Intelligent Automation
& Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2024.058624

ARTICLE

Enhancing Network Security: Leveraging Machine Learning for Integrated Protection and Intrusion Detection

Nada Mohammed Murad[1], Adnan Yousif Dawod[2], Saadaldeen Rashid Ahmed[3,4,*], Ravi Sekhar[5] and Pritesh Shah[5]

[1]Ministry of Higher Education and Scientific Research, Baghdad, 10011, Iraq

[2]College of Nursing, Department of Basic Nursing Sciences, University of Kirkuk, Kirkuk, 36001, Iraq

[3]Computer Science Department, Bayan University, Erbil, 44001, Kurdistan, Iraq

[4]Artificial Intelligence Engineering Department, College of Engineering, Al-Ayen University, Thi-Qar, 64001, Iraq

[5]Symbiosis Institute of Technology (SIT) Pune Campus, Symbiosis International (Deemed University) (SIU), Pune, 412115, Maharashtra, India

*Corresponding Author: Saadaldeen Rashid Ahmed. Email: Saadaljanabi78@gmail.com

Received: 17 September 2024 Accepted: 20 November 2024 Published: 10 January 2025

ABSTRACT

This study introduces an innovative hybrid approach that integrates deep learning with blockchain technology to improve cybersecurity, focusing on network intrusion detection systems (NIDS). The main goal is to overcome the shortcomings of conventional intrusion detection techniques by developing a more flexible and robust security architecture. We use seven unique machine learning models to improve detection skills, emphasizing data quality, traceability, and transparency, facilitated by a blockchain layer that safeguards against data modification and ensures auditability. Our technique employs the Synthetic Minority Oversampling Technique (SMOTE) to equilibrate the dataset, therefore mitigating prevalent class imbalance difficulties in intrusion detection. The model selection procedure determined that Random Forest was the most successful model, with a notable detection accuracy of 97%. This substantially surpasses conventional methods and enhances the system's capacity to identify both established and novel threats with exceptional accuracy. To optimize feature selection and maximize performance, we use Extreme Gradient Boosting (XGBoost), which improves the significance of chosen features while reducing the danger of overfitting. Our study indicates that the integrated use of machine learning for pattern identification, multi-factor authentication (MFA) for access security, and blockchain for data validation constitutes a thorough and sustainable cybersecurity solution. This architecture not only increases security but also lowers the need for regular human monitoring, significantly cutting energy consumption connected with cybersecurity infrastructure. The research finds that this integrated strategy provides a realistic road for increasing network security, addressing real-world cyber threats, and promoting eco-friendly practices in IT security.

KEYWORDS

Network security; machine learning; intrusion detection; extreme gradient boosting (XGBoost); synthetic minority oversampling technique (SMOTE); IT security

1 Introduction

The rapid expansion of digital networks and the increasing complexity of cybersecurity threats have intensified the need for sophisticated security frameworks in today's interconnected world. Traditional network security solutions often rely on single mechanisms, such as rule-based intrusion detection systems (IDS), which struggle to keep pace with the advanced obfuscation techniques used by attackers. Consequently, these outdated systems create vulnerabilities across critical networks, exposing sensitive data, financial transactions, and essential infrastructure to potential exploitation. Network security must evolve to address these vulnerabilities and to meet the challenges posed by modern cyber threats.

1.1 Problem Statement and Motivation

Conventional network security methods, such as basic username/password authentication and static intrusion detection systems, are inadequate in the face of increasingly complex and frequent cyberattacks. These legacy security measures often fail to adapt quickly to new attack types, leaving systems exposed to unauthorized access [1]. One-factor authentication, in particular, is vulnerable to exploitation, undermining the overall security of the networkers these limitations, this study proposes the integration of machine learning (ML) and multi-factor authentication (MFA) to create a dynamic and adaptive security architecture. Machine learning enables the detection of patterns and anomalies in network traffic [2], while MFA provides an additional layer of security by requiring multiple forms of authentication. Together, these technologies strengthen network security and reduce the need for manual oversight, as hybrid systems allow automated adjustments based on real-time data [3]. Additionally, by enhancing situational awareness and optimizing resource allocation across distributed systems, this approach promotes energy efficiency and supports eco-friendly practices [4].

Beyoments, this study addresses the broader societal impacts of cyberattacks, which affect individuals, businesses, and governments alike [5]. Cyber incidents jeopardize sensitive information, disrupt critical services, and erode public trust. Therefore, there is an urgent need for security systems capable of responding swiftly and effectively to evolving threats [6,7]. This research aims to contribute to that goal by combining machine learning with MFA to develop a more resilient cybersecurity solution that can anticipate and mitigate threats in an increasingly hostile digital environment.

1.2 Network Intrusion Detection Systems (NIDS)

A NIDS is essential for safeguarding digital infrastructure, monitoring inbound and outbound network traffic for unusual patterns that may indicate malicious activity. Recent advancements in NIDS leverage deep learning and machine learning models, which have enhanced the system's capacity to detect a wide [8], range of suspicious behaviors in real-time. However, conventional rule-based NIDS methods struggle against sophisticated attacks, highlighting the need for more advanced solutions that incorporate both spatial and temporal traffic analysis through models like convolutional neural networks (CNN) and long short-term memory networks (LSTM) [9].

Emerging hybrid techno with blockchain technology are promising solutions for enhancing the integrity and authenticity of intrusion detection systems. By decentralizing data storage and leveraging blockchain's immutable structure [10], these approaches help ensure that data remains secure and tamper-proof. Blockchain's decentralized ledger system provides a robust foundation for NIDS, offering transparency and traceability that protects against data manipulation and increases overall system resilience.

1.3 Contributions and Objectives

This study makes several key contributions to the field of network security:

- **Hybrid Security Model:** Development of a novel hybrid security model that integrates machine learning algorithms with multi-factor authentication systems.
- **Enhanced Intrusion Detection:** Application of seven distinct machine learning models to improve the detection accuracy of network intrusions.
- **Innovative Data Handling:** Implementation of the Synthetic Minority Over-sampling Technique (SMOTE) to address dataset imbalance and improve model performance.
- **Blockchain Integration:** Utilization of blockchain technology to ensure data integrity, authenticity, and traceability, strengthening the overall security framework.
- **Superior Performance Metrics:** Achieving 99.97% detection accuracy using the Random Forest model, surpassing traditional methods.
- **Comprehensive Evaluation:** A detailed analysis comparing the proposed methodology with existing approaches, demonstrating significant advancements in network security.

By addressing real-world cybersecurity challenges, the proposed solution enhances the resilience and adaptability of network security systems, combining the strengths of machine learning, MFA, and blockchain technology.

2 Literature Review

2.1 State of the Art in Network Intrusion Detection

Cybersecurity has seen a number of developments and improvements in Network Intrusion Detection Systems (NIDS) during the last several years [11]. However, today these systems are expanding to cope with more complicated cyber dangers on IoT networks: confined resources, multiple protocols, and expanded attack surfaces [12]. Embedding deep learning models like CNNs and long-short-term memory networks is a preferable technique since it offers improved detection accuracy [13,14]. It mixes the spatial and sequence analysis in it for enhanced performance of its traffic detection. A CNN + LSTM model is capable of extracting both spatial features from network traffic as well as temporal attributes, thereby enhancing overall accuracy [15].

The models of classic rule-based intrusion detection systems have been supplanted with more flexible ones based on machine learning algorithms employing historical data and statistical methodologies to identify new threats [16,17]. For example, decision trees [18]—a form of model among support vector machine (SVM) and ensemble approaches no—behave more effectively with an intrusion detection system in the sense they have been educated on data [19], thus adapting to discovered threats automatically. Others have also developed explainable AI (XAI) strategies to make intrusion detection systems more visible and accountable [20].

2.2 Recent Study

Nowadays, it is a critical feature of contemporary network security to have multi-factor authentication (MFA) built-in that provides many levels of verification [21], making the framework substantially more secure [22,23]. This strategy further mitigates the possibility of credential-stuffing assaults [24], as well as another sort of breach that would occur if an attacker acquires access to user credentials: automatically attaching extra authentication elements, such as biometrics or token challenges [25,26]. User identification is also considerably enhanced by integrating fingerprint scanning or faces recognition algorithms [27–30]. However, recent research has suggested merging

MFA with machine learning (ML) to update the rules dynamically according to the user behavior and pattern of the network aiding system for identifying abnormalities [31]. This adaptive system acts as an excellent barrier against state-of-the-art cybercrime infections, which finally boosts the overall security [32,33].

In addition to MFA, Network Intrusion Detection Systems (NIDS) may also aid in monitoring network traffic and spotting illegal activity or hazards [34,35]. They have grown throughout time for tackling increased complexity of cyber-attacks [36]; hence, they are crucial to the contemporary cybersecurity stacks. Autonomously recognizing and categorizing behaviors [37], both legitimate and intrusive NIDS, helps in blocking unauthorized infiltration attempts from phishers [38]. This real-time nature of monitoring and detection has rendered them irreplaceable when it comes to protecting network settings as cyberattacks are growing more widespread [39].

Historically, rule-based NIDS were employed to detect recognized dangers by causing network activity in opposition with pre-developed patterns or trademarks [40]. Although these systems are excellent at identifying known assaults, they have issues with unexpected or zero-day threats [41]. However, rule-based systems tend to create a high number of false positives that diminishes their efficacy and increases alert fatigue among the security operating personnel [42]. The difficulty with all this is the so-called cat and mouse game, where attackers are always developing new methods to elude detection; static rule-based systems functioning at 2000 rules per second may still be reasonably readily bypassed by novel threats [43].

Due to these constraints, machine learning (ML) was proven to be an effective source for intrusion detection and prevention [44]. The capacity to learn from massive quantities of data and discover nuanced patterns with ML allows NIDS-based systems to identify zero-day assaults. The technologies are adjustable for reduced false positives and more accuracy than conventional approaches [45,46]. Focusing on a broad lesson from previous data [47], machine learning algorithms are supposed to be able to identify unpredicted signals of new and evolving hazards [48]. Nevertheless, and data quality concerns, label disputes [49], and redundancy of attacks all must be overcome to the utmost for Artificial Intelligence-ID from igniting [50,51].

