion detection. When coupled with high accuracy and a low false positive rate, it becomes an even more attractive solution. However, scalability is no longer an issue of concern as networks expand in size and complexity because Data Volume and Velocity are adequate. The increase in the amount of data that is generated and stored is rapid, and the level of accuracy actually curbed the known attack or unknown attack that may arise. Real-time Processing is also in place as a result of the execution time for immediate detection of threats without compromising accuracy.

Data Reduction Techniques: Techniques like aggregation, sampling, and anomaly detection are used to drastically reduce the amount of data while keeping important information.

CONCLUSION

Finally, the created intrusion detection model has shown to be useful in effectively identifying and categorising

network intrusions. The combination of machine learning techniques, comprehensive feature selection, and rigorous assessment produced encouraging outcomes that outperform the outcome of Hadeel et al. (2024). The model's strong accuracy, precision, and recall rates show its potential for real-world use in network security. However, it is critical to maintain research and development efforts to improve the model's performance and keep up with increasing network threats. Future work should focus on integrating real-time data streams to enhance the model's applicability in dynamic network environments. Additionally, exploring other deep learning architectures like LSTM networks could further improve detection capabilities and enable smooth integration with a variety of systems and devices. The model may be expanded to function in real-time settings, in which network traffic is continually watched and analysed for quick detection and reaction to possible threats. This would necessitate efficient algorithms and simplified data processing methods.

ACKNOWLEDGEMENT

The authors would like to thank the Department of Computer Science at Federal University Lokoja in Nigeria for creating an atmosphere conducive to conducting this research.

REFERENCE

Adnan, M. M., Venugopal, D., & Shiva, S. G. (2020). Comparative analysis of ML classifiers for network intrusion detection. *Journal of Comparative Analysis, 1027,* 193–207. **[Crossref]**

Ahmad, R., Alsmadi, I. A., Wasim, A., & Tawalbeh, L. (2022). A comprehensive deep learning benchmark for IoT IDS. *Journal of Computer and Security,* 114, 102588. **[Crossref]**

Ahmad, R., Alsamadi, I., Alhamdani, W., & Tawalbeh L. (2022). A deep learning ensemble approach to detecting unknown network attacks. *Journal of Information Security and Applications,* 67, 103196. **[Crossref]**

Alimgeer, K. Yuanqing, G. Ameer, M. Tayyab P. & Khurrum D. (2022). Hybrid of WOA-ABC and proposed CNN for intrusion detection system in wireless sensor network. *Journal of Sensor and Network,* 101, 42-67. **[Crossref]**

Duy, P. T., Tien, L. K., Khoa, N. H., Hien, D. T. T., Ngyen, A, G., & Pham V. (2021). DIGFuPAS: Deceive IDS with GAN and function-preserving on adversarial samples in SDN-enabled networks. *Journal of Computer and Security,* 109, 102367. **[Crossref]**

Esra, A., Mohammed, A. A., Ahmed, A. (2024). Anomaly detection IDS for detecting DoS attacks in IoT Networks Based on Machine Learning Algorithms. *Sensors* 24 (2), 713. **[Crossref]**

Jing C. Y., Hongwei, L., Shuo, S., Futai, Z., & Yue, W., (2022). FS-IDS: A framework for intrusion detection based on few-shot learning. *Journal of Computer and Security*, 122, 102899. **[Crossref]**

Lirim, A., & Cihan, D. (2021). Network intrusion detection system using deep learning. *Procedia Computer Science*, *185*, 239–247. **[Crossref]**

Hadeel, Q. G., & Wathiq, L. A. (2024). e-Prime-Advances in Electrical Engineering, Electronics and Energy. *Journal of Computer Network* , 9, 100673, 2024. **[Crossref]**

Huang, S., & Lei, K. (2020). IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks. *Journal of Ad hoc Network,* 105, 102177. **[Crossref]**

Jayrajsinh, Z., Panchal, A., Thakkar, A., Prajapati, B., & Puvar, P. (2020). Intrusion detection system using machine learning. *Journal* of *Computer Security, 12,* 61–71. **[Crossref]**

Mohamed A. E., Mohammed A. A., Abdelghani D., Ibrahim R. A., Ahmed A. A., (2023). Intrusion detection approach for cloud and IoT environments using deep learning and Capuchin Search Algorithm. *Journal of Advances in Engineering Software* 176, 103402. **[Crossref]**

Mehdi, E., Ali, F., Abdulreza, M. & Zahra, T. (2022). ITL-IDS: incremental transfer leanring for intrusion detection system. *Journal of Knowledge-Base System, 253, 68-69.* **[Crossref]**

Mohammed, A. D. Youcef, D., Mebarek, B., Abdelkader, O., & Hasan, A. (2023). Convolutional neural network-based high-precision and speed detection system on CIDDS-001. *Journal of Data and Knowledge Engineering,* 144, 02130. **[Crossref]**

Saveetha, D. & Maragatham G. (2022). Design of blockchain enabled intrusion detection model for detecting security attacks using deep learning. *Journal of Pattern Recognition Letters, 153*, 24-28. **[Crossref]**

Sung J., Ying-Chin C., Chit-Jie C., Chih-Lung, C., Thu-Nguyet, H., & Chung-Wei K. (2022). CoNN-IDS: Intrusion detection system based on collaborative neural networks and agile training. *Journal of Computer and Security, 122, 102908.* **[Crossref]**

Sidharth, M., & Sharma, P. (2019). Intrusion detection using machine learning and feature selection. *Journal of Feature Selection,* 11(4), 43–52. **[Crossref]**

Taher, K. A., Mohammed Yasin Jisan, B., & Rahman, M. M. (2019). Network intrusion detection using supervised machine learning technique with feature selection. *1st International Conference (ICREST),* 643–646. **[Crossref]**

Unal, C. (2019). A new hybrid approach for Intrusion Detection using Machine Learning Methods. *Applied Intelligence, 49*(7), 2735–2761. **[Crossref]**