International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

*Research Article*

# Scalable Anomaly Detection Frameworks for Network Traffic Analysis in cybersecurity using Machine Learning Approaches

**Mani Gopalsamy**[1*]

[1]Senior Cyber Security Specialist, Louisville, KY, USA- 40220

Received 20 Nov 2022, Accepted 18 Dec 2022, Available online 20 Dec 2022, **Vol.12, No.6 (Nov/Dec 2022)**

***Abstract***

*The capacity to detect facts or observations that differ from what is normally thought of by domain experts is crucial for many contemporary applications. These outliers may be located with the use of anomaly detection, and the system can subsequently implement the required adjustments. This study presents a scalable anomaly detection framework for network traffic analysis in cybersecurity using advanced machine learning. Approaches. Leveraging the NSL-KDD dataset. Before model building, Recursive Feature Elimination (RFE) identifies the most relevant features for classification. Machine learning models—DNN, KNN, RF, and NB—are employed and evaluated using F1-score, recall, accuracy, and precision, with a confusion matrix to assess performance. Results show RF achieves the highest accuracy (99.81%), precision (99.89%), and recall (99.90%), followed closely by KNN. Generally speaking, both NB and DNN perform worse since their metrics are lower. The results reveal the enhanced performance of RF and KNN in terms of identifying and categorising anomalous behaviours in the network traffic and, therefore, providing a viable solution to augment existing real-time cybersecurity systems.*

***Keywords:*** *Anomaly detection, Network traffic analysis, Cybersecurity, Intrusion detection systems (IDS), machine learning.*

## 1. Introduction

It is noteworthy that the problem of network anomaly detection is becoming more and more acute in the context of the rapidly evolving threat space. Conventional security measures are struggling to cope with the burgeoning volume and type of risks, including both techniques and systems[1][2][3]. Since network anomaly detection is one of the most important defensive strategies focused on the detection of anomalous patterns or actions in the networks indicating that it is a potential threat or attack, it is impossible to overestimate the role of the said method in cybersecurity [4][5][6]. When it comes to security its impact has become more serious primarily due to the dependence on the internet and digital systems for privacy to national security[7][8]. Conventional security measures, frequently based on rules and static, are showing themselves to be insufficient in the face of sophisticated and adaptive cyber threats[9][10].

Anomaly detection is a core solution necessary within various industries and systems to deal with various application difficulties, including intrusion detection[11], data cleaning[12], fraud detection systems, health monitoring[13][3], and assortment optimisation [14][15][16][17].

*Corresponding author's ORCID ID: 0000-0000-0000-0000
DOI: https://doi.org/10.14741/ijcet/v.12.6.9

Considering its primary application, anomaly detection is about defining and searching for data patterns significantly different from the norm and can point to experimental errors or fraudulent actions. It is essential for the organisation's Continuity [18][19]. Maintaining the integrity of the systems can significantly affect the various components of the organisation's financial health and efficiency[20].

In the classic scenarios, the anomaly detection techniques were relatively primitive and more or less a manual exercise that is highly static with much dependence on thresholds and rules of thumb[21][22]. This makes them relatively static and often requires updating based on the subject expertise, which results in much human involvement [23]. Machine learning (ML) techniques offer advanced capabilities for enhancing system observability through anomaly detection.

Machine learning (ML) provides a valuable means to address these challenges since the detection process is automated, and the methods can be trained and updated on new data without much intervention [24]. Machine learning algorithms mainly owe superiority in detecting large and intricate patterns and flipping the scale of manoeuvring various data streams with less

reliance on human knowledge[25]. This, in turn, results in accurate detection of anomalous conditions and increased efficiency in the lost detection process. As a result, it reduces the chances of missed anomalous situations and coupled financial impacts.

### *A. Motivation and Contribution of Study*

The growing complexity of cyber threats and the shortcomings of conventional security measures serve as a driving force behind this investigation. As network anomaly detection becomes crucial for cybersecurity, machine learning offers a powerful solution to automate the detection of abnormal patterns, reducing human intervention and enhancing accuracy. This study aims to develop a scalable and efficient framework using the NSL-KDD dataset and advanced ML models to improve intrusion detection and address a limitation of conventional systems. This study's contribution is to create a scalable anomaly detection framework for network traffic analysis using advanced machine learning techniques. The following key contributions are:

- Utilized the NSL-KDD dataset, addressing limitations of earlier intrusion detection datasets like KDD-99.
- Selected features using Recursive Feature Elimination (RFE), which improved model performance by concentrating on the most important characteristics.
- Applied ML models including DNN, KNN, RF, and Naïve Bayes for network anomaly classification.
- F1-score, recall accuracy, and precision were some of the important metrics used to evaluate the model's performance using a confusion matrix.