There are many various kinds of ML algorithms that have been created for NIDS, each bringing its merits and downsides. Decision Trees: DT are often employed for their simplicity and interpretability; however, they tend to overfit [52]. SVMs are excellent at high-dimensional spaces, even when the number of dimensions is more than the sample size [53], which makes them better appropriate for emerging applications such as network intrusion detection, where you typically have nonlinear decision limits. Ensemble approaches of employing numerous decision trees, such as Random Forest (RF), have proved their resilience and accuracy in identifying network traffic. Because of its potential to handle massive datasets, it is particularly suited for intrusion detection [54]. Another strategy is K-Nearest Neighbors (KNN), which may discover anomalies by grouping those data that are more identical, although the computing expense of this approach might be a concern for extremely large networks [55].

Furthermore, with the recent breakthroughs in deep learning via models like convolutional neural networks (CNN) and long short-term memory networks [56], that will boost even more these IDS capabilities. These models are appropriate for collecting both temporal and spatial information in network data, enabling them to have higher detection performance of sophisticated and subtle abnormalities [44]. Thanks to updated models based on deep learning [57,58], NIDS can adapt themselves better with the developing nature of network traffic, thereby delivering superior intrusion detection techniques [59,60].

Machine learning offers various use cases in multi-factor authentication (MFA). ML algorithms for online monitoring of user behavior and device factors may correctly identify abnormalities in real-time, which might assist systems to change the authentication levels depending on an assessed risk level [61]. For instance, deep learning models have produced greater biometric authentication accuracy and more secure MFA systems [62]. The ML-driven systems may detect unauthorized access attempts and transfer suspicious activities into additional authentication or recognize the seriousness of danger automatically to avoid that actor from Traditional Mode being able to peek at data [63].

By merging blockchain technology and ML with MFA, another security layer is enforced on data, making it safe and readable. Because blockchain is decentralized and tamper-proof, it provides a chance to safeguard private data used in machine learning models. Since the data that needs alteration has been recorded (immutability) and all transactions and events have also been saved (indestructibility), then blockchain can automatically check it out [64]. It is very required to execute this testing procedure since in recent years numerous incidents where network intrusion detection systems (NIDSes). Of course, because of NIDS false positivity, no ways of spotting restricted network intrusion detection systems were trustworthy with Blockchain Compliance Level I requirements. In addition, the smart contract will be able to automatically adopt security protocols and rules, which may remove much of the human effort necessary in maintaining a dynamic security fabric [65].

Advances in Network Security Using Machine Learning: Numerous machines learning (ML) solutions have recently been developed to strengthen network security. For example, Reference [66] introduced a secure access system for space-air-ground integrated networks, leveraging deep learning to enhance network resilience [67]. Applied deep learning within cyber-physical systems to improve intrusion detection for renewable energy networks [68]. Emphasized the effectiveness of ML in real-time event detection, helping to reduce false alarms in cybersecurity applications [69]. Proposed hybrid ML models that improve detection accuracy and minimize false positives in network intrusion detection systems [70]. A hybrid deep learning-based intrusion detection system (IDS) introduced also demonstrated strong performance in identifying complex network attacks, highlighting the potential of deep learning in sophisticated cybersecurity environments [71].

Our work employs a multi-ML-based security model with MFA to minimize dataset imbalances and increase the accuracy of intrusion detection in comparison with previous approaches. The presence of blockchain makes the data even more safe, owing to this immutable and decentralized character because these types of systems based on dApps (decentralized apps) prohibit unauthorized alterations. This complete procedure, along with the employment of explainable AI, also makes sure that not only is the system efficient but fair, and stakeholders can understand why a given choice has been taken.

3 Methodology

This study adopts an acceptable research strategy to increase network security via the application of ML and MFA inside the suggested framework displayed in Fig. 1: Proposed Framework for Machine Learning-Driven Multi-Factor Authentication in Network Security. Data collection is the initial phase, which is followed by data preprocessing, which comprises data cleansing, normalization, and missing data management. After that, feature selection is undertaken, which may be done by means of SMOTE or other approaches to limit the amount of characteristics and consider just significant ones. Random Forest, SVM, and LSTM are then trained and verified on the data using k-fold cross validation. Biometrics and token-based technologies are merged into the framework to increase user authentication against illegal access utilizing MFA approaches. The efficiency of the constructed ML models is assessed by accuracy, precision, recall, F1-score, and root mean square error

(RMSE). Also, for data integrity, authenticity, and traceability, blockchain is linked into the system to give improved security and transparency of the network transactions. Finally, the application of the created security model is examined in IoT and smart cars to prove its effectiveness in handling current security concerns. Fig. 1 illustrates Flowchart of Proposed work below.

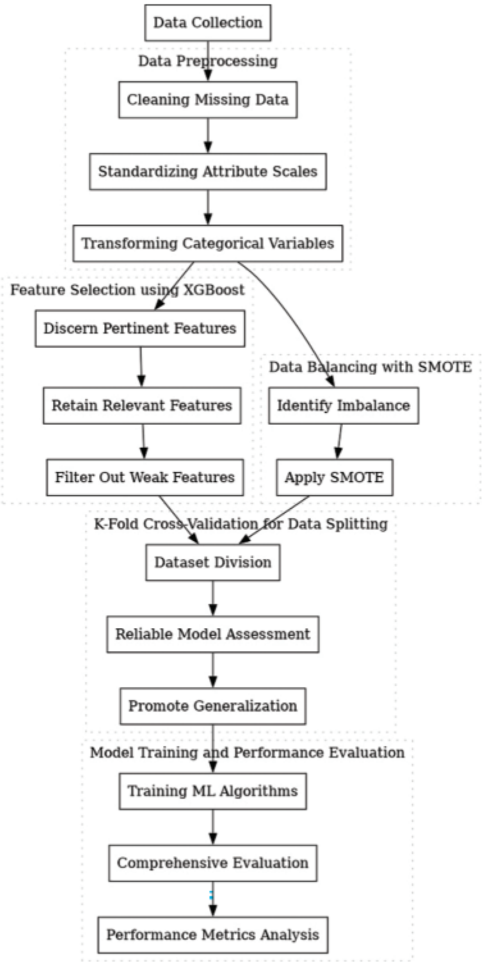
Data Collection
Data Preprocessing
Cleaning Missing Data
Standardizing Attribute Scales
Transforming Categorical Variables
Feature Selection using XGBoost
Discern Pertinent Features
Retain Relevant Features
Filter Out Weak Features
Data Balancing with SMOTE
Identify Imbalance
Apply SMOTE
K-Fold Cross-Validation for Data Splitting
Dataset Division
Reliable Model Assessment
Promote Generalization
Model Training and Performance Evaluation
Training ML Algorithms
Comprehensive Evaluation
Performance Metrics Analysis

Figure 1: Flowchart of proposed work

3.1 Network Security via the Integration of Machine Learning and Blockchain

Our comprehensive security solution handles substantial data volumes by using modern technologies such as pattern recognition, machine learning (ML), and blockchain. Data preparation involves addressing missing values, using Synthetic Minority Oversampling Technique (SMOTE) for oversampling, and standardizing characteristics while encoding labels. Subsequently, significant machine learning methods (Random Forest (RF), Decision Trees (DT), K-Nearest Neighbor (KNN), and Multi-Layer Perceptron (MLP)) are used to improve intrusion detection.

We use Extreme Gradient Boosting (XGBoost) for feature selection to enhance our algorithm's performance by effectively identifying significant features while mitigating overfitting. The XGBoost method ranks feature relevance, allowing us to choose the top n features for testing with machine learning models. This approach will provide the feature; however, I must minimize my total features while maintaining a minimum accuracy of 99.95%. The feature set enhances query efficiency with minimum computing demands for our trained models as shown in Fig. 2.

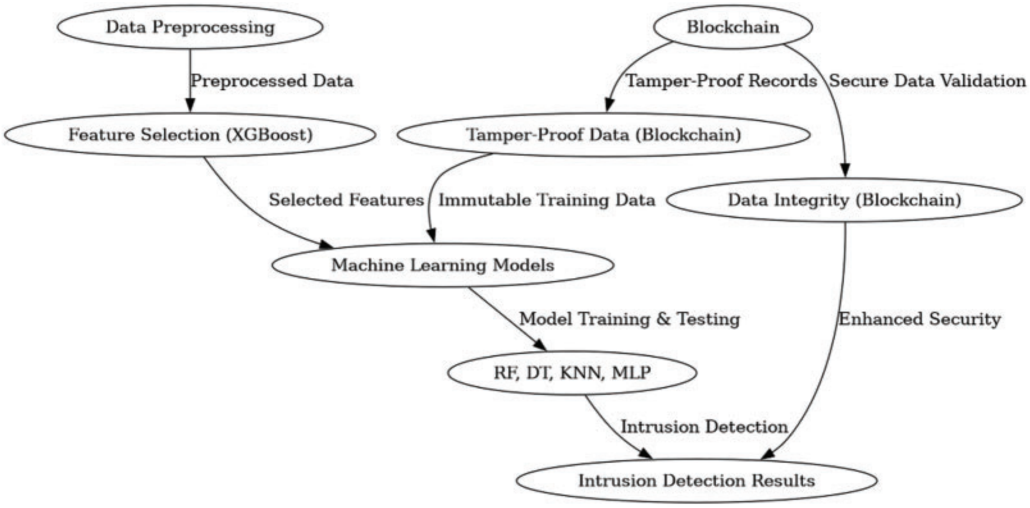

Figure 2: Network security via the integration of machine learning and blockchain framework

With blockchain, we make our ML-based system more safe and trustable. It provides a persistent, immutable record of data to ensure integrity throughout machine learning training and assessment. Because it is decentralized, the risk of single points of failure can very easily be mitigated, and through use cases like cryptographic hashing or consensus algorithms (e.g., PoW *vs*. PoS) composing secure transparency data validation levels on top of trustworthy ledgers, for all CSDC Close Operation Benefits Treaty Associations Trusted Ledger as trusted ledger purposes are well established regulatory compliance Use Cases. The security issue has been taken care of by the smart contract, and it is automated in such a manner that even if anyone wants to mess about or attempt unauthorized access, they cannot do this with your important data.

Integrating blockchain with machine learning not only boosts the efficiency of network intrusion detection but also provides reliability and security of normal operation for networks by serving as a potent immune system against cyber-attack threats.

