Vol. 05, No. 01, pp. 34 –45 (2024)
ISSN: 2708-0757

JOURNAL OF APPLIED SCIENCE AND TECHNOLOGY TRENDS

www.jastt.org

A Comparative Analysis of Intrusion Detection Systems: Leveraging Classification Algorithms and Feature Selection Techniques

Vaman Shakir Sulaiman [1, *], Adnan Mohsin Abdulazeez [2]

[1] *CIS Dept., Zakho Technical College, Duhok Polytechnic University, Iraq, vaman.sulaiman@dpu.edu.krd*
[2] *IT Dept., Duhok Technical College, Duhok Polytechnic University, Iraq, adnan.mohsin@dpu.edu.krd*

Abstract

With the increasing use of the Internet and its coverage of all areas of life and the increasing amount of sensitive and confidential information on the Internet, the number of malicious attacks on that information has increased with the aim of destroying, changing, or misusing it. Consequently, the need to discover and prevent these kinds of attacks has increased in order to maintain privacy, reliability, and even availability. For this purpose, intelligent systems have been developed to detect these attacks, which are called Intrusion Detection System (IDS). These systems were tested and applied to special benchmark datasets that contain a large number of features and a massive number of observations. However, not all the features are important, and some are not relevant. Therefore, applying feature selection techniques becomes crucial, which select the features with the most importance and relevance in order to enhance the performance of the classification model. The aim of this review paper is to conduct a comparative analysis of various state-of-the-art IDS that use algorithm classifications to detect network attacks with the cooperation of feature selection techniques that have been applied to various well-known IDS datasets, such as KDD cup99, NSL-KDD, etc. This comparison is based on several factors, including the utilized classification technique, feature selection used, employed evaluation metrics, datasets used, and finally the highest accuracy rate obtained by each study.

Keywords: *Classification Algorithm, IDS, Feature Selection, NSL-KDD, Machine Learning, Deep Learning, Network Attacks*
Received: March 15, 2024 / Accepted: May 21, 2024 / Online: May 23, 2024

I. INTRODUCTION

The Internet is regarded as a virtual environment that provides services to individuals and organizations to conduct practices such as education, entertainment, and e-commerce. Additionally, the Internet is a world with a high amount of data related to financial transactions and people's privacy. Having an impact on how data is exchanged in terms of Availability, Integrity, and Confidentiality has a dangerous effect on systems and networks. As internet utilization rises continuously, there is a corresponding increase in cyber-attacks. Given that numerous of these attacks are new, the need for an intelligent system, specifically an Intrusion Detection System (IDS), becomes crucial for effectively detecting and mitigating such threats [1]. The IDS is designed to analyze network traffic, enabling it to discern the nature of various attacks. Within every security policy, the inclusion of an IDS is vital for the overall security infrastructure. The IDS serves to protect the system by efficiently detecting the intrusion on both the network and host levels. Regarding the detection approach [2], IDS employs anomaly-based and signature-based detection techniques. The first utilizes intelligent and statistical patterns to identify abnormal and normal behavior, while the second relies on signatures to recognize attacks [2], [3], [4].

Traditional intrusion detection techniques prove ineffective when applied to large datasets. The use of Machine Learning (ML) methods can enhance intrusion detection efficiency [5]. Numerous approaches for constructing IDS have been proposed in different studies; these approaches rely on intelligent classification strategies that utilize Artificial Intelligence (AI) algorithms to distinguish abnormal behavior from normal behavior [2], [6]. In the field of the network IDS, numerous researches have been carried out employing various Deep Learning (DL) and ML techniques including but not limited to Logistic Regression (LR), Artificial Neural Network (ANN), Deep Neural Network (DNN), Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT), Convolutional

doi:10.38094/jastt501186

Neural Network (CNN), K-Nearest Neighbor (KNN), Naive Bayes (NB), among others.

To effectively detect network attacks, a significant amount of data (such as the IDS benchmark datasets) is necessary to create a model that distinguishes between normal and anomalous patterns. Additionally, utilizing supervised learning, especially classification algorithms for effective data comprehension, facilitates the development of robust models with high detection rates for predicting new attacks [7]. The high dimensionality of the data presents a challenge, as an expanded feature space with a relatively limited number of records can lead to the "curse of dimensionality" problem, which adversely affects classification performance. To handle such issues, there is a concerted effort to employ Feature Selection (FS) and extraction techniques to improve classification results [7].

The remaining of this paper is organized as follows: Section Two provides a theoretical background of the ML, classification technique, IDS, benchmark datasets, and feature selection technique. Section Three conducts a review on numerous studies regarding the IDS field. Section Four presents a discussion of the reviewed papers in the literature. Finally, the conclusion is presented in Section Five.

II. BACKGROUND THEORY

This section provides a brief overview of the theoretical background of the ML concept and classification algorithm, IDS, various IDS benchmark datasets, and feature selection techniques.

A. Machine Learning and Classification

Machine Learning (ML), a subset of Artificial Intelligence (AI), encompasses a collection of algorithms and methods that empower computer systems to learn from existing data and autonomously make predictions or decisions, all without requiring explicit programming [8], [9]. There are three major categories of machine learning techniques: supervised learning, unsupervised learning, and semi-supervised learning. In supervised learning, the system is trained using input data that has been labeled, allowing it to differentiate between different classes in the dataset [10]. On the other hand, unsupervised learning involves using input data that is unlabeled to help the system identify patterns of similarity within the data [5]. Semi-supervised learning combines both approaches by using a limited amount of labeled data and a large amount of unlabeled data, which bridges the gap between supervised and unsupervised learning. By leveraging both labeled and unlabeled samples, the performance of semi-supervised learning can be improved significantly [5].

Classification, a form of supervised learning, involves training an algorithm to predict the category or class of a given input data point. The algorithm learns from a labeled dataset, which includes input data points along with their corresponding labels or target classes [11]. The goal of classification algorithms is to construct a model capable of accurately classifying new, previously unseen data points into predefined categories or classes [12].

B. Intrusion Detection Systems

Intrusion Detection System (IDS) are the techniques that allow for the detection of intrusions in network environments. It detects malicious utilization of the resources of the network [13]. In other words, IDS is a security technology designed to analyze and monitor network behaviors, aiming to detect and react to security violations, unauthorized access, or potential risks. IDSs work by examining network traffic or system actions, matching them with predetermined signatures or behavioral patterns that could signify an intrusion or malicious behavior [14]. The primary objectives of an IDS include monitoring and analyzing host and network behaviors, providing alerts, and taking action in response to suspicious activities [7].

As shown in Fig. 1, IDSs can be broadly categorized into two types: deployment-based methods and detection-based methods, depending on the employed recognition techniques [15], [16]. Within the deployment-based techniques, IDSs can be further classified into two types: Network IDS (NIDS) and Host-Based IDS (HIDS). HIDS is implemented on individual information hosts, where its role is to scrutinize all activities on a specific host, checking for security policy violations and any suspicious behavior. However, a notable drawback is the necessity to deploy HIDS on all hosts requiring intrusion protection, leading to additional processing overhead on each node and, consequently, a performance degradation of the IDS [17]. Conversely, NIDS is deployed on the network itself with the objective of safeguarding the entire network and all devices from intruders. NIDS consistently monitors network traffic, conducting scans to detect potential security violations and breaches. This approach offers a broader perspective by monitoring overall network activity rather than concentrating on individual hosts [17].

As for detection-based methods, there are two subcategories, namely, anomaly detection and signature-based detection [15], [16]. Signature-based detection, often referred to as rule-based detection, relies on a set of pre-defined signatures stored in a database. This method operates by comparing a sample's signature with those in the database. However, a drawback of this approach is the challenge of crafting well-organized signatures. Despite this limitation, signature-based detection has gained more popularity due to its ability to report on specific attack types along with their causes, offering a low false alarm rate. On the downside, this method has a higher missed alarm rate and struggles with detecting unknown attacks, requiring the maintenance of an extensive signature database [16], [17]. On the other hand, Anomaly-based detection is employed to identify changes in behavior. This method creates a profile of normal behavior and compares ongoing activities against that profile. Any deviation triggers an alert. The key requirement in this detection technique is the creation of a normal behavior profile. Its advantages include robust generic support and efficiency in detecting new attacks. However, it comes with a higher false alarm rate and is inefficient in providing explanations for detected irregularities [15], [17].

