الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research
المدرسة الوطنية العليا للتكنولوجيات المتقدمة
National Higher School of Advanced Technologies

Department of Electrical Engineering and Industrial Computing

Final Year Project to Obtain the Diploma of Master

– Field –

Telecommunication

– Specialty –

Teleommunications Systems and Networking

– Subject –

A State of The Art: AI-Enabled Intrusion Detection Systems

Realized by

AIT MIMOUNE Yasmine

REBAHI Khadidja

Presented publicly, the 22 / 06 /2025

Members of The Jury :

Name	Establishment	Grade	Quality
Mrs. BOUTERFAS Malika	ENSTA	MCA	President
Mrs. LAKHDARI Kheira	ENSTA	MCB	Supervisor
Mr. BEGHAMI Sami	ENSTA	MAA	Examinator
Mr. CHERIFI Tarek	ENSTA	MCA	Examinator

2024 / 2025

State of the Art: AI-Enabled Intrusion Detection Systems

Yasmine AIT MIMOUNE[1] , Khadidja REBAHI[2]
[1,2]*Electrical Engineering and Industrial Computing*
National Higher School of Advanced Technologies
Algiers, Algeria
[1]y_aitmimoune@ensta.edu.dz, [2]k_rebahi@ensta.edu.dz

Abstract

As digital networks expand at an unprecedented rate, alongside with the progression of cyber-attacks which present significant challenges to traditional security measures. Although intrusion detection systems (IDSs) have long served as the primary line of defense, they often struggle to keep pace with novel ,zero-day, and unknown threats. Recent advancements in Artificial Intelligence—particularly through ML and and DL approaches—By learning from evolving attack patterns, AI-powered IDS can dynamically adapt, offering faster and more precise threat detection. This paper provides a state of the art review of AI-enabled IDS approaches by examining architectures, detection techniques, and performance metrics across a range of benchmark datasets.Our comparative analysis highlights the superior accuracy of deep learning approaches on modern datasets, while also examining the impact of dataset quality, detection of rare attacks, and model efficiency.It also analysis demonstrates that while AI-driven methods markedly enhance detection accuracy and reduce false alarm rates, persistent challenges remain, especially in reliably classifying rare and novel attack types due to imbalanced datasets and computational constraints.This study offers valuable insights for future advancements toward robust, real-world intrusion detection systems.

Keywords: Intrusion Detection Systems (IDS); Machine Learning; Deep Learning;Attacks;Cyber Security.

1 Introduction

In today's digital world, rapid technological advancements and the massive volume of cloud-handled data have significantly reshaped cybersecurity strategies. Intrusion detection systems (IDS) have evolved since the late 1980s; early systems struggled with high resource demands and failed to detect zero-day attacks. In the 1990s, anomaly detection, which focused on detecting unusual activity patterns instead of relying on known threat signatures; however, variable network traffic led to high false alarm rates and reduced reliability.

Recent advances in network infrastructure, computational power, and machine learning have refreshed IDS capabilities. Modern AI-based systems continuously learn from network data to improve threat detection while aiming to minimize false positives. Despite these improvements, challenges remain, especially when applying these models to diverse datasets and dynamic real-world environments, particularly in terms of detection accuracy, false positive rates, and computational efficiency.

This paper presents a state of the art review of AI-enabled IDS, focusing on two main criteria: *Algorithm Performance*—comparing different AI models across various IDS datasets—and *Practical Limitations*—exploring the challenges current AI-based IDS face in dynamic network settings.

2 Intrusion Detection Systems (IDS)

Cyber threats aim to compromise systems by stealing, altering, or disabling data and services. They are commonly categorized into four types [1]: DoS/DDoS, Probe, U2R, and R2L attacks. DoS/DDoS flood resources, Probe scans networks for vulnerabilities, U2R and R2L exploit access levels to gain unauthorized control, often bypassing detection [2].

An IDS is a security tool, hardware or software, that monitors network traffic and raises alarms when it detects malicious activity by logging traffic and providing real-time details. There are three types of IDS: Network IDS (NIDS), Host IDS (HIDS) and Hybrid IDS [3]. NIDS are designed to continuously monitor network traffic to detect various threats, including DDoS attacks, unauthorized attempts, and port scanning activities [3]. NIDS face challenges in processing large volumes of data, particularly when encryption is used. In contrast, HIDS operates directly on individual devices or hosts [4]. They monitor local system logs, file integrity, and system calls to detect suspicious activities. Hybrid systems integrate the strengths of both HIDSs and NIDSs by combining detailed insights from individual hosts with a broad perspective on overall network activity [5], [6]. Figure 1 illustrates how the three types of Intrusion Detection Systems can be integrated to operate collaboratively within a network infrastructure.

An IDS architecture could be either *Centralized*—consolidate all monitoring data at a single location for unified analysis and management—or *Distributed*—deploy independent monitoring nodes across different network segments, enabling localized threat detection and improved scalability. Hybrid IDS models combine the strengths of centralized and distributed systems; they distribute detection tasks among various nodes while centrally aggregating data for analysis. For IDS types , a classification taxonomy is given in Figure 2 [7] , this classification is based on the the perspective of IDS deployment or detection methods.There are two main

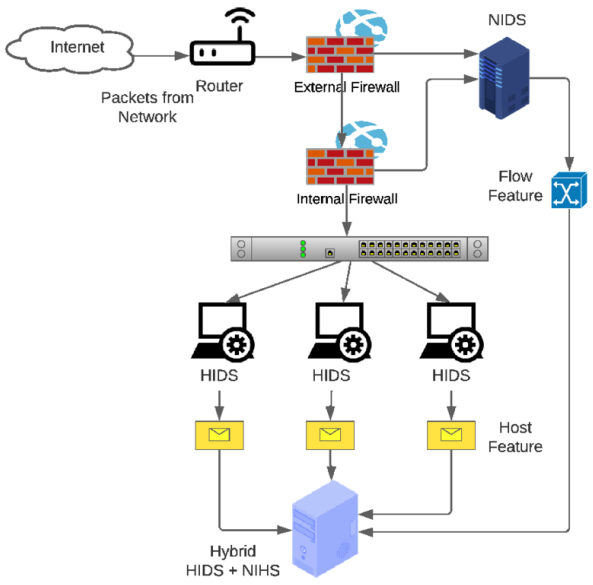

Figure 1: Types of IDS.

detection techniques for IDS, *Misuse-based* and *Anomaly-based*. Misuse-based detection comprises two subcategories [8]: the first one being *Signature-Based Detection*, which relies on a predefined database of attack signatures, raising alerts when a match is found. The second approach is *ML-Based Misuse Detection*, developed to overcome the rigidity of signature-based systems; ML models learn from historical attack data, understanding the general structure of known attacks and potentially predicting evolving variants. *Anomaly-Based Detection* builds a model of normal behavior by analyzing network traffic features. Deviations from this model are flagged as anomalies, indicating potential security threats [4]. It can be implemented usin ML, statistical techniques, or finite-state machines [8]. For better efficiency, *Hybrid systems* were developed to combine misuse-based and anomaly-based methods, using signature matching to detect known threats and anomaly detection to identify novel attacks [9].