### *B. Structure of paper*

The paper's structure is set up like this: In Section II, previous studies on Network Traffic Analysis in Cybersecurity are reviewed. Section III details the research approach. The performance of the models and the experimental findings are shown in Section IV. Section V concludes by discussing the key findings and the implications of these findings.

## 2. Literature Review

This section provides a literature study of a previous work on the topic of cybersecurity in anomaly detection using ML-based classification techniques was given below:

In this study, Xu et al., (2021) provide a novel model built on a 5-layer AE more suited for use cases requiring the detection of abnormalities in networks. They built our method on top of a thorough examination of several performance measures included in an AE model. Our suggested model employs an innovative data pre-processing method that finds and removes the most influential outliers from the input samples in order to mitigate the bias that results from data imbalance among the different types of feature sets. Our proposed approach employs the optimal reconstruction error function to ascertain the normalcy or abnormality of a given sample of network data. By combining the top model architecture with these sets of novel techniques, our model becomes better at learning features and reducing dimensions, which in turn improves its f1-score and detection accuracy. We found that our suggested model had the best detection accuracy (90.61 percent) and f1-score (92.26%), compared to competing approaches on the NSL-KDD dataset [26].

In this paper, Gandhi, (2021) evaluated a number of ML approaches using the Stacked ensemble learning model that we proposed. In order to compare different ML methodologies, they propose a stacked ensemble learning model and employ metrics such as F1 score, accuracy, precision, recall, and area under the ROC curve. With a precision of 99.8 percent, the suggested strategy surpasses the majority of conventional ML techniques. To the current anomaly detection system, the suggested stacked ensemble learning model might be useful [27].

In this paper, Gandhi, (2021) evaluated our suggested Stacked ensemble learning model against several ML techniques. Our suggested stacked ensemble learning model is compared to other ML algorithms using evaluation measures such as F1 score, accuracy, precision, recall, and area under ROC curve. The suggested technique outperforms the majority of conventional ML algorithms with a remarkable accuracy rate of 99.8 percent. The current anomaly detection system might be enhanced with the help of the suggested stacked ensemble learning model [27].

In this work Alrawashdeh and Goldsmith, (2020) to lessen the impact of adversarial instances and backdoor assaults, provide a DL defensive strategy that integrates activation function with neurones pruning. They test the method's performance on a DBN and Coupled GAN anomaly detection application. The technique decreases the accuracy loss due to assaults by an average of 10% with DBN and 14% with CoGAN. Two benchmark datasets are used to assess the method: Ransomware and NSL-KDD[28].

This article Abdulhammed et al., (2019) construct a robust intrusion detection system using the most recent CIDDS-001 dataset by using diverse methods for dealing with skewed datasets. Using DNNs, stacking ML classifiers, variational autoencoder, voting, and RF, the efficacy of sampling techniques on CIDDS-001 is empirically examined and investigated in detail. As a result, the suggested approach is useful for real-time data fusion issues involving data classification, as it achieved an accuracy of 99.99% while dealing with the unbalanced class distribution using less samples [29].