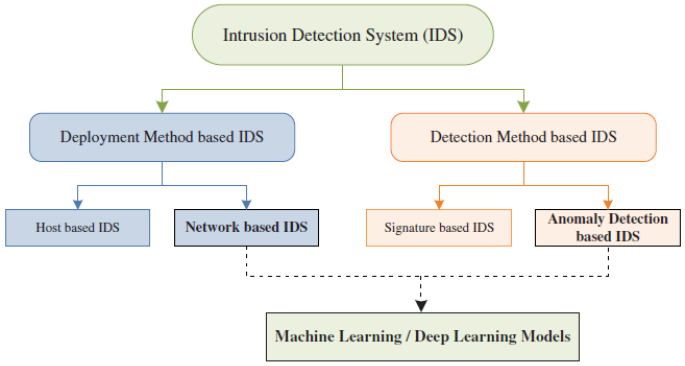

Fig. 1. IDS classification taxonomy [17].

C. Benchmark Datasets

The benchmark datasets for evaluating IDS approaches are essential to validate their effectiveness in identifying intrusive behavior. Various private and public datasets are combined as benchmark data to evaluate IDS. To prevent incomplete training datasets, self-produced and private datasets are created, but they remain unreachable, and difficult to guarantee their efficacy. Due to privacy concerns, the availability of datasets used for analyzation of network packets, in commercial products is often restricted. Nonetheless, there are publicly accessible benchmark datasets like NSL-KDD, KDDCup99, and CISID2017, which are widely utilized benchmarks in the field of IDS. TABLE I presents various well-known benchmark datasets [18], [19], [20].

TABLE I. SUMMARIZATION OF THE SEVERAL IDS BENCHMARK DATASETS

Dataset	Year	Number of features	Number of instances	Attacks
KDDCup99	1999	41	4,900,000	DoS, Probe, R2L, U2R
Kyoto 2006+	2006	24	-	Unknown and Known Attacks
CAIDA	2007	20	-	DDoS
NSL-KDD	2009	41	148,517	DoS, Probe, R2L, U2R
UNSW-NB15	2015	49	-	Backdoors, DoS, worms, Exploits, Port scans, Shellcode, Generic, Reconnaissance, Fuzzers
CICIDS2017	2017	80	2,830,743	HeartBleed, DoS, Botnet, Brute Force (FTP, and SSH), Web, DDoS, Infiltration
CSE-CICIDS2018	2018	80	16,000,000	DoS, HeartBleed, Brute Force (FTP, and SSH), Botnet, Web, DDoS, Infiltration

D. Feature Selection

Feature Selection (FS) is a highly skilled method for reducing the dimensionality of data by choosing the most significant features without repetition and unrelated while maintaining a high level of classification accuracy [21], [22]. It is concerned with minimizing computational complexity and avoiding the overfitting issue [23]. Utilizing FS provides simpler and faster predive models with less False Alarm Rate (FAR).

1) Feature Selection Types

Features selection techniques have been classified into three groups [20], [22], [24], [25]: filter, wrapper, and hybrid methods. In the filter method, the features are evaluated by utilizing statistical approaches instead of using classification algorithms [26], [27], [28]. This type is implemented before fitting ML models and it measures the correlation of input features with each other as well as with the output variable [20]. In contrast, the wrapper method utilizes ML algorithms to pick the most significant features from the whole features in the dataset [21], [29]. Though wrapper approach performs better than filter approach in terms of classification efficiency, they are more computationally costly [30]. Otherwise, in the processing with high-dimensional data, a filter method is preferred [21]. As a result, the hybrid approach is a mixture of wrapper and filter approaches that was evolved to cover the disadvantages while making use of advantages [31], [32]. According to the literature conducted in this study, various researches applied a variety of FS techniques including Correlation Coefficient, Entropy, Particle Swarm Optimization (PSO), Principle Component Analyzes (PCA), Random Forest (RF), among others.

As illustrated in Fig. 2, the procedure of feature selection consists of four essential steps including subset generation, subset evaluation, stopping criteria, and result validation. Subset generation is a search technique that identifies the search space for the purpose of choosing the best subset, which includes the most important features. Each subset is evaluated using the criterion measures. While the validation of the results is the stage in which the efficacy of the classifier algorithm is determined. The search technique (subset generation) and the evaluation process (subset evaluation) are primary steps in the feature selection process since they determine the efficacy of the selected feature subset(s).

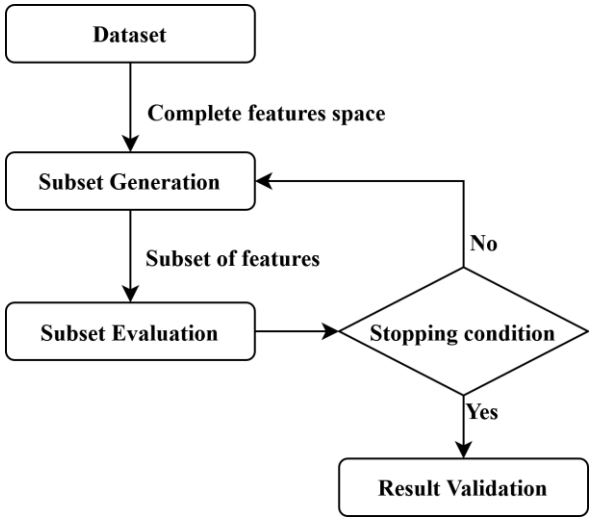

Fig. 2. Feature Selection Steps.

III. LITERATURE REVIEW

This section conducts a review of various state-of-the-art IDSs that utilize classification algorithms in cooperation with FS techniques to train the model for the prediction process. Moreover, TABLE II presents a summary of the reviewed works based on several factors including, the used classification method, FS technique, evaluation metrics, dataset, and accuracy result.

The authors in [33] suggested DL-based IDS using Long Short Term Memory (LSTM) and a Fully Connected Neural Network (FCNN), consisting of several stages. The initial phase of the methodology involves the utilization of a CNN for the extraction of spatial information. This entails employing two convolution layers with output dimensions of 32 and 64 bits, respectively. Both convolution layers utilize a 3x3 kernel. Subsequent to each convolution layer, dimensionality reduction is accomplished by integrating a Max-pooling layer with a size of 2x2. Following the CNN stage, the information is forwarded to the subsequent phase, comprising an LSTM layer, a fully connected layer, and an output layer. The output layer serves the pivotal role of synthesizing the information. Both the fully connected layer and the long short-term memory (LSTM) layer consist of 128 nodes. Eventually, a softmax layer is applied for classification purposes, ensuring accurate depiction of the probability associated with each input stream. Notably, when addressing five-class classification tasks, the proposed deep learning model yields enhanced accuracy. The authors utilized various metrics to evaluate the performance of the model including accuracy, False Positive Rate (FPR), and True Positive Rate (TPR). Specifically, upon evaluation with the KDDCup99 dataset, an accuracy of 99.99% is attained, while on the NSL-KDD dataset, an accuracy of 99.95% is achieved.

The authors in [34] employed three ML classifiers including SVM, RF, and DT in order to build the model to predict the various attacks included in NSL-KDD dataset. The system implemented ANOVA F-Test and Recursive Features Elimination (RFE) as a FS technique to reduce the number of features from 41 to 13 leading to an improvement in model performance. From the experimentation conducted using the three classifiers, RF algorithm provides higher results as compared to SVM and DT achieving an accuracy of 98%, 87%, 86%, and 76% for User to Root (U2R), Denial of Service (DoS), Probe, and Remote to Local (R2L) categories, respectively representing average overall accuracy of 86.75%.

The researchers in [35] introduced a method named IGRF-RFE to identify network anomalies using Multi-Layer Perceptron (MLP) classifier. This method combines two different types of feature selection methods: wrapper and filter techniques. Initially, they applied two filter techniques, Information Gain (IG) and Random Forest (RF), to narrow down the selected features. This combination effectively drops less significant attributes (identified by IG) with the help of RF. Following this, they employed a wrapper method called RFE to further decrease the number of features while considering the relevance of similar features. They evaluated the proposed method using various measures such as Precision, Recall, F1-Score, FPR, and Accuracy, on a dataset called UNSW-NB15. The results indicate that IGRF-RFE improves the accuracy of anomaly detection by selecting more permanent features while reducing the total feature space. Specifically, the number of features decreased from 42 to 23, leading to an improvement in the MLP's multi-classification accuracy from 82.25% to 84.24%.

The researchers in [36] presented another technique that merges CNN and Gated Recurrent Unit (GRU) architectures. They explored different sequences of CNN–GRU combinations to optimize network parameters. The study utilized the CICIDS2017 benchmark dataset and assessed performance using metrics such as accuracy, recall, precision, TPR, FPR, and other related metrics. Additionally, the authors enhanced the original dataset by selecting features based on Pearson's Correlation coefficient and eliminating redundant instances. The results highlighted a significant improvement, achieving an impressive accuracy of 98.73% in detecting various network attacks, along with a low FPR rate of 0.075.