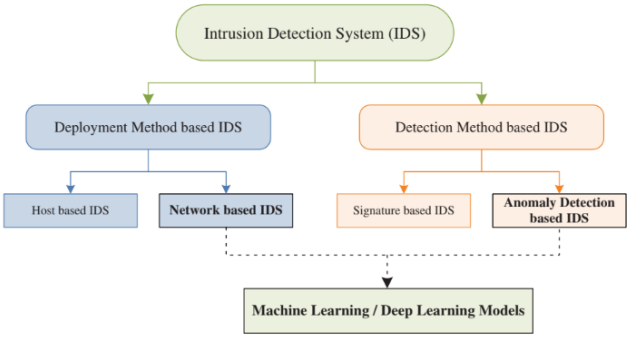

Figure 2: IDS Classification [7].

Figure 3 shows the differences in how the two techniques Misuse-based and Anomaly-based generate the alerts.

Table 1: Summary of Intrusion Detection Systems Characteristics [10]

Characteristic	**Signature-based IDS**	**Anomaly-based IDS**
Detection Capability	Uses predefined signatures and contextual analysis to recognize known threats.	Detects both known and unknown attacks by identifying deviations from established network norms.
System Dependency	Relies on specific system software and OS details to identify vulnerabilities.	Less dependent on system-specific details; instead, it focuses on overall network traffic patterns.
Update Requirements	Requires regular updates of its signature database to remain effective.	Builds dynamic profiles of normal network behavior, eliminating the need for constant database updates.
Protocol Analysis	Offers limited in-depth protocol insights.	Performs comprehensive protocol analysis to examine detailed packet information.

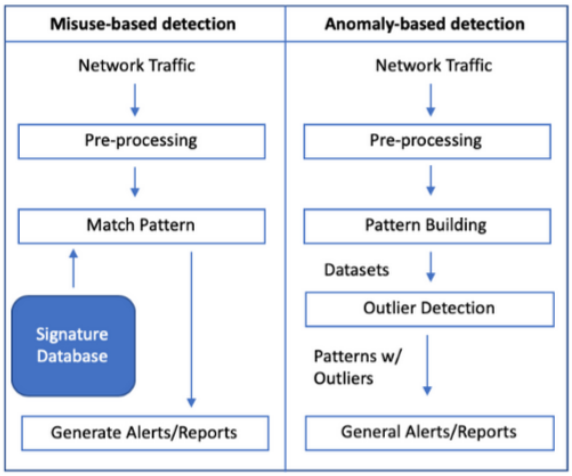

Figure 3: Comparison of IDS Techniques

Modern cybersecurity demands more than passive detection (IDS). Organizations increasingly adopt Intrusion Prevention Systems (IPS), which actively block attacks by operating inline and often merging firewall and IDS functionalities [4]. While this enhances protection, IPSs can generate false positives that block legitimate traffic and risk DoS conditions. As potential single points of failure, especially in network setups, IPSs must remain robust and stable [4].

3 AI-Based Intrusion Detection Systems

AI-based Intrusion Detection Systems is using AI approaches to identify malicious activities within a network. By learning from how network traffic behaves, these systems can spot both known and new threats when they happen.

3.1 Machine Learning Models

Machine learning algorithms are used in IDS by learning from historical data, spot patterns, and make smart decisions. These models are categorized into supervised, unsupervised, and semi-supervised learning. The diagram in Figure 4 illustrates this categorization.

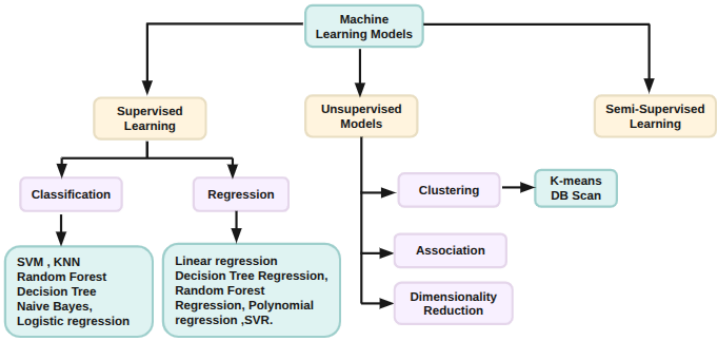

Figure 4: Machine learning Models Categorization.

Supervised machine learning: The system is trained on a labeled datasets where we know both the input data and corresponding output. The model learns how different features relate to specific outcomes, so it can recognize and classify new data. The supervised learning is divided into two categories: classification techniques, where the output is a discrete class label. The common classification algorithms are: support vector machines which finds the optimal hyperplane that separates data points of different classes, making it effective for high-dimensional and sparse data classification, K-Nearest Neighbor algorithm (KNN), which classifies a given data sample based on feature similarity [10]. Decision trees have a conventional tree structure for rule-based classification, it can represent the result of a test of one or more attributes [7]. Random Forest (RF) is based on building multiple decision trees using the bagging method to enhance classification accuracy and reduce overfitting [11]. Naïve Bayes classification estimates feature likelihood from data using Bayes' theorem [10].Logistic regression predicts probabilities and class labels using the logit (sigmoid) function. On the other hand, we have regression techniques where the output is predicted as a continuous value , by modeling the relationship between independent variables and a dependent variable [4]. Common regression algorithms include Linear Regression, Decision Tree Regression, Random Forest Regression, Polynomial regression and Support Vector Regression (SVR).

Unsupervised machine learning: Used when labeled data are unavailable, in this model the algorithm independently identifies hidden patterns and relationships within the data. Typically, it groups data points based on their similarities or differences. This type of learning is effective for analyzing large datasets in tasks such as clustering algorithms like k-means and DB scan that group similar data points together, association rule learning when the model looks for relationships between variables, and dimensionality reduction that reduces the number of variables in data without losing significant information [4].

Semi-supervised machine learning: A combination of supervised and unsupervised machine learning models. The dataset is partially labeled in semi-supervised learning. The ability to use methods and algorithms from both forms of machine learning is one of the primary benefits of semi-supervised learning. To improve accuracy, new machine learning algorithms can be developed. Additionally, because semi-supervised learning does not need the usage of a whole set of labeled data, it takes less time. Nevertheless, semi-supervised learning also bears the drawbacks of the two methods mentioned above [4].

3.2 Deep Learning Models

Kimanzi et al. [12] define deep learning as a subset of machine learning that is inspired by biological neural networks. It interprets, classifies, and organizes data into different categories, mimicking how the brain processes information. Deep learning incorporates artificial neural networks (ANNs) with multiple layers, allowing different levels of abstraction to extract complex patterns. The key characteristic lies in its deep structure, consisting of multiple hidden layers that enable automatic feature extraction from raw data, which has also made significant contributions to the development of AI-based IDS. Common deep learning models include convolutional neural networks (CNNs) for analyzing structured network data. CNNs process packet headers and log files to detect intrusion attempts and can automatically learn relevant features from raw traffic data, reducing the need for manual feature engineering [13]. Deep Belief Networks (DBNs) combine unsupervised learning for feature extraction with supervised fine-tuning to improve classification. Their layered architecture enables DBNs to automatically learn high-level abstract features from raw network traffic, making them effective for detecting complex and unknown threats[12]. Deep Neural Networks (DNNs), often implemented as feedforward or multilayer perceptrons [12], are widely used in large-scale IDS applications due to their scalability and ability to generalize across diverse datasets. Recurrent neural networks (RNNs) are used for sequential data processing, making them effective in identifying evolving threats across time-series network traffic [13]. Their ability to capture temporal dependencies improves detection accuracy, particularly in dynamic environments [13]. Long Short-Term Memory (LSTM), as a type of RNN, uses gating mechanisms to allow the network to remember or forget information from its memory selectively. LSTMs are a solid approach when it comes to analyzing network traffic data in real-time to identify anomalies and potential intrusions, taking into account both short-term and long-term patterns [14]. Autoencoders learn to compress data into a smaller representation and then reconstruct it, minimizing the difference between the input and output, reducing reconstruction loss [10].They are particularly effective for anomaly detection, as abnormal traffic often produces higher reconstruction errors.