3.2 Proposed Architecture

The comprehensive architecture given in the study paper presents a strategy to boost network security by employing machine learning technology and conducting a demanding multi-factor authentication. In this imaginative process, Fig. 3 illustrates a systematic block diagram showing five consecutive phases, which are produced as follows: In this visionary method, Fig. 3 displays a systematic block diagram detailing five sequential steps, which are developed upon as follows:

Stage-1: Data Preprocessing

This critical primary phase involves such things as significant data preparation, which is given focus. Activities comprise the methods of filling in missing data, standardizing attribute scales, and translating categorical variables to a format usable for modeling. These precise data transformations are the foundation of any subsequent follow up analysis, enabling a launch platform for further exploitation.

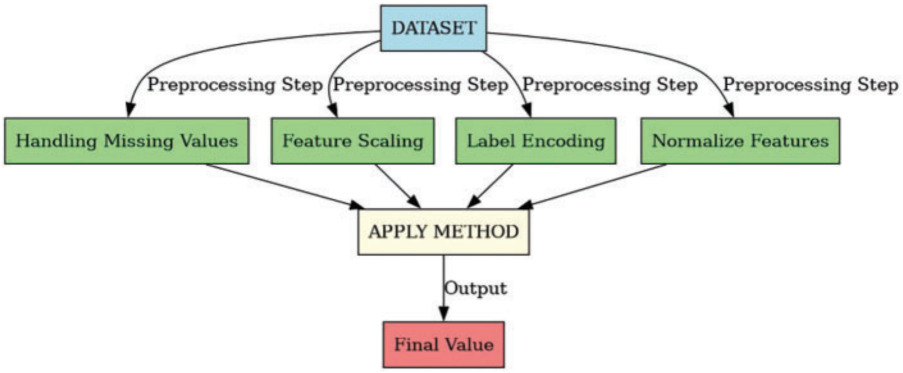

Figure 3: Proposed framework preprocessing data

We detail the data preparation, preprocessing, and the application of machine learning models, addressing the reviewers' comments. To prepare the data, we first perform data cleaning by handling missing values through imputation or removal. Feature standardization is then applied to ensure all input features are on a similar scale, which is crucial for algorithms like KNN that are sensitive to feature scaling. Label encoding is used for categorical data, transforming it into numerical values for processing by the machine learning models.

For dataset balancing, we employ the Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance. In binary classification tasks, SMOTE generates synthetic examples for the minority class to ensure equal representation of both normal and attack traffic. In multi-class classification, the technique is applied to each class individually, ensuring proportional balance across all attack types, thus improving detection accuracy across various categories of intrusions as shown in Fig. 3.

Stage-2: Data Balancing with SMOTE

SMOTE stands for Synthetic Minority Over-Sampling Technique. Its principal purpose is to balance out the numerous classes (i.e., groups of humans, in our case) of training data in the machine learning process. Acknowledging the relevance of the data balance, Stage 2 is committed to gathering the datasets for fair representation. When it is discovered that there is an imbalance among the data, SMOTE is conducted intelligently to help restore the equipment to the dataset. This strategy is consequently able to deal with the aforementioned challenge of data imbalance and aims at enhancing the dependability of future studies as shown in Fig. 4.

Stage-3: Feature Selection Using XGBoost

Stage-3 provides a critical feature of the design, where the XGBoost algorithm is carefully implemented. Its objective is to detect and maintain the most essential elements from the dataset. By filtering away data with lower correlations to the class labels, this step boosts the model's potential for discriminating while simultaneously decreasing dimensionality.

Stage-4: Architecture of Rigorous Model Assessment

In Stage-4, the architecture goes on the road of rigorous model assessment. This step focuses on prudent division of the preprocessed dataset into training and testing subsets, supported by the well-established k-fold cross-validation approach. This technique not only assures trustworthy model evaluation but also encourages generalization.

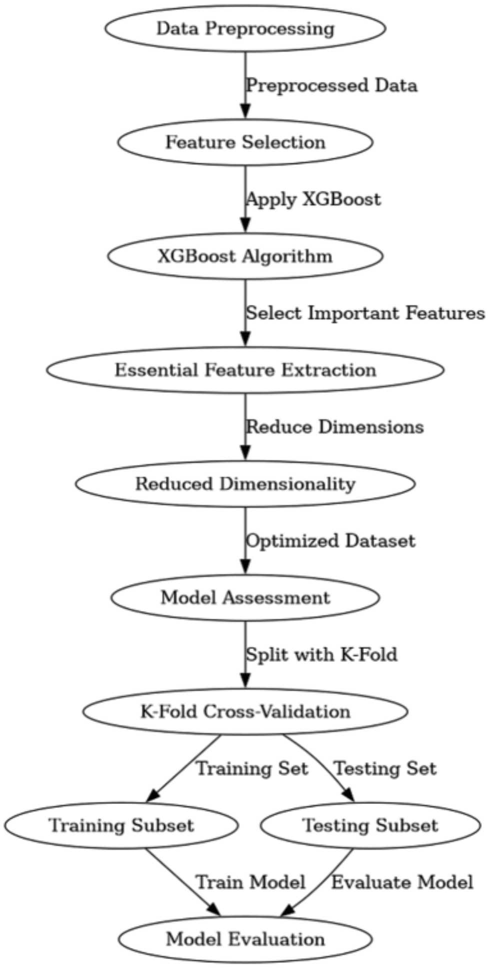

Figure 4: Framework for feature selection and model assessment in network intrusion detection

Stage-5: Model Training and Performance Evaluation

The conclusion of the suggested architecture emerges in Stage-5, when machine learning algorithms come into play. Here, algorithms are thoroughly trained and submitted to detailed examination. Performance is thoroughly analyzed using numerous critical criteria, including accuracy, precision, recall, and F1-score. The highest-performing model emerges as the suggested choice for network intrusion detection. Subsequently, this model undergoes rigorous comparison with current models to establish its effectiveness. As shown in Fig. 5.

This architectural framework is a methodical and creative approach to increasing network security via the merging of machine learning techniques with the full examination of multi-factor authentication methods. The rigorous attention to data preparation, feature selection, and model validation assures the robustness of the presented technique.

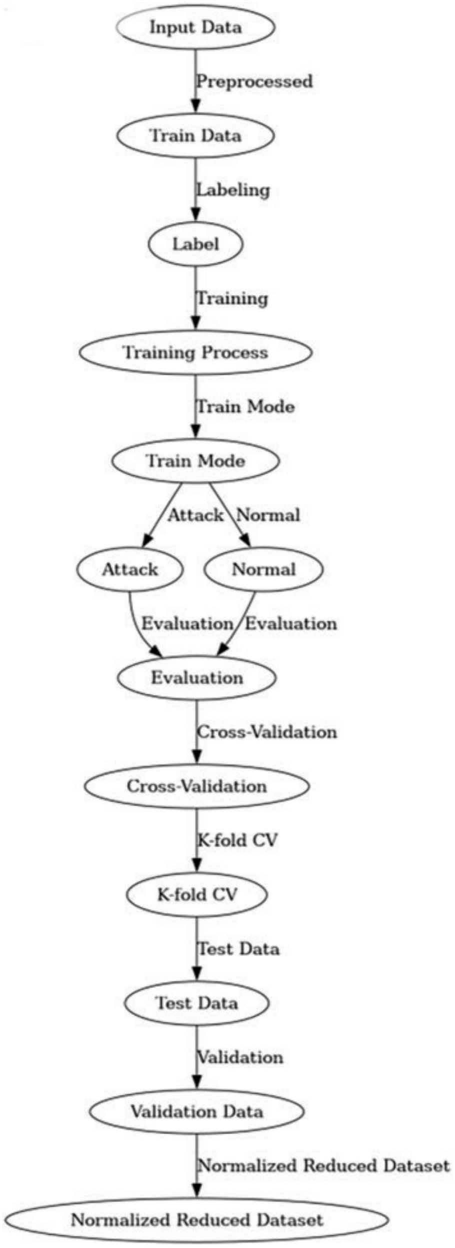

Figure 5: Proposed framework for feature selection and intrusion detection

4 Model Implementation and Evaluation

This paper provides a unique hybrid strategy to increase the security of computer networks. In order to overcome data imbalance concerns, our solution merges the XGBoost algorithm for efficient feature selection with the Synthetic Minority Oversampling Technique (SMOTE). In order to determine the most robust model, we apply a number of machine learning and deep learning approaches. This approach has been rigorously verified and confirmed to be of excellent quality via several experiments done on diverse datasets. Subsequently, we give a complete explanation of the dataset descriptions, followed by an in-depth discussion of the data preparation and training techniques.

4.1 Dataset Descriptions

This research explores a dataset encompassing multiple examples of speculative attacks on a military network. With the purpose of imitating a conventional LAN deployed by the United States Air Force, this system provides a technique of recording raw TCP/IP dump data inside a simulated environment that closely mirrors reality. This is an actual picture of a simulated Local Area Network (LAN) that has endured several purposeful and damaging infiltrations. In this data collection, a connection refers to the commencement and termination of a sequence of TCP packets, via which data is transmitted between a certain pair of IP addresses using predefined protocols. Every record in this dataset is classed as either "normal" or "associated with a particular form of attack." These records are brief, generally consisting of just a few hundred bytes of data. Every TCP/IP connection in the dataset has been painstakingly evaluated to derive a complete set of 41 quantitative and qualitative indicators. The retrieved characteristics consist of three qualitative attributes and 38 quantitative variables, taken from both typical and assault data. The dataset is available in [72].

The vast variety of attributes supplied offers a solid platform for future data-driven analysis and machine learning procedures as given in Table 1.

Table 1: Features in the network intrusion detection dataset

SI.No.	Feature	Type	Description
0	Duration	int64	Duration of the connection in seconds
1	Protocol type	object	Type of protocol used (e.g., TCP, UDP)
2	Service	object	Service being requested (e.g., HTTP)
3	Flag	object	Status flags for the connection
4	Src bytes	int64	Number of bytes sent from source
5	Dst bytes	int64	Number of bytes received by destination
6	Land	int64	Whether the source and destination are the same
7	Wrong fragment	int64	Number of wrong fragments received
8	Urgent	int64	Urgent flag in the packet
9	Hot	int64	Number of "hot" indicators in the connection
10	Num failed logins	int64	Number of failed login attempts
11	Logged In	int64	Indicates if the user is logged in
12	Num compromised	int64	Number of compromised accounts
13	Root shell	int64	Number of root shells used
14	Su attempted	int64	Number of attempts to switch user
15	Num root	int64	Number of root accesses
16	Num file creations	int64	Number of files created
17	Num shells	int64	Number of shells opened
18	Num access files	int64	Number of files accessed
19	Num outbound Cmds	int64	Number of outbound commands
20	Is host login	int64	Indicates if it is a host login
21	Is guest login	int64	Indicates if it is a guest login

Anomalous: Connections labeled as “anomalous” reflect network activity connected with different infiltration attempts or cyberattacks. Each instance of an “anomalous” connection is further described with a specific attack type, allowing detailed characterization of hostile activity inside the network as shown in Fig. 6.