The authors in [37] proposed a system for IDS that combines techniques for selecting important features and classifying data. They use Pearson's correlation coefficient to identify highly relevant features and the KNN classifier to detect attacks in Internet of Things (IoT) networks. The aim of this approach is to improve classification performance by increasing the detection accuracy and decreasing time complexity via selecting only nineteen relevant features from the original forty-one in the dataset. They tested the performance of this IDS on the NSL-KDD dataset and compared it with other ML models like DT, SVM, RF, KNN, and NB. Additionally, they used metrics such as Accuracy, Precision, Recall, and F1-score to evaluate the system's performance. The results showed that the proposed system effectively reduced the number of features while achieving a higher classification accuracy of 79.24%.

The authors in [38] introduced another model for IDS aimed at classifying network traffic, utilizing a DNN. FS was employed initially as a preprocessing step to decrease the number of features, and K-fold cross-validation was used to partition the data into training and testing sets. The experimental evaluation involved two datasets: CICIDS2017, which contains new attacks, and the commonly used NSL-KDD dataset. Various evaluation metrics were utilized such as Accuracy, DR, and FAR to analyze the findings, indicating that the model achieved an accuracy of 99.43% and 99.63% for the CICIDS2017 and NSL-KDD datasets, respectively.

The researchers in [39] used a technique for FS called Stack Denoising Auto Encoder (SDAE) to enhance the protection rate. In addition, the authors implemented various well-known classification techniques to distinguish normal from abnormal network traffic, including KNN, DT, NB, SVM, and LSTM. All experiments are applied to the NSL-KDD dataset after implementing the preprocessing stage, which consists of two processes; dropping the outliers using the Median Absolute Deviation Estimator (MADE) technique and making all features on the same scale using the Min-Max scaler. For the evaluation of the models, several evaluation metrics were used, such as Accuracy, F1-score, Precession, and Recall. The results demonstrated that LSTM outperforms the remaining used classification algorithm, achieving an accuracy of 85.2%.

The authors in [40] introduced a method called the DLNID (Deep Learning Network Intrusion Detection) model to identify anomalies in network traffic. This model combines an attention mechanism with a bidirectional LSTM (Bi-LSTM) network. Initially, it uses a CNN to extract sequence features from the data traffic. Then, the attention mechanism adjusts the weights of each channel, followed by the Bi-LSTM to learn the sequence features network. To address the issue of imbalanced data in intrusion detection public datasets, the paper employs Adaptive Synthetic Sampling (ADASYN) to increase minority class samples, thereby creating a more balanced dataset. Furthermore, a modified stacked autoencoder is used to reduce data dimensionality, aiming to improve data integration. Experimental evaluations on the NSL-KDD public benchmark dataset for network intrusion detection show that the proposed model achieved F1-score of 89.65% accuracy of 90.73%.

The authors in [41] introduced an IDS, characterized by a low FAR and simultaneously high Detection Rate (DR). Their system employs a binary Pigeon Inspired Optimizer (PIO) to select highly important features. Additionally, it integrates two main subsystems, each trained independently using the One-Class SVM (OCSVM) classifier. The first subsystem focuses on normal packets while the other deals with attack packets. Following this, a combination of the results of both subsystems is utilized to provide each network packet with a comprehensive assessment. The performance of this proposed NIDS is evaluated using three commonly used datasets (KDDCup99, NSL-KDD, and UNSW-NB15) across various metrics including DR, Accuracy, FAR, and F1-score. The results show high accuracy rates of 99.7%, 99.8%, and 99.3% on the KDDCup-99, NSL-KDD, and UNSW-NB15 datasets, respectively.

The researchers in [42] presented an IDS employing a range of ML and DL algorithms, such as KNN, Adaboost, DT, LSTM, and MLP, applied to the TON_IoT dataset. Additionally, the approach involved using a hybrid FS technique consisting of RF and Pearson Correlation Coefficient to improve performance. Furthermore, they evaluated the system's performance using various metrics including Recall, Accuracy, Precision, F1-score, and Receiver Operating Characteristic (ROC) curve. The authors noted that DT (for ML) and MLP (for DL) achieved the highest results with lower False Negative Rate (FNR) and FPR compared to other algorithms, achieving accuracies of 99.6 and 99.2, respectively.

The authors in [43] presented a new approach called Boruta Feature Selection with Grid Search Random Forest (BFS-GSRF) for the purpose of recognizing network intrusions. The evaluation of the model is conducted using KDDCup99; a familiar benchmark dataset. The classifiers, namely RF, Classification And Regression Tree (CART), Linear Discriminant Analysis (LDA), and SVM, achieve performance rates of 99%, 97.7%, 98%, and 98.5%, correspondingly. The suggested study introduced the BFS-RF algorithm to further enhance the performance of the classifier which helps in finding the optimal value for each hyper-parameter. The BFS method is employed to effectively select the most relevant features through the utilization of wrapper techniques. The accuracy of BFS-RF performance was evaluated and found to be 99.9%.

The researchers in [44] suggested several techniques to develop IDS employing the UNSW-NB15 dataset. They identified optimal feature subsets from the dataset by analyzing their relationships using the correlation coefficient technique. Making use of the selected features, they introduced an IDS approach employing AdaBoost. In their study, SVM and MLP were also utilized for comparison, but the AdaBoost model, based on the DT classifier, was identified as the best-performing model for distinguishing potential threats from normal activities. Furthermore, various performance metrics, including Precision, Accuracy, Recall, and F1-score, were used for comparison purposes. The experimental findings highlighted the effectiveness of the proposed method in detecting various network intrusion forms, achieving a high accuracy rate of 99.3%.

The researchers in [45] introduced a fresh perspective on intrusion detection systems by employing PCA for feature selection and incorporating a variety of SVM kernels. The authors explored how different kernel functions (Polynomial, Sigmoid, Linear, and Gaussian radial basis function) affect evaluation metrics such as TPR, Precision, True Negative Rate (TNR), Sensitivity, F1-score, and Accuracy. The model undergoes evaluation using both the UNSW-NB15 and KDDCup99 datasets. Findings revealed that the Gaussian radial basis function kernel surpasses the Polynomial, Sigmoid, and Linear kernels in both employed datasets, achieving an Accuracy, F1-score, and Sensitivity of 93.94%, 94.44%, and 93.23% for UNSW-NB15 dataset, and 99.11%, 99.03%, and 98.97% for KDD CUP'99 dataset, respectively.

The authors in [46] presented IDS using a combination of LSTM and CNN algorithms. Their model comprised stacked layers of CNN and LSTM, leveraging LSTM's capability of extracting temporal features and CNN's ability of capturing spatial features. To enhance the model's performance, the authors incorporated batch normalization, dropout layers, and standardization techniques. They evaluated the proposed model using the WSN-DS, CIC-IDS2017, and UNSW-NB15 datasets. Initially, they tested the behaviors of these datasets using various combinations of LSTM-CNN, CNN, CNN-LSTM, and LSTM models. The findings revealed that the hybrid model (CNN-LSTM) achieved the highest accuracy and DR. Subsequently, the authors evaluated the hybrid model in both binary and multiclass classification scenarios. After 5 epochs, they achieved accuracy rates of 99.67%, 94.53%, and 99.64% for binary classification using the WSN-DS, UNSW-NB, and CIC-IDS2017 datasets, respectively.

The researchers in [47] proposed SMO-HPSO, a hybrid FS model that combines two optimization methods including Spider Monkey Optimization (SMO) and Hierarchical Particle Swarm Optimization (HPSO), improving the detection accuracy as well as minimizing FAR. After selecting the optimal feature subset, RF classifier was utilized for the purpose of classifying the network attacks that existed in both UNSW-NB15 and NSL-KDD datasets. The SMO-HPSO incorporated the feature importance along with Rosenbrock's banana function, while integrating hierarchical PSO's velocity with the searching process of SMO. The introduced work examined various classifiers including DT, SVM, and RF on both recently mentioned datasets by utilizing different evaluation metrics such

as Precision, Recall, F-Score, Accuracy, FAR, Area Under the Curve (AUC), Matthews Correlation Coefficient (MCC). The findings demonstrated that using the hybrid optimization method followed by using RF classifier achieved high accuracy represented as 99.175% and 99.18 for NSL-KDD and UNSWNB15 datasets, respectively.

The authors in [48] proposed a method called PSO-DNN (Particle Swarm Optimization combined with Deep Neural Network) to develop a highly efficient and accurate IDS for the Internet of Medical Things (IoMT). PSO is applied to identify the most significant features influencing the system's performance, while DNN is employed for modeling. This approach achieves a detection accuracy of 96% in identifying network intrusions by utilizing a comprehensive dataset that integrates patient sensing information and network traffic. Additionally, the study extensively analyzes various DL and ML approaches, including LR, KNN, DT, RF, SVM, Adaboost, DNN, CNN, and LSTM, for network intrusion detection in IoMT. The findings confirm that DL models demonstrated a slight performance advantage over ML models. Evaluation metrics employed in the study include Accuracy, Recall, Precision, and F1-score.

The researchers in [49] aimed to train and assess various ML classifiers, including DT, MLP, Gradient Boosting Tree (GBT), Gated Recurrent Unit (GRU), LSTM, and AdaBoost, for the binary classification task of ML-based IDS. Employing both UNSW-NB15 and Network TON_IoT datasets, a Gini Impurity-based Weighted Random Forest (GIWRF) was introduced for FS, considering the potential impact of imbalanced class distributions on this process. This FS method reduced the feature sets of UNSW-NB 15 and Network TON_IoT datasets from 42 to 20 and from 41 to 10 features, respectively. The models' performance was evaluated using various metrics such as Accuracy, Precision, FPR, F1-score, and Recall to identify intrusions. Initially, experiments were conducted using the complete feature sets of both datasets. Subsequently, the experiments were repeated using only the features selected through the FS method. A comparative analysis of the models' performance was conducted between the full feature space and the reduced feature set for both datasets. When the FS technique was applied, for TON_IoT dataset, GBT model exhibited superior performance achieving an accuracy of 99.98%, while for UNSW-NB15 dataset, DT was found to have the highest accuracy achieving 93.01%.

The authors in [50] introduced a model for IDS based on Ensemble Learning (EL) algorithms. The study utilized an effective FS strategy, merging the Correlation coefficient Feature Selection technique with Forest Panelized Attributes (CFS–FPA). To enhance intrusion detection, AdaBoosting and bagging EL algorithms were employed to improve four classifiers: NB, SVM, KNN, and RF. These enhanced classifiers were initially utilized via AdaBoosting and then through bagging, employing the aggregation method with the voting average technique. The performance of the model was assessed in both binary and multi-class classification scenarios to ensure comprehensive evaluation. Experimental findings on the CICIDS2017 dataset indicate satisfying outcomes, achieving an accuracy of 99.7%, a, FAR of 0.004, and FNR of 0.053.

The authors in [51] introduced a method to enhance the reliability of Network Intrusion Detection (NID) by employing a DT and enhancing data quality. This approach consists of three main phases: a data quality phase, a model building phase, and an intrusion detection deployment phase. The methodology involves preprocessing network data and applying entropy-based FS to enhance data quality and facilitate training effectively. Then, a DT algorithm is used to achieve reliable intrusion detection. Three evaluation metrics, including DR, Accuracy, and FAR, were utilized to evaluate the performance of the proposed model. Through experimentation on both datasets: CICIDS2017 and NSL-KDD, the suggested model exhibited robust performance, achieving an accuracy of 98.80% and 99.42%, respectively.

The authors in [52] proposed IDS using ANN with the help of PCA. PCA is employed to reduce the number of features that will be considered as input to the neural network. The proposed system experiments were performed and evaluated through the use of two benchmark datasets, including NSL-KDD and CICIDS2017. Furthermore, K-fold cross validation is used to split the data into train-test splits, with K = 10 to produce 10 splits. Among all the classifiers applied in this work, ANN is found to provide the highest accuracy as compared to the other classification algorithms used, including AdaBoost, NB, and SVM, by achieving an accuracy of 99.91% and 97.69% when applied to NSL-KDD and CICIDS2017, respectively. For the first dataset, NSL-KDD, the best parameter values were (hidden layers: 1, neurons: 25, epochs: 100, activation function: tanh, optimizer: adam, batch size: 100), while for the second dataset, CICIDS2017 dataset, the optimal parameter values were (hidden layers: 4, number of neurons for each: 50, epochs: 30, activation function: Relu, batch size: 50, optimizer: rmsprop).

The authors in [53] presented IDS using Recurrent Neural Network (RNN) and DNN algorithms for multiclass and binary classification, respectively. moreover, NSL-KDD was used to build the protection model. For binary classification, DNN is used to classify normal from attack network traffic. In contrast, RNN is used for classifying the 5 classes (normal, U2R, Probe, DoS, and R2L), included in the NSL-KDD dataset. Before applying the classification model, RFE technique is utilized for feature selection, and normalization (Min-Max scaler) is employed to make all selected 25 features on the same scale, ranging from 0 to 1. DNN (with four hidden layers) achieved an accuracy rate of 94%. The FPR was found to be 0.08, while the TPR reached 92%. Additionally, using RNN with four hidden layers, for the Normal category, the accuracy and TPR were 96% and 77%, respectively. For DOS, the accuracy and TPR were 96% and 94%. The Probe category showed an accuracy of 87% with a TPR of 87%. R2L achieved an accuracy of 70% with a TPR of 87%. Lastly, the U2R category exhibited an accuracy of 94% with a TPR of 99%.

The researchers in [54] introduced a method called Double-Layered Hybrid Approach (DLHA) developed to address the challenge of effectively identifying rare attacks while simultaneously enhancing the overall detection performance. Within DLHA, an Intersectional Correlated Feature Selection (ICFS) mechanism is integrated to eliminate irrelevant features and maintain only the important ones, thereby reducing dimensionality and accelerating real-time detection. The

detection mechanism operated in two layers: the initial layer employed NB classifier for classifying Probe and DoS attacks across all connections, while the subsequent layer utilized SVM to distinguish R2L and U2R attacks from normal traffic, which is found to be a more difficult task. The suggested system assessed the performance of the utilized models using various metrics such as Accuracy, F1-score, Precision, DR, and FAR. Evaluations conducted on the NSL-KDD dataset demonstrated the effectiveness of DLHA, achieving an accuracy of 88.97%.

The authors in [55] employed two distinct sets of features to train four ML classifiers. These feature sets were derived using the PSO and Genetic Algorithm (GA) methods, both of which are effective for solving optimization problems. The classifiers utilized in this study include KNN, NB, SVM, and DT. These classifiers were trained and evaluated utilizing the most well-known IDS benchmark dataset, the NSL-KDD dataset. The experimental findings reveal that implementing the selected 20 sub-features derived from applying PSO leads to an improvement of approximately 1.55% in detection accuracy compared to the GA-based features (11 features). Notably, the DT classifier trained with PSO-based features surpassed other utilized classifiers, achieving impressive results in Accuracy, F1-score, Recall, and Precision of 99.38%, 99.34%, 99.32%, and 99.36%, respectively.

The authors in [56] presented an IDS developed for detecting various types of attacks in Internet of Things (IoT) networks. Their approach involved a combination of Grey Wolf Optimization (GWO) and PSO to select highly relevant features. These chosen features were then fed into RF classification algorithm, enhancing the system's accuracy in detecting attacks. To address the issues related to data imbalance, the study implemented an oversampling technique. The experimentation was conducted using the CICIDS2017, NSL–KDD, and KDDCup99 datasets. Evaluation metrics encompassed Accuracy, Recall, Precision, and F1-score. The GWO–PSO–RF Network IDS (NIDS) model showed a notable average accuracy of 99.66% in multiclass classification.

The authors in [57] suggested an IDS method for enhancing cloud service providers' ability to model their users' behavior. For the purpose of process recognition and detection, a combination of a probabilistic neural network with PSO was applied in this system. For feature selection, the authors employed PCA. As a preprocessing phase, the data were normalized using Min-Max method. The authors began the recognition process by meaningfully converting behaviors of the user to an understandable format, and then recognized and classified malicious behaviors by utilizing a multi-layer ANN. The authors validated their approach using the UNSW-NB15 dataset by characterizing various kinds of malicious behaviors. The experimental results showed that the suggested technique provided a DR of 96.4%.

The researchers in [58] presented a hybrid IDS that combines two ML classifiers, specifically J48 DT and SVM. Relevant FS from the KDDCup99 dataset is achieved using PSO. The classification process on the KDDCup99 dataset is implemented using the WEKA tool, with the dataset divided into training and testing sets at ratios of 60:40, 70:30, and 80:20. Experimental results demonstrated high-performance model, with the 60:40 dataset achieving a DR of 99.6%, an accuracy of 99.1%, and a FAR of 1.0%. Similarly, the 70:30 datasets showed an accuracy of 99.2%, a DR of 99.6%, and a FAR of 0.9%. The 80:20 datasets exhibited comparable performance, with an accuracy of 99.1%, a DR of 99.6%, and a FAR of 0.9%. These findings highlight the effectiveness of the proposed hybrid IDS framework.

The authors in [59] used RF algorithm to perform FS in order to reduce the number of features and exclude irrelevant features. It improved the efficiency and effectiveness of the main task of intrusion detection. A comparative analysis was conducted using a variety of classifiers, such as SVM, DT, NB, KNN, and LR to evaluate the various IDS metrics. The PSO algorithm was applied to the features of NSL-KDD dataset that were selected, resulting in a decrease in the number of false alarms and an improved accuracy and DR of the IDS as compared to the aforementioned classifiers. As performance measures for IDSs, this study included DR, Precision, Accuracy, and FPR. By selecting 10 features out of 41 features in the proposed system, the experimental results demonstrated an accuracy of 99.32% using PSO, and 97.18% using KNN.

The authors in [60] introduced Network IDS using DL by conducting a performance comparative analysis across three publicly available benchmark datasets. PCA was utilized for feature extraction, and ANN was employed for the classification phase. The evaluation of the model was based on four key metrics: Recall, Accuracy, Precision, and F1-score. The results indicated that the model performed most effectively on the NSL-KDD dataset, followed by UNSW-NB15 and CSE-CIC-IDS2018, achieving accuracies of 97.89%, 89.99%, and 76.47%, respectively.

The researchers in [61] suggested IDS that employed NSL-KDD dataset for training a classification model using RF algorithm. Before applying RF (consisting of 1000 Decision trees) to the abovementioned dataset, the feature selection method called Gini impurity was applied, resulting in lower feature dimension. The experiment showed that the suggested method achieved a high accuracy of 99.88%.

TABLE II. SUMMARIZATION OF THE REVIEWED WORKS

Ref	Classification Algorithm	FS Technique	Evaluation Metrics	Dataset	Accuracy %
[33], 2023	**LSTM+FCNN**	CNN	Accuracy, TPR, FPR	KDDCup99	99.99
				NSL-KDD	99.95
[34], 2023	**RF,** SVM, DT	ANOVA F-Test and RFE	Accuracy	NSL-KDD	86.75
[35], 2023	**MLP**	IGRF-RFE	Precision, Recall, F1-score, FPR, Accuracy	UNSW-NB15	84.24
[36], 2023	**CNN+GRU**	Correlation Coefficient	Accuracy, Recall, Precision, TPR, FPR	CICIDS2017	98.73

[37], 2023	**KNN,** DT, SVM, RF, NB	Correlation Coefficient	Accuracy, Precision, Recall, F1-score	NSL-KDD	79.24
[38], 2022	**DNN**	Not mentioned	Accuracy, DR, FAR	NSL-KDD	99.63
				CICIDS2017	99.43
[39], 2022	**LSTM,** KNN, DT, NB, SVM	SDAE	Accuracy, Precision, Recall, F1-score	NSL-KDD	85.2
[40], 2022	**LSTM**	CNN	Accuracy, Precision, Recall, F1-score	NSL-KDD	90.73
[41], 2022	**OCSVM**	PIO	Accuracy, DR, FAR, F1-score	KDDCup99	99.7
				NSL-KDD	99.8
				UNSW-NB15	99.3
[42], 2022	**AdaBoost**, DT, KNN, MLP, LSTM	RF, Correlation Coefficient	Accuracy, Recall, Precision, F1-score, ROC	TON_IoT	99.8
[43], 2022	**RF,** CART, LDA, SVM	Boruta	Accuracy, Kappa	KDDCup99	99.9
[44], 2022	**AdaBoost**, MLP, SVM	Correlation Coefficient	Accuracy, Recall, Precision, F1-score	UNSW-NB15	99.3
[45], 2022	**SVM**	PCA	TPR, FNR, FPR, TNR, Accuracy, Precision, Sensitivity, F1-score	UNSW-NB15	93.94
				KDDCup99	99.11
[46], 2022	**CNN+LSTM**, CNN, LSTM, LSTM+CNN	SelectKBest	Accuracy, FAR, DR, Precision, F1-score	WSN-DS	99.67
				UNSW-NB15	94.53
				CICIDS2017	99.64
[47], 2022	**RF**, DT, SVM	SMO-HPSO	Precision, Recall, F1-score, Accuracy, FAR, AUC, MCC	NSL-KDD	99.175
				UNSW-NB15	99.18
[48], 2022	**DNN**, LR, DT, KNN, RF, SVM, CNN, Adaboost, LSTM	PSO	Accuracy, Precision, Recall, F1-score	WUSTL EHMS 2020	96
[49], 2022	**DT**, MLP, **GBT**, GRU, LSTM, AdaBoost	GIWRF	Accuracy, Precision, FPR, F1-score, Recall	TON_IoT	GBT=99.98
				UNSW-NB15	DT=93.01
[50], 2022	(**NB+SVM+KNN+RF**)	CFS–FPA	Accuracy, FAR, F1-score, DR, Precision	CICIDS2017	99.7
[51], 2021	**DT**	Entropy	Accuracy, DR, FAR	NSL-KDD	99.42
				CICIDS2017	98.80
[52], 2021	**ANN,** NB, AdaBoost, SVM	PCA	Accuracy, Precision, Recall, F1-score	NSL-KDD	99.91
				CICIDS2017	97.69
[53], 2021	**DNN,** RNN	RFE	Accuracy, TPR, FPR, Precision, Recall, F1-score	NSL-KDD	94
[54], 2021	**NB+SVM**	ICFS	Accuracy, Precision, Recall, F1-score, DR, FAR	NSL-KDD	88.97
[55], 2021	**DT**, KNN, NB, SVM	PSO	Accuracy, F1-score, Recall, Precision	NSL-KDD	99.38
[56], 2021	**RF**	GWO+PSO	Accuracy, Recall, Precision, F1-score	CICIDS2017	99.88
				NSL-KDD	99.24
				KDDCup99	99.66
[57], 2020	**PSO-PNN**	PCA	DR, Recall, FPR, Precision, F1-score	UNSW-NB15	99.4 (DR)
[58], 2020	**DT+SVM**	PSO	Accuracy, DR, FAR	KDDCup99	99.2
[59], 2020	**KNN**, **PSO,** SVM, DT, NB, and LR	RF	F1-score, TPR, TNR, FPR, FNR, Precision, Accuracy	NSL-KDD	KNN=97.18 PSO=99.32
[60], 2020	**ANN**	PCA	Accuracy, Recall, Precision, F1-score	NSL-KDD	97.89
				UNSW-NB15	89.99
				CSE-CICIDS2018	76.47
[61], 2019	**RF**	Gini	Accuracy	NSL-KDD	99.88

IV. Discussion and Comparison

In the context of intrusion detection systems, a variety of research approaches have been employed by diverse researchers for the purpose of analyzing network traffic, thereby improving network security. These intelligent systems use various classification algorithms to obtain prediction models trained on various IDS benchmark datasets. These datasets contain different types of network attacks. In addition, they comprise a large number of features representing training data. Despite these datasets containing a large number of features, the presence of relevant features for the trained model affects the detection performance. To address this issue, feature selection techniques are key. They are employed to maintain only the most important ones, resulting in an improvement in classification performance, including a decrease in computational complexity as well as increasing detection rate which enhances the model's ability to distinguish meaningful patterns from noisy data.

This paper offers a comprehensive comparison among the state-of-the-art IDS that utilized classification algorithms in cooperation with FS methods. In the previous table, TABLE II, each entry outlines the classification algorithm used, the feature selection technique applied, evaluation metrics utilized, datasets employed, and the resulting accuracy percentages. The table demonstrates a wide array of classification algorithms and feature selection techniques utilized in IDS. Moreover, it shows that intrusion detection is rich with methodological diversity, starting from traditional algorithms like DT and SVM to more modern approaches such as DNN and RNN. This diversity reflects ongoing efforts by researchers to reach the optimal model for detecting various types of cyber threats. Feature selection plays a crucial role in enhancing the performance of

IDS. Techniques like Correlation Coefficient, RFE, and PSO are commonly employed to identify relevant features. The choice of feature selection technique often depends on the characteristics of the dataset and the specific requirements of the detection task.

The evaluation metrics used to assess the performance of IDS models include traditional metrics such as Accuracy, Precision, Recall, and F1-score, as well as more specialized metrics like TPR, FPR, and FAR. These metrics provide comprehensive insights into the model's ability to accurately classify instances of both normal and malicious activities. Additionally, the use of metrics like Kappa coefficient and AUC reflects a deep understanding of model performance beyond simple accuracy metrics. The choice of datasets, including NSL-KDD, KDDcup, UNSW-NB15, CICIDS2017, and TON_IoT, highlights the importance of benchmarking and generalizability in IDS research. These datasets comprise a wide range of network traffic scenarios and attack types, allowing researchers to evaluate the robustness and effectiveness of intrusion detection models under varying conditions. Moreover, the consistent use of well-known datasets enables meaningful comparisons between different approaches and facilitates the identification of the best approach for detecting the intruder. Focusing solely on high accuracy on a single dataset might not translate well to unseen attack scenarios. The studies evaluating models on multiple datasets provide a better assessment of generalizability.

