Machine and deep learning classifiers for binary and multi-class network intrusion detection systems

Ahmad Aloqaily[1], Emad Eddien Abdallah[1], Esraa Abu Elsoud[2], Yazan Hamdan[3], Khaled Jallad[3]
[1]Department of Information Technology, Faculty of Prince Al-Hussein Bin Abdullah II for Information Technology,
The Hashemite University, Zarqa, Jordan
[2]Department of Cybersecurity and Cloud Computing, Faculty of Information Technology, Applied Science Private University,
Amman, Jordan
[3]Department of Computer Information Systems, Faculty of Prince Al-Hussein Bin Abdullah II for Information Technology,
The Hashemite University, Zarqa, Jordan

Article Info

Article history:

Received Sep 23, 2024
Revised Jun 24, 2025
Accepted Oct 18, 2025

Keywords:

Cyber attacks
Cyber security
Deep learning
Intrusion detection
Machine learning

ABSTRACT

The rapid proliferation of the internet and advancements in communication technologies have significantly improved networking and increased data volume. This phenomenon has subsequently caused a multitude of novel attacks, thereby presenting significant challenges for network security in the intrusion detection system (IDS). Moreover, the ongoing threat from authorized entities who try to carry out various types of attacks on the network is a concern that must be handled seriously. IDS are used to provide network availability, confidentiality, and integrity by employing machine learning (ML) and deep learning (DL) algorithms. This research aimed to study the impacts of the binary and multi-attack instances label by establishing IDS that leverages hybrid algorithms, including artificial neural networks (ANN), random forest (RF), and logistic model trees (LMTs). The paper addresses challenges such as data preprocessing, feature selection, and managing imbalanced datasets by applying synthetic minority oversampling technique (SMOTE) and Pearson's correlation methodologies. The IDS was tested using network security laboratory knowledge discovery datasets (NSL-KDD) and catalonia independence corpus intrusion detection system (CIC-IDS-2017) datasets, achieving an average F1-score of 96% for binary classification on NSL-KDD and 85% for binary classification on CIC-IDS-2017, while for multi-classification, the proposed model achieved an average F1-score of 82% and 96% for NSL-KDD and CIC-IDS-2017 successively.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Ahmad Aloqaily
Department of Information Technology
Faculty of Prince Al-Hussein Bin Abdullah II for Information Technology, The Hashemite University
P.O. Box 330127, Zarqa 13133, Jordan
Email: aloqaily@hu.edu.jo

1. INTRODUCTION

The exponential increase in internet usage in daily life has led to an increase in cyberattacks, such as the SolarWinds breach in 2020 have highlighted the increasing sophistication of network intrusions. According to the Internet Security Threat Report (ISTR), malware is found in one of every thirteen Web queries'. A cyberattack starts with target reconnaissance and ends with using vulnerabilities to carry out a harmful operation. These cyberattacks result in system intrusions, which are characterized as unauthorized system access that compromises the confidentiality, integrity, and availability (CIA) of security measures protecting computer or network resources. In recent years, we have seen the emergence of numerous new cyberattacks, including cross-site scripting, brute force, botnets, distributed denial of service, and others, where in 2023, the worldwide number of malware attacks reached 6.06 billion, an increase of 10% compared to the preceding year [1]. These intrusions raised more serious than ever concerns regarding cybersecurity [2]. However, securing the networks becomes essential; one of the most effective ways to identify these threats is intrusion detection system (IDS), which depends on analyzing and monitoring the network traffic.

A host intrusion detection system (HIDS) is an IDS approach that uses system activities that appear through a variety of log files created on the local host computer to identify possible intrusions, whereby these log files are collected through local sensors [3]. On the other hand, a network intrusion detection system (NIDS) analyzes the contents of packets within network traffic streams, whereas HIDS primely employs data derived from log files, system logs, sensor logs, file system data, disk resource allocation, and other relevant information from each system. Many organizations use a hybrid approach that combines both NIDS and HIDS techniques [4]. The employment of stateful protocol analysis, anomaly detection, and signature detection techniques are used for analyzing network traffic flows. Signature detection depends on human involvement to refresh the signature database continuously and uses pre-established signatures and filtration algorithms to identify attacks. This methodology works well for identifying known threats, but it is completely ineffective against unknown attacks. However, anomaly detection often leads to a significantly higher percentage of false positives. Most organizations choose to apply hybrid approaches to get a more effective detection model [5]. Depending on the standard framework of communication TCP/IP model, analysis protocols on the network, application, and transport layers are the most powerful techniques to detect any potential threats [6].

Machine learning (ML) methods have shown excellence in achieving high detection accuracy. Although there are some limitations, such as handling raw, unlabeled, high dimensional data and manual feature extraction, these limitations affect the accuracy of IDS [7], to address these drawbacks, deep learning (DL) emerged. This research aims to enhance security through IDSs by applying both ML and DL algorithms to network security laboratory knowledge discovery datasets (NSL-KDD) and catalonia independence corpus intrusion detection system (CIC-IDS-2017) datasets to improve overall system architecture and detection performance. These datasets provide a foundation for benign and attack network traffic, although they have shortcomings such as labeling issues, duplicate flows, and insufficient attack variation. The proposed model in this research seeks to address these limitations and develop a more resilient IDS, by applying a comprehensive experiment including two phases, study the effects of binary class and multi-class in the performance of the IDS. Where several researchers highlight this issue due to its importance in the performance of the IDS [8], [9]. Additionally, we identify the major gap in the literature regarding the integration of ML and DL approaches in the context of HIDS and NIDS. While previous studies have examined the effectiveness of different detection strategies, they often do not explicitly investigate how detection accuracy could be improved by combining ML and DL to work together on different attack vectors. This is especially true when it comes to reducing high false positive rates and the challenges that come with data labeling.

The remainder of this paper is organized as follows: section 2 reviews recent studies on IDS. Section 3 outlines the research methodology. Section 4 presents the results and discussion. Lastly, section 5 contains the conclusion.

2. LITERATURE REVIEW

Many researchers studied IDS by proposing different approaches where the IDS could be different in technology used, the dataset, feature selection techniques, and many more criteria that affect the performance of the proposed model. In this section, we will illustrate these differences by mentioning some of these studies. The poor performance of conventional intrusion detection techniques prompted the research in [10] to suggest a neural network methodology. A multi-layer convolutional neural network (CNN) is used for feature extraction and selection. To categorize the network attacks, a soft-max classifier is used. To do additional analysis, a multi-layer deep neural network (DNN) is utilized for network intrusions. Two commonly utilized benchmark intrusion detection datasets, NSL-KDD and KDDCUP'99, have been used in the research investigations. Four performance metrics—accuracy, recall, F1-score, and precision—are used to evaluate the suggested model's performance. Comparing the suggested method to other IDSs, the testing findings demonstrate that it attained an accuracy of 99%.

The research in [11] examined the applicability of DL to internet of things (IoT) data security and conducted a comparison analysis using three DL models, including CNN, long short-term memory (LSTM), and DNN. Based on the results, DNN achieves 94.61% accuracy, while CNN and LSTM achieve 98.61% and 97.67%, respectively. It has been established through this comparative study and literature review that DL models perform better in the IoT IDS setting than other approaches. Although the DL models exhibit better accuracy, their future work should focus on creating a hybrid DL model for IoT ID that can anticipate attacks more accurately while experimenting with real-time datasets. The hybrid model is used for IoT IDS installation strategy and detection techniques.