**Table 1** Background study comparison on Network Traffic Analysis in Cybersecurity using Machine Learning Approaches

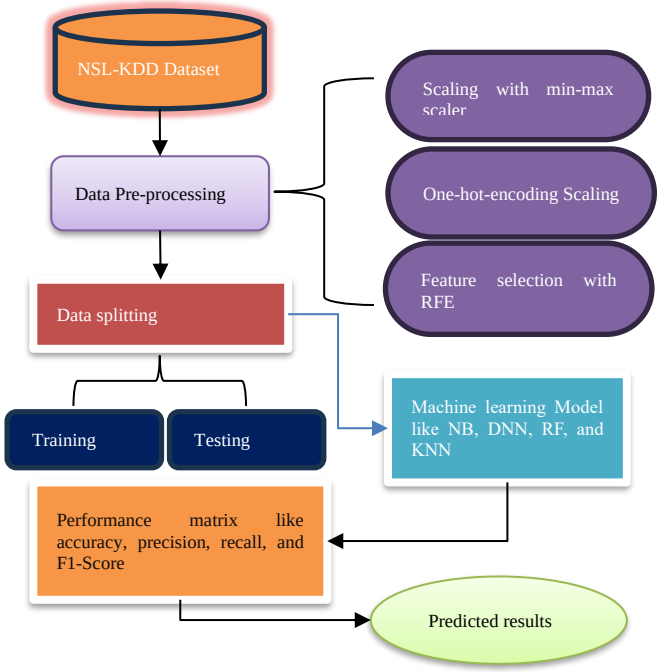
| Author | Methodology | Dataset | Performance | Limitation/Gap |
|---|---|---|---|---|
| Xu et al. (2021) | 5-layer autoencoder model with new preprocessing methods and reconstruction error function. | NSL-KDD | Accuracy: 90.61%, F1-Score: 92.26% | Limited to NSL-KDD; may not generalise to other datasets. |
| Gandhi (2021) | Comparison of various ML algorithms with proposed stacked ensemble model. | Public data | Accuracy: 99.8% | Lacks detailed analysis of specific algorithms compared. |
| Gandhi, (2021) | Utilised LSTM and various ML classifiers for detection. | Traditional Machine Learning Models | accuracy of 99.8% | Might not generalise well to unseen data without proper validation |
| Alrawashdeh and Goldsmith (2020) | Defensive technique combining activation function and pruning in deep learning models. | NSL-KDD, Ransomware | Reduced accuracy loss from attacks significantly. | May require more extensive evaluation across different models and datasets. |
| Abdulhammed et al. (2019) | Techniques for handling imbalanced datasets using various ML methods. | CIDDS-001 | Accuracy: 99.99% | Primarily focused on imbalanced datasets; broader implications need exploration. |

*A. Research gaps*

The field of network traffic analysis in cybersecurity using machine learning approaches has made significant strides, yet several research gaps remain. Firstly, many existing studies focus on specific types of attacks or datasets, limiting the generalizability of findings across different environments, as presented in Table I. Additionally, while advanced models like deep learning have shown promise, there is still a lack of comprehensive evaluations comparing these models against traditional techniques under varied conditions. Furthermore, issues related to data imbalance and the incorporation of real-time analysis techniques have not been sufficiently addressed, potentially impacting the effectiveness of detection systems. Addressing these gaps can improve a robustness and applicability of ML methods in cybersecurity.

## 3. Methods And Materials

The research methodology for Anomaly Detection Frameworks for Network Traffic Analysis in Cybersecurity using Machine Learning Approaches involves utilising the NSL-KDD dataset, a refined version of KDD-99, to address common shortcomings in intrusion detection systems (IDS). Standardisation of data techniques, including the Min-Max normalisation and One Hot Encoding, are used for preprocessing the data. The most important elements for categorisation are found by feature selection utilising RFE. A dataset is split into 80:20, where 80% is used to train, and 20% is utilised to further test various ML algorithms such as DNN, KNN, RF, and NB. The suggested models are assessed using recall, accuracy, precision, and F1-score, and their performance is measured using the confusion matrix. The purpose is to develop a large-scale anomaly detection model for the real-time monitoring of network traffic for real-time detection of security threats. Figure 1 is a flow diagram depicting the stages and procedures of the research process.




**Fi.1** Flowchart for Network Traffic Analysis in Cybersecurity

Each step of the following flowchart for Network Traffic Analysis in Cybersecurity is provided below:

*A. Data source*

The NSL-KDD-99 dataset is widely utilised in the fields of ML and IDSs, while it contains valuable data that does not need any kinds of conversions prior to analysis. While the KDD-99 datasets had some limitations, all such problems were addressed in the NSL-KDD dataset. The training dataset has about 1,074,992 vectors, which is comparable to KDD-99. There are 41 distinguishing characteristics of each. There are two categories for each vector: "Normal" and "Attack." Attacks are further split into a variety of subtypes, including DoS, U2R, R2L, and probing.