The obtained accuracy results vary across different studies, ranging from 76.47% to nearly 100%. The performance of IDS models can be influenced by factors such as classification algorithms, feature selection, and hyperparameter tuning, highlighting the need for continued research and innovation in the field. The accuracy results are impressive, with many studies reporting results above 99%. This suggests that the combination of sophisticated algorithms and feature selection techniques can lead to highly accurate models. However, it's important to note that high accuracy does not always guarantee a model's effectiveness in real-world scenarios, where data may be more imbalanced or noisy.

From the above table, it can be observed that DL models, particularly those combining LSTM with other neural network architectures, appear to achieve high accuracy, indicating their potential for capturing complex patterns in data. Additionally, ensemble methods and hybrid models are also prominent, suggesting that combining multiple models or techniques can enhance performance.

As listed in the summarization table (see TABLE II), some works perform better than others in terms of having higher accuracies since they utilize the proper classification algorithm and the feature selection technique that fits the used dataset. For instance, by comparing the results obtained by the works that employed NSL-KDD dataset (see Fig. 4), the highest accuracy achieved is 99.95% which was by [33], outperforming all other studies including [34], [37], [38], [39], [40], [41], [47], [51], [52], [53], [54], [55], [56], [59], [60], and [61]. This indicates the power of deep learning as a classifier, as well as a feature selector. Moreover, showing the power of using a hybrid feature selection method including GWO and PSO, the work of [56] obtained the highest result when CICID2017 dataset was utilized (see Fig. 5), outperforming the other works including [36], [38], [46], [50], [51], and [52]. Furthermore, for the UNSW-NB15, the two works [41] and [44] provide the same highest accuracy of 99.3%, surpassing the works of [35], [45], [46], [47], [49], [57], and [60] (see Fig. 6). Another less frequent dataset, KDDCup99, is used by 6 works including [33], [41], [43], [45], [56], and [58]. Notably, the highest reported accuracy 99.99% was obtained by [33] (see Fig. 7). Finally, the remaining datasets are rarely used, indicating the need to apply more work to those datasets.

Based on the comparison table above, as shown in Fig. 3, it is worth mentioning that the most utilized dataset is NSL-KDD, which was employed by 17 works out of a total of 29 reviewed studies. The reason behind this is due to its pre-processing, realism, balanced class distribution, and public availability. Derived from the KDDCup99 dataset, it offers a mix of normal and attack instances while addressing issues such as redundancy and class imbalance, making it suitable for training and evaluating machine learning models. However, it's important to acknowledge that while the NSL-KDD dataset provides valuable insights, it may not fully capture the complexity of real-world network traffic, necessitating the use of multiple datasets for comprehensive evaluation. Moreover, the most applied feature selection technique is the "Correlation Coefficient", since it has the ability to identify redundant features, it is simple to calculate, and effectively calculates the strength of the relationship between each feature with the other as well as with the class target.

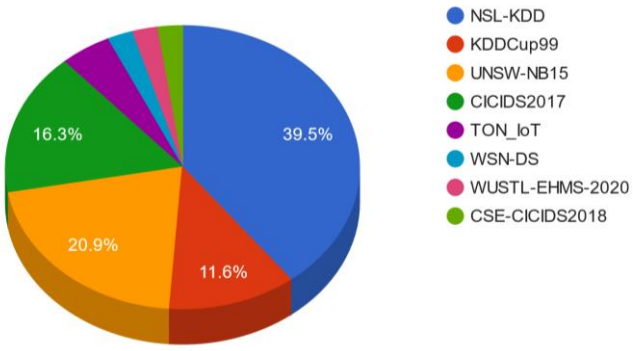

Fig. 3. Dataset Utilization Percentages

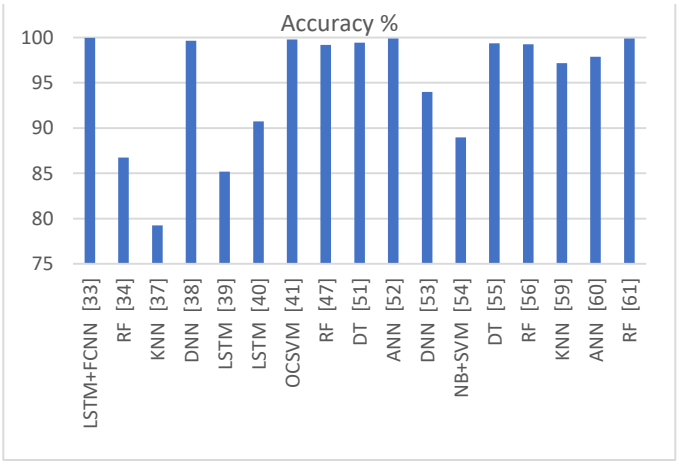

Fig. 4. Accuracy Comparison of Works Utilizing NSL-KDD

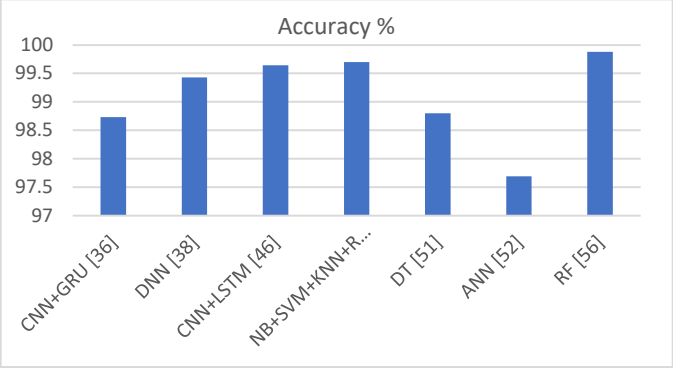

Fig. 5. Accuracy Comparison of Works Utilizing CICIDS2017

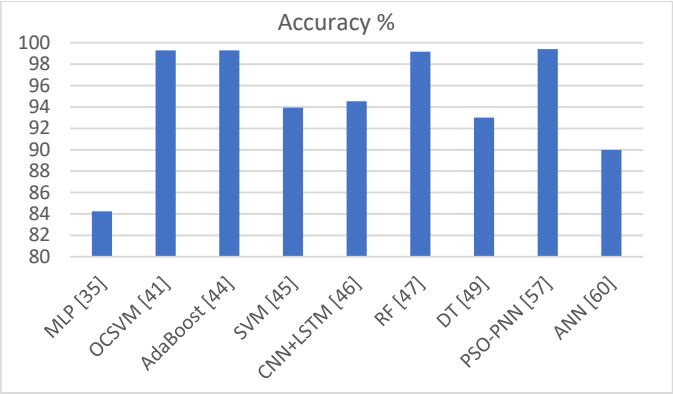

Fig. 6. Accuracy Comparison of Works Utilizing UNSW-NB15

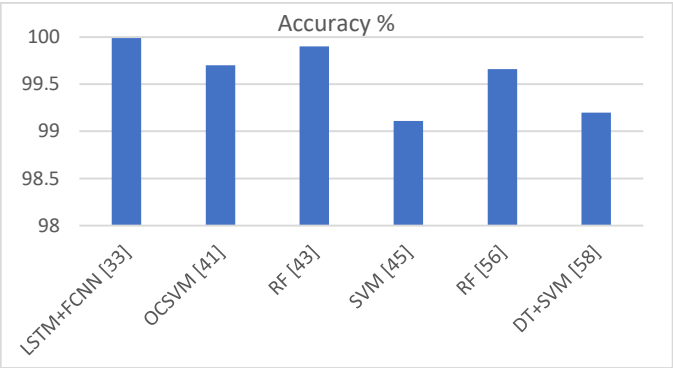

Fig. 7. Accuracy Comparison of Works Utilizing KDDCup99

In the realm of IDSs, researchers encounter several key challenges that must be addressed. Among these challenges, dataset imbalance, evaluation metric standardization, and overfitting stand out as crucial areas requiring attention to gain valuable insights into the current state of IDS research. Dataset imbalance is a major concern. Imbalanced datasets pose a significant challenge in IDS because they can skew the performance of machine learning models. In cybersecurity data, there is often a large amount of one class compared to a smaller number of another class. This imbalance can lead to models that are biased towards the majority class, resulting in poor detection of minority classes. Resampling methods like oversampling the minority class or undersampling the majority class are common strategies to address this issue, allowing the model to learn more effectively from the minority class samples and make better predictions for both classes.