Patil *et al.* [12] presented an IDS model that enables the use of ML algorithms like support vector machine (SVM), random forests (RF), and decision trees. Following the model's training, an ensemble method known as a voting classifier was included, and it was able to attain 96.25% accuracy. The study suggests that trust is necessary for human-machine interactions to be productive. Local interpretable model-agnostic explanation (LIME) is an extendable, modular technique that provides concise, comprehensible descriptions of predictions. An explanation of prediction is highly useful for the selection of representative models. It is employed in model selection, trust evaluation, model improvement for unreliable models, and prediction analysis for both system experts and non-experts. To comprehend the model's prediction, the paper suggests deploying a LIME explainable framework after employing an ensemble of ML models. The ensemble of ML models showed an improved accuracy of 96.25%.

Meng [13] examines the use of supervised and unsupervised learning methods to improve cybersecurity threat detection accuracy in his research. Additionally, the study emphasizes the use of reinforcement learning in adaptive threat modeling. The approach helps systems discover the best methods to respond to threats, making them more adaptive to changing cyber threats. The article also addresses real-time threat identification using neural networks and DL algorithms.

Hnamte and Hussain [14] describe an advanced and efficient network-based NIDS that uses DL techniques to detect attacks. CIC-IDS-2018 and Edge_IIoT are two real-time datasets on which the model has been painstakingly trained. Multiclass classification is used to examine the model's performance, and the results show remarkable accuracies of 100% and 99.64%. In contrast, Qazi *et al.* [15] implemented a hybrid DL-based NIDS, which leverages neural network architectures, applying it to the CIC-IDS-2018 dataset, and attained an accuracy of 98.9%. Musleh *et al.* [16] seek to present a comprehensive study on ML-based IDS within the IoT context, employing various feature extraction techniques and ML algorithms to enhance their proposed model. The investigation evaluates an array of feature extractors, including image filtering techniques and transfer learning frameworks. The study culminates in an assessment utilizing the IEEE Dataport dataset, achieving an accuracy rate of 98.3%.

Moving to research that focuses on the effectiveness of binary and multi-class in IDS, Acharya *et al.* [8] create a unique and reliable heterogeneous ensemble ML model, to identify abnormalities in NIDS. To address the class-imbalance issue with NIDS datasets, the suggested model initially uses subsampling. Then, applying the Min-Max technique for normalization translated the input data into the 0–1 range, reducing overfitting and promoting convergence. Often employed in meta-heuristic-based techniques, feature reduction is utilized to decrease the features while retaining the most appropriate features and avoiding computational overheads. To accomplish both two-class and multi-class classification across feature-selected NSL-KDD, KDD99, and UNSW-NB-15 datasets, the suggested NIDS approach ultimately created a heterogeneous ensemble learning model using J48, k-nearest neighbors (k-NN), SVM, Bagging, AdaBoost, and RF algorithms as base-classifiers. Bacevicius and Taraseviciene [17] aims to address the difficulties that arise when testing multi-class classification performance for network intrusions using highly imbalanced raw data, such as the CIC-IDS-2017 and CSE-CIC-IDS-2018 datasets. The main objective of the study is to examine several ML models, such as CNNs, artificial neural networks (ANN), RF, decision trees, and logistic regression. It also uses explainable artificial intelligence (XAI) tools to examine potential interpretations of the data. With an average macro F1-score of 0.96878, the results showed that decision trees using the classification and regression trees (CART) strategy performed better than other methods on the 28-class classification task.

Tseng and Chang [18] presented an ensemble feature selection framework that combines three feature scoring techniques—classification and regression tree, random forest, and extra tree—with two different feature selection methodologies to produce six distinct feature sets. The framework determines the best feature set based on accuracy for each binary model. By utilising random sampling and offering a customised sample size based on the target class dimensions in each binary model, the proposed ensemble data balancing technique significantly enhances conventional data balancing approaches. Random sampling, the synthetic minority oversampling technique (SMOTE), and Tomek Link methods are all included in this framework. It also incorporates four encoder modes to identify the best feature extraction configuration for each binary model. Experimental findings demonstrate that ensemble binary detection models achieve higher accuracy in identifying three types of wireless attacks in the Aegean Wi-Fi intrusion dataset (AWID) compared to similar studies using traditional multi-class detection frameworks [18].

In addition, a data resampling method based on the adaptive synthetic (ADASYN) and Tomek links algorithms is presented in [2], combined with several DL models. Using the benchmark NSL-KDD dataset, the proposed model is evaluated through accuracy, precision, recall, and F-score metrics. Experimental results indicate that the approach achieves 99.8% accuracy in binary classification, outperforming existing models. Its performance in multi-class classification also improves, surpassing state-of-the-art accuracy levels of 99.9%.

3. RESEARCH METHODOLOGY

Our proposed methodology consists of five phases illustrated in Figure 1. We used two datasets, NSL-KDD [19] and CIC-IDS-2017 [20], the data were cleaned by removing the noise instances and any duplicated data. The third phase aimed to convert features into numerical data using an ordinal encoder. To ensure that features are treated equally during the training phase, MinMax scaling scales data in the standard range between 0 and 1. Furthermore, we used Pearson's correlation coefficient to evaluate the linear relationships between features in both the NSL-KDD and CIC-IDS-2017 datasets. A correlation coefficient threshold of 0.8 (in absolute value) was chosen to identify highly correlated features. Features with correlation coefficients greater than this threshold were considered redundant and removed, as they did not provide additional information for model training. This threshold was selected to balance between reducing dimensionality and retaining informative features. To ensure that features were treated equally during model training, we applied MinMax scaling to scale all features to the range [0, 1]. In our study, we applied SMOTE after feature selection to ensure that the generated synthetic data was based on relevant features. The technique was crucial in improving the classifier's performance, particularly for detecting rare attack types in the NSL-KDD and CIC-IDS-2017 datasets, which were otherwise underrepresented. SMOTE is a powerful technique used to address class imbalance by generating synthetic samples for the underrepresented class. The algorithm works by selecting a sample from the minority class, finding its k-NN, and then creating synthetic instances by interpolating between the selected sample and its neighbors. This approach helps to increase the decision boundary complexity for the minority class, thus improving the classifier's ability to distinguish between the classes.

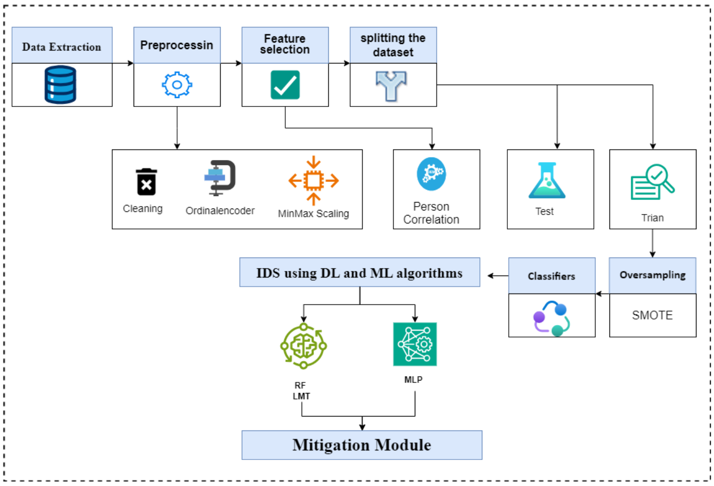


Figure 1. Proposed methodology

According to Table 1 and Figure 2 Benign traffic is disproportionately predominant (454,495 instances), indicating a major imbalance in the distribution. There have been a lot of DDoS attacks (25,545 instances), DoS Hulk attacks (45,887 instances), and PortScan attacks (31,702 instances), but very few Heartbleed attacks (2 occurrences), infiltration attacks (9 instances), and SQL injection attacks (5 instances). Because there is insufficient data to train algorithms, this mismatch makes it difficult to classify attacks accurately, especially for under-represented attack types. Overall, the approach exceeds other classifiers in binary and multi-class cases, especially when applied to handling rare attack types, making it the most effective model for the dataset.

Table 1. Number of instances for each attacks type

Standards	Number of Instances
BENIGN	454495
Bot	388
DDoS	25545
DoS GoldenEye	2020
DoS Hulk	45887
DoS slowhttptes	1140
DoS slowloris	1180
FTP-Patator	1620
Heartbleed	2
Infiltration	9
PortScan	31702
SSH-Patator	1164
Brute force	281
Sql injection	5
XSS	142

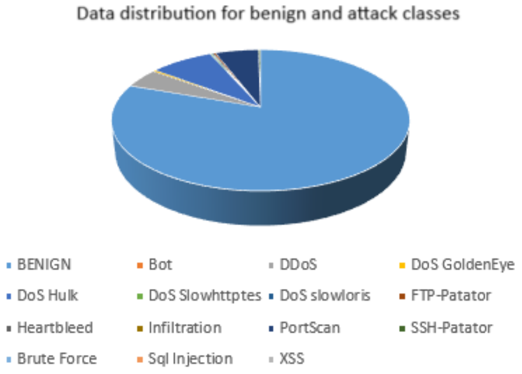


Figure 2. Data distribution for BENIGN and attack classes

Finally, we applied different classification algorithms to the train data to evaluate the proposed model performance. The methodology phases can be outlined in these five steps:

i) Data extraction: the experiments conducted depend on two datasets, NSL-KDD and CIC-IDS-2017. Table 2 summarizes the used dataset. For model training and evaluation, we divided the datasets into training and test sets. In the case of the NSL-KDD dataset, we used 80% of the instances for training (100,000 instances) and reserved the remaining 20% (25,000 instances) for testing. Similarly, for the CIC-IDS-2017 dataset, 80% of the instances (2,264,594) were allocated for training, and the remaining 20% (566,149 instances) were used for testing.

Table 2. Summary of NSL-KDD and CIC-IDS-2017 datasets

Dataset name	Number of instances	Number of features	Attack
NSL-KDD	125,000	41	DOS, Probe, R2L and U2R
CIC-IDS-2017	2,830,743	79	Brute force FTP, Brute force SSH, DoS, Heartbleed, Web attack, infiltration, Botnet, and DDoS

ii) Preprocessing: the initial step in the preparation of data is to remove constant features that add no meaningful value to the dataset. Subsequently, data encoding is applied to convert non-numeric properties into numeric representations. This is especially useful for ordinal data, which are categorical data with a particular hierarchy. After encoding, the data is normalized using the MinMaxScaler, which scales features to a predetermined range (usually 0-1) while preserving the structure of the original distribution [21]. By ensuring that every variable contributes equally to the model, this normalization helps to prevent bias and improves the stability and speed of DL and ML algorithms during training. The MinMaxScaler operates by applying (1) to feature values to fit them into the specified range [22].

$$X_{\text{scaled}} = \frac{X - \min(X)}{\max(X) - \min(X)} \tag{1}$$

iii) Feature selection: Pearson's correlation coefficient is used to determine the correlations between the variables in the datasets to select features. This statistical tool produces a correlation coefficient that ranges from -1 to +1 by evaluating the linear relationship between two continuous variables [23], [24]. When a coefficient is close to ±1, it represents a strong linear link; when it is close to 0, it denotes no linear association. The methodology assumes that the variables involved have a normal distribution, are independent, and are linear. Table 3 shows the features that were identified based on the chosen algorithm.

Table 3. Top features from NSL-KDD and CIC-IDS-2017 Datasets

NSL-KDD features		CIC-IDS-2017 features	
duration	protocol_type	Flow IAT Std	Max Packet Length
service	flag	Init_Win_bytes_forward	act_data_pkt_fwd
src_bytes	dst_bytes	SubflowFwdBytes	TotalBackwardPackets
land	wrong_fragment	Flow IAT Mean	ACK Flag Count
urgent	hot	Avg Bwd Segment Size	URG Flag Count
num_failed_logins	logged_in	Fwd Packet Length Max	ECE Flag Count
num_compromised	root_shell	Packet LengthStd	IdleMean
su_attempted	num_root	Init_Win_bytes_backward	PacketLength Mean
num_file_creations	num_shells	RST Flag Count	Fwd Header Length
num_access_files	is_host_login	Bwd Packet Length Max	min_seg_size_forward
is_guest_login	count	IdleMax	BwdPackets/s
srv_count	serror_rate	TotalFwd Packets	Fwd Packet Length Mean
srv_serror_rate	rerror_rate	Fwd Header Length.1	Fwd Packet Length Std
srv_rerror_rate	same_srv_rate	PSH Flag Count	Fwd IAT Max
diff_srv_rate	srv_diff_host_rate	Active Mean	Idle Min
dst_host_count	dst_host_srv_count	Bwd Packet Length Mean	Average Packet Size
dst_host_same_srv_rate	dst_host_diff_srv_rate	Fwd PSH Flags	Total Length of Fwd Packets
dst_host_same_src_port_rate	dst_host_srv_diff_host_rate	Fwd IAT Std	Flow IAT Max
dst_host_serror_rate	dst_host_srv_serror_rate	Bwd Packet Length Std	Avg Fwd Segment Size
dst_host_rerror_rate	dst_host_srv_rerror_rate	Flow Packets/s	Down/Up Ratio
		Destination Port	Packet Length Variance
		Subflow Fwd Packets	SYN Flag Count

iv) Over-sampling: the datasets' class imbalance was solved using the SMOTE. Rather than just copying samples from the existing dataset, this technique generates new, synthetic samples. We specifically employed SMOTE to reduce the size of the CIC-IDS-2017 dataset to 24,607,475 instances and the NSL-KDD dataset to 308,830 instances.

v) Classifiers: in the context of ML, a classifier is an algorithm that automatically sorts or groups data into one or more ”classes.” Data is categorized or classified according to specific features [25]. In this research, we have used three classifiers: multi-layer perceptron (MLP), RF, and logistic model trees (LMTs).

- MLP is a type of ANN that consists of multiple layers of interconnected nodes, called neurons. It is one of the simplest and most used neural network architectures [26]. For binary classification tasks, the output layer of the MLP typically uses the sigmoid activation function. This activation function outputs a value between 0 and 1, which can be interpreted as the probability of the instance belonging to one of the classes. A threshold of 0.5 is commonly used to assign the class label: values above 0.5 are classified as class 1, and values below 0.5 as class 0. In the hidden layers, rectified linear unit (ReLU) is often employed to introduce non-linearity, helping the model to learn complex patterns in the data.

– RF is one of the popular ML algorithms that belong to the ensemble learning category. It is used for both classification and regression tasks and is based on the concept of decision tree [27].
– LMTs combine decision tree structures with logistic regression functions. A logistic regression model is stored in each leaf node of the LMT and is used to categorize occurrences that fall into the appropriate region. LMTs divide the instance space into discrete regions, each represented by a leaf node with a logistic regression function on it [28].

4. RESULTS AND DISCUSSION

We evaluate the classification models using binary and multi-class labels to identify the most effective IDS model. In the binary classification setup, all attack instances are labeled as 1, and all normal instances are labeled as 0. On the other hand, for multi-classification the targeted attack instances are labeled as 1, other types of attack instances are labeled as 2, and all normal instances are labeled as 0. We illustrate the effect of the label on accuracy by performing an extensive performance analysis of the models on the NSL-KDD 2017 and CIC-IDS 2017 datasets. The results of applying the selected classifiers mentioned in the previous section which applied to the NSL-KDD dataset are presented in Table 4.

Table 4. Performance metrics for binary and multi-class NSL-KDD

Classifiers	Performance metrics	Binary NSL-KDD	Multi class NSL-KDD			
			U2R	Dos	R2L	propel
MLP	Precision	0.99	0.84	0.99	0.24	0.99
	Recall	0.99	0.94	0.99	0.43	0.99
	F1-score	0.99	0.89	0.99	0.31	0.99
RF	Precision	0.99	0.91	0.99	0.65	0.99
	Recall	0.99	0.94	0.99	0.62	0.99
	F1-score	0.99	0.93	0.99	0.63	0.99
LMT	Precision	0.88	0.34	0.98	0.02	0.76
	Recall	0.94	0.88	0.97	0.71	0.94
	F1-score	0.91	0.49	0.97	0.04	0.84

Table 4 provides information on the precision, recall, and F1-score performance measures for three different classifiers: MLP, RF, and LMT. These classifiers were evaluated using the binary and multi-class NSL-KDD datasets. When it comes to binary classification, both RF and MLP perform almost optimally with similar metrics. They both achieve an F1-score, precision, and recall of 0.99, which indicates an extraordinary ability to identify instances with minimal errors. On the other hand, LMT performs somewhat worse than the other classifiers with a precision of 0.88, a recall of 0.94, and an F1-score of 0.91. This suggests that although it can detect true positives, it produces more false positives than the other classifiers, as shown in Figure 3.

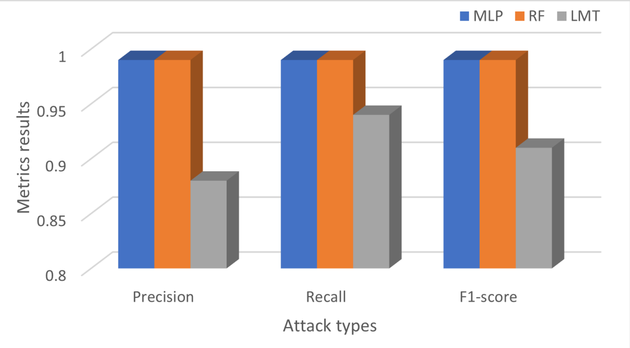


Figure 3. Binary NSL-KDD classification

Within multi-class classification, there is a significant difference in the performance among the four different attack categories (U2R, DoS, L2R, and Probe). In most classes, the MLP performs effectively; it achieves high metrics for DoS and Probe (all approximately 0.99); however, it has significant issues with L2R, as indicated by a low F1-score of 0.31, which is due to poor precision (0.24) and recall (0.43). On the other hand, in multi-class scenarios, the RF algorithm typically outperforms MLP, attaining high precision, recall, and F1 scores in most categories. While in U2R, it records an F1-score of 0.93 and performs well in the DoS and Probe classifications. However, even with L2R, it still faces some moderate issues. There is a noticeable drop in metrics for U2R and L2R and inconsistent performance across many multi-class categories for LMT. In U2R, it achieves a comparatively high recall (0.88) but low accuracy (0.34), resulting in a lower F1 score of 0.49. With a precision of 0.02 and an F1-score of 0.04 indicating that LMT is almost useless in correctly identifying the L2R category, this category presents significant challenges for LMT. LMT performs satisfactorily in the DoS and Probe categories despite these difficulties, especially in recall. Finally, while RF and MLP both perform exceptionally well in binary classification, RF is the more robust model in multi-class classification, especially when dealing with a variety of attack types, whereas LMT clearly shows deficiencies, particularly concerning less prevalent attack classes, Figure 4 illustrate the performance of multi-class classification for the NSL-KDD dataset.

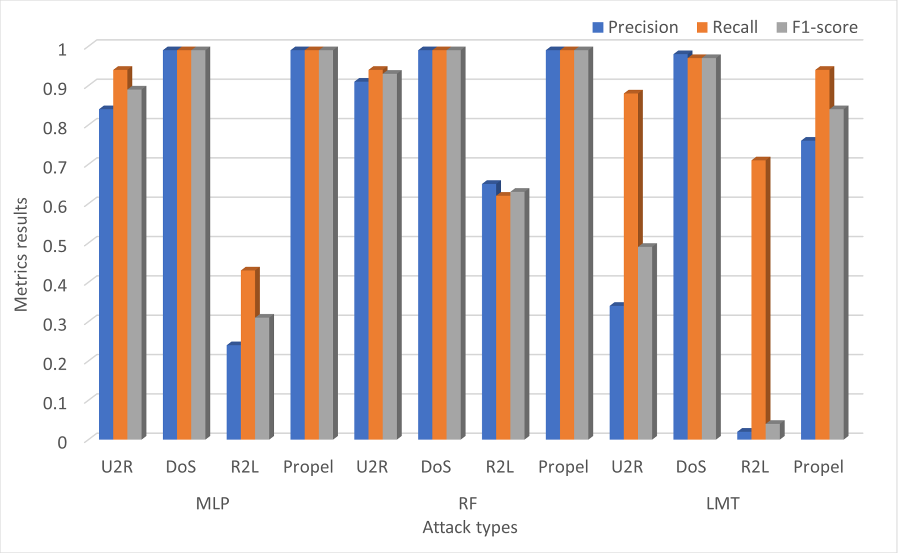


Figure 4. Multi-class NSL-KDD classification

For the CIC-IDS-2017 dataset, Tables 5 and 6 provided a comprehensive comparison of MLP, RF, and LMT—across different attack categories and general types. In the binary classification framework of the CIC-IDS-2017 dataset, we found that the MLP exhibits outstanding results, with a precision of 0.92, a recall of 0.99, and an F1-score of 0.95. This implies that the MLP is usually good at distinguishing between legitimate and malicious traffic. However, it should be noted that its precision is less than that of the RF precision. On the other hand, the RF classifier performs better on all evaluation measures, achieving an F1 score of 0.99, a precision of 0.99, and a recall of 0.99. This indicates that RF can identify and classify cases as malicious or legitimate instances with remarkable precision and reliability. However, LMT has a significantly lower performance, with a precision of 0.52, a recall of 0.80, and an F1-score of 0.63. The suboptimal precision and F1 score imply that LMT encounters greater challenges in accurately classifying instances while sustaining a balance between precision and recall, as shown in Figure 5.

Table 5. Performance metrics for binary and multi-class CIC-IDS-2017

Classifiers	Metrics	Binary	Bot	DDoS	DoS GoldenEye	DoS Hulk	DoS Slowhttptest	DoS Slowloris
MLP	Precision	0.92	0.99	0.99	0.99	0.99	0.99	0.99
	Recall	0.99	0.67	0.99	0.99	1.00	0.99	0.98
	F1-score	0.95	0.80	0.99	0.99	0.99	0.99	0.99
RF	Precision	0.99	1.00	1.00	1.00	1.00	1.00	0.99
	Recall	0.99	0.96	1.00	1.00	1.00	0.99	1.00
	F1-score	0.99	0.98	1.00	1.00	1.00	1.00	1.00
LMT	Precision	0.52	0.00	0.99	0.91	0.97	0.88	0.87
	Recall	0.80	0.00	0.97	0.85	0.95	0.71	0.81
	F1-score	0.63	0.00	0.98	0.88	0.96	0.79	0.84

Table 6. Performance metrics for binary and multi-class CIC-IDS-2017

Classifiers	Metrics	FTP-Patator	Heartbleed	Infiltration	PortScan	SSH-Patator	Brute Force	SQL Injection	XSS
MLP	Precision	1.00	0.00	1.00	1.00	0.98	0.61	0.00	0.70
	Recall	1.00	0.00	0.22	1.00	0.99	0.19	0.00	0.10
	F1-score	1.00	0.00	0.36	1.00	0.99	0.29	0.00	0.10
RF	Precision	1.00	1.00	0.83	1.00	1.00	0.71	1.00	0.50
	Recall	1.00	1.00	0.56	1.00	1.00	0.76	0.20	0.40
	F1-score	1.00	1.00	0.67	1.00	1.00	0.73	0.33	0.40
LMT	Precision	0.84	0.50	0.00	0.90	0.87	0.00	0.00	0.00
	Recall	1.00	1.00	0.00	1.00	0.51	0.00	0.00	0.00
	F1-score	0.91	0.67	0.00	0.94	0.64	0.00	0.00	0.00

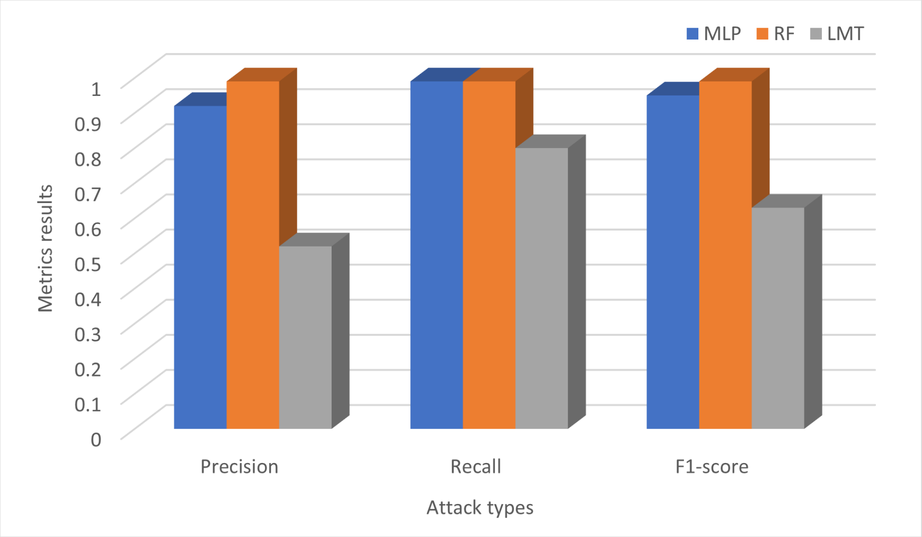
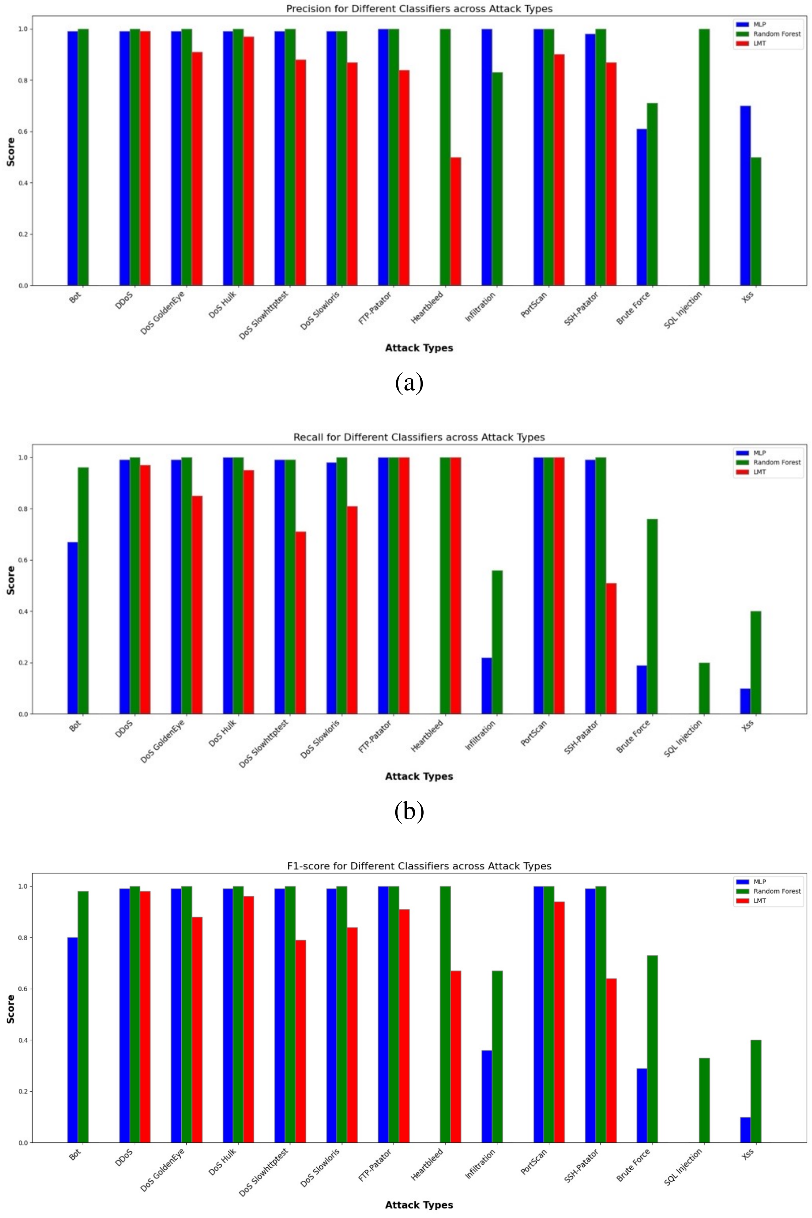


Figure 5. Binary CIC-IDS-2017 classification

In terms of multi-class classification, we found that the MLP classifier performs well ($x \geq y$ 0.99) in most categories; nevertheless, in terms of 'Heartbleed', 'Brute Force', 'SQL Injection', and 'XSS', Precision significantly decreases (from 0.00 towards 0.73). The MLP recall numbers show some variation; for example, it performs well in the "DDoS" category (0.99) and the "DoS" attack category (98–1.00); at this point, it falls poorly in the "Heartbleed," "Infiltration," and "XSS" categories (0.00–0.06). In several categories, the F1-scores for MLP are high (0.99 for some). Still, they are significantly lower in 'Heartbleed', 'SQL Injection', and 'XSS', indicating difficulties in finding a balance between precision and recall for various attack types. RF consistently maintains a high recall ($x \geq y$ 0.96), although some lower values (0.56 to 0.39) are shown in 'Infiltration', 'SQL Injection', and 'XSS', indicating an ability to ignore some rare attacks. The F1-score for RF are consistently high ($x \geq y$ 0.98) in all categories; nevertheless, they show lower scores in 'Infiltration', 'SQL Injection', and 'XSS', showing certain domains where it fails to balance precision and recall. LMT displays a varied precision profile with high scores in 'DDoS', 'DoS GoldenEye', 'DoS Hulk', and 'FTP-Patator' (from 0.84 to 0.97), and very low scores in 'Bot', 'Heartbleed', 'Infiltration', and 'XSS' (from 0.00 to 0.50). LMT performs poorly in areas like "Bot," "SQL Injection," and "XSS," but it obtains good recall in "DDoS," "DoS GoldenEye," and "DoS Hulk" (ranging from 0.71 to 1.00). The F1-scores of LMT are highest in 'DDoS', 'DoS GoldenEye', 'DoS Hulk', and 'FTP-Patator' (from 0.84 to 0.96), but they are low or nonexistent in other categories, indicating that LMT faces significant difficulties when handling less frequent or rare attack scenarios. Figure 6 presents the performance metrics for multi-class classification on the CIC-IDS-2017 dataset. Figure 6(a) illustrates the precision values for each attack type across different classifiers, Figure 6(b) shows the corresponding recall performance, and Figure 6(c) displays the F1-scores, which summarize the balance between precision and recall. Overall, the RF classifier achieved consistently higher scores across most attack categories.



(a)

(b)

(c)

Figure 6. Performance metrics for multi-class in CIC-IDS-2017 dataset: (a) precision for multi-class, (b) recall for multi-class, and (c) F1-score for multi-class

There can be notable differences in IDS performance between binary and multi-class classification methods. By concentrating on differentiating between benign and malicious communications, binary classification frequently improves the detection process and can increase the detection rates of minority classes. On the other hand, multi-class categorization seeks to distinguish between different types of attacks, which may make it difficult to reliably identify attacks that occur rarely. Table 7 compares our findings with other recent studies that focus on analyzing the effects of binary class and multi-class classification in the performance of the IDS.

Using DL models, Singh *et al.* [29] demonstrated two cutting-edge IDS. While the second combines temporal convolutional network (TCN), CNN, and bidirectional long short-term memory (Bi-LSTM), the first employs Bi-LSTM and LuNet. The systems outperform conventional ML models in tests conducted on the NSL-KDD and UNSW-NB15 datasets. Classification accuracy of up to 99% was achieved by using ensemble approaches, namely CNN + BiLSTM combinations. To improve detection for minority classes such as U2R and R2L, inside IDS datasets, Gao *et al.* [30] presented the ensemble-based binary classification technique known as GLM-IDS. This technique increases detection rates for under-represented attack classes by transforming multi-class problems into binary classifications and implementing feature selection. It achieves higher accuracy over conventional ML and DL models by striking a compromise between training and performance, especially in minority class detection.

Table 7. Summary of findings, limitations, and datasets from various references

Reference	Findings	Limitations	Dataset
[29]	IDSs exceed ML methods and current DL models in accuracy, whereas, ensemble methods like CNN and BiLSTM achieve 99% accuracy rates.	X	NSL-KDD, UNSW-NB15
[30]	Enhance detection rate of U2R and R2L classes and maintained detection performance of majority classes.	Low detection rate of minority class due to the unbalanced dataset.	U2R and R2L minority classes
[31]	Longer training time. The decision forest model achieved the best performance, with 99.2% accuracy.	-	UNSW NB-15 dataset
[32]	80.4% accuracy for binary classification using SVM and MLP. 77.5% accuracy for multi-class prediction using SVM and MLP.	Low accuracy in multi-class classification.	NSL-KDD
[33]	Decision tree ensembles performed with best accuracy.	The model will not detect zero day attacks.	CIC-IDS-2017, CSE-CIC-IDS-2018, LITNET-2020
[34]	Unable to detect attacks with limited training data. Improved the results for all performance metrics.	X	NSL-KDD dataset, CIDDS-001 dataset

Rajagopal *et al.* [31] compared eight binary and multi-class models on the UNSW-NB15 dataset through a performance analysis carried out with Azure Machine Learning. The multi-class model distinguished between attack types such as generic and shellcode with high recall rates, whereas the two-class decision forest model obtained 99.2% accuracy with time efficiency. Using the NSL-KDD dataset, Tapsoba and Ouedraogo [32] assessed supervised learning techniques for binary and multi-class network anomaly detection. Their method, which combined the use of SVM and MLP, produced a multi-class detection accuracy of 77.5% and a binary classification accuracy of 80.4%. The problem of class imbalance in intrusion detection datasets was investigated by Bulavas *et al.* [33] to improve the identification of uncommon malicious activity, they concentrated on multi-class classification rather than lumping uncommon attack classes into a single category. The goal of this study was to accurately discover different classes without eliminating minority classes by evaluating the performance of several classifiers under severely imbalanced data scenarios.

To address class imbalances in network intrusion detection, Bedi *et al.* [34] introduced I-SiamIDS, an improved version of the Siam-IDS system. Their approach, which is not dependent on data balancing techniques, detects both majority and minority classes using a two-layer ensemble of binary extreme gradient boosting (b-XGBoost), Siamese neural network (Siamese-NN), and DNN. When evaluated on NSL-KDD and CIDDS-001 datasets, I-SiamIDS showed improvements in accuracy, recall, precision, and F1-score. Finally, our work focused on developing a hybrid IDS to deal with unbalanced data, we used SMOTE, and Pearson's correlation techniques. Our model performed well on the NSL-KDD and CIC-IDS-2017 datasets, with an F1 score of 96% for binary classification and 82% for multi-class classification using NSL-KDD, while the model obtained an F1-score of 85% and 96% for multi-class detection in CIC-IDS-2017 datasets.