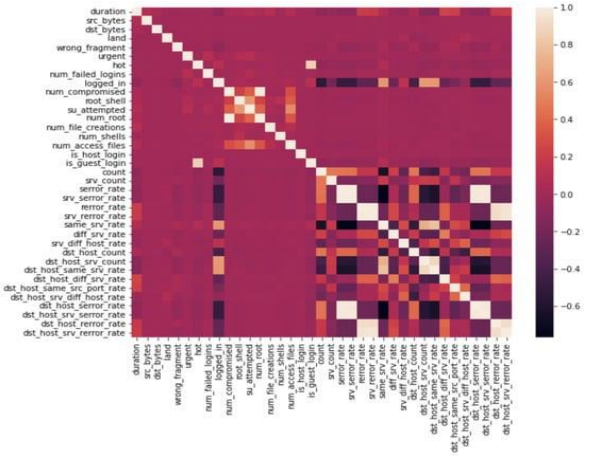


**Fig.2** Heat map of the correlated features

Figure 2 displays the linked characteristics heat map. The features are listed along both the x-axis and y-axis, with colour intensity representing a degree of correlation. A high positive connection is shown by dark red, a strong negative correlation by dark blue, and no link at all by white.

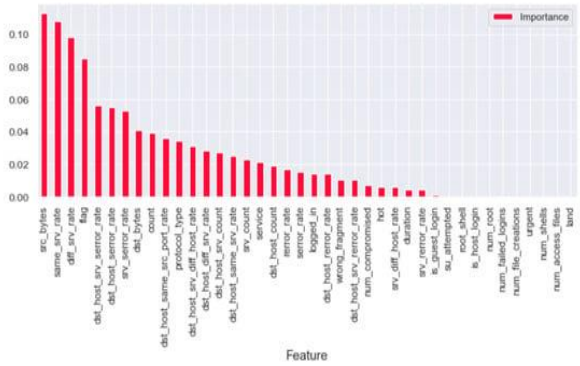



**Fig.3** Features and their importance in KDD Dataset

The bar chart in Figure 3 displays the significance of different features within the dataset. The horizontal axis lists the features, and a vertical axis shows their importance, ranging between 0 and 0.10. Red-coloured bars indicate the level of importance, with taller bars reflecting features that have a higher significance in the dataset, aiding in identifying the most influential factors for the analysis.

*B. Data preprocessing*

Data preprocessing is a cleansing procedure that transforms raw, unstructured data into a tidy, organised dataset that may be utilised for further study[30]. Preprocessing removes outliers and scales the features to an equivalent range. Scaling is used in the KDD-99 dataset as a preprocessing technique. The key pre-processing steps are listed below:

*C. Scaling with minmax scaler*

Machine learning makes extensive use of min-max scaling, also known as normalisation, one of the most basic feature scaling methods. Reducing the feature ranges to a predetermined interval while keeping the data distribution same is the essence of the technique[31]. The data structure and technique utilised determine the most common use cases for ranges; such examples are [0, 1] and ['1, 1]. When applied to the interval [0, 1], the min-max scaling is defined in Equation 1.

$$x_{scaled} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

where: xscaled = scaled sample point
x = sample point

*D. One-Hot-Encoding*

A kind of hot encoding is one that uses binary representation of the data as a feature. A popular approach compares the numerical variable's values at each level to a predetermined baseline. The data set used in this thesis makes use of one hot encoding to convert category variables into binary vectors [32][9].

*E. Feature Selection with Recursive Feature Elimination (RFE)*

A feature selection approach called RFE gradually eliminates the dataset's least essential characteristics based on how effectively an ML model operates. The estimator we used to identify the most important attributes was RFE. RFE fits a model to an existing set of attributes and eliminates the least important feature or features in every iteration until a certain number of features are attained. To choose the least significant attribute or feature, use Equation (2):

$$argmin x_i \in X\big(score(x \setminus i)\big) \quad (2)$$

where X represents the whole collection of features, Xi the i-th feature, X\i the set of all features except the i-th feature, and score(X\i) is a model's performance measure when fitted to a set of features X\i. Until the required number of features is obtained, the feature with the lowest score is removed at each iteration.

*F. Data splitting*

In this the dataset splits in an 80:20 ratio involves partitioning two parts that are training and testing. One compares a Training data set 80% in machine Learning models, and other contains of testing dataset in 20% a model`s performance.

*G. Machine learning models*

In this step, applied ML models named DNN, KNN, RF, and Naïve Bayes algorithms, which were described below in detail.