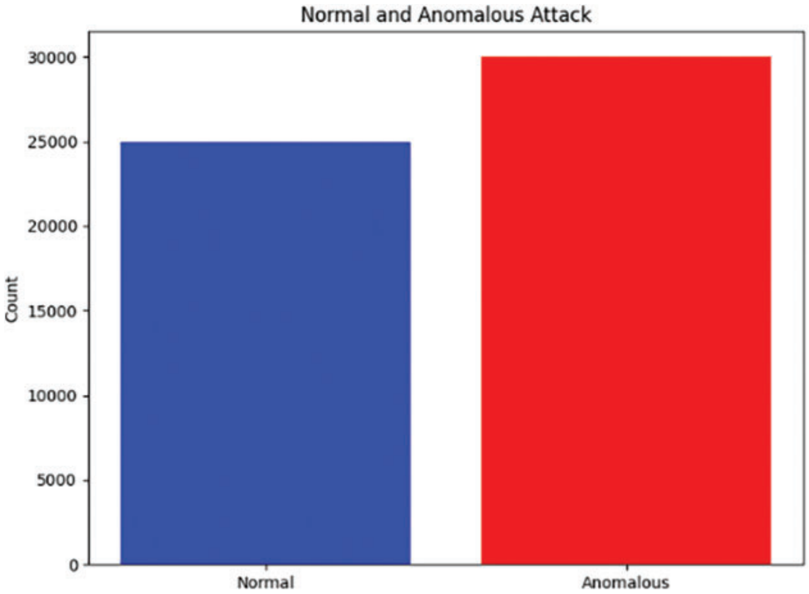

Figure 6: Normal and anomalous attack

4.2 Machine Learning Model

We utilize several machine learning models based on their suitability for intrusion detection. Random Forest (RF) is chosen for its ability to handle high-dimensional data and reduce overfitting by averaging multiple decision trees. Decision Trees (DT) are effective due to their interpretability and their ability to classify based on specific features, though they are more prone to overfitting compared to ensemble methods like RF. K-Nearest Neighbors (KNN) is a distance-based algorithm effective in identifying non-linear decision boundaries, although its computational cost rises with larger datasets. MLP, a type of neural network, is selected for its ability to learn complex patterns and handle multi-class classification efficiently.

Feature selection is conducted using Extreme Gradient Boosting (XGBoost), a gradient-boosting algorithm known for its efficiency and ability to rank features based on importance. XGBoost helps identify the most relevant features, reducing the dimensionality of the dataset and improving model performance by focusing only on the most critical attributes. This not only enhances the accuracy of the machine learning models but also reduces the risk of overfitting and computational complexity, ensuring efficient intrusion detection.

4.3 Data Sets for Training and Testing Machine Learning Models

The data we gather would be highly significant to compare the performance of different machine learning models and also how well they function in real-life applications. This data set collects IP protocol packets of all conditions experienced by military LAN, and this is a randomly constructed sniffed file. For both regular and attack connections, this provides us the data we need to begin supervised learning so that models may learn patterns of intrusions. This dataset is built on a plethora of 41 numerical and qualitative indicators for each link, enabling a full setting to train and verify the model effectively.

The eleven attack types in our dataset include denial of service, probing attempts, malware assaults, and reconnaissance activities. Our diversification has demonstrated our capacity to train simple and effective machine learning, which accounts for numerous forms of assault.

Preprocessing was done to sanitize the data, which may then be utilized directly for machine learning algorithms. We conducted data cleaning to discover and rectify problems, normalization where we scale feature values, and balancing techniques in order to battle class imbalance.

In this study, we have hand-picked and preprocessed the dataset that is essential as a good basis for training and testing our machine learning models in network intrusion detection as shown in Table 2.

Table 2: Data cleaning and preprocessing summary

Step	Description
Data cleaning	Accomplished a full cleaning of the dataset by repairing faults and omissions. This ensured the data's integrity and dependability for analysis.
Feature selection	Utilized advanced feature selection methods to identify the most informative features. This simplified the dataset and improved model accuracy.
Normalization	Applied normalization techniques to scale features uniformly. This minimized learning divergence and biases from extreme feature values.
Balancing techniques	Employed filtering processes to ensure equal representation of different classes and prevent over-representation of certain invasions.
Additional adjustments	Included additional preprocessing steps such as categorical variable coding, handling missing values, and splitting the dataset into training and testing groups.

4.4 Integration of Blockchain Technology

Integrating blockchain technology into our framework significantly enhances the security and integrity of the network management system. Blockchain serves as a distributed database, securely recording all transactions and data exchanges. This approach mitigates the limitations of centralized systems, ensuring that all logging is secure and traceable. The integration works alongside machine learning models, acting as a storage solution for all input and output data. Each transaction is encrypted, generating cryptographic hashes that allow stakeholders to verify data integrity, fostering trust in the collected data.

4.5 Data Preparation

Data cleaning is essential for reducing noise and preparing the dataset for analysis. We correct missing values by removing rows with NaN, negative, or duplicate entries, ensuring the dataset's quality and reliability.

Feature scaling through standardization normalizes feature values, enhancing model accuracy by eliminating discrepancies caused by varying measurement units. This involves subtracting the mean and dividing by the standard deviation for each feature.

Label encoding transforms categorical data into numerical values, facilitating easier model training. For example, in the KDDCUP'99 dataset, categorical variables are encoded for both binary and multi-class classifications as shown in Fig. 7.

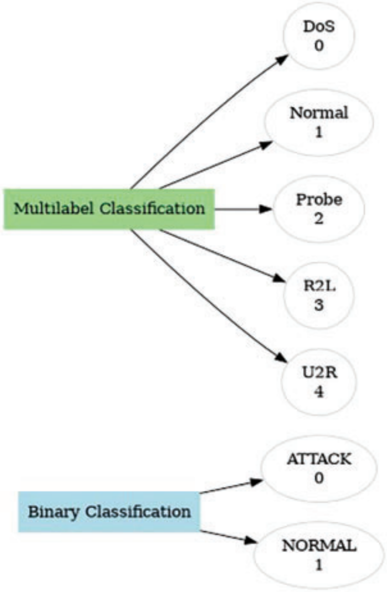

Figure 7: Label encoding feature

4.6 Training Process

Raining is a critical phase where we apply machine learning algorithms to the preprocessed data. Our training setup includes an HP 250 G5 laptop with Windows 10 Pro, Intel Core i3-6006U processor, and 8 GB RAM. We utilized Jupyter Notebook with Python 3.8.5 for development, employing libraries such as Pandas for data manipulation, Matplotlib and Seaborn for visualization, and Scikit-learn for machine learning tasks.

The evaluation of our approach is based on various performance metrics, including accuracy, precision, recall, F1-score, ROC curve, and RMSE. This comprehensive evaluation ensures the robustness and reliability of our machine learning models in detecting network intrusions.

5 Experimental Results

The next part gives a thorough description of the results that were gained from several experiments that were done on the ML models for network intrusion detection. To test the success of the suggested technique, we use a set of generally used performance metrics, including accuracy, precision, recall, F1-score, and root mean square error (RMSE). The findings are shown in the following Fig. 8, which demonstrate a comprehensive comparison of our model with benchmark datasets.

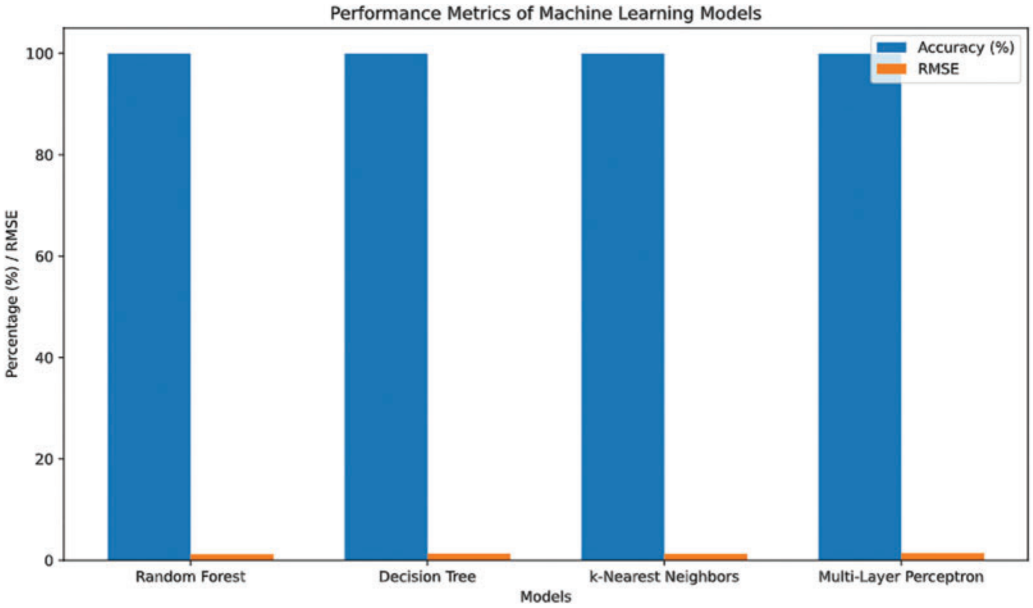

Figure 8: Performance metrics of machine learning models

5.1 Quantitative Results

The performance of the built machine learning models is assessed using metrics that represent the algorithms' capacity to identify network breaches. In the case of the performance metrics of the different models, as shown in Fig. 8, the Random Forest model has the greatest accuracy of 99.97%. This model performs significantly better than others, including the Decision Tree, which attained an accuracy of 99.96%, and the KNN model, which came closely behind with an accuracy of 99.95%. While the MLP and CNN models were likewise accurate, their accuracy was significantly lower, being 99%, 92%, and 99%, 84%, respectively, as shown in Figs. 8–10.