Figure 5 illustrates the key difference between traditional machine learning and deep learning approaches. In ML (top), the process requires manual feature extraction before classifica-

tion—meaning human experts must define relevant features from raw input data. In contrast, DL (bottom) integrates both feature extraction and classification into a single model, automatically learning hierarchical patterns directly from raw data. This end-to-end capability makes DL particularly effective for complex tasks such as intrusion detection, where high-dimensional and dynamic patterns are common.

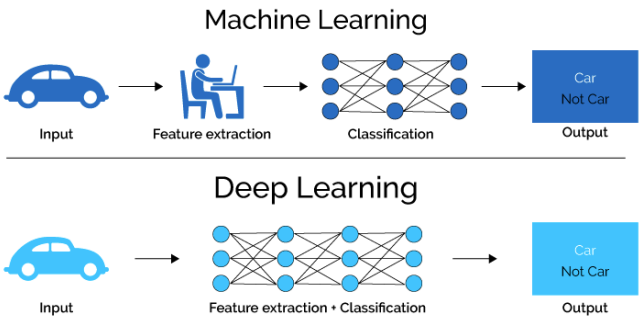

Figure 5: Main difference between Machine and deep learning approaches [15].

3.3 Ensemble Learning

This technique improves prediction accuracy by combining predictions from several base learners (e.g., by voting or averaging). Ensemble learning is widely used for tasks like classification and regression ,some ensembles (e.g., Random Forest) also provide feature-importance scores that can be used for feature selection.Common approaches include bagging (e.g., Random Forest) , it creates random subsets of the data to train each base model in parallel, which helps reduce variance and can improve efficiency on large datasets, boosting (e.g., AdaBoost, XGBoost) which trains models sequentially to correct the errors of previous ones and reduce bias, and stacking which trains multiple base learners independently and then uses a meta-learner to optimally combine their outputs. In intrusion detection, ensemble methods help balance detection across diverse attack types and have shown increased resilience in handling imbalanced or noisy data without heavily increasing model complexity [10].

3.4 Hybrid Approaches

Combining multiple techniques enhances the accuracy and robustness of IDS [4]. A hybrid model might combine the strengths of supervised and unsupervised approaches using labeled and unlabeled data. Some studies have proposed combinations like CNN-LSTM where CNNs were used for structured data and LSTMs were used for long-term dependency, or RF-Autoencoder, showing that hybrid models can offer better balance between detection capability and resource efficiency, especially in complex environments like IoT and cloud systems [12].

3.5 Reinforcement Learning

This is another theoretical model applied in AI-based IDS. This model is inspired by behavioral psychology, where an agent learns to make decisions by interacting with its environment and receiving feedback in the form of rewards or penalties [16]. Reinforcement learning can be used to develop adaptive security policies that evolve based on the threat landscape.

Figure 6 represents AI-based IDS techniques in detail from a ML, DL, and ensemble learning point of view.

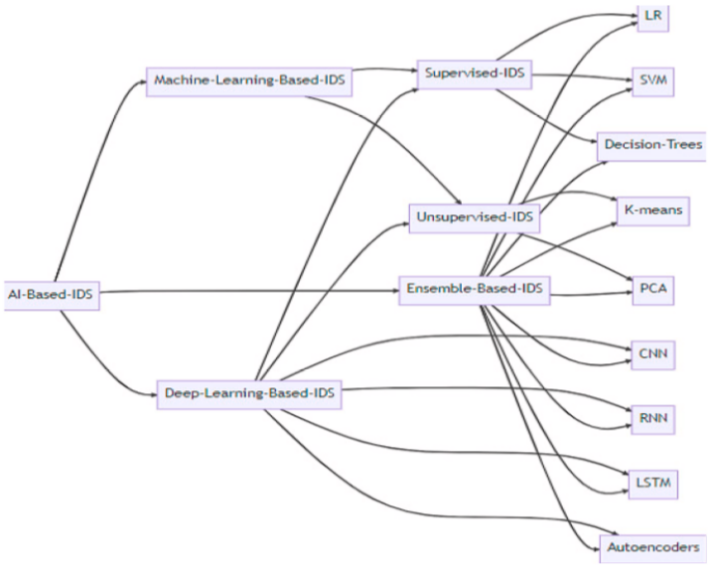

Figure 6: AI-based IDS Algorithms[10].

4 Datasets

Datasets consist of real-world or simulated network traffic data, containing labeled examples of both normal and malicious activities. These datasets play a crucial role in training and evaluating AI-based IDS models. This section presents the most widely utilized datasets for various IDS testing and highlights their key characteristics and applications.

KDDCup99, derived from the DARPA 98 dataset, is widely used for evaluating IDS. It comprises roughly 4,900,000 samples, each with 41 features, labeled as either Normal or Attack. Attacks are categorized into four types: Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L), and Probe. The dataset is available in three versions: the full dataset, a 10% subset, and a test set with 311,029 samples. A key drawback is its imbalance; while classes like DoS and Probe are well-represented, R2L and U2R are sparse, and some subsets may entirely lack these classes [17].

NSL-KDD was developed to address the primary shortcomings of KDDCup99. Introduced by Tavallaee et al. in 2009 [18], it retains the four attack categories of its predecessor. However, it may not perfectly represent existing real networks due to the lack of public datasets for network-based IDSs, but it remains an effective benchmark for comparing different IDS models. The dataset is split into two subsets: a training set with 126,620 instances covering 21 attack types, and a testing set with 22,850 instances representing 37 attack types [17]. It has 41 features, including 38 numeric and 3 categorical features, specifically protocol type, service, and flag [19].

UNSW-NB15 was developed by the Australian Centre for Cyber Security to simulate traffic that blends normal activities with various attack behaviors. It categorizes nine attack types: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. The total number of records is approximately 2,540,044, stored in four CSV files. A partition from this dataset was configured as a training set including 175,341 instances, while the testing set contains 82,332 records [20].
CIC-IDS-2017 is a dataset developed by the Canadian Institute for Cybersecurity in 2017. It consists of five days of data collection with 225,745 packets, containing over 80 features. The dataset captures more than seven days of network activity, including both normal and attack samples. It was analyzed using CICFlowMeter, extracting metrics such as timestamps, source and destination IP addresses, protocols, and attack types (stored in CSV files). It categorizes seven attack types: Brute Force FTP, Brute Force SSH, DoS, HeartBleed, Web attacks, Infiltration, Botnet, and DDoS [21].
CSE-CIC-IDS-2018, developed by the Canadian Institute for Cybersecurity, introduces the concept of profiles. It is the most recent intrusion detection dataset designed for big data applications [22], enabling both automated agents and individuals to generate network events across various protocols and topologies [23]. Updated with standards from CIC-IDS-2017, this dataset minimizes duplicate entries and ambiguous data while being provided in CSV format for immediate use [24].
To build an ideal Dataset or to choose what's adequate for the application or the model studied, certain characteristics are considered. Table 2 outlines these critical characteristics for IDS applications.

Table 2: Key Characteristics for Datasets [25]

Characteristic	**Description**
Network Traffic	Packets from hosts, firewalls, destinations, and web applications should be captured to enable detailed flow analysis and robust dataset creation.
Network Configuration	The network's topology and device connections are essential for accurately simulating real-world attack scenarios.
Network Interaction	Both internal and external communication in the network provides a complete view of network activity.
Labeled Dataset	Each data instance must be precisely tagged to clearly distinguish between normal and malicious behavior, ensuring comprehensive insight into network interactions (supervised learning).
Capturing the Traffic	Collect operational and non-operational traffic to effectively measure the IDS's detection rates and false positive rates.
Protocols	The dataset should include every communication protocol used, including both legitimate and malicious exchanges.
Attacks	A wide and current range of attack types should be included to reflect evolving threat landscapes.
Anonymity	The dataset must include details from both packet headers and payloads to ensure full data representation while maintaining privacy.
Heterogeneity	Data should be sourced from diverse origins to capture the full image of attack detection procedures.
Features	Datasets should have a complete and well-defined set of attributes for accurate classification of attack types.
Metadata	Comprehensive documentation detailing the testing environment, network infrastructures (both attack and victim), and specific attack scenarios is required.

5 Methodology and Comparative Approach

This study employs a comparative analysis to evaluate the performance of AI-enabled Intrusion Detection Systems using both Machine Learning (ML) and Deep Learning (DL) models. The algorithms and models analysis was based on their relevance in recent research, frequent use in cybersecurity studies, and availability of reproducible results. The evaluation focused on three main dimensions:

- Performance Metrics such as Accuracy, detection rate (recall), false alarm rate, precision, F1-score, and ROC, these metrics were selected due to their comprehensive representation of IDS performance, especially in distinguishing between major and minor attack classes.

- Datasets: Standard benchmarks such as NSL-KDD, CICIDS2017, CIC-IDS-2018, KDDcup99, and UNSW-NB15, the datasets vary in terms of complexity, balance, and attack diversity, offering a broad spectrum for evaluation.

- Attack Coverage,models were assessed based on their ability to detect various threats, including DoS, Probe, R2L, U2R, Botnet, and Brute Force attacks.

This approach supports a fair and consistent comparison of IDS models under diverse conditions.

5.1 Evaluation metrics

IDSs that use ML models are evaluated using four metrics: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). TP is the number of attack records correctly identified as attacks, TN is the number of normal records accurately classified as normal, FP represents normal records mistakenly flagged as attacks, and FN is the count of attack records incorrectly labeled as normal [26]. These metrics are then used to derive performance indicators such as detection rate (DR), false alarm rate (FAR), and accuracy (ACC), Precision, Recall, True Negative Rate, F-measure, and Receiver Operating Characteristics (ROC) [4].

5.2 Outcomes

Tables 4 and 5 present a categorized summary of ML- and DL-based IDS performance, detailing strengths, limitations, and specific attack detection capabilities. These comparisons form the basis for identifying promising techniques and outlining existing research gaps in the field.

Table 4 provides a comparative overview of various machine learning models—such as SVM, RF and decision trees, logistic regression, KNN, Naive Bayes and K-Means clustering. Some of these studies were combined with feature extraction methods such as Stacked Sparse Auto-Encoder (SSAE) and Non-symmetric Deep Auto-Encoder (NDAE).

Many studies have explored SVM performance for intrusion detection, highlighting the impact of feature selection. According to [4], Yan et al. [27] combined SVM with a Stacked Sparse Auto-Encoder (SSAE) for feature extraction, achieving an accuracy of 99.35% on NSL-KDD dataset and reduced training time, though detection rates for R2L and U2R attacks were lower. Also, Gu et al. [28] used Naïve Bayes for feature selection before SVM, improving accuracy to 98.92% on CICIDS2017 and 93.75% on UNSW-NB15, but failing to classify specific attacks. And according to [10],Kim et al. [29] evaluated SVM on KDD'99, achieving high detection for DoS 91.6%, but poor performance for Probe 35.65%, U2R 12%, and R2L 22%. which shows that feature selection improves SVM performance, but attack classification remains a challenge, particularly for minority attack classes. According to [4], in a study of Shone et al. [30], they combined non-Symmetric deep Auto-Encoder (NDAE) with Random Forest, achieving 97.85% on KDDcup99 dataset and 85.42% on NSL-KDD dataset, but had some difficulties with small attack classes like R2L and U2R. Also Yiping et al. [31] improved RF for wireless network attacks, integrating a signal detection model and reinforcement learning, achieving 96.93% accuracy. According to [32], Huancayo Ramos et al. [33] used RF and Decision Trees for botnet detection, achieving 99.99% accuracy, 100% precision and a recall of 100 on CICIDS2018 and ISOT HTTP, using feature importance [34] for selection and Grid Search [35] for optimization, but lacked details on data preprocessing. Also, Filho et al. [36] applied RF to detect DoS/DDoS attacks, achieving 100% accuracy and precision across multiple datasets but relied on outdated ISCX2012 for normal traffic. According to [10], Waskle et al. [37] achieved 96.78% accuracy with RF and Logistic Regression. Belouch M et al. [38] reached 97.49% on UNSW-NB15 using RF.

According to [4], Pan et al. [26] tried in their study to optimize KNN with PM-CSCA (Polymorphic Mutation-Compact SCA) to enhance intrusion detection in wireless networks, leveraging fog computing to reduce cloud workload and reduce time response, achieving 99.33% on NSL-KDD and 98.27% on UNSW-NB15, but lacking attack type classification. According to [10], Lin et al. [39] used in their study KNN on KDD dataset to detect DoS, Probe, U2R, and R2L attacks, achieving 99.89% accuracy. Also, Wenchao Li et al. [40] also implemented KNN achieving 98.5% accuracy and a False Alarm Rate FAR of 4.63%, details on dataset or methodology were not provided. Karatas et al. [24] evaluated various ML algorithms, including KNN, on the CSE-CIC-IDS2018 dataset, addressing the issue of class imbalance using SMOTE. This approach enhanced detection rates for minority attacks, which could have further impacted KNN performance positively.

Table 3: IDS Performance Metrics

Metric	Description	Formula
ACC	Ratio of all correctly predicted samples (both attacks and normals) to the total number of samples.	$\frac{TP+TN}{TP+FN+TN+FP}$
Precision	Ratio of correctly predicted attack samples to all samples predicted as attacks.	$\frac{TP}{TP+FP}$
FAR	Ratio of normal samples incorrectly flagged as attacks to all actual normal samples (also known as False Positive Rate).	$\frac{FP}{TN+FP}$
Recall (DR)	Ratio of correctly predicted attack samples to all actual attack samples (also known as Detection Rate).	$\frac{TP}{TP+FN}$
F1-score	The harmonic mean of Precision and Recall, providing a balanced measure of performance.	$2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$
TNR	Ratio of correctly predicted normal samples to all actual normal samples.	$\frac{TN}{TP+FN}$
ROC	Graphical analysis representing the trade-off between detection rate and false alarm rate. A curve closer to the top-left indicates a more effective IDS.	—

TP: True Positives; TN: True Negatives; FP: False Positives; FN: False Negatives.

According to [10], Monika Vishwakarma et al. [41] applied a Naive Bayes classifier on the NSL-KDD, UNSW-NB15, and CICIDS2017 datasets. achieving accuracies of 97%, 86.9%, and 98.59%, respectively, for detecting DoS, Probe, U2R, and R2L attacks. Also, Sharmila B.S et al. [42] implemented in their study Naive Bayes method on the NSL-KDD dataset, obtaining an accuracy of 83% for the same attack types. This accuracy diminution may be attributed to dataset-specific challenges, like class imbalance.

According to [10], K. Samunnisa et al. [43] combined K-Means clustering with Random Forest on NSL-KDD, achieving 92.77% accuracy. Also Vipin et al. [44] used K-Means in their study, they achieved 82.29% accuracy on the NSL-KDD dataset. So k-means showed lower accuracy in detecting attacks compared to the combined method with Random Forest, highlighting the benefit of using ensemble techniques.

Table 5 compares deep learning models like CNN, RNN, autoencoders, DBN, and LSTM.

According to [4] and [32], CNNs have consistently demonstrated high accuracy in identifying intrusions, with reported results up to 99.99% on CIC-IDS-2018 [22] and over 99.56% on CICIDS2017 [45]. Their ability to learn spatial features from network traffic makes them effective, although some studies note challenges in detecting low-frequency attacks (U2R, R2L) in older datasets like KDDcup99 [46].

In a study by C. Yin et al. [10], [47], an RNN was evaluated on the NSL-KDD dataset achieving a training accuracy of 99.81% but only 83.28% on testing data. RNNs are widely used for supervised classification and feature extraction in IDS, but they can struggle with long sequences due to short-term memory limitations [48]. To overcome this, variants such as Long Short-Term Memory (LSTM) networks and their bidirectional counterpart (BLSTM) have been developed, enabling models to capture context from both past and future inputs. Lin et al. [32], [49] reported that an LSTM model achieved 96.2% accuracy, precision, and detection rate on the CIC-IDS-2018 dataset, while another study [50] demonstrated that an LSTM model on CICDOS2017 reached 99.47% accuracy and precision—with nearly perfect F1 score and recall 99.74%, and an FPR of only 0.389% for DDoS attacks. Also BLSTM variants, excel at handling sequential data [32], [51], achieving accuracies around 98–99% on CICIDS2017. However, they often require large amounts of training data, and performance can drop when transitioning from training to testing phases suggesting potential overfitting.

Also, Autoencoder-based models and their deep variants have proven highly effective for dimensionality reduction and anomaly detection in IDS [10], a stacked autoencoder achieved over 94% accuracy on KDDcup99 [10], [15], while more advanced deep autoencoder approaches have demonstrated impressive performance on recent datasets, particularly for detecting botnet and DoS attacks. Specifically, Ferrag et al. [32], [52] employed an RNN-AE on CIC-IDS-2018, achieving 97.38% accuracy and 98.18% recall across various attack types, including DoS, DDoS, web attacks, botnet, and brute force. Similarly, studies by Catillo et al. [32], [53] and Li et al. [32], [54] reported that deep autoencoder models evaluated on CIC-IDS-2018 and CICIDS2017 achieved 99.20% accuracy with a precision of 95.0% and a detection rate of 98.90%, with one approach reaching a 100% detection rate for botnet attacks.also, Khan et al. [4], [55] achieved 99.996% accuracy on KDDcup99 but only 89.134% on UNSW-NB15, revealing a notable gap with modern data

Deep Belief Networks (DBNs) use unsupervised pre-training to detect network anomalies. For instance, Z. Alom et al. [10], [56] reported 97.5% accuracy on NSL-KDD for DoS, Probe, U2R, and R2L attacks, though performance may drop on larger datasets. Additionally, Wei et al. [57] proposed an optimization for DBNs with swarm and genetic algorithms, which significantly improved the detection rate for U2R and R2L classes, but with increased training time due to its complex structure.

ANNs mimic the human brain's function to recognize complex patterns in network traffic. In IDS, studies have shown that ANNs can achieve around 95.45% accuracy on datasets like NSL-KDD and UNSW-NB15, and even higher accuracies on KDD99 (e.g., 99.93% for DoS and 96.51% for U2R) [10], [58]. Despite their strong performance and suitability for large datasets, ANNs require extensive preprocessing and are computationally intensive due to their complex architecture. Also For DNNs Deep Neural Networks models are complex nonlinear functions, and increasing the number of hidden layers enhances their abstraction capability [59]. Yu et al. [4], [60] evaluated a DNN on NSL-KDD and UNSW-NB15, achieving accuracies of 92.34% and 92%, respectively. Their study, which measured metrics like detection rate, false alarm rate, precision, and F-measure, reported particularly strong performance in detecting U2R and R2L attacks.

5.3 Comparative Analysis and Discussion

This section synthesizes the findings from the previous section, providing a comparative evaluation of the most prominent ML and DL-based intrusion detection approaches based on key factors: accuracy, detection of rare attacks, dataset relevance, and computational cost.

Performance Summary: Deep Learning models generally outperformed traditional ML models in terms of accuracy and detection rate, especially on modern datasets such as CICIDS2017 and CICIDS2018. For example, CNNs and LSTMs consistently achieved over 98% accuracy on these datasets, while many ML models struggled with detection of rare attacks like U2R and R2L.

Detection of Minority Attack Classes: DL approaches (especially LSTM and DBN) demonstrated stronger results in detecting low-frequency attacks. However, optimization strategies like SMOTE and hybrid models (e.g., RF + autoencoders) helped improve ML model performance on imbalanced datasets.

Dataset Impact: Model performance was heavily influenced by dataset choice. Older datasets like KDDcup99 and NSL-KDD often yielded inflated results but lacked modern attack signatures. In contrast, CICIDS2017 and CICIDS2018 provided more realistic traffic and diverse attack types, leading to more reliable evaluation outcomes.

Model Complexity and Practicality: While DL models provide higher accuracy, they demand more training data and computational resources, making them harder to deploy in real-time systems. ML models like Random Forest or KNN are easier to train and deploy but may require careful tuning or ensemble strategies to handle complex attack patterns effectively.

Best Approaches by Criteria: Based on the research we concluded some best approaches but each within certain criteria

- Best overall performance: LSTM and CNN on CICIDS2017 and CICIDS2018.
- Best for rare attacks: Deep Autoencoders.
- Best lightweight ML model: Random Forest with feature selection or optimized KNN.
- Best hybrid approach: RF + Autoencoder, or SVM + feature extraction (SSAE).

This analysis shows there is no single "best" IDS model. The choice depends on deployment context, dataset realism, resource constraints, and the importance of detecting rare attacks versus achieving high overall accuracy.

Table 4: Evaluation of Machine Learning-based IDS.

ML Model	Dataset	Ref	Attacks addressed	Performance Metrics	Advantages	Limitation
SVM	NSL-KDD	Yan et al. [27] 2018	DoS, Probe, R2L, U2R	ACC=99.35%	Reduces training and testing duration	Old dataset NSL-KDD, low detection rate for some attacks.
Random Forests	KDDcup99, NSL-KDD	Shone et al. [30] 2018	DoS, Probe, R2L, U2R	ACC=97.85% (KDDcup99), ACC=85.42% (NSL-KDD)	High accuracy, combines non-symmetric Deep Auto-Encoder and RF models for anomaly detection	Difficulty detecting smaller attack classes, lower accuracy on NSL-KDD, old dataset used.
KNN using PM-CSCA	NSL-KDD, UNSW-NB15	Pan et al. [26] 2021	DoS, Sniffing (Probe), U2R, R2L	ACC=99.33% , ACC=98.27%	Optimized KNN model, high accuracy, cloud-enhanced speed in wireless networks	No classification of attack types.
SVM	CICIDS2017, UNSW-NB15	Gu et al. [28] 2021	—	ACC=98.92% (CICIDS2017), ACC=93.75% (UNSW-NB15)	Use of Naïve Bayes, high accuracy	No identification of attack types.
Improved RF Algorithm	—	Yiping et al. [31] 2022	Wireless network attacks	ACC=96.93%	RF model for wireless networks	No known datasets used.
SVM	KDD'99	Kim et al. [29]	DoS, Probe, U2R, R2L	Performance (DoS-91.6, Probe-35.65, U2R-12, R2L-22)	Good performance in DoS attack detection, reduces training detection	Poor detection for U2R and R2L, sensitive to parameter selection, old dataset used.
K-NN	KDD	Lin et al. [39]	DoS, Probe, U2R, R2L	ACC=99.89%	High accuracy achievement	Old dataset KDD, limited to specific attack types.
Random Forest	UNSW-NB15	Belouch M et al. [38]	—	ACC=97.49%	High accuracy on UNSW-NB15 dataset	No identification of attack types.
RF, Decision Tree	CICIDS2018, ISOT HTTP	Huancayo Ramos et al. [33]	Botnet	ACC=99.99%, Recall=100%	Extremely high accuracy and precision	Unclear explanation of data preparation.
RF	CICIDS2018, CICIDS2017, ISCX2012, CIC-DoS	Filho et al. [36]	DoS/DDoS	ACC=100%, Recall=100%	Perfect accuracy in detecting DDoS attacks	Outdated ISCX2012 dataset, limited traffic protocols.
RF, Logistic Regression	—	S. Waskle et al. [37]	—	ACC=96.78%, Error rate=0.21	Good classification performance, low error rate	No known datasets used, no identification of attack types.
LMRDT-SVM	NSL-KDD	Huiwen Wang et al. [61]	—	ACC=99.31%, Detection rate=99.20%	Excellent accuracy and detection rate	No identification of attack types.
Naïve Bayes	NSL-KDD, UNSW-NB15, CICIDS2017	Monika Vishwakarma et al. [41]	DoS, Probe, U2R, R2L	NSL-KDD: 97%, UNSW-NB15: 86.9%, CIC-IDS2017: 98.59%	Simple, fast classification, high accuracy for NSL-KDD, CIC-IDS2017	Low accuracy on UNSW-NB15 compared to other datasets.
K-NN	—	Wenchao Li et al. [40]	—	ACC=98.5%, FAR=4.63%	High accuracy	No known datasets used.
Naïve Bayes	NSL-KDD	Sharmila B.S et al. [42]	DoS, Probe, U2R, R2L	ACC=83%	Simple, fast classification for IDS	Low accuracy compared to other models.
K-Means + RF	NSL-KDD	K. Samunnisa et al. [43]	DoS, Probe, U2R, R2L	ACC=92.77%	Hybrid models improve performance	—
K-Means	NSL-KDD	Vipin et al. [44]	DoS, Probe, U2R, R2L	ACC=82.29%	Effective for clustering-based anomaly detection	Low accuracy compared to other studies.

Table 5: Evaluation of Deep Learning based IDS

DL Model	Dataset	Ref	Addressed Attacks	Performance Metrics	Advantages	Limitation
CNN	KDDcup99	Xiao et al. [46] (2019)	DoS, Probe, U2R, R2L	FAR, DR, ACC=94%	CNN for IDS using PCA and Auto-Encoder for feature extraction.	A low detection rate of U2R and R2L attacks.
CNN	CICIDS2017	Lin et al. [45] (2020)	FTP Brute Force, SSH Brute Force, DoS, Web attacks, penetration attacks	ACC: 99.56%	NIDS using CNN achieves excellent results on one of the most recent datasets, the CICIDS2017.	No feature extraction methods (no accuracy details on each kind of attacks).
CNN	CIC-IDS-2018	Kim et al. [22]	DoS, DDoS, Web attacks, Botnet, Brute force	ACC: 99.99%, Precision: 81.75%, DR: 82.25%	Highest accuracy score in DDoS 100%	Requires large amounts of training data.
RNN	NSL-KDD	C. Yin et al. [47]	DoS, Probe, U2R, R2L	ACC: Training data 99.81%, Testing data 83.28%	High accuracy on training data.	Low performance on testing data.
RNN-BLSTM	CICIDS2017	S. Sivamohan et al. [51]	Brute force, DoS, DDoS	ACC: 98.48%	Perfect handling for sequential data.	Requires large amounts of training data.
ANN	KDD99	Akashdeep [58]	DoS, U2R, R2L, Probe	DoS-99.93%, U2R-96.51%, R2L-92.54%, Probe-98.7%	Works well with large datasets.	Requires extensive preprocessing and complex nature.
AE	KDDcup99	Farahna Kian et al. [62]	DoS, Probe, U2R, R2L	ACC: 94.71%	Effective in feature reduction and identifying anomalies.	May not perform well with complex and diverse datasets.
Deep Auto-Encoders	CIC-IDS-2018, CICIDS2017	Catillo et al. [53], Li et al. [54]	DoS, DDoS, Web attacks, Botnet, Brute force	(ACC=99.20%, Precision: 95.0%, DR: 98.90%) [53], DR: 100% [54]	Highest accuracy in Botnet type.	
LSTM	CIC-IDS-2018	Lin et al. [49]	DoS, DDoS, Web attacks, Botnet, Brute force	ACC: 96.2%, Precision: 96%, DR: 96%	Excels at handling sequential data.	Requires large amounts of training data.
DNN and CNN	NSL-KDD, UNSW-NB15	Yu et al. [60] (2020)	DoS, Probe, U2R, R2L, Other attack (normal, generic, fuzzers, worms, backdoor)	ACC, DR, FAR, Precision, F-measure: 92.34% (NSL-KDD), 92% (UNSW-NB15)	Good results for U2R and R2L compared to other methods.	Complex architecture.
Deep Auto-encoders	KDDcup99, UNSW-NB15	Khan et al. [55]	Normal, DoS, Probe, R2L, U2R (22 different categories of attacks tested)	99.996% (KDDcup99), 89.134% (UNSW-NB15)	Very high accuracy on old dataset.	Old dataset; there is a 10% gap compared to the accuracy with the recent dataset.
DBN	NSL-KDD	Z. Alom et al. [56]	DoS, Probe, U2R, R2L	ACC: 97.5%	High accuracy in identifying attacks.	May not perform well with large datasets.
RNN-AE	CIC-IDS-2018	Ferrag et al. [52]	DoS, DDoS, Web attacks, Botnet, Brute force	ACC: 97.38%, Recall: 98.18%	Impressive performance on recent datasets.	
LSTM	CICDOS2017	Noe et al. [50]	DoS, DDoS, Web attacks, Botnet, Brute force	ACC: 99.473%, Precision: 99.47%, F1 score: 99.473%, Recall: 99.473%, FPR: 0.389%	Achieves the highest performance metrics and the lowest FPR.	Requires large amounts of training data.