*H. Random forest (RF)*

RF is a technique for supervised ensemble ML. When combining several ML algorithms, the ensemble technique is used [33][25]. The name of this classifier gives it away: it's made up of many decision trees that work together to improve accuracy by averaging their results from different parts of the dataset. The RFC is often favoured because of its short training time, reliable outcomes, and ability to retain a high degree of accuracy even with larger datasets.

$$R_{yi} = \frac{1}{N}\sum_{n=1}^{N} P_n(yi) \quad (3)$$

*I. K Nearest Neighbor (KNN)*

Using the sample's Euclidean distance from its KNN in the dataset, the kNN classifier determines the sample's class. To determine which class the test data belongs to, we look at the average distance between each sample of k-nearest training points and their respective Euclidean distances. Equation (4) describes the Euclidean distance among the training sample xa and the test sample xb for 'f' number of feature vectors, the dataset comprises a total of 'n' samples.

$$d(x_{a1} - x_{b1})^2 + (x_{a2} - x_{b2})^2 + \cdots . . + (x_{af} - x_{bf})^2 \quad (4)$$

*J. Deep Neural Network (DNN)*

A multi-layer ANN including input, hidden, and output layers is called a DNN. By adjusting the relative importance of its connections, this network architecture may learn to do distributed processing in parallel.

*K. Naïve bayes (NB)*

The Naïve Bayes (NB) classifier, which originated from the Bayes theorem, was among the first approaches to resolving classification problems. It determines which class a test falls into using probability values. It is referred to as naive since it makes the assumption that the dataset's characteristics are independent of one another and do not interact [34] [35].

*L. Performance matrix*

Models constructed using classification algorithms were assessed in this study by means of a confusion matrix. For this performance review, we used 4 statistical measures: F-score, accuracy, precision, and precision. While "Y" stands for "No" in the True Negative (TN) class, the possibility of successfully detecting the True Positive (TP) class is represented by "Y" in the sensitivity class. A false positive (FP) happens when the model predicts a positive class when in fact the real class is negative, while a false negative (FN) happens when the model predicts a negative class when in fact the genuine class is positive. Following performance measures are as follows:

**Accuracy:** It is an indicator of the classifier's overall performance. It is a metric showing the rate of total correctly classified instances. Accuracy is defined as (5):

$$Accuracy = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (5)$$

**Precision:** The proportion of accurately detected positive samples compared to all positive samples is known as precision. A concise explanation of precision may be found in the following formula (6).

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

**Recall:** Recall, which may be expressed as a ratio of positively categorised samples to a total number of samples in an actual class, is given by Equation (7).

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

**F1-Score:** Precision and recall are the two main components of the F1-score. The F1-score accounts for categorised samples that are false positives as well as false negatives. The following Equation (8) represents the F1-score:

$$F1 - Score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

The following equations are utilised for determine the model efficiency.

## 4. Result Analysis and Discussion

This section offers two machine learning model names RF and KNN performance across performance matrices like f1-score, recall, accuracy, and precision. Table II displays both model performance on NSL-KDD data.
ML model performance for anomaly detection for network traffic analysis.

**Table 2** ML model performance for anomaly detection for network traffic analysis

| Matrix | KNN | RF |
|---|---|---|
| Accuracy | 99.51 | 99.81 |
| Precision | 99.30 | 99.89 |
| Recall | 99.38 | 99.90 |
| F1-Score | 99.34 | 99.90 |

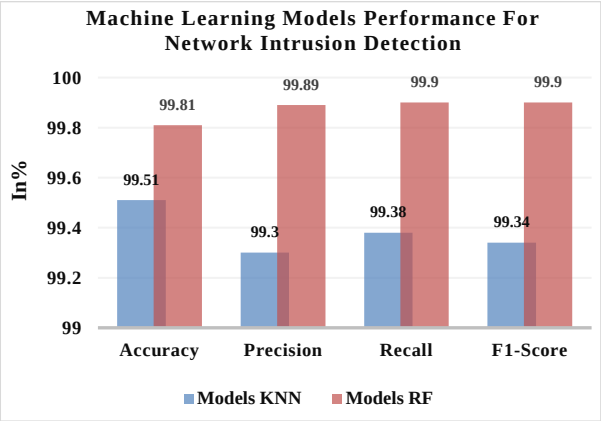



**Fig.4** ML Models Performance on NSL-KDD data

Figure 4 shows the ML model's performance on a NSL-KDD dataset. RF models exhibit an accuracy99.81%, while KNN models show an accuracy of 99.51%. The Random Forest model, in particular, consistently achieves an accuracy of 99.81%, making it the most accurate among the models compared.