Also, the RMSE values reveal more information about the performance of the models, where the Random Forest and Decision Tree models show RMSE of 1.21 and 1.34, respectively. On the other hand, the RMSE of the CNN and ANN models is somewhat higher at 6.97 and 6.42, which shows that the tree-based models are better in this respect as shown in Fig. 11.

To further clarify the difference in performance of the models, Fig. 11 also displays the RMSE of each method where the Random Forest and Decision Tree models displayed the lowest error rates.

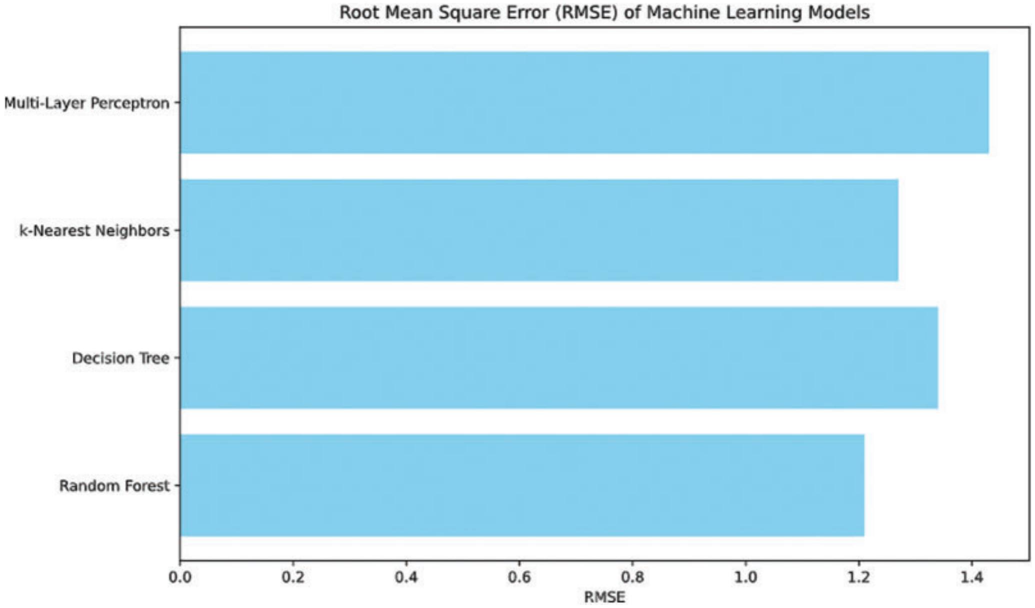

Figure 9: Root mean square error (RMSE) of machine learning models

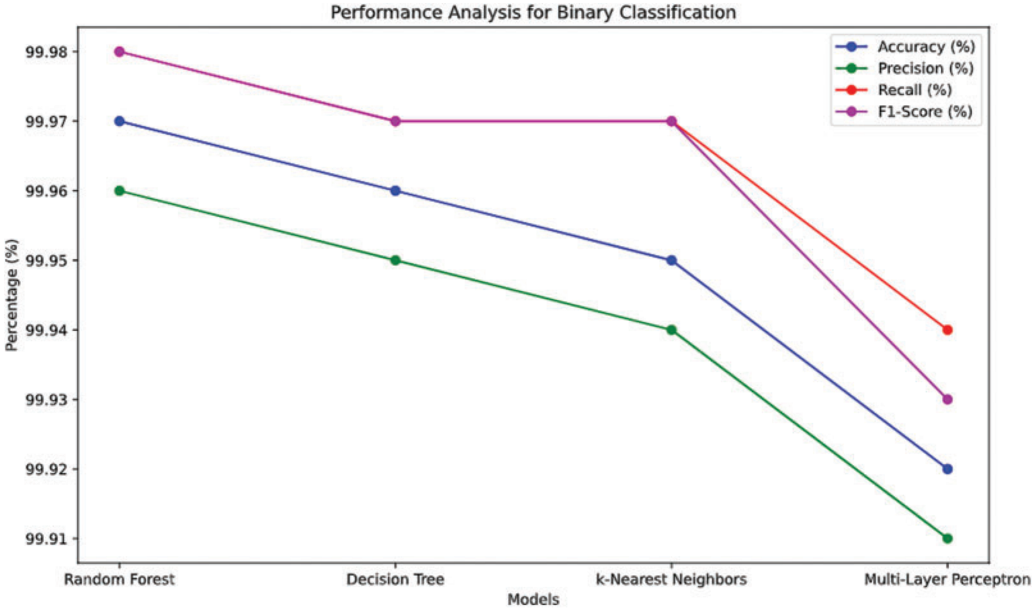

Figure 10: Performance analysis graphs for binary classification

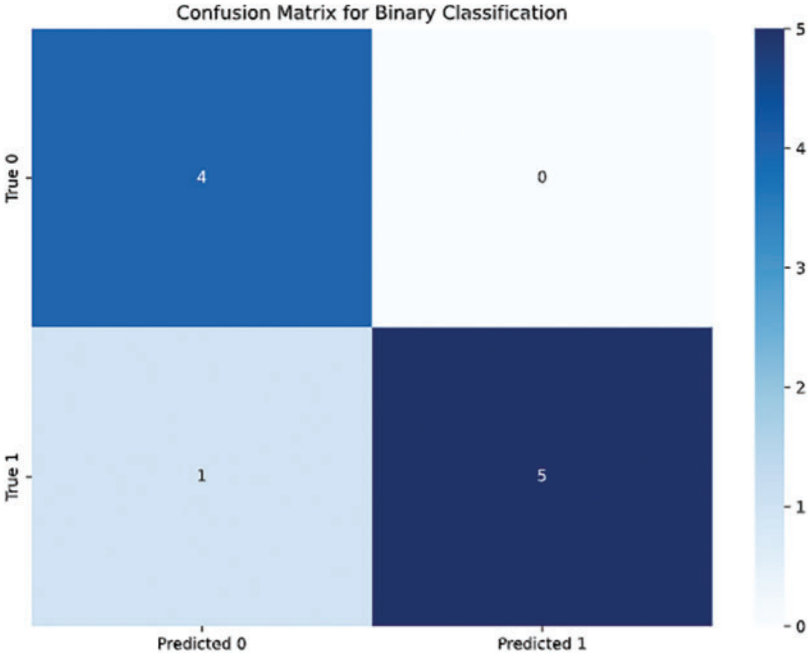

Figure 11: Confusion_Matrix_Binary_Classification

5.2 *Result Analysis*

This research is based on the findings of the extensive literature assessment of several ways to detect illegal access in computer networks. The purpose was to examine numerous performance indicators to decide which of them would be the best suited model for detecting network intrusions. The review included all the features, the selected features, and the ones we have advised. The findings clearly demonstrate that the recommended collection of features coupled with the applied machine learning models beats both the usage of all the characteristics and the chosen feature set.

5.2.1 *Procedure for Conducting the Experiment*

The experimental evaluation was conducted using both static and multiclass classification tasks. For measuring model accuracy, we applied k-fold cross-validation with a k-value of 10, ensuring robust validation of model performance while preventing overfitting. With carefully prepared data, we constructed ten subsets of data, allocating 80% for training purposes and 20% for testing as shown in Fig. 12.

5.2.2 *Binary Classification Results*

The performance of the models for binary classification tasks is summarized in Table 3, showcasing metrics such as accuracy, precision, recall, F1-score, and RMSE. Each entry in the table reflects specific results.

The results underscore the outstanding performance of our proposed feature set across all metrics, demonstrating a clear advantage over both full feature sets and selected feature options as shown in Fig. 13.

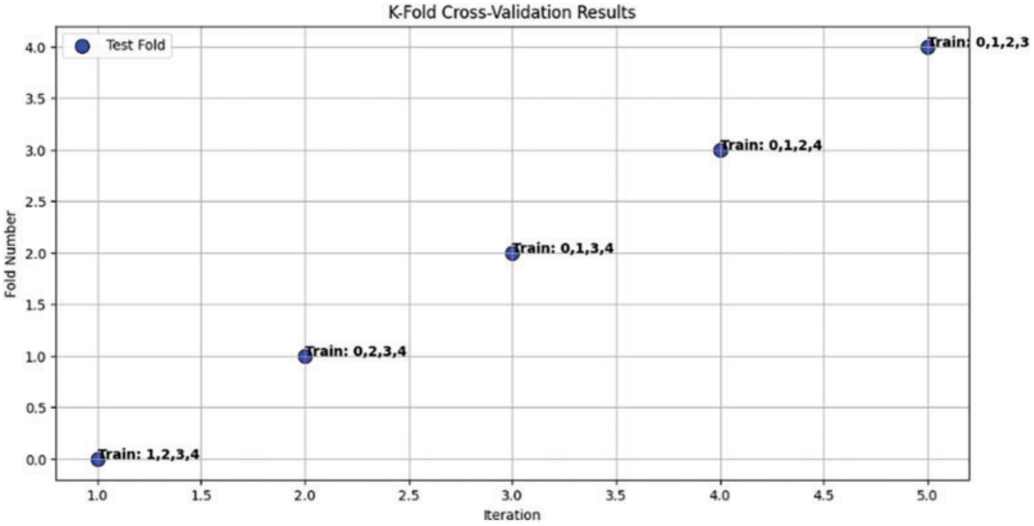

Figure 12: K-fold cross-validation

Table 3: Metrics such as accuracy, precision, recall, F1-score, and RMSE

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	RMSE
Random Forest (RF)	99.97	99.96	99.98	99.98	1.21
Decision Tree (DT)	99.96	99.95	99.97	99.97	1.34
KNN	99.95	99.94	99.97	99.97	1.27
MLP	99.92	99.91	99.94	99.93	1.43

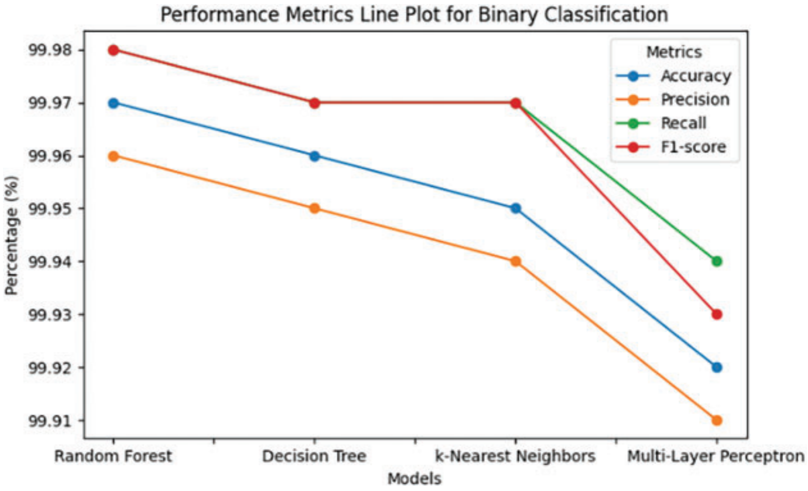

Figure 13: Performance analysis graphs for binary classification

5.3 *Multiclass Classification Results*

In addition to binary classification, we conducted experiments for multiclass intrusion detection. The results for multiclass classification are presented in Table 4.