5.3.1 Challenges and Limitations:

Despite their promise, AI-based intrusion detection systems face key limitations. Many models emphasize overall accuracy but overlook critical metrics like F1-score, precision, and recall—especially important for detecting rare or simultaneous attacks. Model performance also depends heavily on the quality and relevance of datasets, many of which are outdated or imbalanced, reducing detection effectiveness for minority attack classes [10] and missing modern threats like zero-day exploits. Additionally, few studies consider the computational demands of real-world deployment [3], including time complexity and resource usage. While accuracy is often reported in ideal conditions, real-world validation remains limited, and practical, adaptive IDS frameworks are still lacking.

6 Conclusion

In conclusion, this research highlights the significant progress made by AI-enabled intrusion detection systems, especially deep learning models, which have achieved high accuracy on recent benchmark datasets. Models such as LSTM, CNN, and deep autoencoders consistently outperform traditional machine learning approaches in detecting complex and evolving attack patterns. Our findings also emphasize the critical role of dataset quality—modern, balanced datasets like CICIDS2017 and CIC-IDS-2018 yield more reliable results compared to outdated datasets such as KDDcup99. Despite these advances, key challenges remain. Many models still struggle with detecting rare attack types, depend on outdated data, or are not optimized for real-time deployment. Additionally, high accuracy in controlled environments does not always guarantee robust performance in real-world networks. Looking ahead, leveraging the complementary strengths of deep learning and machine learning can lead to intrusion detection systems that are not only highly effective but also practical for real-world use.

References

[1] A. Bendovschi, "Cyber-attacks – trends, patterns and security countermeasures," *Procedia Economics and Finance*, vol. 28, pp. 24–31, 2015.

[2] S. Ndichu, S. Okoth, H. Okoyo, and C. Wekesa, "Detecting remote access network attacks using supervised machine learning methods," *Int. J. of Computer Network and Information Security*, pp. 48–61, 2023.

[3] V. Shanmugam, R. Razavi-Far, and E. Hallaji, "Addressing class imbalance in intrusion detection: A comprehensive evaluation of machine learning approaches," *Electronics*, vol. 14, no. 69, 2025.

[4] P. Vanin, T. Newe, L. L. Dhirani, *et al.*, "A study of network intrusion detection systems using artificial intelligence/machine learning," *Applied Sciences*, vol. 12, no. 11752, 2022.

[5] A. Jamalipour and S. Murali, "A taxonomy of machine-learning-based intrusion detection systems for the internet of things: A survey," *IEEE Internet of Things Journal*, vol. 9, pp. 9444–9466, 2021.

[6] C. M. Ou, "Host-based intrusion detection systems inspired by machine learning of agent-based artificial immune systems," in *Proc. 2019 IEEE Int. Symp. on Innovations in Intelligent Systems and Applications (INISTA)*, Sofia, Bulgaria, Jul. 3–5, 2019, pp. 1–5.

[7] Z. Ahmad, A. S. Khan, C. W. 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, 2021.

[8] A. Naik, *Issues and recent advances in machine learning techniques for intrusion detection systems*, *Washington University in St. Louis*, Available: https://www.cse.wustl.edu/~jain/cse570-19/ftp/ml_ids.pdf, Dec. 2019.

[9] A. Nisioti, A. Mylonas, P. Yoo, and V. Katos, "From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3369–3388, 2018.

[10] T. Sowmya and E. A. M. Anita, "A comprehensive review of ai based intrusion detection system," *Measurement: Sensors*, vol. 28, p. 100 827, 2023.

[11] S. Patil, V. Varadarajan, S. M. Mazhar, *et al.*, "Explainable artificial intelligence for intrusion detection system," *Electronics*, vol. 11, no. 3079, 2022. DOI: 10.3390/electronics11193079.

[12] R. Kimanzi, P. Kimanga, D. Cherori, and P. K. Gikunda, *Deep learning algorithms used in intrusion detection systems – a review*, arXiv:2402.17020, 2024.

[13] Z. Iqbal, A. Sajid, M. Zakki, A. Zafar, and A. Mehmood, "Role of machine and deep learning algorithms in secure intrusion detection systems (ids) for iot & smart cities," *International Journal of Information Technology, Research and Applications*, vol. 3, pp. 1–16, 2024.

[14] S. Muneer, U. Farooq, A. Athar, M. A. Raza, T. M. Ghazal, and S. Sakib, "A critical review of artificial intelligence based approaches in intrusion detection: A comprehensive analysis," *Journal of Engineering*, vol. 2024, pp. 1–16, Article ID 3909173.

[15] U. Odi and T. Nguyen, "Geological facies prediction using computed tomography in a machine learning and deep learning environment," in *2018 URTeC*, Paper 2901881, 2018.

[16] H. Wu, "Reinforcement learning inspired by psychology and neuroscience," *Journal of Education, Humanities and Social Sciences*, vol. 8, pp. 2164–2170, 2023. DOI: 10.54097/ehss.v8i.4673.

[17] L. Ashiku and C. Dagli, "Network intrusion detection system using deep learning," *Procedia Computer Science*, vol. 185, pp. 239–247, 2021.

[18] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *Proc. 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, Canada, Jul. 8–10, 2009, pp. 1–6.

[19] C. Ieracitano *et al.*, *Statistical analysis driven optimized deep learning system for intrusion detection*, arXiv:1808.05633, 2018.

[20] D. Protic, "Review of kdd cup '99, nsl-kdd and kyoto 2006+ datasets," *Vojnotehnički Glasnik*, vol. 66, pp. 580–596, 2018.

[21] N. Moustafa, *The unsw-nb15 dataset*, UNSW Canberra Cyber, Available: https://research.unsw.edu.au/projects/unsw-nb15-dataset, 2022.

[22] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, "Cnn-based network intrusion detection against denial-of-service attacks," *Electronics*, vol. 9, no. 6, p. 916, 2020.

[23] S. Songma, T. Sathuphan, and T. Pamutha, "Optimizing intrusion detection systems in three phases on the cse-cic-ids-2018 dataset," *Computers*, vol. 12, no. 245, 2023. DOI: 10.3390/computers12120245.

[24] G. Karatas, O. Demir, and O. K. Sahingoz, "Increasing the performance of machine learning-based idss on an imbalanced and up-to-date dataset," *IEEE Access*, vol. 8, pp. 32 150–32 162, 2020.

[25] A. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "An evaluation framework for intrusion detection dataset," in *2016 Int. Conf. on Information Science and Security (ICISS)*, 2016, pp. 1–6.

[26] J.-S. Pan, F. Fan, S.-C. Chu, H. Zhao, and G.-Y. Liu, "A lightweight intelligent intrusion detection model for wireless sensor networks," *Security and Communication Networks*, vol. 2021, pp. 1–15, 2021.

[27] B. Yan and G. Han, "Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system," *IEEE Access*, vol. 6, pp. 41 238–41 248, 2018.

[28] J. Gu and S. Lu, "An effective intrusion detection approach using svm with naïve bayes feature embedding," *Computer & Security*, vol. 103, p. 102 158, 2021.

[29] D. S. Kim and J. S. Park, "Network-based intrusion detection with support vector machines," in *Information Networking: Int. Conf. ICOIN 2003, Revised Selected Papers*, Cheju Island, Korea, Feb. 12–14, 2003, pp. 747–756.

[30] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. on Emerging Topics in Computational Intelligence*, vol. 2, pp. 41–50, 2018.

[31] Y. Chen and F. Yuan, "Dynamic detection of malicious intrusion in wireless network based on improved random forest algorithm," in *Proc. 2022 IEEE Asia-Pacific Conf. on Image Processing, Electronics and Computers (IPEC)*, Dalian, China, Apr. 14–16, 2022, pp. 27–32.

[32] J. L. Leevy and T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on cse-cic-ids2018 big data," *Journal of Big Data*, vol. 9, no. 1, pp. 1–41, Jan. 2022.

[33] K. S. H. Ramos, M. A. S. Monge, and J. M. Vidal, "Benchmark-based reference model for evaluating botnet detection tools driven by traffic-flow analytics," *Sensors*, vol. 20, no. 16, p. 4501, 2020.

[34] V. Gupta and A. Bhavsar, "Random forest-based feature importance for hep-2 cell image classification," in *Ann. Conf. on Medical Image Understanding and Analysis*, 2017, pp. 922–934.

[35] Y. Sun, Y. Wu, L. Gong, Z. Ma, and J. Shan, "The comparison of optimizing svm by ga and grid search," in *2017 13th IEEE Int. Conf. on Electronic Measurement & Instruments (ICEMI)*, 2017, pp. 354–360.

[36] F. S. L. Filho, F. A. Silveira, A. de Medeiros Brito Júnior, G. Vargas-Solar, and L. F. Silveira, "Smart detection: An online approach for dos/ddos attack detection using machine learning," *Security and Communication Networks*, vol. 2019, pp. 1–15, 2019.

[37] S. Waskle, L. Parashar, and U. Singh, "Intrusion detection system using pca with random forest approach," in *Int. Conf. on Electronics and Sustainable Communication Systems (ICESC)*, 2020, pp. 803–808.

[38] M. Belouch, S. E. Hadaj, and M. Idhammad, "Performance evaluation of intrusion detection based on ml using apache spark," *Procedia Computer Science*, vol. 127, pp. 1–6, 2018.

[39] W.-C. Lin, K.-S. Shih-Wen, and C.-F. Tsai, "Cann: An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowledge-Based Systems*, vol. 78, pp. 13–21, 2015.

[40] W. Li, P. Yi, Y. Wu, L. Pan, and J. Li, "New intrusion detection system based on knn classification algorithm in wireless sensor network," *Journal of Electrical and Computer Engineering*, vol. 2021, pp. 1–8, Article ID 1752.

[41] M. Vishwakarma and N. Kesswani, "A new two-phase intrusion detection system with naïve bayes ml for data classification and elliptic envelope method for anomaly detection," *Decision Analytics Journal*, vol. 7, p. 100 233, 2023.

[42] B. S. Sharmila and N. Rohini, "Intrusion detection system using naïve bayes algorithm," in *2019 IEEE Int. WIE Conf. on Electrical and Computer Engineering (WIECON-ECE)*, 2019, pp. 1–4.

[43] K. Samunnisa, G. S. V. Kumar, and K. Madhavi, "Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods," *Measurement: Sensors*, vol. 25, p. 100 612, 2023.

[44] V. Kumar, H. Chauhan, and D. Panwar, "K-means clustering approach to analyze nsl-kdd intrusion detection dataset," *Int. J. Soft Comput. Eng.*, vol. 3, no. 4, pp. 1–4, 2013.

[45] L. Chen, X. Kuang, A. Xu, S. Suo, and Y. Yang, "A novel network intrusion detection system based on cnn," in *Proc. 2020 8th Int. Conf. on Advanced Cloud and Big Data (CBD)*, Taiyuan, China, Dec. 5–6, 2020, pp. 243–247.

[46] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An intrusion detection model based on feature reduction and convolutional neural networks," *IEEE Access*, vol. 7, pp. 42 210–42 219, 2019.

[47] C. Yin, Y. Zhu, J. Fei, and X. He, "A dl approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21 954–21 961, 2017.

[48] X. Liu, A. Gherbi, W. Li, and M. Cheriet, "Multi-features and multi-time steps lstm based methodology for bike sharing availability prediction," *Procedia Computer Science*, vol. 155, pp. 394–401, 2019.

[49] P. Lin, K. Ye, and C.-Z. Xu, "Dynamic network anomaly detection system by using deep learning techniques," in *International Conference on Cloud Computing*, Springer, 2019, pp. 161–176.

[50] N. Yungaicela-Naula, C. Vargas-Rosales, and J. A. Perez-Diaz, "Sdn-based architecture for transport and application layer ddos attack detection by using machine and deep learning," *IEEE Access*, vol. 9, pp. 108 495–108 512, 2021.

[51] J. Zhang, X. Zhang, Z. Liu, F. Fu, Y. Jiao, and F. Xu, "A network intrusion detection model based on bilstm with multi-head attention mechanism," *Electronics*, vol. 12, p. 4170, 2023. DOI: 10.3390/electronics12194170.

[52] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102 419, 2020.

[53] M. Catillo, M. Rak, and U. Villano, "2l-zed-ids: A two-level anomaly detector for multiple attack classes," in *Workshops of the Int. Conf. on Advanced Information Networking and Applications*, 2020, pp. 687–696.

[54] X. Li, W. Chen, Q. Zhang, and L. Wu, "Building auto-encoder intrusion detection system based on random forest feature selection," *Computers & Security*, vol. 95, p. 101 851, 2020.

[55] F. A. Khan, A. Gumaei, A. Derhab, and A. Hussain, "A novel two-stage deep learning model for efficient network intrusion detection," *IEEE Access*, vol. 7, pp. 30 373–30 385, 2019.

[56] M. Z. Alom, V. R. Bontupalli, M. Tarek, and T. Taha, "Intrusion detection using deep belief networks," in *2015 National Aerospace and Electronics Conference (NAECON)*, 2015, pp. 339–344.

[57] P. Wei, Y. Li, Z. Zhang, T. Hu, Z. Li, and D. Liu, "An optimization method for intrusion detection classification model based on deep belief network," *IEEE Access*, vol. 7, pp. 87 593–87 605, 2019. DOI: 10.1109/ACCESS.2019.2925828.

[58] I. Manzoor and N. Kumar, "A feature reduced intrusion detection system using ann classifier," *Expert Systems with Applications*, vol. 88, pp. 249–257, 2017.

[59] S. Gu and L. Rigazio, *Towards deep neural network architectures robust to adversarial examples*, arXiv:1412.5068, 2014.

[60] Y. Yu and N. Bian, "An intrusion detection method using few-shot learning," *IEEE Access*, vol. 8, pp. 49 730–49 740, 2020.

[61] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on svm with feature augmentation," *Knowledge-Based Systems*, vol. 136, pp. 130–139, 2017.

[62] F. Farahnakian and J. Heikkonen, "A deep auto-encoder based approach for intrusion detection system," in *2018 20th Int. Conf. on Advanced Communication Technology (ICACT)*, 2018, pp. 178–183. DOI: 10.23919/ICACT.2018.8323688.