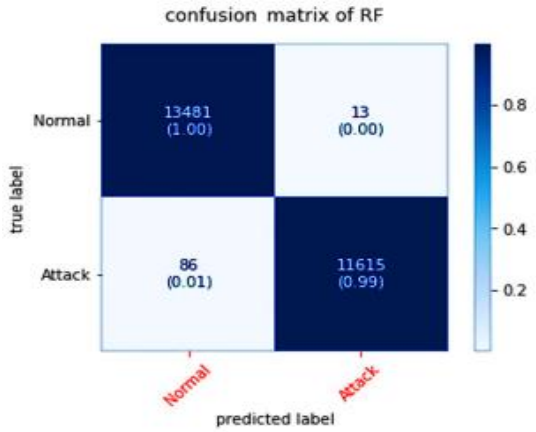



**Fig.5** Confusion Matrices of the RF model

The RF model's confusion matrix in Figure 5 shows strong performance with 13,481 TP and 11,615 TN. It has minimal misclassifications, with only 13 FN and 86 FP, indicating high accuracy in distinguishing between normal and attack cases.

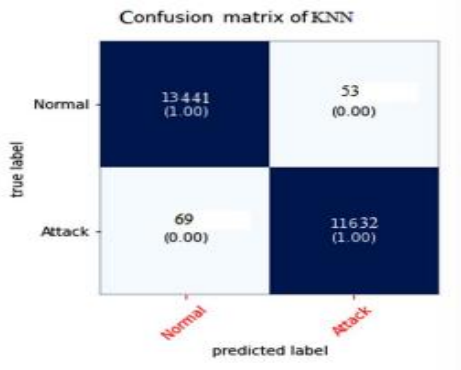



**Fig.6** Confusion Matrices of KNN model

The KNN model's confusion matrix in Figure 6 shows excellent performance, with 13,441 TP and 11,632 TN. It had minimal misclassifications, with only 53 false positives and 69 FN. The normalised values indicate perfect accuracy, with true positives and true negatives at 1.00, and FP and FN at 0.00.

*A. Comparative Analysis*

Using recall, F1-score, accuracy, and precision as important performance measures, this section compares the RF and KNN models with the NB and DNN models. Table III, shown below, summarises the performance of these models across these parameters specifically for network intrusion detection tasks. The comparison highlights the strengths of each model in

**Table 3** Comparison between different ML models for anomaly detection

| Performance Parameters | Machine learning models | | | |
|---|---|---|---|---|
| | ***KNN*** | ***RF*** | ***NB***[36] | ***DNN***[37] |
| Accuracy | 99.51 | 99.81 | 88.85 | 75.75 |
| Precision | 99.30 | 99.89 | 99.9 | 83 |
| Recall | 99.38 | 99.90 | 94.2 | 76 |
| F1-Score | 99.34 | 99.90 | 96.0 | 75 |

In comparing the performance of the ML model across all metrics present in Table III. RF achieves the highest accuracy at 99.81%, followed by KNN at 99.51%, while NB and DNN show much lower accuracies of 88.85% and 75.75%, respectively. RF also leads in precision of 99.89% and recall of 99.90, indicating strong identification of positive instances, compared to KNN's precision of 99.30% and recall of 99.38%. The F1-Score is highest for RF at 99.90%, with KNN at 99.34%, while NB and DNN lag with scores of 96.0% and 75. This analysis highlights the superior effectiveness of KNN and RF in classification tasks compared to NB and DNN.

## Conclusion and Future Scope

The number of malicious attacks on computer systems and networks has been steadily rising with the expansion of the Internet and other communication technologies. Network security has been seriously threatened to a certain extent, and network security technology has also attracted more and more attention from the public. This study developed a scalable anomaly detection framework for network traffic analysis using ML models, demonstrating the effectiveness of RF and KNN in identifying cyber threats with high F1-score, recall, accuracy, and precision. By utilising the NSL-KDD dataset, data preprocessing techniques, and feature selection through Recursive Feature Elimination (RFE), the research addressed the limitations of traditional intrusion detection systems (IDS). The results highlight RF as the best-performing model, with an accuracy of 99.81%, followed by KNN at 99.51%, while NB and Deep Neural Networks (DNN) showed significantly

lower performance. The findings underscore the potential of ML techniques to improve real-time anomaly detection in cybersecurity. Future work could explore the integration of more recent and diverse datasets to better capture real-world network behaviour.

## References


[1] R. Arora, "Mitigating Security Risks on Privacy of Sensitive Data used in Cloud-based Mitigating Security Risks on Privacy of Sensitive Data used in Cloud-based ERP Applications," *8th Int. Conf. "Computing Sustain. Glob. Dev.*, no. March, pp. 458–463, 2021.

[2] Mani Gopalsamy, "Enhanced Cybersecurity for Network Intrusion Detection System Based Artificial Intelligence (AI) Techniques," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 12, no. 01, pp. 671–681, Dec. 2021, doi: 10.48175/IJARSCT-2269M.

[3] M. Gopalsamy, "Artificial Intelligence (AI) Based Internet-ofThings (IoT)-Botnet Attacks Identification Techniques to Enhance Cyber security," *Int. J. Res. Anal. Rev.*, vol. 7, no. 4, pp. 414–420, 2020.

[4] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab, "Machine Learning for Anomaly Detection: A Systematic Review," *IEEE Access*. 2021. doi: 10.1109/ACCESS.2021.3083060.

[5] S. K. R. A. Sai Charan Reddy Vennapusa, Takudzwa Fadziso, Dipakkumar Kanubhai Sachani, Vamsi Krishna Yarlagadda, "Cryptocurrency-Based Loyalty Programs for Enhanced Customer Engagement," *Technol. Manag. Rev.*, vol. 3, no. 1, pp. 46–62, 2018.

[6] M. Gopalsamy, "Advanced Cybersecurity in Cloud Via Employing AI Techniques for Effective Intrusion Detection," *Int. J. Res. Anal. Rev.*, vol. 8, no. 01, pp. 187–193, 2021.

[7] R. Bishukarma, "The Role of AI in Automated Testing and Monitoring in SaaS Environments," *IJRAR*, vol. 8, no. 2, 2021, [Online]. Available: https://www.ijrar.org/papers/IJRAR21B2597.pdf

[8] V. Kumar, V. V. Kumar, N. Mishra, F. T. S. Chan, and B. Gnanasekar, "Warranty failure analysis in service supply Chain a multi-agent framework," in *SCMIS 2010 - Proceedings of 2010 8th International Conference on Supply Chain Management and Information Systems: Logistics Systems and Engineering*, 2010.

[9] V. V. Kumar, A. Sahoo, and F. W. Liou, "Cyber-enabled product lifecycle management: A multi-agent framework," in *Procedia Manufacturing*, 2019. doi: 10.1016/j.promfg.2020.01.247.

[10] V. K. Y. Nicholas Richardson, Rajani Pydipalli, Sai Sirisha Maddula, Sunil Kumar Reddy Anumandla, "Role-Based Access Control in SAS Programming: Enhancing Security and Authorization," *Int. J. Reciprocal Symmetry Theor. Phys.*, vol. 6, no. 1, pp. 31–42, 2019.

[11] J. Thomas, K. V. Vedi, and S. Gupta, "The Effect and Challenges of the Internet of Things (IoT) on the Management of Supply Chains," *Int. J. Res. Anal. Rev.*, vol. 8, no. 3, pp. 874–879, 2021.

[12] V. K. Yarlagadda and R. Pydipalli, "Secure Programming with SAS: Mitigating Risks and Protecting Data Integrity," *Eng. Int.*, vol. 6, no. 2, pp. 211–222, 2018.

[13] V. K. Yarlagadda, "Harnessing Biomedical Signals: A Modern Fusion of Hadoop Infrastructure, AI, and Fuzzy Logic in Healthcare," *Malaysian J. Med. Biol. Res.*, vol. 2, no. 2, pp. 85–92, 2021.

[14] M. Z. Hasan, R. Fink, M. R. Suyambu, M. K. Baskaran, D. James, and J. Gamboa, "Performance evaluation of energy efficient intelligent elevator controllers," in *IEEE International Conference on Electro Information Technology*, 2015. doi: 10.1109/EIT.2015.7293320.

[15] V. V Kumar, "An interactive product development model in remanufacturing environment : a chaos-based artificial bee colony approach," 2014.

[16] V. V. Kumar, M. K. Pandey, M. K. Tiwari, and D. Ben-Arieh, "Simultaneous optimization of parts and operations sequences in SSMS: A chaos embedded Taguchi particle swarm optimization approach," *J. Intell. Manuf.*, 2010, doi: 10.1007/s10845-008-0175-4.

[17] V. V. Kumar, F. T. S. Chan, N. Mishra, and V. Kumar, "Environmental integrated closed loop logistics model: An artificial bee colony approach," in *SCMIS 2010 - Proceedings of 2010 8th International Conference on Supply Chain Management and Information Systems: Logistics Systems and Engineering*, 2010.

[18] V. Zavrtanik, M. Kristan, and D. Skočaj, "Reconstruction by inpainting for visual anomaly detection," *Pattern Recognit.*, 2021, doi: 10.1016/j.patcog.2020.107706.

[19] V. Kumar and F. T. S. Chan, "A superiority search and optimisation algorithm to solve RFID and an environmental factor embedded closed loop logistics model," *Int. J. Prod. Res.*, vol. 49, no. 16, 2011, doi: 10.1080/00207543.2010.503201.

[20] S. K. R. Anumandla, V. K. Yarlagadda, S. C. R. Vennapusa, and K. R. V. Kothapalli, "Unveiling the Influence of Artificial Intelligence on Resource Management and Sustainable Development: A Comprehensive Investigation," *Technol. \& Manag. Rev.*, vol. 5, no. 1, pp. 45–65, 2020.

[21] V. V. Kumar, S. R. Yadav, F. W. Liou, and S. N. Balakrishnan, "A digital interface for the part designers and the fixture designers for a reconfigurable assembly system," *Math. Probl. Eng.*, 2013, doi: 10.1155/2013/943702.

[22] P. Pathak, A. Shrivastava, and S. Gupta, "A survey on various security issues in delay tolerant networks," *J Adv Shell Program.*, vol. 2, no. 2, pp. 12–18, 2015.

[23] A. Diro, N. Chilamkurti, V. D. Nguyen, and W. Heyne, "A comprehensive study of anomaly detection schemes in iot networks using machine learning algorithms," *Sensors*. 2021. doi: 10.3390/s21248320.

[24] A. S. Ramakrishna Garine, Rajeev Arora, Anoop Kumar, "Advanced Machine Learning for Analyzing and Mitigating Global Supply Chain Disruptions during COVID-19," *SSRN*, pp. 1–6, 2020.

[25] J. Thomas and V. Vedi, "Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 9, 2021.

[26] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3116612.

[27] N. Gandhi, "Stacked ensemble learning based approach for anomaly detection in IoT environment," in *2nd International Conference on Range Technology, ICORT 2021*, 2021. doi: 10.1109/ICORT52730.2021.9581549.

[28] K. Alrawashdeh and S. Goldsmith, "Defending Deep Learning Based Anomaly Detection Systems against White-Box Adversarial Examples and Backdoor Attacks," in *International Symposium on Technology and Society, Proceedings*, 2020. doi: 10.1109/ISTAS50296.2020.9462227.

[29] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. Abumallouh, "Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic," *IEEE Sensors Lett.*, 2019, doi: 10.1109/LSENS.2018.2879990.

[30] K. Patel, "Quality Assurance In The Age Of Data Analytics: Innovations And Challenges," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 12, pp. f573–f578, 2021.

[31] A. P. A. Singh, "Streamlining Purchase Requisitions and Orders : A Guide to Effective Goods Receipt Management," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 5, pp. g179–g184, 2021.

[32] K. Potdar, T. S., and C. D., "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers," *Int. J. Comput. Appl.*, 2017, doi: 10.5120/ijca2017915495.

[33] A. O. Alzahrani and M. J. F. Alenazi, "Designing a network intrusion detection system based on machine learning for software defined networks," *Futur. Internet*, 2021, doi: 10.3390/fi13050111.

[34] S. Shaukat *et al.*, "Intrusion Detection and Attack Classification Leveraging Machine Learning Technique," in *Proceedings of the 2020 14th International Conference on Innovations in Information Technology, IIT 2020*, 2020. doi: 10.1109/IIT50501.2020.9299093.

[35] S. Pandey, "Transforming performance management through ai: advanced feedback mechanisms, predictive analytics, and bias mitigation in the age of workforce optimizatioN," *Int. J. Bus. Quant. Econ. Appl. Manag. reseacrh*, vol. 6, no. 7, pp. 1–10, 2020.

[36] V. Pai, Devidas, and N. D. Adesh, "Comparative analysis of Machine Learning algorithms for Intrusion Detection," in *IOP Conference Series: Materials Science and Engineering*, 2021. doi: 10.1088/1757-899X/1013/1/012038.

[37] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for Network Intrusion Detection in Software Defined Networking," in *Proceedings - 2016 International Conference on Wireless Networks and Mobile Communications, WINCOM 2016: Green Communications and Networking*, 2016. doi: 10.1109/WINCOM.2016.7777224.