Another challenge lies in the lack of standardized evaluation metrics. Researchers often employ various metrics, making it difficult to compare the performance of different IDS approaches. This inconsistency hinders the ability to definitively assess the effectiveness of new methods. Establishing a common set of evaluation metrics, along with clear reporting guidelines, would significantly improve the comparability and transparency of IDS research.

Finally, overfitting remains a persistent threat. Overfitting occurs when a model learns the training data too well, including its noise and outliers, which reduces its ability to generalize to new, unseen data. This is particularly problematic in IDS, where the model needs to detect novel attack patterns. Overfitting can lead to a high number of false alarms, reducing the reliability of the IDS3. Addressing overfitting requires careful selection of model architectures, regularization techniques, cross-validation, and the use of diverse, well-balanced datasets.

V. Conclusion and Future Directions

This paper provides an overview of various IDSs that utilize ML and DL to predict network attacks for enhanced network security. It explores how these systems choose relevant features and employ different datasets. The analysis reveals the considerable advancement and diversity in this area, showing diverse approaches in leveraging classification algorithms to improve prediction capabilities. This review underscores the pivotal role of feature selection in refining data classification accuracy, thereby enhancing the overall efficiency of intrusion detection systems. Furthermore, the paper delves into dataset selection, demonstrating the importance of using datasets that reflect real-world scenarios. The review also outlines various evaluation criteria for systematically assessing the performance of the studies. By considering metrics like Accuracy, Recall, and F1-score, among others, it offers a comprehensive insight into the strengths of different classification methods and feature selection techniques within the realm of network intrusion detection. The comparison showed that DL and ensemble methods are promising directions for achieving high accuracy in IDS, as well as that feature selection plays a vital role in improving model performance and interoperability. Furthermore, a diverse set of evaluation metrics is necessary for robust IDS evaluation, especially when dealing with imbalanced datasets. Additionally, utilizing more recent and comprehensive benchmark datasets strengthens the generalizability of the findings.

For future research directions, we propose exploring novel feature selection methods by integrating optimization-based techniques with statistical and machine learning-based approaches (for example, a combination of RF and PSO) can select the optimal feature subset. Furthermore, incorporating bagging and ensemble learning strategies such as Bagged Neural Networks could significantly boost model performance. Finally, to improve the generalizability, the researcher could develop a dataset that combines more than one dataset. While this approach offers significant benefits, it also introduces new challenges.

REFERENCES

[1] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," Expert Syst Appl, vol. 42, no. 5, pp. 2670–2679, Apr. 2015, doi: 10.1016/j.eswa.2014.11.009.

[2] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," Applied Soft Computing Journal, vol. 10, no. 1, pp. 1–35, 2010, doi: 10.1016/j.asoc.2009.06.019.

[3] H. J. Liao, C. H. Richard Lin, Y. C. Lin, and K. Y. Tung, "Intrusion detection system: A comprehensive review," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 16–24, 2013, doi: 10.1016/j.jnca.2012.09.004.

[4] C. H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," Pattern Recognit, vol. 40, no. 9, pp. 2373–2391, 2007, doi: 10.1016/j.patcog.2006.12.009.

[5] S. Subbiah, K. S. M. Anbananthen, S. Thangaraj, S. Kannan, and D. Chelliah, "Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm," Journal of Communications and Networks, vol. 24, no. 2, pp. 264–273, Apr. 2022, doi: 10.23919/jcn.2022.000002.

[6] S. Ganapathy, K. Kulothungan, S. Muthurajkumar, M. Vijayalakshmi, L. Yogesh, and A. Kannan, "Intelligent feature selection and classification techniques for intrusion detection in networks: A survey," EURASIP J Wirel Commun Netw, vol. 2013, no. 1, pp. 1–16, 2013, doi: 10.1186/1687-1499-2013-271.

[7] E. E. Abdallah, W. Eleisah, and A. F. Otoom, "Intrusion Detection Systems using Supervised Machine Learning Techniques: A survey," in Procedia Computer Science, Elsevier B.V., 2022, pp. 205–212. doi: 10.1016/j.procs.2022.03.029.

[8] A. A. Saleem, M. M. Hassan, and I. A. Ali, "INTELLIGENT HOME: EMPOWERING SMART HOME WITH MACHINE LEARNING FOR USER ACTION PREDICTION," Science Journal of University of Zakho, vol. 11, no. 3, pp. 403–420, Aug. 2023, doi: 10.25271/sjuoz.2023.11.3.1145.

[9] P. K. Singh, A. K. Kar, Y. Singh, M. H. Kolekar, and S. Tanwar, Eds., Proceedings of ICRIC 2019: Recent Innovations in Computing, First Edit. Springer Nature, 2020. doi: https://doi.org/10.1007/978-3-030-29407-6.

[10] A. A. Saleem, M. M. Hassan, and I. A. Ali, "Smart Homes Powered by Machine Learning: A Review," Proceedings of the 2nd 2022 International Conference on Computer Science and Software Engineering, CSASE 2022, pp. 355–361, 2022, doi: 10.1109/CSASE51777.2022.9759682.

[11] A. A. Salih and A. M. Abdulazeez, "Evaluation of Classification Algorithms for Intrusion Detection System: A Review," Journal of Soft Computing and Data Mining, vol. 2, no. 1, pp. 31–40, Apr. 2021, doi: 10.30880/jscdm.2021.02.01.004.

[12] M. Bertolini, D. Mezzogori, M. Neroni, and F. Zammori, "Machine Learning for industrial applications: A comprehensive literature review," Expert Syst Appl, vol. 175, no. February, p. 114820, 2021, doi: 10.1016/j.eswa.2021.114820.

[13] C. Kalimuthan and J. Arokia Renjit, "Review on intrusion detection using feature selection with machine learning techniques," Mater Today Proc, vol. 33, no. xxxx, pp. 3794–3802, 2020, doi: 10.1016/j.matpr.2020.06.218.

[14] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative Analysis of Intrusion Detection Systems and Machine Learning-Based Model Analysis Through Decision Tree," IEEE Access, vol. 11, pp. 80348–80391, 2023, doi: 10.1109/ACCESS.2023.3296444.

[15] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[16] N. Unnisa A, M. Yerva, and K. M Z, "Review on Intrusion Detection System (IDS) for Network Security using Machine Learning Algorithms," International Research Journal on Advanced Science Hub, vol. 4, no. 03, pp. 67–74, Mar. 2022, doi: 10.47392/irjash.2022.014.

[17] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Transactions on Emerging Telecommunications Technologies, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.

[18] S. Choudhary and N. Kesswani, "Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT," in Procedia Computer Science, Elsevier B.V., 2020, pp. 1561–1573. doi: 10.1016/j.procs.2020.03.367.

[19] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Transactions on Emerging Telecommunications Technologies, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.

[20] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[21] H. Bostani and M. Sheikhan, "Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems," Soft comput, vol. 21, no. 9, pp. 2307–2324, 2017, doi: 10.1007/s00500-015-1942-8.

[22] G. Cybenko and T. G. Allen, "Parallel Algorithms For Classification And Clustering," Advanced Algorithms and Architectures for Signal Processing II, vol. 0826, no. 4, p. 126, 1988, doi: 10.1117/12.942023.

[23] P. Dhal and C. Azad, "A comprehensive survey on feature selection in the various fields of machine learning," Applied Intelligence, vol. 52, no. 4, pp. 4543–4581, Mar. 2022, doi: 10.1007/s10489-021-02550-9.

[24] A. Salappa, M. Doumpos, and C. Zopounidis, "Feature selection algorithms in classification problems: An experimental evaluation," Optim Methods Softw, vol. 22, no. 1, pp. 199–212, 2007, doi: 10.1080/10556780600881910.

[25] Y. Chen, Y. Li, X. Q. Cheng, and L. Guo, "Survey and Taxonomy of Feature Selection Algorithms in Intrusion Detection System," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 4318 LNCS, pp. 153–167, 2006, doi: 10.1007/11937807_13.

[26] G. Qu, S. Hariri, and M. Yousif, "A new dependency and correlation analysis for features," IEEE Trans Knowl Data Eng, vol. 17, no. 9, pp. 1199–1206, 2005, doi: 10.1109/TKDE.2005.136.

[27] M. A. Ambusaidi, X. He, Z. Tan, P. Nanda, L. F. Lu, and U. T. Nagar, "A novel feature selection approach for intrusion detection data classification," Proceedings - 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2014, pp. 82–89, 2015, doi: 10.1109/TrustCom.2014.15.

[28] B. Xue, L. Cervante, L. Shang, W. N. Browne, and M. Zhang, "A multi-objective particle swarm optimisation for filter-based," Connection Science 31:4, vol. 24, no. September, pp. 91–116, 2012.