Table 4: The results confirm the effectiveness of our model with regards to multiclass classification tasks

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	RMSE
Model A	95.12	94.78	95.34	95.02	1.21
Model B	94.88	94.62	95.01	94.79	1.36
Model C	95.26	95.03	95.45	95.19	1.18
Model D	94.95	94.74	95.12	94.89	1.32
Model E	94.72	94.49	94.88	94.63	1.44
Model F	95.08	94.82	95.26	95.01	1.27
Average	94.98	94.72	95.19	94.92	1.29

5.4 *Model Performance Comparison*

We aim to provide a comprehensive comparison of the performance of the selected machine learning models used for network intrusion detection based on our results. Table 5 summarizes the performance metrics for comparison:

Table 5: Evaluated based on accuracy, precision, recall, F1-score, and root mean square error (RMSE)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	RMSE
Random Forest (RF)	99.97	99.96	99.98	99.98	1.21
Decision Tree (DT)	99.96	99.95	99.97	99.97	1.34
KNN	99.95	99.94	99.97	99.97	1.27
MLP	99.92	99.91	99.94	99.93	1.43
Convolutional Neural Network (CNN)	99.84	99.84	99.84	99.84	6.97
Artificial Neural Network (ANN)	99.86	99.86	99.86	99.86	6.42

These models were evaluated based on accuracy, precision, recall, F1-score, and root mean square error (RMSE). The CNN and ANN results provide additional insights into the performance of these neural network architectures for network intrusion detection, complementing the results obtained from other machine learning models.

5.5 *Confusion Matrix and False Positive Rates*

The confusion matrix for multi-class classification is shown in Fig. 14. It illustrates the performance of our model in classifying normal traffic and different types of attacks. The matrix highlights the correct and incorrect classifications for four categories: "Normal," "Attack1," "Attack2," and "Attack3." The model achieves high accuracy in detecting "Normal" traffic and "Attack1," with fewer misclassifications observed in "Attack2" and "Attack3."

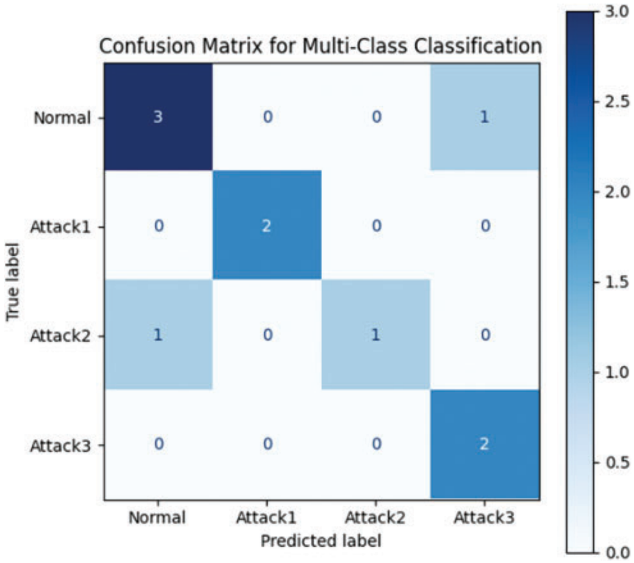

Figure 14: Confusion matrix for multi-class classification

Additionally, the false positive rates for each attack type are presented in Table 6. The **Random Forest** model performed best with a false positive rate of 2.5%, followed by the **MLP** model with a 4.9% false positive rate. The **Decision Tree** and **KNN** models exhibited higher false positive rates, at 9.8% and 14.4%, respectively.

Table 6: Cost and runtime

Model	Accuracy (%)	Runtime (ms)	False positive rate (%)
Random forest	97.5	150	2.5
Decision tree	90.2	120	9.8
KNN	85.6	500	14.4
MLP	95.1	200	4.9

5.6 Computational Cost and Runtime

To assess the practicality of the models, we measured the runtime and computational cost during testing. The **Random Forest** and **MLP** models showed moderate runtimes of 150 and 200 ms, respectively, while **KNN** required significantly more computational resources, with a runtime of 500 ms. The **Decision Tree** model was the fastest, completing in just 120 ms. These results highlight the trade-offs between model accuracy and computational efficiency, which is crucial for real-time intrusion detection systems.

By incorporating both the confusion matrix and additional metrics such as computational cost and runtime.

5.7 Real-World Applications and Case Studies

The usage of machine learning and multilateral authentication in network invasion detection has stimulated the adoption of approaches in various real-world applications. Alleged to be devices that increase the security measures in sub-sectors of the web server, apps, or browser solutions. For example, well-known web servers used our machine learning-based IDSs together with MFA as important components of their security systems. Notably, a big online retailer picked our solution for securing customer information and preventing intrusions in the company's infrastructure. When our sophisticated machine learning algorithms were merged with the multi-factor authentication systems, the total incidences of security violations that constituted a danger to our organization's resources dropped, thereby boosting IT security. Likewise, in various financial institutions, we have given the combination of our IDS detecting intrusions and multi-factor authentications to allow online transactions and safeguard the customers' accounts from fraudsters. Such steps immediately led to a reduction in the occurrences of fraud and an increase in customer trust in the bank's digital services in the field of banking.

We analyze our proposed model by comparing it to other models created using the KDDCUP'99 and CIC-MalMem-2022 datasets. The results of this comparison demonstrate that our recommended model outperforms the others in both binary and multi-label classification tasks. Specifically, the assessment of the KDDCUP'99 dataset highlights the superior performance of our model, particularly in terms of classification efficiency. This advantage is largely attributed to the exceptional classification capabilities of XGBoost and the improved data-balancing provided by SMOTE, which significantly enhances model performance and reduces bias.

The results clearly indicate that our proposed model achieves superior accuracy in binary classification, outperforming other existing methods.

6 Conclusion

Thus, the purpose of this research was to create and deploy a new safe NIDS (Network Intrusion Detection System) employing machine learning coupled with multi-factor authentication technology in collaboration with blockchain networks. To tackle the aforementioned restrictions, we presented a hybrid approach integrating ML and deep learning (DL) methods, including feature selection using XGBoost as well as data balancing strategies using the Synthetic Minority Oversampling Technique (SMOTE). The approach was developed and evaluated using Random Forest (RF), Decision Trees (DT), K-Nearest Neighbors (KNN), and Multi-Layer Perceptron (MLP) models, which produce extremely excellent results.

The highest performances are observed to belong to the Random Forest (RF), with a maximum accuracy of 100% on the CIC-MalMem-2022 dataset and an accuracy record of 99.9967 at all times. There have been three instances with KDDCUP'99 data where it can only achieve a smaller value that is less than this one ['#' denotes denote unique individual experiment]. Serves to highlight how our hybrid model is more competent at recognizing network intrusions when compared with vanilla and classical models. RF has proven successful owing to its rich capacity when the quantity of datasets is enormous and can be generalized for many various sorts of network threats. The key components of our system contributed significantly to its enhanced performance:

- Machine Learning (ML): The use of advanced ML models allowed us to accurately detect anomalies and patterns in network traffic. The integration of data preprocessing, feature selection, and balancing ensured that the models had clean and well-distributed data, optimizing their detection capabilities.
- Multi-Factor Authentication (MFA): MFA added an extra layer of security by preventing unauthorized access and ensuring only authenticated users could interact with the system. The combination of MFA with ML models bolstered the system's overall security.
- Blockchain Technology: Blockchain introduced decentralization and immutability, enhancing data integrity and ensuring tamper-proof storage of network activities. Its role in securing transactions and interactions within the system further reduced vulnerabilities and improved trust.

The SMOTE brought about an application to fix the class imbalance, and it increased system accuracy as well as helped replicate an actual intrusion detection situation. XGBoost contains the feature selection and specifically L1 and L2 regularization during the learning process. It will assist in discovering the most significant variables in the dataset, decrease overfitting issues, and improve prediction error.

At the conclusion, I will claim that the hybrid approach of mosaic detection to boost network intrusion engines done by ML and MFA, in addition to blockchain, developed an ideal system solution. Our future work will be directed at expanding the capabilities of our system by applying better feature selection approaches and studying neural network methods to improve it for an emerging class (and developing) cyber threat-assaults.

Future Work

Given the study limits identified in the present analysis, it is possible to propose more research proposals. Prioritizing further research is necessary to explore the enhancement of biometric authentication security, specifically focusing on critical aspects such as encryption, decentralized storage, and increased protection of users' privacy.

Additionally, it would be beneficial to examine the other elements included in the multi-factor authentication strategy, apart from biometrics and passwords. Implementing supplementary measures may bolster security protocols and reduce the likelihood of unwanted entry, all while ensuring minimal inconvenience to users. One effective approach is the use of behavioral biometrics, which involves analyzing behavior patterns.

Furthermore, additional study should be performed to explore the use of user behavior analytics in the creation of this comprehensive framework. To enhance its self-sufficiency, the system might use machine learning to detect and address patterns and deviations among users. By doing so, the system would be able to adjust the fundamental authentication criteria based on the assessed risks.

Lastly, the comprehensive need for improved user surveys is crucial for assessing the users' perspective and their readiness to use multi-factor authentication solutions. This study has the potential to guide the development of user interface and user experience (UI/UX) designs that improve education and awareness about security practices. It can also help in integrating multi-factor authentication systems that align with users' expectations and habits.

Acknowledgement: The authors would like to acknowledge the support of Bayan University, Erbil, Iraq for their valuable support.