In comparison to the scholarly works examined, our research presents a comprehensive hybrid IDS model that adeptly navigates the complexities of both binary and multi-class classification tasks. The implementation of SMOTE and Pearson's correlation coefficient for feature selection, in conjunction with the amalgamation of ANN, RF, and LMT, rests in commendable performance metrics on both the NSL-KDD and CIC-IDS-2017 datasets. While other works explore the domains of ensemble approaches, DL, and binary classification strategies for minority class identification, our approach provides a comprehensive and useful solution to the problems related to feature selection and class imbalance. This puts our work in a prominent position within the field, especially in terms of improving detection effectiveness across various attack types by leveraging hybrid techniques.

5. CONCLUSION

Over the years, ML and DL have proven their specialization in the varied fields of research, including security and intrusion detection. This research endeavor investigated the challenges related to interpreting the results of multi-class and binary classification tasks concerning network intrusions within the context of imbalanced datasets, such as the CIC-IDS-2017 and NSL-KDD datasets. In the research, ML and DL models were included to evaluate and understand whether the binary or multi-class classification is more effective in the detection of cyberattacks. Also, the paper addressed the issue of significant imbalance in the data distribution highlighting the need for advanced techniques, including data augmentation or customized cost-sensitive learning methods, to improve model performance, particularly about under-represented attack categories. The results show that multi-class classification performs better than binary class classification, with an average F1-score of 96% for binary classification on NSL-KDD and 85% for binary classification on CIC-IDS-2017, while for multi-classification the proposed model achieved an average F1-score of 82% and 96% for NSL-KDD and CIC-IDS-2017 successively.

FUNDING INFORMATION

The authors state no funding is involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ahmad Aloqaily	✓	✓	✓	✓		✓			✓	✓		✓	✓	
Emad Eddien Abdallah	✓				✓					✓		✓	✓	
Esraa Abu Elsoud					✓	✓				✓	✓			
Yazan Hamdan	✓		✓	✓	✓	✓	✓	✓	✓					
Khaled Jallad	✓		✓	✓	✓	✓	✓	✓	✓					

C	: **C**onceptualization	I	: **I**nvestigation	Vi	: **Vi**sualization
M	: **M**ethodology	R	: **R**esources	Su	: **Su**pervision
So	: **So**ftware	D	: **D**ata Curation	P	: **P**roject Administration
Va	: **Va**lidation	O	: Writing - **O**riginal Draft	Fu	: **Fu**nding Acquisition
Fo	: **Fo**rmal Analysis	E	: Writing - Review & **E**diting		

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY

The datasets used in this study are publicly available and maintained by the Canadian Institute for Cybersecurity. The NSL-KDD dataset can be accessed at https://www.unb.ca/cic/datasets/nsl.html, and the CIC-IDS-2017 dataset is available at https://www.unb.ca/cic/datasets/ids-2017.html. Both datasets were preprocessed to remove redundant features, normalize numerical attributes, and encode categorical values prior to model training and evaluation. No proprietary or confidential data were used in this research.

REFERENCES

[1] A. Petrosyan, "Number of malware attacks per year 2023," *statista.com*, 2025. Accessed: Oct. 29, 2025. [Online]. Available: https://www.statista.com/statistics/873097/malware-attacks-per-year-worldwide/

[2] A. Abdelkhalek and M. Mashaly, "Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning," *The Journal of Supercomputing*, vol. 79, no. 10, pp. 10611–10644, Jul. 2023, doi: 10.1007/s11227-023-05073-x.

[3] H. Satilmiş, S. Akleylek, and Z. Y. Tok, "A systematic literature review on host-based intrusion detection systems," *IEEE Access*, vol. 12, pp. 27237–27266, 2024, doi: 10.1109/ACCESS.2024.3367004.

[4] A. Alsaffar, M. N. -Baygi, and H. Zolbanin, “Shielding networks: enhancing intrusion detection with hybrid feature selection and stack ensemble learning,” *Journal of Big Data*, vol. 11, no. 133, 2024, doi: 10.1186/s40537-024-00994-7.

[5] J. M. Kizza, “System intrusion detection and prevention,” in *Guide to Computer Network Security*, Cham: Springer International Publishing, 2024, pp. 295–323, doi: 10.1007/978-3-031-47549-8_13.

[6] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, “Deep learning approach for intelligent intrusion detection system,” *IEEE Access*, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[7] G. Kocher and G. Kumar, “Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges,” *Soft Computing*, vol. 25, no. 15, pp. 9731–9763, Aug. 2021, doi: 10.1007/s00500-021-05893-0.

[8] T. Acharya, I. Khatri, A. Annamalai, and M. F. Chouikha, “Efficacy of machine learning-based classifiers for binary and multi-class network intrusion detection,” in *2021 IEEE International Conference on Automatic Control & Intelligent Systems (I2CACIS)*, Shah Alam, Malaysia: IEEE, Jun. 2021, pp. 402–407, doi: 10.1109/I2CACIS52118.2021.9495877.

[9] C. Iwendi, S. Khan, J. H. Anajemba, M. Mittal, M. Alenezi, and M. Alazab, “The use of ensemble models for multiple class and binary class classification for improving intrusion detection systems,” *Sensors*, vol. 20, no. 9, 2020, doi: 10.3390/s20092559.

[10] M. B. Umair *et al.*, “A network intrusion detection system using hybrid multilayer deep learning model,” *Big Data*, vol. 12, no. 5, pp. 367–376, Oct. 2024, doi: 10.1089/big.2021.0268.

[11] J. Jose and D. V. Jose, “Deep learning algorithms for intrusion detection systems in internet of things using CIC-IDS 2017 dataset,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 1, pp. 1134-1141, Feb. 2023, doi: 10.11591/ijece.v13i1.pp1134-1141.

[12] S. Patil *et al.*, “Explainable artificial intelligence for intrusion detection system,” *Electronics*, vol. 11, no. 19, Sep. 2022, doi: 10.3390/electronics11193079.

[13] X. Meng, “Advanced AI and ML techniques in cybersecurity: supervised and unsupervised learning, reinforcement learning, and neural networks in threat detection and response,” *Applied and Computational Engineering*, vol. 82, no. 1, pp. 24–28, Jul. 2024, doi: 10.54254/2755-2721/82/2024GLG0054.

[14] V. Hnamte and J. Hussain, “DCNNBiLSTM: an efficient hybrid deep learning-based intrusion detection system,” *Telematics and Informatics Reports*, vol. 10, June 2023, doi: 10.1016/j.teler.2023.100053.

[15] E. U. H. Qazi, M. H. Faheem, and T. Zia, “HDLNIDS: hybrid deep-learning-based network intrusion detection system,” *Applied Sciences*, vol. 13, no. 8, Apr. 2023, doi: 10.3390/app13084921.

[16] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, “Intrusion detection system using feature extraction with machine learning algorithms in IoT,” *Journal of Sensor and Actuator Networks*, vol. 12, no. 2, Mar. 2023, doi: 10.3390/jsan12020029.

[17] M. Bacevicius and A. P.-Taraseviciene, “Machine learning algorithms for raw and unbalanced intrusion detection data in a multi-class classification problem,” *Applied Sciences*, vol. 13, no. 12, June 2023, doi: 10.3390/app13127328.

[18] C. H. Tseng and Y.-T. Chang, “EBDM: Ensemble binary detection models for multi-class wireless intrusion detection based on deep neural network,” *Computers & Security*, vol. 133, Oct. 2023, doi: 10.1016/j.cose.2023.103419.