[29] F. Salo, A. B. Nassif, and A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," Computer Networks, vol. 148, pp. 164–175, 2019, doi: 10.1016/j.comnet.2018.11.010.

[30] B. Xue, A. K. Qin, and M. Zhang, "An archive based particle swarm optimisation for feature selection in classification," Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014, pp. 3119–3126, 2014, doi: 10.1109/CEC.2014.6900472.

[31] I. Ahmad, "Feature selection using particle swarm optimization in intrusion detection," Int J Distrib Sens Netw, vol. 2015, 2015, doi: 10.1155/2015/806954.

[32] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," J Comput Sci, vol. 25, pp. 152–160, 2018, doi: 10.1016/j.jocs.2017.03.006.

[33] A. Chakrawarti, D. Shiv, and S. Shrivastava, "International Journal of INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING Intrusion Detection System using Long Short-Term Memory and Fully Connected Neural Network on Kddcup99 and NSL-KDD Dataset," International Journal of Intelligent Systems and Applications in Engineering IJISAE, vol. 11, no. 9s, pp. 621–635, 2023, [Online]. Available: www.ijisae.org

[34] Srinath Venkatesan, "Design an Intrusion Detection System based on Feature Selection Using ML Algorithms," Mathematical Statistician and

Engineering Applications, vol. 72, no. 1, pp. 702–710, 2023, [Online]. Available: http://philstat.org.ph

[35] Y. Yin et al., "IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset," J Big Data, vol. 10, no. 1, Dec. 2023, doi: 10.1186/s40537-023-00694-8.

[36] A. Henry et al., "Composition of Hybrid Deep Learning Model and Feature Optimization for Intrusion Detection System," Sensors, vol. 23, no. 2, Jan. 2023, doi: 10.3390/s23020890.

[37] S. Konde and S. B. Deosarkar, "A NOVEL INTRUSION DETECTION SYSTEM (IDS) FRAMEWORK FOR AGRICULTURAL IOT NETWORKS," J Theor Appl Inf Technol, vol. 15, no. 21, 2023, [Online]. Available: www.jatit.org

[38] H. Azzaoui, A. Z. E. Boukhamla, D. Arroyo, and A. Bensayah, "Developing new deep-learning model to enhance network intrusion classification," Evolving Systems, vol. 13, no. 1, pp. 17–25, Feb. 2022, doi: 10.1007/s12530-020-09364-z.

[39] A. Sunyoto and Hanafi, "Enhance Intrusion Detection (IDS) System Using Deep SDAE to Increase Effectiveness of Dimensional Reduction in Machine Learning and Deep Learning," International Journal of Intelligent Engineering and Systems, vol. 15, no. 4, pp. 125–141, 2022, doi: 10.22266/ijies2022.0831.13.

[40] Y. Fu, Y. Du, Z. Cao, Q. Li, and W. Xiang, "A Deep Learning Model for Network Intrusion Detection with Imbalanced Data," Electronics (Switzerland), vol. 11, no. 6, Mar. 2022, doi: 10.3390/electronics11060898.

[41] H. Alazzam, A. Sharieh, and K. E. Sabri, "A lightweight intelligent network intrusion detection system using OCSVM and Pigeon inspired optimizer," Applied Intelligence, vol. 52, no. 4, pp. 3527–3544, Mar. 2022, doi: 10.1007/s10489-021-02621-x.

[42] I. Hidayat, M. Z. Ali, and A. Arshad, "Machine Learning-Based Intrusion Detection System: An Experimental Comparison," Journal of Computational and Cognitive Engineering, Jul. 2022, doi: 10.47852/bonviewJCCE2202270.

[43] S. Subbiah, K. S. M. Anbananthen, S. Thangaraj, S. Kannan, and D. Chelliah, "Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm," Journal of Communications and Networks, vol. 24, no. 2, pp. 264–273, Apr. 2022, doi: 10.23919/jcn.2022.000002.

[44] I. Ahmad, Q. E. U. Haq, M. Imran, M. O. Alassafi, and R. A. Alghamdi, "An Efficient Network Intrusion Detection and Classification System," Mathematics, vol. 10, no. 3, Feb. 2022, doi: 10.3390/math10030530.

[45] M. A. Almaiah et al., "Performance Investigation of Principal Component Analysis for Intrusion Detection System Using Different Support Vector Machine Kernels," Electronics (Switzerland), vol. 11, no. 21, Nov. 2022, doi: 10.3390/electronics11213571.

[46] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," IEEE Access, vol. 10, pp. 99837–99849, 2022, doi: 10.1109/ACCESS.2022.3206425.

[47] S. Ethala and A. Kumarappan, "A Hybrid Spider Monkey and Hierarchical Particle Swarm Optimization Approach for Intrusion Detection on Internet of Things," Sensors, vol. 22, no. 21, Nov. 2022, doi: 10.3390/s22218566.

[48] R. Chaganti, A. Mourade, V. Ravi, N. Vemprala, A. Dua, and B. Bhushan, "A Particle Swarm Optimization and Deep Learning Approach for Intrusion Detection System in Internet of Medical Things," Sustainability (Switzerland), vol. 14, no. 19, Oct. 2022, doi: 10.3390/su141912828.

[49] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique," Cybersecurity, vol. 5, no. 1, Dec. 2022, doi: 10.1186/s42400-021-00103-8.

[50] D. N. Mhawi, A. Aldallal, and S. Hassan, "Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems," Symmetry (Basel), vol. 14, no. 7, Jul. 2022, doi: 10.3390/sym14071461.

[51] A. Guezzaz, S. Benkirane, M. Azrour, and S. Khurram, "A Reliable Network Intrusion Detection Approach Using Decision Tree with Enhanced Data Quality," Security and Communication Networks, vol. 2021, 2021, doi: 10.1155/2021/1230593.

[52] M. Choraś and M. Pawlicki, "Intrusion detection approach based on optimised artificial neural network," Neurocomputing, vol. 452, pp. 705–715, Sep. 2021, doi: 10.1016/j.neucom.2020.07.138.

[53] B. Mohammed and E. Gbashi, "Intrusion Detection System for NSL-KDD Dataset Based on Deep Learning and Recursive Feature Elimination," Engineering and Technology Journal, vol. 39, no. 7, pp. 1069–1079, Jul. 2021, doi: 10.30684/etj.v39i7.1695.

[54] T. Wisanwanichthan and M. Thammawichai, "A Double-Layered Hybrid Approach for Network Intrusion Detection System Using Combined Naive Bayes and SVM," IEEE Access, vol. 9, pp. 138432–138450, 2021, doi: 10.1109/ACCESS.2021.3118573.

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

[56] P. K. Keserwani, M. C. Govil, E. S. Pilli, and P. Govil, "A smart anomaly-based intrusion detection system for the Internet of Things (IoT) network using GWO–PSO–RF model," J Reliab Intell Environ, vol. 7, no. 1, pp. 3–21, Mar. 2021, doi: 10.1007/s40860-020-00126-x.

[57] M. Rabbani, Y. L. Wang, R. Khoshkangini, H. Jelodar, R. Zhao, and P. Hu, "A hybrid machine learning approach for malicious behaviour detection and recognition in cloud computing," Journal of Network and Computer Applications, vol. 151, Feb. 2020, doi: 10.1016/j.jnca.2019.102507.

[58] A. Kumari and A. K. Mehta, "A Hybrid Intrusion Detection System Based on Decision Tree and Support Vector Machine," in 2020 IEEE 5th International Conference on Computing Communication and Automation, ICCCA 2020, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 396–400. doi: 10.1109/ICCCA49541.2020.9250753.

[59] N. Kunhare, R. Tiwari, and J. Dhar, "Particle swarm optimization and feature selection for intrusion detection system," Sadhana - Academy Proceedings in Engineering Sciences, vol. 45, no. 1, pp. 1–14, Dec. 2020, doi: 10.1007/S12046-020-1308-5/METRICS.

[60] Gabriel Chukwunonso Amaizu, Cosmas Ifeanyi Nwakanma, Jae-Min Lee, and Dong-Seong Kim, "Investigating Network Intrusion Detection Datasets Using Machine Learning," ICTC 2020 : the 11th International Conference on ICT Convergence : "Data, Network, and AI in the Age of 'Untact'" , 2020, doi: 10.1109/ICTC49870.2020.9289329.

[61] Y. and M. R. Negandhi Prashil and Trivedi, "Intrusion Detection System Using Random Forest on the NSL-KDD Dataset," in Emerging Research in Computing, Information, Communication and Applications, L. M. and N. H. C. and H. P. N. and N. N. Shetty N. R. and Patnaik, Ed., Singapore: Springer Singapore, 2019, pp. 519–531.