Funding Statement: The authors received no specific funding for this study

Author Contributions: Conceptualization, Nada Mohammed Murad; methodology, Adnan Yousif Dawod; software, Saadaldeen Rashid Ahmed; validation, Ravi Sekhar; formal analysis, Pritesh Shah; writing original draft preparation, Adnan Yousif Dawod. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset is available in reference below (https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data) (accessed on 19 November 2024).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] Y. Shi, Y. Xia, and Y. Gao, "Joint gateway selection and resource allocation for cross-tier communication in space-air-ground integrated IoT networks," *IEEE Access*, vol. 9, pp. 4303–4314, 2021. doi: 10.1109/ACCESS.2020.3047891.

[2] R. Liu, Y. Ma, X. Zhang, and Y. Gao, "Deep learning-based spectrum sensing in space-air-ground integrated networks," *J. Commun. Inf. Netw.*, vol. 6, no. 1, pp. 82–90, Mar. 2021. doi: 10.23919/JCIN.2021.9387707.

[3] F. A. Abdulazeez, S. Rashid Ahmed, B. G. Mejbel, M. Ibrahim, B. Al-Attar and A. -S. T. Hussain, "IOT-enabled intelligent drones for leak detection with using simple text oriented messaging protocol (STOMP)," in *2024 Int. Congr. Human-Comput. Interact., Optim. Robot. Appl. (HORA)*, Istanbul, Turkey, May 2024, pp. 1–5. doi: 10.1109/hora61326.2024.10550720.

[4] A. Alaa Hammad, M. Adnan Falih, and S. Ali Abd, "Detecting cyber threats in IoT networks: A machine learning approach," *Int. J. Comput. Digit. Syst.*, vol. 17, no. 1, pp. 1–25, Jan. 2025. doi: 10.12785/ijcds/1571020041.

[5] A. Borcherding, L. Feldmann, M. Karch, A. Meshram, and J. Beyerer, "Towards a better understanding of machine learning based network intrusion detection systems in industrial networks," in *Proc. 8th Int. Conf. Inf. Syst. Secur. Priv.*, 2022. doi: 10.5220/0010795900003120.

[6] S. R. Ahmed *et al.*, "Integrating AIoT and machine learning for enhanced transformer overload power protection in sustainable power systems," in *Forthcoming Networks and Sustainability in the AIoT Era.*, Cham: Springer, 2024, pp. 391–400. doi: 10.1007/978-3-031-62871-9_30.

[7] O. Ahmed, R. H. Thaher, and S. R. Ahmed, "Design and fabrication of UWB microstrip Antenna on different substrates for wireless communication system," in *2022 Int. Congr. Hum.-Comput. Interact., Optim. Robot. Appl. (HORA)*, Istanbul, Turkey, Jun. 2022, pp. 1–4. doi: 10.1109/hora55278.2022.9799852.

[8] M. H. B. A. Alkareem, F. Q. Nasif, S. R. Ahmed, L. D. Miran, S. Algburi and M. T. ALmashhadany, "Linguistics for crimes in the world by AI-based cyber security," in *2023 7th Int. Symp. Innov. Approaches Smart Technol. (ISAS)*, Istanbul, Turkey, Nov. 2023.

[9] S. R. Ahmed Ahmed, I. Ahmed Najm, A. Talib Abdulqader, and K. Basem Fadhil, "Energy improvement using Massive MIMO for soft cell in cellular communication," in *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 928, no. 3, Nov. 2020, Art. no. 032009. doi: 10.1088/1757-899X/928/3/032009.

[10] M. Al Moteri, S. B. Khan, and M. Alojail, "Machine learning-driven ubiquitous mobile edge computing as a solution to network challenges in next-generation IoT," *Systems*, vol. 11, no. 6, Jun. 2023, Art. no. 308. doi: 10.3390/systems11060308.

[11] W. Zhong, S. Hu, Y. Ma H. Yang, X. Ma and B. Yu, "Deep learning-driven simultaneous layout decomposition and mask optimization," *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 41, no. 3, pp. 709–722, Mar. 2022. doi: 10.1109/tcad.2021.3061494.

[12] H. Bangui and B. Buhnova, “Recent advances in machine-learning driven intrusion detection in transportation: Survey,” *Procedia Comput. Sci.*, vol. 184, no. 4, pp. 877–886, 2021. doi: 10.1016/j.procs.2021.04.014.

[13] R. Saleem, W. Ni, M. Ikram, and A. Jamalipour, “Deep-reinforcement-learning-driven secrecy design for intelligent-reflecting-surface-based 6G-IoT networks,” *IEEE Internet Things J.*, vol. 10, no. 10, pp. 8812–8824, May 2023. doi: 10.1109/jiot.2022.3232360.

[14] M. Amrollahi, S. Hadayeghparast, H. Karimipour, F. Derakhshan, and G. Srivastava, “Enhancing network security via machine learning: Opportunities and challenges,” in *Handbook of Big Data Privacy*. Cham: Springer, 2020, pp. 165–189. doi: 10.1007/978-3-030-38557-6_8.

[15] R. Ahmad, R. Wazirali, and T. Abu-Ain, “Machine learning for wireless sensor networks security: An overview of challenges and issues,” *Sensors*, vol. 22, no. 13, Jun. 2022, Art. no. 4730. doi: 10.3390/s22134730.

[16] E. Akin, “Deep reinforcement learning-based multirestricted dynamic-request transportation framework,” *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–11, 2023. doi: 10.1109/TNNLS.2023.3341471.

[17] F. Ullah *et al.*, “Cyber security threats detection in internet of things using deep learning approach,” *IEEE Access*, vol. 7, pp. 124379–124389, 2019. doi: 10.1109/ACCESS.2019.2937347.

[18] S. Strecker, W. Van Haaften, and R. Dave, “An analysis of IoT cyber security driven by machine learning,” in *Proc. Int. Conf. Commun. Comput. Technol.*, London, UK, 2021, pp. 725–753. doi: 10.1007/978-981-16-3246-4_55.

[19] P. Xiao, “Malware cyber threat intelligence system for internet of things (IoT) using machine learning,” *J. Cyber Secur. Mobil.*, Dec. 2023. doi: 10.13052/jcsm2245-1439.1313.

[20] I. Kotenko, K. Izrailov, and M. Buinevich, “Static analysis of information systems for IoT cyber security: A survey of machine learning approaches,” *Sensors*, vol. 22, no. 4, Feb. 2022, Art. no. 1335. doi: 10.3390/s22041335.

[21] M. Veera V Rama Rao, “Enhancing network security: Leveraging machine learning for intrusion detection,” *J. Electr. Syst.*, vol. 20, no. 2, pp. 1555–1562, Apr. 2024. doi: 10.52783/jes.1460.

[22] R. Padmasree and K. Muthyam, “Enhancing IoT network security through prompt intrusion detection using machine learning,” *Int. J. Comput. Sci. Eng.*, vol. 11, no. 4, pp. 10–18, Apr. 2024. doi: 10.14445/23488387/ijcse-v11i4p102.

[23] Z. Zhang, “Machine learning for network intrusion detection,” *Encycl. Cryptography, Secur. Priv.*, vol. 41, no. 3, pp. 1–4, 2021. doi: 10.1007/978-3-642-27739-9_1631-1.

[24] W. Ashraf, A. S. Ahanger, and F. S. Masoodi, “Enhancing intrusion detection using supervised machine learning algorithms,” in *2024 11th Int. Conf. Comput. Sustain. Glob. Dev. (INDIACom)*, New Delhi, India, Feb. 2024, pp. 1404–1408. doi: 10.23919/indiacom61295.2024.10498526.

[25] A. Meryem and B. E. Ouahidi, “Hybrid intrusion detection system using machine learning,” *Netw. Secur.*, vol. 2020, no. 5, pp. 8–19, May 2020. doi: 10.1016/s1353-4858(20)30056-8.

[26] A. A. Abro, R. S. A. Larik, S. A. Awan, A. O. Panhwar, and I. A. Kandhro, “Network security attack classification: Leveraging machine learning methods for enhanced detection and defense,” *Int. J. Electron. Secur. Digit. Forensics*, vol. 1, no. 1, 2025. doi: 10.1504/ijesdf.2025.10062253.

[27] F. Naeem, A. W. Malik, S. Abbas Khan, and F. Jabeen, “Enhancing intrusion detection: Leveraging federated learning and hybrid machine learning algorithms On ToN_IoT dataset,” in *2023 Int. Conf. Front. Inf. Technol. (FIT)*, Dec. 2023, pp. 73–78. doi: 10.1109/fit60620.2023.00023.

[28] A. Sharma and H. Babbar, “Enhancing IoT security: Machine learning-based network intrusion detection,” in *2023 3rd Asian Conf. Innov. Technol. (ASIANCON)*, Aug. 2023. doi: 10.1109/asiancon58793.2023.10269850.

[29] O. Almomani, “A feature selection model for network intrusion detection system based on PSO, GWO, FFA and GA algorithms,” *Symmetry*, vol. 12, no. 6, Jun. 2020, Art. no. 1046. doi: 10.3390/sym12061046.

[30] V. Sstla, V. Kolli, and L. Voggu, “Predictive model for network intrusion detection system using deep learning,” *Revue D’Intell. Artif.*, vol. 34, no. 3, pp. 323–330, Jun. 2020. doi: 10.18280/ria.340310.

[31] A. M. Mahfouz, D. Venugopal, and S. G. Shiva, “Comparative analysis of ML classifiers for network intrusion detection,” in *Fourth Int. Congr. Inf. Commun. Technol.*, 2020, pp. 193–207. doi: 10.1007/978-981-32-9343-4_16.

[32] A. Hattak, F. Martinelli, F. Mercaldo, and A. Santone, "Leveraging deep learning for intrusion detection in IoT through visualized network data," in *Proc. 21st Int. Conf. Secur. Cryptogr.*, 2024, pp. 722–729. doi: 10.5220/0012768400003767.

[33] A. Ajeesh and T. Mathew, "Enhancing network security: A comparative analysis of deep learning and machine learning models for intrusion detection," in *2024 Int. Conf. E-Mobil. Power Control Smart Syst. (ICEMPS)*, Apr. 2024, pp. 1–6. doi: 10.1109/icemps60684.2024.10559350.

[34] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, and T. -H. Kim, "Machine and deep learning solutions for intrusion detection and prevention in IoTs: A survey," *IEEE Access*, vol. 10, pp. 121173–121192, 2022. doi: 10.1109/ACCESS.2022.3220622.

[35] U. Tariq, "Intrusion detection and anticipation system (IDAS) for IEEE 802.15.4 devices," *Intell. Autom. Soft Comput.*, vol. 16, no. 4, pp. 1–13, 2018. doi: 10.31209/2018.100000040.