[19] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” *2009 IEEE Symposium On Computational Intelligence For Security And Defense Applications*, Ottawa, Canada, 2009, pp. 1-6, doi: 10.1109/CISDA.2009.5356528.

[20] I. Sharafaldin, A. Lashkari, A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)*, pp. 108-116, 2018, doi: 10.5220/0006639801080116.

[21] B. Deepa and K. Ramesh, “Epileptic seizure detection using deep learning through min max scaler normalization,” *International Journal of Health Sciences*, vo. 10, no. S1, pp. 10981–10996, May 2022, doi: 10.53730/ijhs.v6nS1.7801.

[22] K. N. A. Halim, A. S. M. Jaya, and A. F. A. Fadzil, “Data pre-processing algorithm for neural network binary classification model in bank tele-marketing,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 3, pp. 272–277, Jan. 2020, doi: 10.35940/ijitee.C8472.019320.

[23] P. Schober, C. Boer, and L. A. Schwarte, “Correlation coefficients: appropriate use and interpretation,” *Anesthesia & Analgesia*, vol. 126, no. 5, pp. 1763–1768, May 2018, doi: 10.1213/ANE.0000000000002864.

[24] H. Gong, Y. Li, J. Zhang, B. Zhang, and X. Wang, “A new filter feature selection algorithm for classification task by ensembling pearson correlation coefficient and mutual information,” *Engineering Applications of Artificial Intelligence*, vol. 131, May 2024, doi: 10.1016/j.engappai.2024.107865.

[25] J. Beyerer, R. Hagmanns, and D. Stadler, *Pattern recognition: introduction, features, classifiers and principles*, 2nd edition. in De Gruyter Textbook. Berlin, Germany: De Gruyter, 2024, doi: 10.1515/9783111339207.

[26] T. Sathish *et al.*, “Characteristics estimation of natural fibre reinforced plastic composites using deep multi-layer perceptron (MLP) technique,” *Chemosphere*, vol. 337, Oct. 2023, doi: 10.1016/j.chemosphere.2023.139346.

[27] Z. Sun, G. Wang, P. Li, H. Wang, M. Zhang, and X. Liang, “An improved random forest based on the classification accuracy and correlation measurement of decision trees,” *Expert Systems with Applications*, vol. 237, Mar. 2024, doi: 10.1016/j.eswa.2023.121549.

[28] B. Ghasemkhani, R. Yilmaz, D. Birant, and R. A. Kut, “Logistic model tree forest for steel plates faults prediction,” *Machines*, vol. 11, no. 7, Jun 2023, doi: 10.3390/machines11070679.

[29] T. P. Singh, G. D. Kumar, M. Mutharasu, K. T. Rao, and S. M. Imran, “YARS-IDS: a novel IDS for multi-class classification,” in *2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, Chennai, India: IEEE, May 2024, pp. 1–6, doi: 10.1109/ACCAI61061.2024.10601966.

[30] X. Gao, T. Wang, Q. Wu, and J. Wu, “An intrusion detection mothod based on feature selection and binary classification grouped learning,” in *2022 IEEE 6th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Beijing, China: IEEE, Oct. 2022, pp. 1723–1730, doi: 10.1109/IAEAC54830.2022.9929759.

[31] S. Rajagopal, K. S. Hareesha, and P. P. Kundapur, “Performance analysis of binary and multiclass models using azure machine learning,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 1, pp. 978-986, Feb. 2020, doi: 10.11591/ijece.v10i1.pp978-986.

[32] A. R. Tapsoba and T. F. Ouedraogo, "Evaluation of supervised learning algorithms in binary and multi-class network anomalies detection," in *2021 IEEE AFRICON*, Arusha, Tanzania, United Republic of: IEEE, Sept. 2021, pp. 1–6, doi: 10.1109/AFRICON51333.2021.9570886.

[33] V. Bulavas, V. Marcinkevičius, and J. Rumiński, "Study of multi-class classification algorithms' performance on highly imbalanced network intrusion datasets," *Informatica*, vol. 32, no. 3, pp. 441–475, 2021, doi: 10.15388/21-INFOR457.

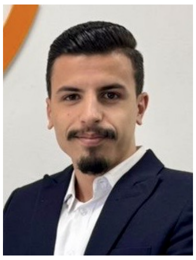
[34] P. Bedi, N. Gupta, and V. Jindal, "I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems," *Applied Intelligence*, vol. 51, no. 2, pp. 1133–1151, Feb. 2021, doi: 10.1007/s10489-020-01886-y.

BIOGRAPHIES OF AUTHORS

Ahmad Aloqaily is an Associate Professor in Computing Science at the Hashemite University, Jordan. He earned his Master's and Ph.D. in Computer Science from the University of Technology Sydney, Australia. His research focuses on data science, machine learning, sentiment analysis, and deep learning, with publications in international journals and funding from local and international grants. He played a key role in securing ABET accreditation for the computer science and computer information systems programs. He is a board member of the Jordan Computer Society, a member of the Australian Computer Society, and serves as a reviewer for various renowned journals and conferences. He can be contacted at email: aloqaily@hu.edu.jo.

Emad Eddien Abdallah is currently a Full Professor in the Department of Information Technology at Hashemite University (HU), Jordan. He received his Ph.D. Computer Science from Concordia University in 2008, where he worked on multimedia security, pattern recognition, and 3D object recognition. He received his B.S. in Computer Science from Yarmouk University, Jordan, and M.S. in Computer Science from the University of Jordan in 2000 and 2004, respectively. Prior to joining HU, he was a Software Developer at SAP Labs Montreal. His current research interests include machine learning, multimedia security, pattern recognition, cyber forensics, and data privacy. He can be contacted at email: emad@hu.edu.jo.

Esraa Abu Elsoud received the B.Sc. degree in Electrical Engineering from The Hashemite University, Jordan, in 2013, and the M.Sc. degree in Cyber Security from The Hashemite University in 2023. She works as a lecturer at Applied Science Private University in the fields of cybersecurity and cloud computing. Her current research interests include cybersecurity, machine learning, big data, and mobile networks. She can be contacted at email: e_abuelsoud@asu.edu.jo.



Yazan Hamdan is a skilled software developer currently working at Ideal Value Management Consulting in Amman, Jordan. He holds a Bachelor's degree in Computer Information Systems from the Hashemite University, where he gained in-depth knowledge of algorithms, data structures, and software engineering principles. He has extensive experience in web application development, database-driven systems, and designing artificial intelligence models. He also possesses advanced expertise in machine learning and deep learning. His professional background includes developing customized enterprise resource planning (ERP) systems, from user requirements analysis to the design and implementation of front-end, back-end, and database components. He can be contacted at email: Info@iValueConsult.com.

Khaled Jallad is a Computer Information Technology graduate from Hashemite University (2023) and a Full Stack Web Developer at Wasaq, Amman. He excels in maintaining and optimizing web applications, integrating APIs, and designing database schemas. He is skilled in Python, machine learning, and responsive design, and has worked on an IDS project using deep learning techniques. He can be contacted at email: info@Philadelphiaconsulting.com.