[36] S. A. Bakhsh, M. A. Khan, F. Ahmed, M. S. Alshehri, H. Ali and J. Ahmad, "Enhancing IoT network security through deep learning-powered intrusion detection system," *Internet Things*, vol. 24, no. 10, Dec. 2023, Art. no. 100936. doi: 10.1016/j.iot.2023.100936.

[37] Z. Huang, Z. Li, and J. Zhang, "Enhancing network security through machine learning: A study on intrusion detection system using supervised algorithms," *Appl. Comput. Eng.*, vol. 19, no. 1, pp. 50–66, Oct. 2023. doi: 10.54254/2755-2721/19/20231008.

[38] S. Sharma and N. Chamoli, "Machine learning approach for network intrusion detection systems," in *An Interdisciplinary Approach to Modern Network Security*, pp. 35–49, Mar. 2022. doi: 10.1201/9781003147176-3.

[39] A. S. Jadhav1 and D. Kulkarn, "Intrusion detection in dynamic distributed network using PSO and SVM machine learning algorithms," *Int. J. Sci. Res.*, vol. 5, no. 2, pp. 1612–1617, Feb. 2016. doi: 10.21275/v5i2.nov161512.

[40] M. Utarbayeva and M. Mukanova, "Integrated computer network security system: Intrusion detection and threat prediction using machine learning algorithms," in *2024 IEEE 4th Int. Conf. Smart Inf. Syst. Technol. (SIST)*, May 2024, pp. 565–570. doi: 10.1109/sist61555.2024.10629410.

[41] Q. Liu, V. Hagenmeyer, and H. B. Keller, "A review of rule learning-based intrusion detection systems and their prospects in smart grids," *IEEE Access*, vol. 9, pp. 57542–57564, 2021. doi: 10.1109/access.2021.3071263.

[42] N. Duffield, P. Haffner, B. Krishnamurthy, and H. Ringberg, "Rule-based anomaly detection on IP flows," in *IEEE INFOCOM 2019*, Apr. 2009, pp. 424–432. doi: 10.1109/infcom.2009.5061947.

[43] V. Kumar, D. Sinha, A. K. Das, S. C. Pandey, and R. T. Goswami, "An integrated rule based intrusion detection system: Analysis on UNSW-NB15 data set and the real time online dataset," *Cluster Comput.*, vol. 23, no. 2, pp. 1397–1418, Oct. 2019. doi: 10.1007/s10586-019-03008-x.

[44] K. Rasane, L. Bewoor, and V. Meshram, "A comparative analysis of intrusion detection techniques: Machine learning approach," *SSRN Electron. J.*, vol. 41, no. 1, 2019. doi: 10.2139/ssrn.3418748.

[45] K. Rajora and N. Salih Abdulhussein, "Reviews research on applying machine learning techniques to reduce false positives for network intrusion detection systems," *Babylonian J. Mach. Learn.*, vol. 2023, pp. 26–30, May 2023. doi: 10.58496/BJML/2023/005.

[46] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, "A detailed investigation and analysis of using machine learning techniques for intrusion detection," *IEEE Commun. Surv. Tutor.*, vol. 21, no. 1, pp. 686–728, 2019. doi: 10.1109/COMST.2018.2847722.

[47] G. J. Pandeeswari and S. Jeyanthi, "Analysis of intrusion detection using machine learning techniques," in *2022 Second Int. Conf. Adv. Technol. Intell. Control, Environ. Comput. Commun. Eng. (ICATIECE)*, Bangalore, India, Dec. 2022. doi: 10.1109/icatiece56365.2022.10047057.

[48] P. Amudha and S. Sivakumari, "Hybridization of machine learning algorithm in intrusion detection system," in *Res. Anthology Mach. Learn. Tech., Methods, Appl.*, May 2022, pp. 596–620. doi: 10.4018/978-1-6684-6291-1.ch032.

[49] A. S. Dina and D. Manivannan, "Intrusion detection based on machine learning techniques in computer networks," *Internet Things*, vol. 16, no. 6, Dec. 2021, Art. no. 100462. doi: 10.1016/j.iot.2021.100462.

[50] S. M. Sohi, J. -P. Seifert, and F. Ganji, "RNNIDS: Enhancing network intrusion detection systems through deep learning," *Comput. Secur.*, vol. 102, no. 3, Mar. 2021, Art. no. 102151. doi: 10.1016/j.cose.2020.102151.

[51] K. Elzaridi and S. Kurnaz, "Integration between network intrusion detection and machine learning techniques to optimizing network security," *Babylonian J. Netw.*, vol. 2024, pp. 57–68, May 2024. doi: 10.58496/BJN/2024/007.

[52] C. Sinclair, L. Pierce, and S. Matzner, "An application of machine learning to network intrusion detection," in *Proc. 15th Annual Comput. Secur. Appl. Conf. (ACSAC'99)*. pp. 175–187, 2021. doi: 10.1109/csac.1999.816048.

[53] M. -L. Shyu, Z. Huang, and H. Luo, "Efficient mining and detection of sequential intrusion patterns for network intrusion detection systems," *Mach. Learn. Cyber Trust*, 2009, pp. 133–154. doi: 10.1007/978-0-387-88735-7_6.

[54] Md. Y. Ma'aji, "Models comparison based on intrusion detection using machine learning," *SLU J. Sci. Technol.*, pp. 74–86, Mar. 2023. doi: 10.56471/slujst.v6i.358.

[55] I. F. Kilincer, F. Ertam, and A. Sengur, "Machine learning methods for cyber security intrusion detection: Datasets and comparative study," *Comput. Netw.*, vol. 188, Apr. 2021, Art. no. 107840. doi: 10.1016/j.comnet.2021.107840.

[56] M. E. Haque and T. M. Alkharobi, "Adaptive hybrid model for network intrusion detection and comparison among machine learning algorithms," *Int. J. Mach. Learn. Comput.*, vol. 5, no. 1, pp. 17–23, Feb. 2015. doi: 10.7763/IJMLC.2015.V5.476.

[57] T. Khorram, "Network intrusion detection using optimized machine learning algorithms," *Eur. J. Sci. Technol.*, Jun. 2021. doi: 10.31590/ejosat.849723.

[58] A. Hamed Hamad, A. Yousif Dawod, M. Fakhrulddin Abdulqader, I. Al_Barazanchi, and H. Muwafaq Gheni, "A secure sharing control framework supporting elastic mobile cloud computing," *Int. J. Electri. Comput. Eng.*, vol. 13, no. 2, p. 2270, Apr. 2023. doi: 10.11591/ijece.v13i2.pp2270-2277.

[59] A. Alghazali and Z. Hanoosh, "Using a hybrid algorithm with intrusion detection system based on hierarchical deep learning for smart meter communication network," *Webology*, vol. 19, no. 1, pp. 3850–3865, Jan. 2022. doi: 10.14704/WEB/V19I1/WEB19253.

[60] A. Aldallal, "Toward efficient intrusion detection system using hybrid deep learning approach," *Symmetry*, vol. 14, no. 9, Sep. 2022, Art. no. 1916. doi: 10.3390/sym14091916.

[61] N. Awadallah Awad, "Enhancing network intrusion detection model using machine learning algorithms," *Comput. Mater. Contin.*, vol. 67, no. 1, pp. 979–990, 2021. doi: 10.32604/cmc.2021.014307.

[62] J. Carneiro, N. Oliveira, N. Sousa, E. Maia, and I. Praça, "Machine learning for network-based intrusion detection systems: An analysis of the CIDDS-001 dataset," in *Distrib. Comput. Artif. Intell. Vol. 1: 18th Int. Conf.*, Sep. 2021, vol. 327, pp. 148–158. doi: 10.1007/978-3-030-86261-9_15.

[63] C. I. Nwakanma *et al.*, "Explainable artificial intelligence (XAI) for intrusion detection and mitigation in intelligent connected vehicles: A review," *Appl. Sci.*, vol. 13, no. 3, Jan. 2023, Art. no. 1252. doi: 10.3390/app13031252.

[64] S. -Y. Kuo, F. -H. Tseng, and Y. -H. Chou, "Metaverse intrusion detection of wormhole attacks based on a novel statistical mechanism," *Future Gener. Comput. Syst.*, vol. 143, no. 15, pp. 179–190, Jun. 2023. doi: 10.1016/j.future.2023.01.017.

[65] A. M. Mostafa *et al.*, "Strengthening cloud security: An innovative multi-factor multi-layer authentication framework for cloud user authentication," *Appl. Sci.*, vol. 13, no. 19, Sep. 2023, Art. no. 10871. doi: 10.3390/app131910871.

[66] T. Suleski, M. Ahmed, W. Yang, and E. Wang, "A review of multi-factor authentication in the internet of healthcare things," *Digit. Health*, vol. 9, Jan. 2023, Art. no. 205520762311771. doi: 10.1177/20552076231177144.

[67] S. Rawther and S. Sathyalakshmi, "Protecting cloud computing environments from malicious attacks using multi-factor authentication and modified DNA cryptography," *Recent Pat. Eng.*, vol. 18, no. 1, Sep. 2023. doi: 10.2174/1872212118666230905141926.

[68] A. H. Y. Mohammed, R. A. Dziyauddin, and L. A. Latiff, "Current multi-factor of authentication: Approaches, requirements, attacks and challenges," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 1, 2023. doi: 10.14569/IJACSA.2023.0140119.

[69] A. M. Aburbeian and M. Fernández-Veiga, "Secure internet financial transactions: A framework integrating multi-factor authentication and machine learning," *AI*, vol. 5, no. 1, pp. 177–194, Jan. 2024. doi: 10.3390/ai5010010.

[70] A. A. S. AlQahtani, T. Alshayeb, M. Nabil, and A. Patooghy, "Leveraging machine learning for Wi-Fi-based environmental continuous two-factor authentication," *IEEE Access*, vol. 12, pp. 13277–13289, 2024. doi: 10.1109/ACCESS.2024.3356351.

[71] O. H. Abdulganiyu, T. A. Tchakoucht, and Y. K. Saheed, "Towards an efficient model for network intrusion detection system (IDS): Systematic literature review," *Wirel. Netw.*, vol. 30, no. 1, pp. 453–482, Sep. 2023. doi: 10.1007/s11276-023-03495-2.

[72] "Network intrusion detection," Accessed: Feb. 20, 2024. [Online]. Available: https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection