

A Holistic review and performance evaluation of unsupervised learning methods for network anomaly detection

Niharika Sharma¹ Bhavna Arora¹,
Shabana Ziyad^{2,*}, Pradeep Kumar
Singh¹ and Yashwant Singh¹

¹Department of Computer Science
and Information Technology,
Central University of Jammu,
Rahya-Suchani (Bagla),
District Samba-181143, Jammu
(Jammu & Kashmir), India

²Department of Computer Science,
College of Computer Engineering
and Sciences, Prince Sattam Bin
Abdulaziz University, Al-Kharj,
Saudi Arabia

*E-mail: ziyadshabana@gmail.com

Received for publication:
November 24, 2023

Abstract

The evolving cyber-attack landscape demands flexible and precise protection for information and networks. Network anomaly detection (NAD) systems play a crucial role in preventing and detecting abnormal activities on the network that may lead to catastrophic outcomes when undetected. This paper aims to provide a comprehensive analysis of NAD using unsupervised learning (UL) methods to evaluate the effectiveness of such systems. The paper presents a detailed overview of several UL techniques, lists the current developments and innovations in UL techniques for network anomaly and intrusion detection, and evaluates 13 unsupervised anomaly detection algorithms empirically on benchmark datasets such as NSL-KDD, UNSW-NB15, and CIC-IDS 2017 to analyze the performance of different classes of UL approaches for NAD systems. This study demonstrates the effectiveness of NAD algorithms, discusses UL approaches' research challenges, and unearths the potential drawbacks in the current network security environment.

Keywords

unsupervised learning, machine learning, unsupervised anomaly detection techniques, network anomaly detection, intrusion detection

1. Introduction

With the rapid growth of the Internet, massive data are generated daily. With evolving hacking tools and methods, robust security measures are crucial. Businesses have suffered significant financial losses due to security attacks and network intrusions [1, 2]. To address these challenges, network anomaly detection (NAD) and intrusion detection systems (IDSs) safeguard network systems from unauthorized access and infiltration activities. An IDS is a device that monitors and detects attacks or intrusions in a computer or network [3]. It provides alerts to administrators for corrective actions. Traditional detection systems were manual and could not keep up with the increasing data volume and speed. Misuse detection techniques

were introduced but were limited to known attacks. Anomaly detection (AD) techniques have replaced traditional signature-based schemes, working with unlabeled data to detect unknown attacks. This eliminates the need for costly labeled traffic generation and allows for identifying new attack versions [4].

AD techniques capture deviations from normal behavior, detecting unknown attacks without prior knowledge [5]. However, they can have a high false positive rate (FPR) due to overfitting. False positives in anomaly detection occur when normal situations are mistakenly classified as abnormal due to unrecognized patterns. Data mining techniques, particularly unsupervised learning (UL), overcome these limitations by using unlabeled data and detecting previously unseen attacks [3]. UL algorithms used in

NAD improve the quality of data representations by removing newly derived variables before the categorization process. This means these algorithms eliminate unnecessary or redundant variables, resulting in more informative and accurate representations of the original data. While unsupervised NAD techniques have received less attention, their combination with supervised learning has shown promising results in research. Unsupervised and hybrid techniques have recently gained popularity for effective AD systems.

This paper emphasizes UL's significance in enhancing the security of NAD systems. Notably, UL offers several benefits, including:

- UL is analogous to how a human learns to contemplate through their knowledge, bringing it closer to evident AI.
- For obtaining meaningful information from data, UL is effective.
- Data that are not classified or categorized are used in UL, which makes it more significant to deploy in real environments.
- Input data are not always available in a structured format that conforms to output data in the real world. So, UL is needed to derive patterns from unstructured data.

AD schemes have gained increased attention due to the rise of zero-day attacks [5]. Researchers have extensively explored supervised and unsupervised approaches to effectively identify intrusions and anomalies since the early days of networking and the Internet [6]. Classification and clustering methods, such as support vector machine (SVM), K-nearest neighbor (KNN), and k-means, are commonly employed for modeling normal network behavior. Clustering and outlier detection play a significant role in UL schemes for NAD [5–8]. Promising techniques for high-dimensional datasets include density and grid-based clustering [9, 10]. Hybrid approaches like k-means with decision trees [11] and genetic algorithms (GAs) [12] are effective in detecting network attacks. Self-organizing maps (SOMs) aid in data visualization and dimensionality reduction [13]. Recent developments in artificial neural networks (ANNs), particularly deep learning (DL) architectures, have sparked a revolution [14–16].

Therefore, this paper offers an in-depth analysis of NAD using unsupervised approaches, highlighting recent developments while identifying key research limitations. The evaluation of 13 UL methods across benchmark intrusion detection datasets aims to assess their effectiveness and provide valuable insights

for future research and development in NAD systems. Figure 1 depicts the results of a bibliometric analysis conducted on AD literature. Subfigure (a) illustrates the publication trend of bibliometric papers from 2010 to 2023. Subfigure (b) highlights the data-retrieval process, focusing on publications from reputable publishers such as Springer, IEEE, and Elsevier, which collectively provided most of the analyzed data.

a. Survey timeline

This section reviews and evaluates the performance of UL methods for detecting network anomalies and intrusions. It highlights significant contributions from selected survey papers published between 2000 and 2023. The surveys are categorized into three time intervals for evaluation.

a.i 2000–2007

Various studies have explored unsupervised AD algorithms for IDS. In 2000, one study compared cluster-based estimation, KNN, and one-class SVM using the KDD Cup 99 dataset [17]. In another analysis in 2003, the authors evaluated AD approaches on the DARPA98 dataset, showing promising results with local outlier factor (LOF) and unsupervised SVM [18]. In 2007, the authors reviewed different clustering schemes and analyzed that K-harmonic means performed well in speed, while population-based stochastic techniques were suitable for cluster quality [19]. Additionally, in a study in 2007, seven distinct IDS techniques for HTTP anomalies were tested and compared under simulated and real-world settings. It was found that the token-based approaches exhibited better discrimination and generalization using heuristics-based approaches [20].

a.ii 2008–2015

In 2009, the authors presented literature studies to review machine learning (ML) methods for IDS, including single, hybrid, and ensemble classifiers [21]. In 2012, the authors reviewed and compared different clustering techniques based on parameters such as data type, cluster type, complexity, dataset, and measures, along with their advantages and disadvantages [22]. In another study in 2014, the authors present an overview and comparison of four novelty detection techniques. The experimental outcomes indicate that the KNN novelty detection scheme performs better than other methods [23]. Moreover, in 2015, the authors provided a comprehensive review

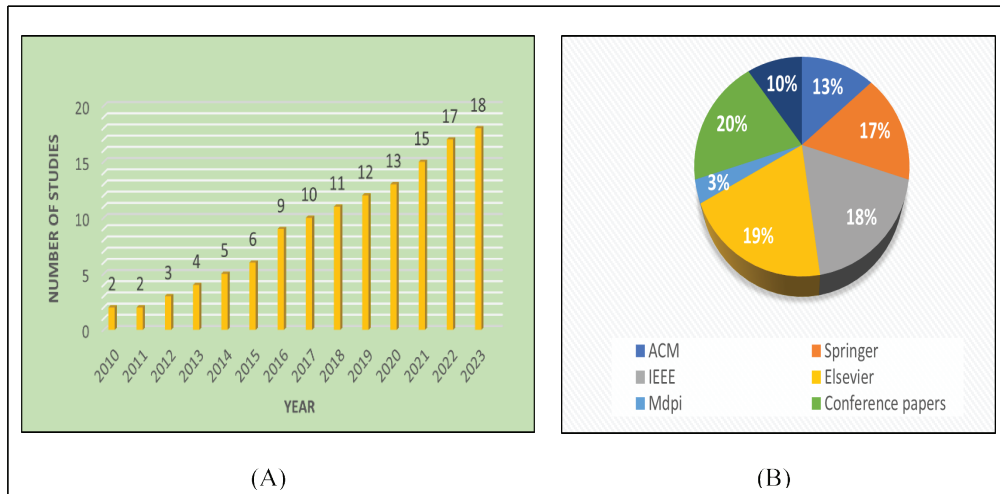


Figure 1: Bibliometric analysis of AD literature (A) Chronological distribution of bibliometric papers (B) Percentage distribution of publication types. AD, anomaly detection.

of the 19 most popular clustering algorithms with a detailed analysis of traditional and modern approaches to emphasize the importance of the essential data-analysis technique, clustering [24].

a.iii 2016–2023

In 2016, a comprehensive assessment of 19 unsupervised AD techniques was conducted on 10 datasets, showcasing their practical significance across diverse applications [1]. Another study in 2016 compared four AD algorithms, highlighting the superior performance of one-class SVM in smart city wireless sensor networks [2]. In 2018, a survey focused on comparing feature techniques and bridging the gap between IDS and forensic investigation [3]. A 2019 study evaluated unsupervised AD algorithms against unified attack models and addressed research questions related to performance. A detailed review in 2019 explored UL methods in various networking domains, discussing recent advances, challenges, and future directions [4]. A general introduction to the ML algorithms applied to IDS can be found in (2020). The proposed work also performs experiments on the KDD-CUP dataset to evaluate the performance of ML algorithms [5]. In 2021, the authors presented a comprehensive survey on IDS with special attention to feature-selection methods, different types of datasets, challenges, and possible future directions [6].

The survey conducted in 2021 categorizes anomaly detection algorithms into seven distinct categories based on their working mechanisms [7]. These

categories include statistical models, density-based methods, distance-based techniques, clustering approaches, isolation methods, ensemble methods, and subspace techniques. For each category, the survey provides insights into the time complexity of the algorithms and outlines their general advantages and disadvantages. Furthermore, a detailed and systematic survey of clustering methods is provided in (2022). This paper covers an in-depth overview of state-of-the-art clustering methods, performance, open issues, and various application areas of clustering methods [8]. A comprehensive overview of UL techniques for AD was conducted in 2022, emphasizing their utility in scenarios with scarce labeled data [9]. It highlights the challenges associated with traditional outlier detection methods and underscores the potential of unsupervised approaches in handling raw data effectively. Another survey was conducted in 2022 to delve into unsupervised anomaly localization (AL) using DL for industrial image inspection [10]. With its in-depth technical insights, the survey emerges as a vital resource for researchers venturing into industrial AL and seeking to extend its applications across diverse domains.

In a 2023 survey, outlier detection strategies from seven families were analyzed, revealing that consensus-based strategies derived from DL outperformed standalone UL methods [11]. Another comprehensive study in 2023 explored anomaly detection techniques within sensor networks and the internet of things (IoT) [12]. It unveiled a significant focus on detecting malicious activities, spanning various ML and DL approaches, with a notable surge in interest between

2017 and 2022. The study showcased a significant adoption of DL and ML techniques, notably MLPs and graph-based methods, across various anomaly detection applications, reflecting the evolving landscape of IoT security measures and the need to address emerging threats comprehensively. A review conducted in 2023 focused on different clustering categories within semisupervised and UL approaches [13]. Furthermore, another survey conducted in 2023 delved into automatic log file analysis for early incident detection, particularly focusing on self-learning anomaly detection techniques [14]. These methods, predominantly leveraging DL neural networks, demonstrated superior performance over traditional approaches and addressed issues with unstable data formats. A comparison of existing anomaly detection surveys with our study is presented in Table 1.

From Table 1, existing literature offers surveys on ML techniques for NAD and other domains but lacks a comprehensive assessment of UL techniques specifically for NAD systems. This paper fills that gap by comparing UL techniques based on relevant parameters, as mentioned in Table 1. It also includes recent advances in unsupervised NAD and empirical evaluations on benchmark datasets to analyze performance across conventional and modern datasets.

b. Motivation

The motivation for this paper stems from the need to address the shortcomings observed in existing research on UL approaches in NAD systems. The paper aims to fill various gaps identified in previous studies by conducting a comprehensive survey. These gaps include the need for more thorough comparative analyses that consider the characteristics of different algorithm classes, the limited generalizability of findings due to analyses on limited datasets, and insufficient emphasis on recent network datasets in evaluations. Moreover, previous reviews require more coverage of recent advances in NAD systems and performance evaluations on benchmark datasets such as NSL-KDD, UNSW-NB15, and CIC-IDS2017. To bridge these gaps, this paper evaluates 13 UL algorithms on popular benchmark datasets, providing a benchmark for anomaly-based IDS. Furthermore, the analysis conducted in this paper highlights the strengths and weaknesses of UL algorithms across varied dataset environments, offering valuable insights for further research and development in the field of NAD systems.

c. Contribution

The key contributions of this paper are highlighted as under:

- It discusses the state-of-the-art unsupervised AD methods and reviews their utilization for detecting network attacks.
- It presents contemporary coverage of recent trends and innovations in the respective field.
- This paper investigates the experimental comparison of 13 selected unsupervised AD methods on three popular benchmark datasets. The analysis of results is also presented.
- The findings of the experiments provide a basis for evaluating the performance of algorithms across different families of AD techniques about network datasets and for forecasting the efficacy of the chosen algorithm under varying parameter settings. It can be utilized to assist in the selection of a suitable UL algorithm for a given task.
- Finally, the paper presents future works highlighting potential pitfalls and research gaps in the literature.

d. Organization

The present paper is presented in different sections. Section 2 introduces 13 selected UL-based methods from the pool of 6 classes of AD. This section also presents an unsupervised NAD detection model. Section 3 is the core section that discusses literary works in detail by proposing the classification of UL approaches into distinct classes. It also presents recent works to highlight the latest developments and innovations. Section 4 performs the experimental evaluation of 13 selected UL algorithms on 3 benchmark datasets. In Section 5, research challenges and future directions are detailed. Finally, the paper is concluded in Section 6. The roadmap of the paper is depicted in Figure 2.

II. UI-Based Techniques

The following section discusses the various UL methods that are used for AD. A generic framework of the NAD model employing K-means clustering is also presented. The focus of this section is to provide an in-depth discussion of six different families of AD and their types.

Table 1: A relative comparison of this paper with existing surveys/review papers.

References	Discussion	Focus	1	2	3	4	5	6
[15]	Focus on literature review and performance evaluation of UL methods.	IDS	≈	×	×	×	✓	×
[16]	It provides a comparative analysis of various unsupervised NAD approaches and investigates standard metrics suitable for many connections.	AD in networks	≈	✓	×	×	✓	×
[17]	Presents a detailed review of different clustering methods and their strengths and weaknesses.	–	≈	✓	×	×	×	×
[18]	Presents a framework for evaluating anomaly detection methods for HTTP.	HTTP AD	≈	≈	×	×	✓	×
[19]	An overview of ML techniques for intrusion detection between 2000 and 2007 is presented.	IDS	✓	×	×	≈	×	×
[20]	It provides an overview and comparison of various clustering techniques, pros, and cons.	AD	≈	≈	×	×	×	×
[21]	Performance evaluation of four novelty detection algorithms is carried out.	Novelty detection	✓	✓	×	×	✓	✓
[22]	Presents a detailed review of popular traditional and modern clustering approaches.	–	≈	≈	×	×	×	×
[1]	Nineteen widely employed unsupervised techniques are evaluated to provide an understanding of their performance in various domains.	AD	≈	✓	×	×	✓	×
[2]	The proposed work focuses on a comparative evaluation of four UL algorithms.	Smart city wireless networks	≈	✓	×	×	✓	×
[3]	A survey on ML techniques focusing on unsupervised and hybrid IDS is presented.	IDS	✓	✓	×	✓	×	✓
[4]	The proposed work aims to provide an experimental comparison of unsupervised methods employed for NAD against five intrusion detection datasets.	Anomaly-based IDS	≈	✓	×	✓	✓	×
[23]	A comprehensive survey of UL techniques in networking is presented.	Anomaly-based in networking	≈	✓	✓	×	×	✓
[5]	Review and compare ML algorithms for IDS.	IDS	✓	≈	×	×	✓	×
[6]	Presents a detailed review of state-of-the-art IDS methods, datasets, performance measures, and research challenges.	IDS	✓	≈	×	✓	✓	✓
[8]	An in-depth review of state-of-the-art clustering methods is presented.	-	≈	✓	✓	×	×	✓
[9]	A comprehensive overview of UL techniques for AD is presented, emphasizing their utility in scenarios with scarce labeled data.	AD in industrial applications	✓	✓	✓	×	×	✓
[11]	Review and compare outlier detection techniques from seven different families.	Outlier detection	≈	≈	×	×	✓	×

(Continued)

Table 1: Continued

References	Discussion	Focus	1	2	3	4	5	6
[12]	AD techniques are explored comprehensively to address the detection of emerging threats.	IoT and sensor data	✓	≈	✓	×	×	✓
[13]	In-depth review of various unsupervised and semisupervised clustering methods.	-	✓	✓	≈	×	✓	×
[14]	Focus on log file analysis for early incident detection, particularly emphasizing self-learning AD techniques.	AD	✓	≈	✓	×	×	×
Our survey	The primary focus is on studying UL NAD methods while considering recent advances.	NAD	✓	✓	✓	✓	✓	✓

DL, deep learning; IDS, intrusion detection systems; IoT, internet of things; ML, machine learning; NAD, network anomaly detection; UL, unsupervised learning.

Note: 1: Review of ML and DL-based NAD approaches; 2: Unsupervised methods; 3: Recent advances; 4: Comparative analysis of network datasets; 5: Experimental analysis; 6: Open issues/future directions; Notations: ✓: Included; ×: Not included; ≈: Partially included.

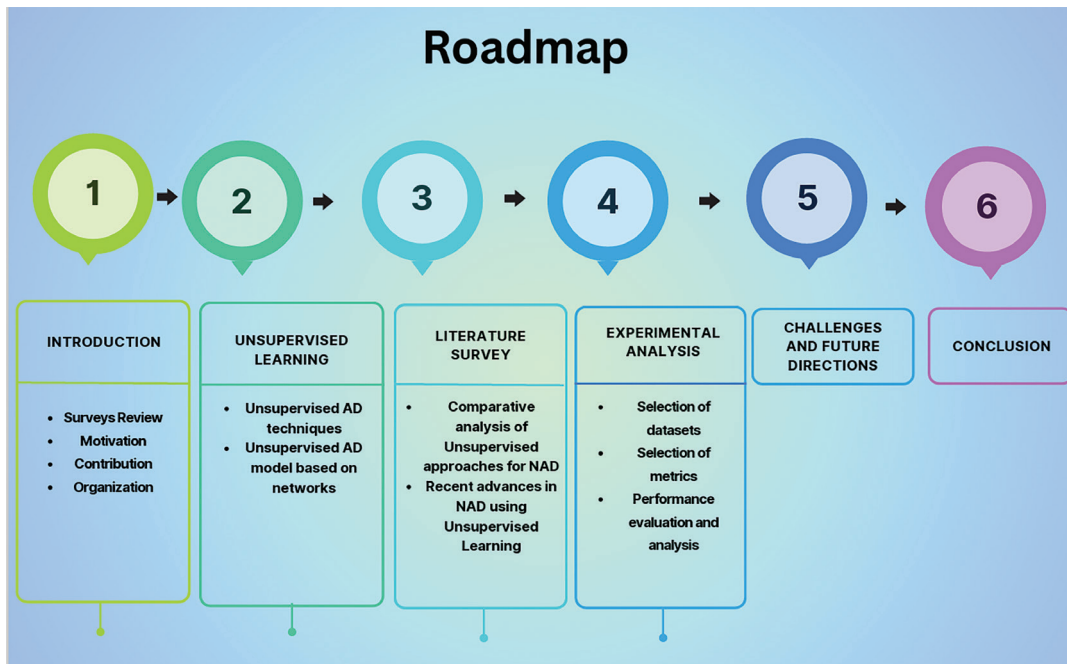


Figure 2: Roadmap of the paper.

a. Selection of the algorithms

The literature review evaluates several UL algorithms based on their performance in NAD systems. A taxonomy of 13 unsupervised algorithms is proposed from the pool of 6 families of AD techniques, as presented in Figure 3, based on the following criteria:

- Based on previous practical applications, primarily employed in intrusion detection/NAD.
- The collection of algorithms should include the following classes of AD techniques: clustering, dimensionality reduction, statistical, neighbor-based, density-based, and classification-based to develop an effective prediction model.

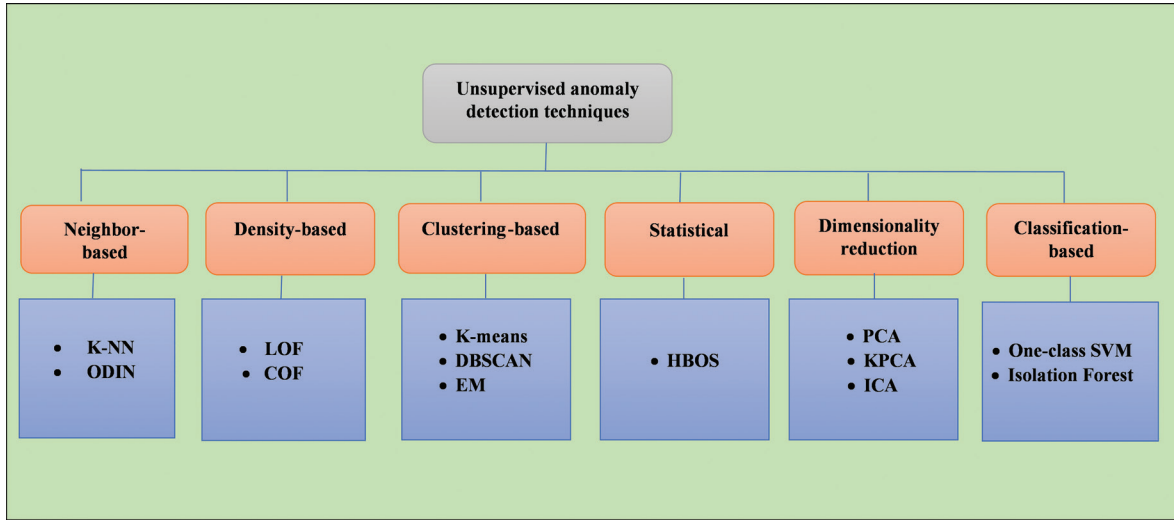


Figure 3: Taxonomy of 13 selected UL methods for anomaly detection from 6 families. UL, unsupervised learning.

- Unsupervised dimensionality reduction algorithms are also included in 13 selected methods to facilitate non-redundant and relevant features in the dataset and support the effective deployment of UL methods in real-world data with higher dimensions. The categorization of 13 UL methods selected from 6 families of AD is presented in Figure 3.

The proposed taxonomy of UL methods from the six families of AD techniques is illustrated in Figure 3. This taxonomy aids in developing AD models for detecting network anomalies and intrusions. The taxonomy of AD techniques is discussed below and empirically evaluated in Section 5, providing insights into their performance across diverse network datasets.

a.i Neighbor based

- KNN* [24]: It is a neighbor-based technique. This algorithm focuses on detecting outliers. The KNNs of a given data point are determined by comparing it to the entire dataset and selecting the items with the most comparable feature values. The data point is considered abnormal if the most nearest neighbor (NN) was previously categorized as abnormal.
- Outlier detection using indegree number (ODIN)* [25]: It derives from the KNN algorithm, which analyzes the entire dataset to find the feature distances to the specified point [26]. This algorithm permits isolating NN, so the KNN graph is produced. Unlike KNN, ODIN classifies the data points as anomalies with a few nearby edges in the KNN graph.

a.ii Density based

- LOF* [27]: It utilizes the concept of local density to identify anomalous points/outliers in the datasets. The NN is calculated for each data point. Next, the local reachability density (LRD) is calculated using the computed neighborhood. The LOF score is determined by comparing the LRD of a data point to the LRD of a previously computed NN to find outliers [28]. A data point is more likely to be an outlier if its LOF value is high.
- Connectivity-based outlier factor (COF)* [29]: The LOF is upgraded. It works by ascribing a degree of outlieriness to each data point. This degree of anomaly is called the COF of the data point. A data point with a high COF value indicates a high likelihood of being an outlier.

a.iii Clustering based

- K-means clustering* [30]: A popular clustering technique that divides data points into k clusters by feature values. The k-cluster centroids are initialized randomly [26]. Then, each data record is assigned to the cluster with the nearest centroid, and the modified cluster centroid is recalculated. This procedure stops when the centroids stop changing anymore. Data points with comparable feature values are clustered using a distance algorithm. The final step involves calculating the scores of each data point within a cluster as the distance to its centroid [31]. Data points distant from their clusters' centers are flagged as anomalies.

- *Expectation maximization (EM)* [32]: It is based on the Gaussian mixture model (GMM). The GMM approach improves the density of sample data represented as a function of a single-density estimation method with multiple Gaussian probability density functions to describe the data distribution. EM and K-means are comparable in facilitating model refinement through an iterative process to discover the optimal configuration. However, the K-means algorithm uses a different method for determining the Euclidean distance when computing the distance between each pair of data points, whereas EM employs statistical methods [32].
- *Density-based spatial clustering of applications with noise (DBSCAN)* [33]: It is a density-based clustering technique that uses the hypothesis that clusters are discrete high-density regions separated by low-density regions [34]. Data points “densely grouped” are aggregated into a single cluster. It can discover clusters in huge spatial datasets by analyzing the local density of the data points [35]. Each object that does not belong to any cluster is considered noise. The most impressive aspect of DBSCAN clustering is its robustness against outliers.

a.iv Dimensionality reduction

- *Principal component analysis (PCA)* [36]: PCA is a dimension reduction strategy that speeds up the learning process with fewer computations. It is a popular unsupervised method for defining the mapping from high- to low-dimensional space. It uses an orthogonal transformation to transform observations of correlated features into a set of linearly uncorrelated features. PCA considers the variance of each feature because a high value indicates a clear separation between classes and, therefore, lowers the dimensionality. It generates a set of principal components ranked by variance (the first component has a greater variance than the second, and so on), uncorrelated, and few.
- *Kernel principal component analysis (KPCA)* [37]: An extension of the PCA technique allows non-dimensionality reduction. If the data structure is nonlinear and cannot fit well in linear subspace, KPCA enables us to generalize PCA in nonlinear datasets. It employs one of the kernel methods to translate the nonlinear features onto a higher-dimensional feature space.
- *Independent component analysis (ICA)* [38]: It is a computational method mainly used in signal processing and differentiating. It finds hidden components inside datasets of random variables,

signals, or measurements. It aids in splitting multi-dimensional signals into discrete parts considered non-Gaussian and independent of one another. In contrast to PCA, which maximizes the variance of the data points, ICA emphasizes independence, i.e., separate components [39].

a.v Statistical based

- *Histogram-based outlier score (HBOS)* [40]: A statistical method creates a histogram for each independent feature in the dataset. Histograms are first constructed using the feature value distributions of all the available data points [41]. Subsequently, the anomaly score for each data point is determined by multiplying the inverse heights of the columns into which its features fall.

a.vi Classification based

- *One-class SVM* [42]: One-class SVM is a UL method for learning how to distinguish test samples of a specific class from those of other categories. It works on the fundamental principle of reducing the hypersphere of a single course of instances in training data, assuming that any sample outside the hypersphere is an outlier or outside the training data distribution.
- *Isolation forest (IF)* [43]: IF works on the premise that anomalies are “few and distinct” data points. The data in an IF are sampled randomly and then processed using a tree structure with features picked randomly. Due to the increased number of cuts required to separate samples from deeper within the tree, there is a lower probability that they include anomalies. Moreover, the examples that fall into shorter branches are anomalous because the tree was able to differentiate them from the other observations more efficiently [44].

b. Unsupervised NAD model

This section presents the NAD model based on UL, which mainly employs K-means clustering for grouping data instances.

Figure 4 displays the framework of the NAD model that utilizes UL. The framework provides a detailed methodology for the NAD model and consists of several phases, each serving a specific purpose to enhance the effectiveness of NAD systems. These phases include preprocessing, feature selection, grouping based on K-means clustering, and detection using a Random Forest classifier. The preprocessing phase ensures data integrity by converting categorical

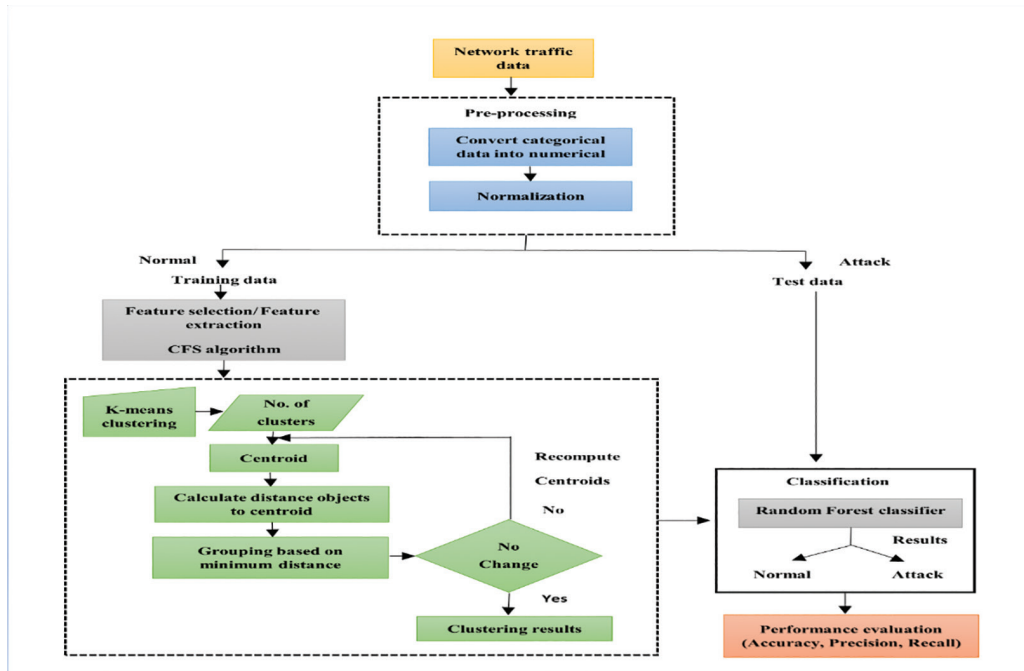


Figure 4: Framework of NAD based on UL [45]. NAD, network anomaly detection; UL, unsupervised learning.

variables into numerical format for improved processing. The feature selection phase employs correlation-based feature selection (CFS) to optimize dataset dimensions while preserving the intrinsic meaning of features. The grouping phase involves K-means clustering, which uses similarity measures to create a comprehensive normal behavioral profile. UL based on clustering enhances exploratory data analysis, exposing underlying structures. In the detection phase, the framework employs a Random Forest classifier, a powerful decision-making tool, to accurately identify data instances. The framework's contribution extends beyond precise categorization; it reveals meaningful patterns within cohesive subgroups, adding depth to understanding network intrusions. A brief description of the unsupervised model is discussed below. The various stages are:

Preprocessing: The preprocessing phase is fundamental for ensuring the quality of the dataset. Cleaning data and transforming categorical variables into numerical form are crucial steps. Incorporating min-max normalization scales data variables within a small range enhances computational efficiency.

Feature selection: Feature selection is introduced to reduce the dimensionality of the dataset cost-effectively. The CFS method is chosen for its quick

processing capabilities and low computational cost, retaining the physical meaning of features.

Grouping: The grouping phase is designed to build a normal behavioral profile using K-means clustering. This phase is pivotal for forming distinct clusters based on similarity measures. The iterative nature of K-means clustering involves centroid initialization, distance estimation, and cluster optimization to ensure convergence and meaningful cluster formation. Centroid initialization, distance estimation, and cluster optimization are fundamental to K-means clustering, vital for convergence and meaningful pattern extraction. The process begins by initializing “K” centroids, influencing convergence, and final cluster assignments. Distance estimation calculates distances between data points and centroids, guiding data point assignments based on similarity. Cluster optimization recomputes centroids iteratively, ensuring convergence with minimal changes or reaching a predefined limit. Convergence assurance is crucial for stable and reliable results. Simultaneously, meaningful patterns emerging as centroids dynamically adjust based on data point assignments, capturing significant dataset structures.

Detection: In the final detection phase, the framework adopts a Random Forest classifier, a choice rooted in its reported excellent accuracy in the literature. The application of Random Forest enables the

categorization of data instances into normal or attack categories. Key metrics such as accuracy, recall, and precision are employed to evaluate the model's performance.

III. Literature Survey

In this section, various literature works employed for network intrusion detection using UL are discussed in detail. The preliminary studies categorize various UL approaches into six classes, providing a comprehensive framework for organizing and analyzing this field's diverse methods.

a. UL techniques

This study aims to classify the existing literature on UL approaches into different categories to improve understanding and facilitate effective comparisons. The identified classes include clustering, SOM, dimensionality reduction, classification-based techniques, hybrid methods, and DL, as depicted in Figure 5. By organizing the diverse range of approaches into these distinct classes, this classification framework facilitates meaningful comparisons and analysis, aiding researchers and practitioners in selecting the most suitable approaches for their specific tasks and advancing the field of UL.

a.i Clustering-based

An unsupervised approach based on density and grid-based clustering was proposed in 2005 [46]. It effectively classified a high-dimensional dataset into a

collection of clusters in which the points not associated with any cluster are flagged as abnormal. The presented strategy achieved comparable outcomes; still, the FPR was high [46]. Another advanced method based on fuzzy rough C-means clustering was introduced in further research [47, 48]. K-means clustering is a well-known methodology for finding anomalies, outperforming prior unsupervised algorithms in accuracy [49]. Later 2012, a new variant was introduced that merged K-means clustering with the C4.5 decision tree approaches to generate more effective outcomes than previous methods [50]. Density-based clustering methods excel at identifying clusters of arbitrary shapes, while grid-based clustering schemes offer fast processing.

Leung and Leckie [46] proposed the merging of adaptive intervals approach to spatial data (MAFIA). This novel mechanism combines density-based and grid-based clustering to discover high-dimensional input space clusters. Gujral et al. [51] employed spectral clustering to identify scattered anomalies and form representative clusters. A sparse symmetric matrix was utilized with a KNN to address scalability in IDS. In another work [52], a two-fold framework for NAD in high-dimensional data was proposed by Ni et al. [52]. The framework uses a clustering technique to approximate cluster centers and the mutual information coefficient to remove redundant and irrelevant features. Comparing experimental results to those of other unsupervised approaches, accuracy was enhanced. To represent normal data into clusters, Bhuyan et al. [53] presented an effective outlier-based technique employing a variation of k-means clustering. Based on outlier scores that exceed a threshold, anomalies are found.

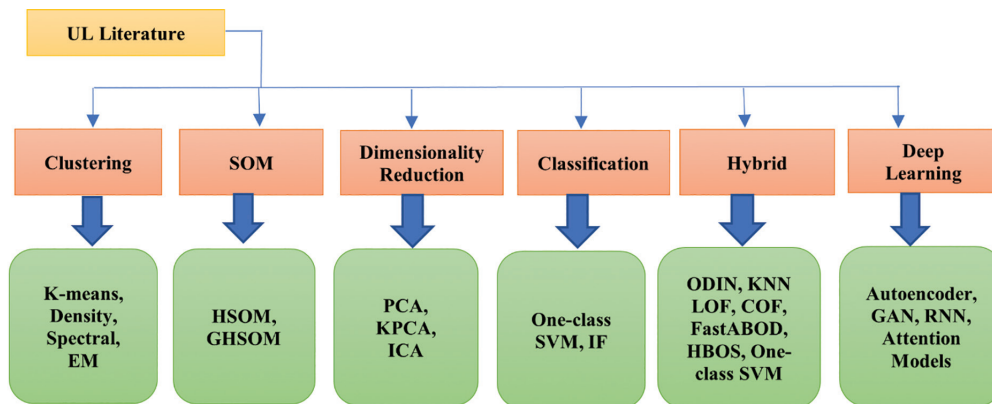


Figure 5: Categorization of UL techniques [23]. UL, unsupervised learning.

Syarif et al. [54] compared misuse detection and AD algorithms using four classifiers (Naïve Bayes, rule induction, NN, and decision tree) for misuse detection and five clustering algorithms (k-means, k-medoids, EM clustering, and distance-based outlier detection) for AD. Evaluation of DARPA98 and NSL-KDD datasets showed that detecting unknown attacks remains challenging for misuse detection. Rule induction achieved 99.58% accuracy with a 0.40% FPR for known attacks, while the decision tree classifier performed poorly with 63.97% accuracy and a 17.90% FPR. In contrast, the distance-based outlier detection algorithm achieved 80.15% accuracy in detecting unknown intrusions in the AD module. Bhuyan et al. [55] proposed Tree-based Subspace Clustering (TreeCLUS), a UL approach for detecting network anomalies with a minimized FPR. It utilizes cluster stability analysis to identify effective clusters and a multi-objective scheme for cluster labeling.

In another work, Prasad et al. [56] proposed an unsupervised feature selection technique for IDS. This technique selects the most informative features by calculating feature scores using entropy and computing the feature histogram. The study also introduced a novel cluster center initialization approach to address the limitations of conventional clustering techniques. The approach computed semi-identical instances, created sets, and avoided outliers as initial cluster centers. Micro-clusters were formed based on spatial distance, and similar micro-clusters were merged into macro-clusters. The presented cluster centralization approach outperformed basic clustering methods, requiring fewer iterations to obtain final clusters and improving the accuracy of the detection model.

UNADA [57] is a distributed computing framework-based scheme designed to enhance the detection of NAD unsupervised algorithms. It leverages a density-based grid algorithm deployed on a large cluster of servers to process high traffic volumes efficiently. However, DBSCAN's limitations in handling high-dimensional data are addressed by introducing subspace clustering and evidence accumulation techniques for parallel processing across multiple servers. Despite its advantages, UNADA's time complexity hampers its suitability for real-time and scalable anomaly detection. In another work [58], a real-time UL anomaly detection scheme for streaming data is proposed. Hierarchical temporary memory (HTM) detects spatial and temporal anomalies in noisy environments. The strategy adapts well to changing data streams while minimizing false positives. However, there is potential for further accuracy improvement by

adopting ensemble-based approaches. Additionally, the exploration of real-time anomaly detection techniques for multivariate data streams is crucial to address the challenges of the current scenario.

a.ii SOM

SOMs are frequently applied as unsupervised tools for identifying malicious and abnormal network behavior. The characteristic of SOMs is that they can automatically organize and infer patterns from a range of inputs and then assess whether a piece of new information conforms to the deduced pattern or not, hence identifying anomalous inputs [6, 59, 60]. Landress [61] proposed a fusion model for UL in intrusion detection to address false positives. The model combines K-means clustering, J48 decision tree, and SOM using the KDDCUP99 dataset. Clustering is performed with k-means, grouping attacks into normal, Neptune, smurf, and ipsweep categories. J48 decision tree is then used for feature selection, reducing false positives by selecting optimal features. SOM is applied to visualize the data and categorize abnormal behavior.

Huang and Huang [62] employed GHSOM to analyze anomalous traffic behavior by analyzing node density and samples. The network traffic data are analyzed via visual representation of attack patterns concerning hierarchical relationships. In the detection phase, GHSOM is combined with SVM, which provides classification rules to identify the network traffic into normal and abnormal behaviors.

a.iii Dimensionality reduction

PCA is commonly used for dimensionality reduction in IDS [63, 64], but ICA is proposed as an alternative for non-Gaussian data [65]. Pattewar and Sonawane [66] present a feature extraction approach based on PCA and KPCA. In the first approach, PCA is utilized for extracting features. KPCA and Bayesian methods are employed in the second approach to eliminate the noisy features, thereby removing the critical features that result in better accuracy and computational efficiency than that of PCA.

In another work proposed by Kuang et al. [67], hybridization of the KPCA, GA, and SVM model is employed. In the proposed scheme, KPCA is applied to elicit the core features and the NRBF kernel is employed to lessen the training time of SVM. At the same time, GA optimizes the parameters of SVM to avoid overfitting. The experimental findings reveal that the proposed approach shows better generalization

performance and a higher accuracy rate than the SVM classifier. In the work of Elkhadir et al. [68], a comparative analysis of PCA and KPCA is performed on the KDDCUP99 dataset. Experiments reveal that when the number of NN is between one and four, a KPCA with a quadratic kernel offers a higher detection rate (DR) than a PCA, notably for identifying DOS and PROBE attacks.

a.iv Classification-based

Unlike attack records, normal connection records are often readily available in real-world applications. As a result, one-class SVM trained on a dataset consisting of normal connection records can effectively differentiate between normal and varied attacks. Amer et al. [69] report that one-class SVM is susceptible to data outliers and suggest two modifications: robust one-class SVM and ETA one-class SVM. The central concept for robust one-class SVM is to minimize the mean square error (MSE) for dealing with outliers by utilizing the class mean as the averaged information. In contrast, ETA one-class SVM employs an outlier suppression mechanism. The proposed approach indicated that the enhanced one-class SVM performed better on two of four datasets than the existing UL AD methods.

Nguyen et al. [70] further improve the performance of one-class SVM by introducing the nested one-class SVM. Radial basis function (RBF) or a Gaussian kernel is commonly used to kernelize one-class SVM. However, selecting a Gaussian kernel hyperparameter based on the attack's availability is impractical. The authors attempted to solve the problem of kernel parameter optimization by proposing data-driven hyperparameter optimization. The presented methodology has achieved high detection accuracy and a lower false alarm rate (FAR) than the original one-class SVM in the experiments.

Verkerken [71] conducted a study on the generalization ability of UL algorithms in NAD. The four UL methods used for evaluation were PCA, IF, autoencoder (AE), and one-class SVM. The models were trained on one dataset and evaluated on another to assess their inter-dataset generalization. Experimental results showed that one-class SVM achieved the best AUROC scores, followed by the AE and PCA. However, when evaluated on different datasets, there was a significant drop in performance for all models, indicating a lack of generalization. The inter-dataset evaluation scheme highlighted the low generalization strength of the models. The study suggests that further research is needed to improve the

adaptability and generalization of UL models for real-time detection in practical applications.

a.v Hybrid

Zoppi et al. [26] conducted an extensive study investigating the relationship between attack families, anomaly classes, and detection methods. Their experiments on various UL algorithms, including ODIN, K-means, LOF, FastABOD, HBOS, and one-class SVM showed that FastABOD is effective for point anomalies, K-means performs well for collective anomalies, and HBOS is lightweight and efficient for limited computing resources. Additionally, the analysis of HBOS in Ref. [72] found that dynamic bins are particularly useful for identifying R2L and U2R attacks. NN-based approaches are commonly used and exhibit strong performance due to their unsupervised nature and outlier detection criteria. However, they may face challenges with high-dimensional data and incur computational complexity when dealing with large datasets. On the contrary, SVM maximizes the machine's accuracy concerning the training dataset while maximizing its capacity to correctly classify subsequent testing datasets, even on high-dimensional datasets. Therefore, combining NN-based approaches and SVM enhances classification performance with reasonable computational cost in high-dimensional data [73]. In contrast, SVMs maximize accuracy on training and testing data, including high-dimensional datasets, with one-class SVMs focusing on discovering decision boundaries with maximum distance from the origin [73].

Falcão et al. [4] compared various UL algorithms for NAD. Classification algorithms were found to exhibit the highest accuracy among the methods considered. Angle-based algorithms showed robustness in parameter selection. However, detecting anomalies with nonrepeatable and unstable behavior remained challenging, and higher detection scores were observed in datasets with rare anomaly events. In the literature, many proposed NAD schemes rely on traditional distance metrics such as Euclidean or Mahalanobis distance, which may not adequately capture the heterogeneous nature of network features.

To overcome this limitation, Aliakbarisani et al. [74] proposed a linear metric learning approach incorporating side information and UL techniques to extract similarity or dissimilarity constraints. Their approach formulates an optimization problem using the constraint trace ratio and utilizes Laplacian Eigenmap for preserving local neighborhood information. Experimental results on Kyoto 2006+ and NSL-KDD

datasets demonstrated improved accuracy and reduced FPRs compared to Euclidean-based distance metrics.

Preprocessing steps are essential in NAD. One common technique is using LOF to remove outliers from the training set. LOF detects network anomalies by comparing them with new data and estimating the LOF value based on preprocessed normal datasets [75]. Additionally, Ding et al. [76] proposed hybridizing the similarity matrix scheme and LOF as preprocessing steps to identify outliers by considering data objects with lower similarity than others. This approach enhances outlier detection adaptability, especially for datasets with irregular shapes.

a.vi DL

With the rapid growth of the IoT, attackers are exploiting vulnerabilities in insecure devices like IP cameras to launch devastating Distributed Denial of Service (DDoS) attacks [77]. Traditional defense methods that rely on predefined features extracted from predefined flows are often ineffective in promptly detecting and blocking malicious flows. Hwang et al. [83] proposed a new scheme for automatically profiling traffic patterns directly from raw traffic to address this. By examining the initial packets and utilizing a convolutional neural network (CNN), the method generates features from raw traffic, enabling auto-learning. An unsupervised AE is then used to create profiles of benign traffic, effectively filtering out abnormal traffic. Experimental results showed a promising outcome with 100% accuracy and low false negative (0.02%) and false positive (0.83%) rates achieved by the proposed method.

Kabir and Luo [78] evaluated the performance of traditional and DL-based unsupervised methods for network flow anomaly detection. The study compared SOM, K-means, deep autoencoding GMM (DAGMM), and adversarially learned anomaly detection ALAD algorithms. Experimental results on the KDDCUP99 dataset showed that DAGMM achieved the lowest FPR (0.9%), while SOM had the highest accuracy (99.9%). SOM and k-means were found to be easy to implement with stable performance. ALAD performed well in detecting minority attacks due to generating adversarial samples.

In another study, Truong-Huu et al. [79] investigated the deployment of generative adversarial networks (GANs) for NAD. They utilized the UL scheme, which does not require prior label information during training. The concepts of traffic sessions and traffic aggregation techniques were introduced to

extract useful traffic features and learn traffic behaviors. The experiments conducted on UNSW-NB15, CIC-IDS2017, and synthetic datasets demonstrated that GAN outperformed existing anomaly detection approaches.

Sovilj et al. [80] addressed the lack of descriptive analysis in intrusion detection models by introducing an attentional model analysis for DL models. This analysis provides explanations and predictions, aiding network operators in understanding potential threats. The authors conducted a comparative analysis of deep neural learning architectures for IDS in sequential data streams. Experimental results on synthetic and CIC-IDS2017 datasets showed that the RNN variant outperformed the nonsequential deep AE for anomaly detection. However, the performance of the attentional model could be further improved by incorporating attentional components to detect anomalous input data segments. Another study by Carrera et al. [81] addresses the critical need for detecting zero-day cyber-attacks, which exploit undisclosed vulnerabilities, posing substantial technical and economic risks to enterprises and technology-dependent systems. Through benchmarking three different approaches, such as deep autoencoding with GMM and IF, the proposed work demonstrates enhanced accuracy with IF. Furthermore, employing the SHAP methodology elucidates crucial features for attack classification, as validated through experiments conducted on diverse datasets, including KDD99, NSL-KDD, and CIC-IDS2017. Sáez-de-Cámara et al. [82] address IoT security challenges by proposing a federated learning (FL) architecture for unsupervised network intrusion detection. Leveraging FL mitigates network overhead and data isolation issues, enhanced by an unsupervised device clustering algorithm. Evaluation on a diverse testbed showcases successful device grouping, although clustering quality may diminish with insufficient local training data. Notably, FL-based models exhibit faster convergence and low FPRs, outperforming ML-based alternatives like Kitsune, signaling promise for real-world deployment. However, challenges remain in adapting the approach to high-mobility IoT settings and addressing compromised devices in adversarial scenarios, warranting further exploration. The comparison analysis of UL approaches employed for network anomaly/intrusion detection is summarized in Table 2, providing an overview of the findings and insights regarding the performance and effectiveness of different methods.

Table 2 offers a comprehensive overview of various network anomaly and intrusion detection strategies. It delineates different methodologies and

Table 2: Relative comparison of UL approaches for network anomaly/intrusion detection.

Authors	Methodology	Algorithm	Dataset	Input data	DoS	Probe	R2L	U2R	Real-time detection
[83]	Unsupervised approach for outlier detection using subspace clustering and evidence accumulation.	DBSCAN	METROSEC	Continuous	95	95	85	85	✓
[84]	Tree-based subspace clustering approach for high-dimensional datasets; includes cluster stability analysis.	Tree-based subspace clustering (TCLUS)	KDDCUP99, TUIDS	Mixed	99	96	86	66	
[85]	Multiclustering scheme incorporating subspace clustering and evidence accumulation to overcome knowledge-based approach limitations.	DBSCAN	KDDCUP99, METROSEC	N/A	-	-	-	-	✓
[86]	Particle swarm optimization clustering strategy based on map-reduce methodology for parallelization in large-scale networks.	PSO clustering	KDDCUP99	Mixed	Max AUC: 96.3				
[87]	Novel strategy for automatic tuning and optimization of detection model parameters in a real network environment.	Clustering and one-class SVM	KDDCUP99 and Kyoto University	Continuous	-	-	-	-	✓
[88]	K-means clustering to generate training subsets; neuro-fuzzy and radial-basis SVM for classification.	K-means, SVM, neuro-fuzzy neural network	KDDCUP99	N/A	98.8	97.31	97.5	97.5	
[89]	Innate-immune strategy via UL for categorizing normal and abnormal profiles.	DBSCAN	KDDCUP99	N/A	FPR: 0.008, TNR: 0.991, ACC: 77.1, Recall: 0.589				✓
[49]	Novel approach using cluster centers and NNs to transform the dataset into a one-dimensional feature set.	Clustering, KNN	KDDCUP99	N/A	99.68	87.61	3.85	57.02	

(Continued)

Table 2: Continued

Authors	Methodology	Algorithm	Dataset	Input data	DoS	Probe	R2L	U2R	Real-time detection
[90]	Nature-inspired meta-heuristic approach to optimize the optimum path forest clustering algorithm.	Optimum path forest clustering	ISCX, KDDCUP99, NSL-KDD	N/A	Purity measure: ISCX: 96.3, KDDCUP99: 71.66, NSL-KDD: 99.8				
[91]	UL technique for real-time detection of fast and complex zero-day attacks.	DBSCAN	DARPA, ISCX	N/A	ACC: 98.39%, Recall: 100%, Precision: 98.12%, FP: 3.61				✓
[92]	Modified optimum path forest algorithm to enhance IDS performance, particularly in detecting less frequent attacks.	K-means, modified optimum path forest	NSL-KDD	N/A	96.89	85.92	77.98	81.13	
[93]	Hierarchical agglomerative clustering applied to SOM network to lower computational cost and sensitivity.	Hierarchical agglomerative clustering, SOM	NSL-KDD	Mixed	DR: 96.66%, ACC: 83.46, Precision: 75, FPR: 0.279, FNR: 0.033				

ACC, accuracy; AUC, area under curve; DBSCAN, density-based spatial clustering of applications with noise; DoS, denial of service; DR, detection rate; FAR, false alarm rate; FNR, false negative rate; FPR, false positive rate; IDS, intrusion detection systems; KNN, K-nearest neighbor; NN, nearest neighbor; SOM, self-organizing map; SVM, support vector machine; TPR, true positive rate; UL, unsupervised learning.

algorithms, ranging from subspace clustering to tree-based clustering and optimization techniques, all tailored to different datasets. The performance metrics exhibited by these approaches vary significantly, with some achieving commendable accuracy and DRs.

The table delineates these methodologies and offers a nuanced insight into their performance metrics. Some methods demonstrate impressive accuracy and DRs, showcasing their efficacy in identifying anomalous activities within network traffic. However, a notable observation is the need for more emphasis on real-time detection capability across these methodologies. Real-time detection is imperative for promptly identifying and mitigating security threats, yet not all approaches prioritize this aspect adequately.

b. Recent advances in NAD using UL

This section delves into the most recent developments in anomaly-based network intrusion detection systems (NIDS) that play a critical role in protecting networks against ever-evolving cyber threats. Table 3 provides a comprehensive overview of recent research efforts to improve network security. It meticulously documents the contributions of various studies, highlighting innovative methodologies, algorithms, and techniques employed in modern IDSs.

Table 3 provides an overview of recent developments in unsupervised network intrusion/anomaly detection. Various techniques and algorithms, such as DBSCAN, one-class SVM, K-means clustering, AE, GANs, and tensor factorization, among others, have been discussed. The studies have utilized different datasets and evaluated the performance based on validation metrics such as accuracy, precision, recall, and F-score. In terms of recent developments, some notable advancements include the utilization of hybrid techniques to reduce FPRs, the application of evolutionary algorithms to optimize detection results, the use of DL models such as CNN and long short-term memory (LSTM) for spatiotemporal correlation analysis, and the exploration of ensemble approaches to improve detection performance.

Moreover, researchers have delved into ensemble approaches to bolster detection performance and have incorporated feature importance and data-reduction techniques to enhance model efficiency. The development of inter-dataset evaluation methods has further facilitated robust performance assessment across diverse datasets. However, there are still some limitations. These include improving DRs for minority-class attacks, investigating real-time packet clustering, and addressing the computational complexity

of certain approaches. Additionally, ongoing research is aimed at exploring multi-class classification techniques to differentiate between various attack types and enhance the generalizability of models across different datasets and network architectures.

IV. Experimental Analyses

In this section, an experimental evaluation of 13 selected UL algorithms from a pool of 6 families is conducted to analyze the performance of UL-based AD methods on network datasets.

a. Selection of the datasets

Researchers have created synthetic datasets used as benchmarks to examine new ML-based intrusion detection models and approaches. Synthetic datasets are generated in a limited, controlled environment and include benign and malicious traffic. These datasets have distinct features, usually presented in a flowchart-like structure. Some benchmark datasets include DARPA [110] (1998), KDD'99 [111] (1998), DEFCON (2000) [112], LBNL [113] (2004), Kyoto [114] (2006), NSL-KDD [115] (2009), CDX [116] (2009), ISCX 2012 [117] (2012), and CAIDA [118] (2013–17). Recently, datasets like UNSW-NB15 [119] (2015), CIDDS-001 [120] (2017), CIDDS-002 [120] (2017), CIC-IDS 2017 [121] (2017), CSE-CICIDS2018 [122] (2018), CICDDoS [123] 2019 (2019), BoT-IoT [124] (2019), and [125] (2020) have emerged as pivotal contributors. However, it is essential to discern their specific characteristics, particularly when comparing the most recent ones. While recent datasets like CICDoS2019, BoT-IoT, and IoT-23 focus on specific DDOS and IoT attacks, CIC-IDS2017 and CSE-CICIDS2018 cover a broader range of attacks.

To develop and evaluate intrusion detection frameworks, critical dataset characteristics include a diversity of attacks, anonymity, available protocols, complete network traffic capture, complete network interaction capture, defined network configuration, feature set, labeled data samples, heterogeneity, and metadata. CIC-IDS-2017 and CSE-CIC-IDS-2018 align with these characteristics and use profiles to generate well-ordered datasets. These datasets offer comprehensive insights into attacks, application models, network devices, and protocols. Although both datasets align with essential characteristics, CSE-CICIDS2018 is larger in size and presents challenges in processing due to the sheer volume of data instances in each file.

Table 3: Recent developments in unsupervised-based network intrusion/anomaly detection.

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
Amoli et al. (2015) [91]	Real-time intrusion detection using adaptive thresholds.	DBSCAN	DARPA, ISCX	DoS, DDoS, POD, SMURF, Mail-bomb, botnet, port scanning-port sweep	Precision: 98.12, ACC: 98.39, FPR: 3.61	Future work: Detecting complex attacks, distinguishing flash crowds and DDoS for better clarity.
Zhang et al. (2016) [42]	Utilizing One-class SVM for network intrusion identification.	One-class SVM	NSL-KDD	DoS, Probe, U2R, R2L	Precision: 99.3, Recall: 91.61, F-value: 95.18	Low DR for minority-class attacks like U2R and R2L.
Landress (2016) [61]	Employing K-means clustering, J48, decision tree, and SOM to reduce false positives.	K-means, J48, decision tree, self-organizing map	KDDCUP99	DoS, Probe, U2R, R2L	ACC: 98.92	Limitation: Increased computational complexity with SOM; explore efficient hybrid techniques for real-time processing.
He et al. (2017) [94]	Exploiting SDN for effective anomaly detection.	DBSCAN	KDDCUP99	DoS, Probe, U2R, R2L	ACC: 94.5	Future work: Focus on real-time packet clustering for timely detection.
Ariaifar and Kiani (2017) [95]	Optimizing clusters using GA, K-means, and decision trees.	K-means, decision tree, GA	NSL-KDD	DoS, Probe, U2R, R2L	DR: 99.1, FAR: 1.8	Evaluate on modern-day datasets like CIC-IDS 201 and CSE-CICIDS2018.
Bigdeli et al. (2018) [96]	Incremental cluster updates with spectral and density-based clustering.	Spectral and density-based clustering	KDDCUP99, NSL-KDD, Darpa98, IUSTsip, DataSetMe	DoS, Probe, U2R, R2L	DR: 94%, FAR: 4%	Introduce concept drift while merging clusters.
Almi' Ani et al. (2018) [97]	Hierarchical agglomerative clustering using k-means for reduced training time.	Hierarchical agglomerative clustering	NSL-KDD	DoS, Probe, U2R, R2L	DR: 96.66, ACC: 83.46, Precision: 75, FPR: 0.279, FNR: 0.033	Low accuracy due to low sensitivity toward normal behavior; investigate fuzzy C-means clustering for accuracy enhancement.

(Continued)

Table 3: Continued

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
Zhou et al. (2019) [98]	Hybrid technique using KPCA and ELM.	KPCA, ELM	KDDCUP99	DoS, Probe, U2R, R2L	DR: DoS: 98.96, Probe: 98.54, R2L: 94.72, U2R: 36.54, Acc: 98.18, FAR: 2.38	The DR of U2R is quite low. To optimize the results of ELM, evolutionary algorithms need to be applied.
Choi et al. (2019) [99]	Extracting key features using AE.	AE	NSL-KDD	DoS, Probe, U2R, R2L	ACC: 91.70	Improve U2R DR; apply evolutionary algorithms to optimize ELM results.
Aliakbarisani et al. (2019) [74]	Formulates a constraint trace ratio optimization problem for Laplacian Eigenmap strategy.	Constraint trace optimization, PCA, LDA	NSL-KDD, Kyoto 2006+	DoS, Probe, U2R, R2L	ACC: 97.84, F-score: 0.878, FPR: 0.001	Train and test with different datasets; explore ensemble strategies based on UL models for enhanced performance.
Paulauskas and Baskys (2019) [72]	Employing HBOS mechanism for identifying rare attacks like U2R and R2L.	HBOS	NSL-KDD	DoS, Probe, U2R, R2L	F-measure: 87	Investigate online learning metrics for newly discovered attacks.
Hwang et al. (2020) [100]	CNN and AE for extracting raw features from network traffic.	CNN, AE	USTC-TFC 2016, Mirai-RGU, Mirai-CCU	SYN flood, UDP flood, ACK flood, and HTTP flood, Mirai C&C	ACC: 99.77, Precision: 99.93, Recall: 99.17, F1 measure: 99.55, FNR: 0.02, FPR: 0.83	Improve performance rate of majority classes.
Zavrak and Iskefiyeli (2020) [101]	Deploys flow-based IDS with One-class SVM as anomaly detector.	AE, VAE, One-Class SVM	CIC-IDS2017	DoS slow loris, DoS slow HTTP-test, DoS Hulk, DoS golden eye, Heart-bleed	AUC: VAE: 75.96, AE: 73.98, One-Class SVM: 66.36	Consider flow-based attributes collected at specified time intervals to improve DR and reduce FAR.

(Continued)

Table 3: Continued

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
Truong-Huu et al. (2020) [79]	Uses GAN strategy to extract useful features and proposes a traffic aggregation technique.	GAN	UNSW-NB15, CIC-IDS2017	Backdoors, Exploits, Worms, Shellcode, generic, DoS slow loris, DoS slow HTTP-test, DoS Hulk, DoS golden eye, Heart- bleed	UNSW-NB15: Precision: 0.84, Recall: 0.85, F1-score: 0.85, AUPRC: 0.8831 CIC-IDS2017: Prec: 0.8260, Recall: 0.8268, F1-score: 0.8264, AUROC: 0.9529, AUPRC: 0.8271	Investigate multi-class classification approach for identifying different types of attacks.
Prasad et al. (2020) [56]	Proposes a novel cluster center initialization approach to overcome shortcomings of conventional clustering techniques.	K-means clustering, cluster center initialization	CIC-IDS2017	DoS slow loris, DoS slow HTTP-test, DoS Hulk, DoS golden eye, Heart-bleed	DR: 88, FR: 88.5, Precision: 88, F-measure: 0.531, ACC: 88.6	Address limitations like manual pre-processing and time/space complexity in MANET deployment.
Megantara and Ahmad (2021) [102]	Utilizes feature importance and data-reduction techniques to improve the prediction performance of NAD.	Decision tree, LOF	NSL-KDD, UNSW-NB15	DoS, Probe, U2R, R2L Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, worms	NSL-KDD: Acc: 99.73 UNSW-NB15: 91.86	The size of the LOF cluster can be optimized and the threshold value for handling outliers can be improved.
Liao et al. (2021) [103]	Presents an ensemble of UL schemes based on a novel weighting scheme.	AE, GANs	UNSW-NB15, CIC-IDS 2017	Backdoors, Exploits, Worms, Shellcode, generic, DoS slow loris, DoS slow HTTP-test, DoS Hulk, DoS golden eye, Heart- bleed	UNSW-NB15: Precision: 97.9, Recall: 92.4, F1-score: 94.9 CIC-IDS2017: Precision: 83.8, Recall: 84, F1-score: 83.5	Address suboptimal results for specific attack families, explore optimization for better performance.

(Continued)

Table 3: Continued

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
Verkerken et al. (2022) [71]	Proposes an inter-dataset evaluation approach for ensemble UL algorithms.	PCA, IF, AE, One-class SVM	CIC-IDS 2017, CSE-IDS 2017	DoS slow loris, DoS slow HTTP-test, DoS Hulk, DoS golden eye, Heartbleed	ACC: PCA: 90.9, IF: 91.1, AE: 99.9, One-class SVM: 98.9 AUROC: PCA: 93.73, IF: 95.8, AE: 97.75, One-class SVM: 94.20	Consider employing supervised learning approaches for generalization strength validation.
Singh and Jang-Jaccard (2022) [104]	Proposes a unified AE architecture based on CNN and LSTM to examine spatiotemporal correlations in traffic data.	MSCNN, LSTM-based AE	NSL-KDD, UNSW-NB15, CICDDoS2019	Normal, Attack (Binary class)	ACC: 97.10, Precision: 95.9, Recall: 96.4, F-score: 96.0 (average)	Fine-tune hyperparameters for enhanced model performance using optimization techniques.
Wang et al. (2022) [105]	Proposes an ensemble of UL algorithms to address challenges in processing overhead and poor detection performance for unseen threats.	AE, IF	CES-CIC-IDS2018, MQTT-IOT-IDS2020	Sparta SSH brute force, DoS attacks-SlowHTTPTest, DoS-Hulk, DDoS attack-LOIC-UDP, DDoS attacks-LOIC-HTTP, Brute force-XSS, Brute force-web, FTP-brute force	Average ACC: 96.43, Average Recall: 95.95, Average F-score: 96.02	Improve DR of specific attacks like “DoS-Hulk” in the proposed work.

(Continued)

Table 3: Continued

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
De C. Bertoli et al. (2022) [106]	Presents an NIDS for generalized detection performance in heterogeneous networks.	Deep AE, FL, Energy Flow Classifier	UNSW-NB15, CSE-CIC-IDS2018, BoT-IoT, ToN-IoT	Normal, Attack (Binary class)	BoT-IoT: ACC: 93, Recall: 93, Precision: 99, F-score: 95 ToN-IoT: ACC: 85, Recall: 85, Precision: 87, F-score: 77, UNSWNB15: ACC: 97, Recall: 99, Precision: 57, F-score: 73 CSE-CIC-IDS2018: ACC: 98, Recall: 88, Precision: 92, F-score: 90	Deploy the proposed approach in distributed NIDS for robustness evaluation.
Eren et al. (2023) [107]	Proposes a novel tensor decomposition method to enhance the detection of unseen attacks in network anomaly and intrusion detection, leveraging tensor factorization for improved generalization and identification of evolving threats.	Unsupervised DL, Tensor factorization algorithm	Neris-botnet, Spam e-mail	-	ROC-AUC, PR-AUC Average ROC-AUC: 0.9661, Average PR-AUC: 0.9152	Using a tensor decomposition algorithm with latent factors enables enhanced initialization of the tensor factorization algorithm for improved performance.

(Continued)

Table 3: Continued

Authors/year	Methodology	Algorithm/ techniques used	Datasets	Attack class	Metrics (%)	Limitations/ future work
Lan et al. (2023) [108]	Presents a novel framework for unsupervised intrusion detection that transfers knowledge from known attacks to detect new ones using a hierarchical attention-based triplet network and unsupervised domain adaptation.	Attention network, unsupervised domain adaption	ISCX-2012, UNSW-NB15, CIC-IDS2017, CTU-13	HTTP DoS, Brute Force SSH, PortScan, Brute Force, Dos slow loris, Web attack XSS, Neris, Rbot, Fuzzers, Generic, Shellcode, Exploits	Average ACC: 96.38, Average DR: 94.16	Utilize tensor decomposition algorithm with latent factors for enhanced initialization and performance.
Boppapa and Bagade (2023) [109]	Synergy of GANs and AE for unsupervised intrusion detection in MQTT networks.	AE, GAN	MQTT-IoT-IDS2020	Normal, Attack (Binary class)	F1-score: 97	Evaluate generalizability to diverse network architectures in future work.

ACC, Accuracy; AE, autoencoders; AUC, area under curve; CNN, convolutional neural network; DBSCAN, density-based spatial clustering of applications with noise; DdoS, distributed denial of service; DL, deep learning; DoS, denial of service; DR, detection rate; ELM, extreme learning machine; FAR, false alarm rate; FL, federated learning; FNR, false negative rate; FPR, false positive rate; GA, genetic algorithm; GAN, generative adversarial network; HBOS, histogram-based outlier score; IDS, intrusion detection systems; IF, isolation forest; IoT, internet of things; KPCA, kernel principal component analysis; LOF, local outlier factor; LSTM, long short-term memory; MSCNN, multi-scale convolution neural network; NAD, network anomaly detection; NIDS, network intrusion detection system; PCA, principal component analysis; PR-AUC, precision-recall curve; ROC, receiver operating characteristic; ROC-AUC, ROC curve; SOM, self-organizing map; SVM, support vector machine; TPR, True positive rate; UL, unsupervised learning; VAE, variational autoencoders.

Amidst these considerations, the selection of the CIC-IDS-2017 dataset for evaluating the UL algorithms becomes apparent. Despite KDD99 and NSL-KDD being commonly used, their outdated content raises concerns. NSL-KDD, while addressing some KDD99 issues, may represent something other than real networks perfectly. Nevertheless, its reasonable number of records facilitates comprehensive experiments. Moreover, UNSW-NB15 is popular and widely employed due to its inclusion of diverse novel attacks. In this paper, we selected NSL-KDD, UNSW-NB15, and CIC-IDS2017 to evaluate UL algorithms based on their popularity, extensive usage, and consistent results. Table 4 provides a comparative analysis of the selected three benchmark datasets with various other datasets.

The following considerations underpin the dataset-selection process:

NSL-KDD:

- Upgraded variant of KDD99.
- Designed to reduce classifier bias.
- Unbiased classification and consistent results.
- No duplicate records, ensuring balance across attack groups.

UNSW-NB15:

- Developed in response to evolving network attack behavior.
- Contemporary substitute for NSL-KDD.
- Proximity in generation processes and functionality.
- Addresses shortcomings of KDD-99, enhancing applicability in modern NIDS.

CIC-IDS2017:

- Meticulously crafted with a focus on essential dataset characteristics.
- Well-ordered format using profiles.
- In-depth understanding of attacks, application models, network devices, and protocols.
- Alignment with critical features, making it a comprehensive resource for evaluation.

Table 4 presents a comprehensive overview of various datasets used for intrusion detection, delineating crucial aspects such as data source, year, emulation status, benchmark or real-life nature, dataset size, attack classes covered, and associated publications. NSL-KDD and UNSW-NB15 datasets, both emulated benchmarks, serve as valuable resources for assessing IDS. NSL-KDD, established in 2009, comprises over 148,000 instances and covers a spectrum of attack classes, including DoS, Probe, U2R, and R2L. Meanwhile, UNSW-NB15, introduced in 2015,

boasts a larger dataset size exceeding 2.5 million instances and encompasses a broader array of attack types, rendering it suitable for evaluating modern intrusion detection techniques.

However, among these datasets, CIC-IDS 2017 emerges as a standout choice. Despite being emulated, it offers an extensive and realistic portrayal of contemporary cybersecurity threats. With over 2.8 million instances, it spans various sophisticated attacks, including DDoS, Dos, Botnet, BruteForce, Infiltration, Web Attack, and Port Scan. This diversity is crucial for evaluating the efficacy of IDS in real-world scenarios. Contrasting with older datasets like DARPA (1998) and KDD CUP 99, it is evident that the field has advanced significantly. While DARPA and KDD datasets were pioneering benchmarks, they need more scalability, diversity, and realism offered by more recent datasets like CIC-IDS 2017. In contrast with datasets like ISCX 2012 and CAIDA, CIC-IDS 2017 provides more extensive coverage of attack types and larger dataset sizes, making it better suited for evaluating the robustness and generalizability of IDS.

The newer datasets, such as CSE-CICIDS2018, maintain relevance in contemporary intrusion detection research. Notably, CSE-CICIDS2018 boasts a larger dataset size than its predecessors, presenting unique challenges in data processing owing to the sheer volume of data instances in each file. On the other hand, recent datasets like CICDoS2019, BoT-IoT, and IoT-23 focus on specific types of attacks prevalent in modern cyber threats, such as DDOS and IoT-based attacks. Contrastingly, CIC-IDS2017 offers a broader spectrum of attack scenarios, covering a wide range of intrusion types beyond just DDOS and IoT attacks. CIC-IDS 2017 stands out for its comprehensive coverage, encompassing a multitude of attack scenarios ranging from network reconnaissance to web-based attacks.

b. Selection of metrics

Performance metrics are used to compare the effectiveness of NAD systems and to evaluate the experimental findings of various approaches. Performance metrics are obtained from the confusion matrix, which aims to categorize normal and abnormal data. The terms TP (true positive) and TN (true negative) refer to accurately predicted conditions, whereas FP (false positive) and FN (false negative) refer to incorrectly classified conditions. TPs and TNs relate to correctly categorized attacks and normal records, while FPs and FNs refer to incorrectly classified attacks and

Table 4: Comparative analysis of network datasets.

Dataset	Data-source	Year	1	2	3	4	5	6	7
DARPA (1998)	MIT Lincoln Laboratory	1998	Emulated	Benchmark	✓	7,000,000	✓	DoS, Probe, U2R, R2L	41
KDD CUP 99	University of California	1999	Emulated	Benchmark	✓	4,900,000	✓	DoS, Probe, U2R, R2L	41
DEFCON	N/A	2000	Real	Benchmark	N/A	N/A	✓	N/A	Flag traces
LBNL	Lawrence Berkeley National Laboratory	2004	N/A	Benchmark	✓	>100 hr	×	Malicious traces	Internet traces
Kyoto	Song et al. (2006) [114]	2006	Real	Benchmark	✓	N/A	✓	Normal and attack sessions	24
CAIDA	Jonker et al. (2017) [118]	2008–2017	Real	Benchmark	✓	Huge	×	DDoS	20
CDX	United States Military Academy	2009	Real	Real-life	N/A	5771	✓	Buffer overflow	5
NSL-KDD	Tavallaee et al. (2009) [126]	2009	Emulated	Benchmark	✓	148,517	✓	DoS, Probe, U2R, R2L	41
ISCX 2012	Shiravi et al. (2012) [117]	2012	Real	Real-life	✓	2,450,324	✓	DoS, DDoS, Brute force, Infiltration	IP flows
UNSW-NB15	Moustafa and Slay (2015) [119]	2015	Emulated	Benchmark	✓	2,540,044	✓	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms	49
CIDDs-001	Ring et al. (2017) [120]	2017	Emulated	Benchmark	✓	31,959,267	✓	Ping scanning, Port scanning, Brute force, and DoS	14
CIDDs-002	Ring et al. (2017) [120]	2017	Emulated	Benchmark	✓	16,161,183	✓	Ping scanning, Port scanning, Brute force, and DoS	14
CIC-IDS2017	Sharafaldin et al. (2018) [121]	2017	Emulated	Benchmark	✓	2,830,743	✓	DDoS, Dos, Botnet, BruteForce, Infiltration, WebAttack, Port scan	80

(Continued)

Table 4: Continued

Dataset	Data-source	Year	1	2	3	4	5	6	7
CSE-CICIDS2018	Bharati and Tamane (2020) [122]	2018	Emulated	Benchmark	✓	16,232,943	✓	DdoS, Dos Botnet, BruteForce, Infiltration, WebAttack, Port scan	80
CICDDoS 2019	Sharafaldin et al. (2019) [123]	2019	Emulated	Benchmark	✓	32,925	✓	DdoS_DNS, DdoS_LDAP, DdoS_MSSQL, DdoS_NetBIOS, DdoS_NTP, DdoS_SNMP, DdoS_SSDP, DdoS_SYN, DdoS_TFTP, DdoS_UDP, DdoS_UDP-Lag, DdoS_WebDDoS	76
BoT-IoT	Koroniotis et al. (2019) [124]	2019	Real	Real-life	✓	73,360,900	✓	DoS, DdoS, Reconnaissance, Theft	29
IoT-23	N/A	2020	Real	Real-life	✓	N/A	✓	Mirai, Torii, Hide and Seek, Muhstik, Hakai, IRCBot, Hajime, Trojan, Kenijiro, Okiru, Gagfyt	21

DdoS, distributed denial of service; DoS, denial of service; IDS, intrusion detection systems; IoT, internet of things; IRCBot, internet relay chat botnet.

Note: 1: Authenticity; 2: Type; 3: Publicly available; 4: No. of instances; 5: Labeled; 6: Attack type; 7: Features.

normal records, respectively [127–129]. The following metrics are selected based on those frequently applied in survey studies.

1. **Accuracy:** It is defined as the ratio of correctly classified instances (TP + TN) to the total number of instances present in the dataset.

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

2. **Precision:** It is defined as the ratio of correctly identified instances (TP) divided by the total number of instances which are classified as correct.

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

3. **Recall:** It is calculated as the ratio of correctly identified positive samples (TP) to the total number of positive samples.

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

4. **F1-score:** It is the harmonic mean of the model's precision and recall, providing information on both aspects of its performance. Thus, a high F1 score shows a high recall and precision value.

$$F - score = 2 \times \left(\frac{Precision \times Recall}{Precision + Recall} \right)$$

c. Results and discussion

The experiments were conducted using the hardware and software specifications given in Table 5. Three datasets are processed by transforming nominal variables to numerical, when necessary, to improve the number of usable features without altering the dataset's semantics. Parameter tuning is utilized to determine the optimal configuration for an algorithm. The tuning process is accomplished by carrying out several different parameter combinations. The results of different parameter combinations are compared to one another to select the best possible combination.

This study extensively evaluates 13 UL methods on three benchmark intrusion detection datasets: NSL-KDD, UNSW-NB15, and CIC-IDS2017. The goal

Table 5: Hardware and software specifications.

Components	Specifications
Operating system	Windows 11
System type	64-bit operating system, x 64-based Processor
Processor	11 th Gen Intel ® Core (TM) i5-1135G7 @ 2.40 GHz 2.42 GHz
RAM	8 GB
Python	3.7
Datasets	NSL-KDD, UNSW-NB15, CICIDS2017

is to assess the performance of these algorithms and establish their effectiveness compared to existing research. These UL algorithms have been implemented for AD, as referenced in [130–134], and tested on both traditional and modern datasets to ensure a thorough analysis. Additionally, unsupervised dimensionality-reduction techniques from the six families of AD techniques are employed to tackle the challenge of high-dimensional real-world data. This allows the representation of data in reduced dimensions.

A decision tree classifier is utilized for result prediction to gauge the performance of dimensionality reduction approaches. For detailed insights into the results, Tables 6–8 present the performance of the 13 selected UL methods for NAD on the three benchmark datasets. These findings provide valuable evidence for the efficacy of these algorithms in AD.

The performance results presented in Tables 6–8 highlight the notable performance of the selected UL methods, particularly on the NSL-KDD dataset. The lack of redundant connections and even data distribution across attack categories and normal connections in the NSL-KDD dataset contribute significantly to these results. As shown in Figures 6–8, the study indicates that ODIN outperforms KNN in detecting anomalies by utilizing an “indegree score” based on the KNN graph. DBSCAN was more effective than K-means and EM in clustering-based techniques. The study observed that nearest-neighbor-based approaches usually outperformed clustering methods, but clustering-based approaches were more computationally efficient for processing large datasets in near real-time environments.

In density-based approaches, LOF surpasses COF, especially in scenarios requiring the detection of local anomalies. However, COF was recommended

Table 6: Results of 13 selected UL methods for NAD on the NSL-KDD dataset.

Algorithms	Family	Accuracy	Precision	Recall	F1-score
K-NN	Neighbor-based	98.4	98.0	97.9	98.5
ODIN	Neighbor-based	99.4	99.0	98.6	98.8
LOF	Density-based	97.0	96.8	95.3	95.6
COF	Density-based	94.0	94.5	96.1	95.1
K-means	Clustering-based	94	92.46	92.3	93
DBSCAN	Clustering-based	94.5	94.0	92.9	94.0
EM	Clustering-based	95.0	93	92.9	94.0
PCA	Dimensionality-reduction	97.8	97.0	96.7	97.0
KPCA	Dimensionality-reduction	99.2	99.0	98.9	98.6
ICA	Dimensionality-reduction	98.0	97.5	97.8	98.1
HBOS	Statistical-based	94.4	93.9	95.8	96.0
One-class SVM	Classification-based	99.4	99.2	99.2	99.5
IF	Classification-based	99.9	99.8	99.6	99.7

COF, connectivity-based outlier factor; DBSCAN, density-based spatial clustering of applications with noise; EM, expectation maximization; HBOS, histogram-based outlier score; ICA, independent component analysis; IF, isolation forest; KNN, K-nearest neighbor; KPCA, kernel principal component analysis; LOF, local outlier factor; NAD, network anomaly detection; ODIN, outlier detection using indegree number; PCA, principal component analysis; SVM, support vector machine; UL, unsupervised learning.

Table 7: Results of 13 selected UL methods for NAD on the UNSW-NB15 dataset.

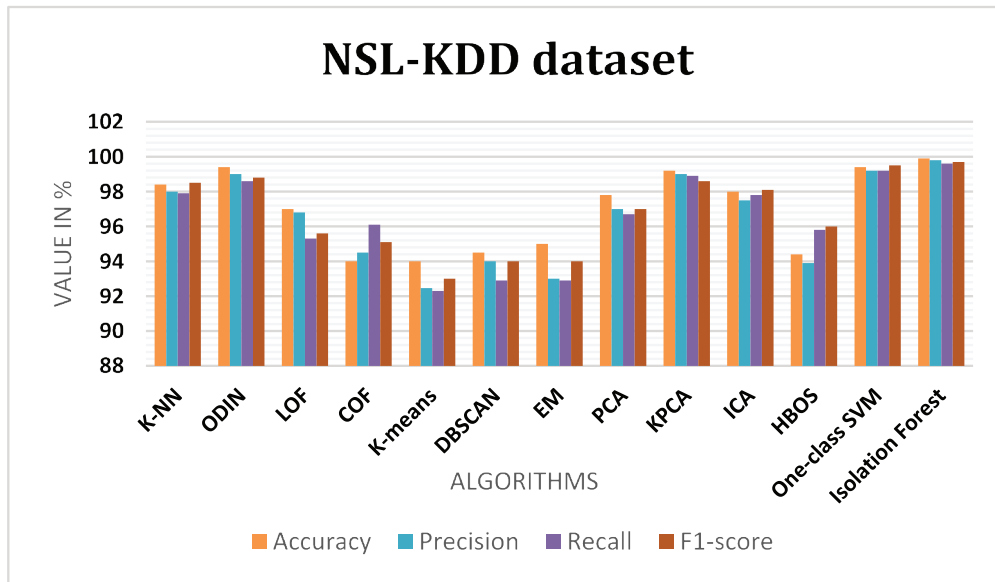
Algorithms	Family	Accuracy	Precision	Recall	F1-score
K-NN	Distance-based	98.2	97.8	97.6	97.7
ODIN	Distance-based	99.2	98.9	98.5	98.6
LOF	Density-based	96.9	95.8	95.1	95.0
COF	Density-based	93.8	94.2	95.9	94.7
K-means	Clustering-based	93.7	92.0	92.1	92.9
DBSCAN	Clustering-based	94.2	93.8	92.7	93.5
EM	Clustering-based	95.5	92.6	92.3	93.4
PCA	Dimensionality-reduction	97.2	96.9	97.3	97.1
KPCA	Dimensionality-reduction	99.1	98.5	98.5	98.7
ICA	Dimensionality-reduction	97.7	97.4	97.1	97.3
HBOS	Statistical-based	92.3	91.9	94.6	94.8
One-class SVM	Classification-based	99.2	99.0	99.2	99.4
IF	Classification-based	99.7	99.5	99.3	99.6

COF, connectivity-based outlier factor; DBSCAN, density-based spatial clustering of applications with noise; EM, expectation maximization; HBOS, histogram-based outlier score; ICA, independent component analysis; IF, isolation forest; KNN, K-nearest neighbor; KPCA, kernel principal component analysis; LOF, local outlier factor; NAD, network anomaly detection; ODIN, outlier detection using indegree number; PCA, principal component analysis; SVM, support vector machine; UL, unsupervised learning.

Table 8: Results of 13 selected UL methods for NAD on the CIC-IDS2017 dataset.

Algorithms	Family	Accuracy	Precision	Recall	F1-score
K-NN	Distance-based	96.4	96.7	96.0	97.0
ODIN	Distance-based	97.6	97.4	97.5	98.0
LOF	Density-based	96	95.3	93.6	94.8
COF	Density-based	93.3	93.2	95	93.7
K-means	Clustering-based	90.2	87.4	88.6	89.5
DBSCAN	Clustering-based	93.7	92.7	90.6	92.3
EM	Clustering-based	94.3	93.2	91.2	92.4
PCA	Dimensionality-reduction	96.3	95.4	94.9	95.1
KPCA	Dimensionality-reduction	97.3	97.2	96.9	97
ICA	Dimensionality-reduction	96.3	96.2	96	96.3
HBOS	Statistical-based	95.9	94.9	93.5	93.0
One-class SVM	Classification-based	98.3	98.1	99	98.4
IF	Classification-based	99.6	99.5	99.2	99.4

COF, connectivity-based outlier factor; DBSCAN, density-based spatial clustering of applications with noise; EM, expectation maximization; HBOS, histogram-based outlier score; ICA, independent component analysis; IDS, intrusion detection systems; IF, isolation forest; KNN, K-nearest neighbor; KPCA, kernel principal component analysis; LOF, local outlier factor; NAD, network anomaly detection; ODIN, outlier detection using indegree number; PCA, principal component analysis; SVM, support vector machine; UL, unsupervised learning.

**Figure 6: Comparison of UL methods on the NSL-KDD dataset. UL, unsupervised learning.**

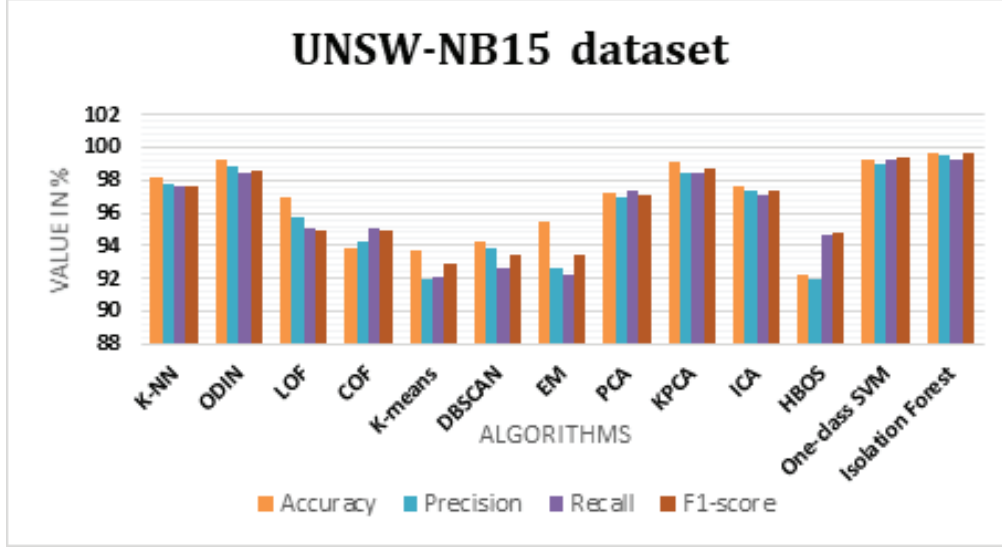


Figure 7: Comparison of UL methods on the UNSW-NB15 dataset. UL, unsupervised learning.

when additional information about the nature of anomalies in the evaluated dataset was unavailable. Dimensionality-reduction techniques, particularly KPCA and ICA, outperformed PCA across all datasets, with KPCA yielding the best results. This superiority can be attributed to the ability of KPCA and ICA to capture higher-order information in the original inputs, unlike PCA. HBOS, a statistical method assuming feature independence, demonstrates prominent results and is well suited for application in large datasets due to its fast computation time. It serves as a valuable option for AD in such scenarios.

The study also observed that algorithms belonging to the classification family consistently outperformed algorithms from other families on all datasets. IF stood out as the top-performing algorithm, achieving impressive average scores of 99.9, 99.7, and 99.6 on the NSL-KDD, UNSW-NB15, and CIC-IDS2017 datasets. The findings of this study have practical implications for real-world NAD systems. IF is recommended for critical infrastructure networks, while ODIN is suitable for healthcare systems or financial institutions where minimizing false negatives is critical. Clustering-based methods such as DBSCAN and density-based techniques like LOF can be employed in network monitoring systems to identify unusual patterns or behaviors that may indicate security breaches or system malfunctions. Finally, dimensionality-reduction techniques like KPCA and ICA can reduce computational overhead and facilitate real-time anomaly detection in dynamic network environments.

By understanding the practical applications of these UL methods, researchers can tailor their selection and implementation strategies to address specific security challenges and operational requirements. The study's findings reveal the superior performance of ODIN, DBSCAN, LOF, KPCA, ICA, and IF. They can be employed to develop more effective NAD systems for various industries, preventing potential cyber threats, enhancing network security, and improving operational efficiency.

V. Open Challenges and Future Directions

This section discusses various unexplored research areas and significant prospects for using UL in network intrusion/anomaly detection.

1. *Improving SOM algorithm:* The SOM algorithm faces challenges related to high time complexity, prompting an amalgamation of methods like k-means [135] to enhance computational efficiency. However, various approaches can still be explored to optimize performance in terms of speed and precision further [136, 137]. Beyond the overarching concerns of time complexity, the algorithm encounters heightened FPRs when confronted with symmetric data. This noteworthy observation prompts a comprehensive exploration of potential solutions. A forward-looking strategy

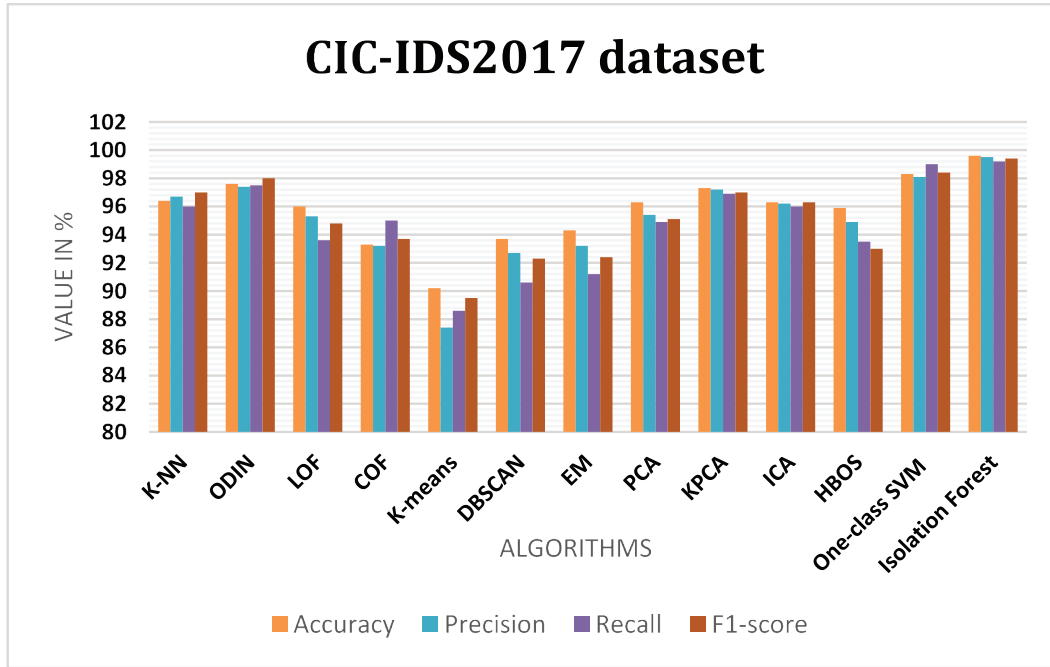


Figure 8: Comparison of UL methods on the CIC-IDS2017 dataset. IDS, intrusion detection systems; UL, unsupervised learning.

involves the integration of hybrid technologies, notably those enriched with topographic measures [138]. These measures are envisioned to instill refined growing rules, laying the groundwork for more nuanced and accurate processing, especially in symmetric datasets [139].

2. *Inter-dataset evaluation*: Ensemble techniques that use UL methods are widely used in NAD to improve system performance. However, it is a challenge to assess the ability of NAD models to perform well across different datasets when relying only on individual assessments. Therefore, it is essential to explore inter-dataset evaluation strategies to determine the true generalization ability of unsupervised methods and enhance adaptability for future developments. To achieve this, a thorough investigation is required to align methodologies with the ever-changing landscape of diverse datasets and ensure the development of resilient and adaptable NAD systems.
3. *Explanation facility*: DL models are great at identifying network anomalies, but they lack the ability to provide clear explanations to end-users. Although attentional models offer insights, there is room for improvement in result comprehension and explanatory clarity [80]. To address this, forthcoming research should delve into unexplored territories by investigating various attentional

components. Moreover, adopting the backward pass-through-the-network approach, well-known in image processing [140, 141], could effectively scrutinize backward techniques and reveal the complex relationship between input features and the final reconstruction values. This exploration has the potential to yield greatly improved results, providing more coherent and meaningful explanations.

4. *Self-taught learning (STL)*: Applying STL with UL is a potent strategy for enhancing intrusion and anomaly detection in network security. This method strategically addresses the challenge of limited labeled data by leveraging STL's adaptive learning capabilities. STL autonomously discerns intricate patterns and anomalies within the network, fostering real-time anomaly recognition and prompt threat response. The model's dynamic evolution ensures continuous adaptation to the evolving threat landscape, enhancing its effectiveness. The fusion of STL and UL is a promising framework for advancing network security and provides a learning-centric approach with significant potential for future research. This approach can strengthen network defense mechanisms against evolving threats.
5. *Interpretability problems with some unsupervised ML algorithms*: Addressing interpretability challenges in unsupervised ML algorithms, particularly

deep neural networks, is a pivotal research focus. While contributing to high accuracy, the inherent complexity of these algorithms often hinders their interpretability, posing challenges for comprehension and decision-making in operational networks [75]. To tackle this issue, ongoing efforts by researchers aim to enhance the transparency and explainability of neural network models [142]. Future research in this domain should delve into developing methodologies and frameworks that augment the interpretability of artificial intelligence and ML models, ensuring their effective integration into operational networks [141].

6. *Lack of functional success in intrusion detection*: Exploring the practical application of UL in network intrusion detection reveals noteworthy challenges hindering its functional success in operational contexts. Despite substantial academic research and practical implementations in diverse fields, adopting ML solutions, specifically UL, for network intrusion detection still needs to be improved in real-world scenarios [143]. This constraint is attributed to various factors: (a) insufficient training data availability, (b) a semantic gap between the obtained results and their operational interpretation, (c) the substantial costs associated with errors in the detection process, and (d) inherent challenges in conducting accurate performance evaluations. Addressing these challenges is crucial for realizing the full potential of UL in network security and establishing its efficacy in operational environments.

VI. Conclusion

This paper provides an in-depth analysis of UL anomaly detection (AD) techniques, highlighting recent developments while identifying key research limitations. The study demonstrates the effectiveness of these approaches in real-world network security scenarios by evaluating 13 UL methods across benchmark intrusion detection datasets. IFs, ODIN, DBSCAN, LOF, KPCA and ICA are some of the methods evaluated, which have versatile applications in anomaly detection in various sectors such as critical infrastructure networks, healthcare, finance, and network monitoring.

The paper suggests several recommendations to guide future research endeavors for advancements of UL methods. One of the recommendations is to address the interpretability challenges inherent in DL algorithms, particularly in the context of NAD, by integrating explainable AI techniques such as attention

mechanisms or back-propagation methods into UL models. This integration would enhance transparency and facilitate informed decision-making in security operations. Another recommendation is to explore ensemble techniques for inter-dataset evaluation, which hold promise for assessing UL methods' generalization capabilities across diverse datasets. In addition, the paper suggests the fusion of self-taught learning with UL as a compelling avenue for enhancing anomaly detection capabilities in network security. Leveraging STL's adaptive learning mechanisms can augment UL models' ability to discern complex patterns and anomalies, bolstering real-time threat detection and response capabilities.

The paper suggests strategies like data augmentation, transfer learning, and feature engineering to address practical constraints hindering UL's functional success, such as data scarcity and semantic gaps between model outputs and operational interpretations. Furthermore, the integration of model explainability techniques such as feature importance analysis, SHAP, LIME, and Integrated Gradients, alongside human-in-the-loop approaches involving expert feedback and domain knowledge integration, can enhance interpretability and align UL models more closely with operational requirements, thereby fostering the development of more robust and actionable anomaly detection solutions. By heeding these recommendations, researchers can navigate today's challenges more effectively and pave the way for advancements in UL-based NAD.

Data Availability

The research has been conducted only with openly available datasets and will be publicly available in (<https://www.unb.ca/cic/datasets/nsl.html>, <https://research.unsw.edu.au/projects/unsw-nb15-dataset>, <https://www.unb.ca/cic/datasets/ids-2017.html>).

Author Contributions

Conceptualization: Niharika Sharma (N.S.), Bhavna Arora (B.A.), Pradeep Kumar Singh (P.K.S.), Surbhi Bhatia Khan (S.B.K.), Yashwant Singh (Y.S.); formal analysis: Niharika Sharma (N.S.), Bhavna Arora (B.A.); investigation & project administration: Niharika Sharma (N.S.), Bhavna Arora (B.A.), Pradeep Kumar Singh (P.K.S.), Surbhi Bhatia Khan (S.B.K.), Yashwant Singh (Y.S.); resources: Niharika Sharma (N.S.), Bhavna Arora (B.A.), Pradeep Kumar Singh (P.K.S.), Surbhi Bhatia Khan (S.B.K.), Yashwant Singh (Y.S.); supervision: Bhavna Arora (B.A.), Pradeep Kumar

Singh (P.K.S.), Yashwant Singh (Y.S.), Shabana Ziyad; visualization: Bhavna Arora (B.A.), Pradeep Kumar Singh (P.K.S.), Surbhi Bhatia Khan (S.B.K.), Yashwant Singh (Y.S.); writing: Niharika Sharma (N.S.), writing—review and editing: Bhavna Arora (B.A.), Pradeep Kumar Singh (P.K.S.), Surbhi Bhatia Khan (S.B.K.), Shabana Ziyad, Yashwant Singh (Y.S.). All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This publication was supported by the Deanship of Scientific Research at Prince Sattam bin Abdulaziz University, Al Kharj, Saudi Arabia.

References

- [1] M. Goldstein and S. Uchida, “A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data,” *PLoS One*, vol. 11, no. 4, p. e0152173, Apr. 2016, doi: 10.1371/JOURNAL.PONE.0152173.
- [2] V. Garcia-Font, C. Garrigues, and H. Rifà-Pous, “A Comparative Study of Anomaly Detection Techniques for Smart City Wireless Sensor Networks,” *Sensors (Basel)*, vol. 16, no. 6, Jun. 2016, doi: 10.3390/S16060868.
- [3] A. Nisioti, A. Mylonas, P. D. Yoo, and V. Katos, “From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods,” *IEEE Commun. Surv. Tutorials*, vol. 20, no. 4, pp. 3369–3388, Oct. 2018, doi: 10.1109/COMST.2018.2854724.
- [4] F. Falcão et al., “Quantitative comparison of unsupervised anomaly detection algorithms for intrusion detection,” *Proc. ACM Symp. Appl. Comput.*, vol. Part F147772, pp. 318–327, 2019, doi: 10.1145/3297280.3297314.
- [5] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, “Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review,” *Procedia Comput. Sci.*, vol. 171, pp. 1251–1260, Jan. 2020, doi: 10.1016/J.PROCS.2020.04.133.
- [6] A. Thakkar and R. Lohiya, “A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions,” *Artif. Intell. Rev.*, vol. 55, no. 1, pp. 453–563, Jan. 2022, doi: 10.1007/S10462-021-10037-9/FIGURES/15.
- [7] D. Samariya and A. Thakkar, “A Comprehensive Survey of Anomaly Detection Algorithms,” *Ann. Data Sci.*, vol. 10, no. 3, pp. 829–850, Jun. 2023, doi: 10.1007/S40745-021-00362-9.
- [8] A. E. Ezugwu et al., “A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects,” *Eng. Appl. Artif. Intell.*, vol. 110, p. 104743, Apr. 2022, doi: 10.1016/J.ENGAPAI.2022.104743.
- [9] U. A. Usmani, A. Happonen, and J. Watada, “A Review of Unsupervised Machine Learning Frameworks for Anomaly Detection in Industrial Applications,” *Lect. Notes Networks Syst.*, vol. 507 LNNS, pp. 158–189, 2022, doi: 10.1007/978-3-031-10464-0_11/COVER.
- [10] X. Tao, X. Gong, X. Zhang, S. Yan, and C. Adak, “Deep Learning for Unsupervised Anomaly Localization in Industrial Images: A Survey,” *IEEE Trans. Instrum. Meas.*, vol. 71, 2022, doi: 10.1109/TIM.2022.3196436.
- [11] M. Q. Ma, Y. Zhao, X. Zhang, and L. Akoglu, “The Need for Unsupervised Outlier Model Selection: A Review and Evaluation of Internal Evaluation Strategies,” Accessed: Jun. 04, 2023. [Online]. Available: <https://github.com/yzhao062/uoms>.
- [12] K. DeMedeiros, A. Hendawi, and M. Alvarez, “A Survey of AI-Based Anomaly Detection in IoT and Sensor Networks,” *Sensors 2023, Vol. 23, Page 1352*, vol. 23, no. 3, p. 1352, Jan. 2023, doi: 10.3390/S23031352.
- [13] K. Taha, “Semi-supervised and un-supervised clustering: A review and experimental evaluation,” *Inf. Syst.*, vol. 114, p. 102178, Mar. 2023, doi: 10.1016/J.IS.2023.102178.
- [14] M. Landauer, S. Onder, F. Skopik, and M. Wurzenberger, “Deep learning for anomaly detection in log data: A survey,” *Mach. Learn. with Appl.*, vol. 12, p. 100470, Jun. 2023, doi: 10.1016/J.MLWA.2023.100470.
- [15] E. Eskin, “Anomaly Detection over Noisy Data Using Learned Probability Distributions,” 2000, doi: 10.7916/D8C53SKF.
- [16] A. Lazarevic, L. Ertöz, V. Kumar, A. Ozgur, and J. Srivastava, “A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection †,” Accessed: Oct. 15, 2022. [Online]. Available: <https://epubs.siam.org/terms-privacy>.
- [17] M. G. H. Omran, A. P. Engelbrecht, and A. Salman, “An overview of clustering methods,” *Intell. Data Anal.*, vol. 11, pp. 583–605, 2007.
- [18] K. L. Ingham and H. Inoue, “Comparing anomaly detection techniques for HTTP,” *Lect. Notes Comput.*

Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 4637 LNCS, pp. 42–62, 2007, doi: 10.1007/978-3-540-74320-0_3.

[19] C. F. Tsai, Y. F. Hsu, C. Y. Lin, and W. Y. Lin, "Intrusion detection by machine learning: A review," *Expert Syst. Appl.*, vol. 36, no. 10, pp. 11994–12000, Dec. 2009, doi: 10.1016/J.ESWA.2009.05.029.

[20] B. G. O. Reddy, B. G. O. Reddy, and D. M. Ussenaiah, "Literature Survey On Clustering Techniques," Accessed: Jun. 07, 2022. [Online]. Available: <https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.467.9888>.

[21] X. Ding, Y. Li, A. Belatreche, and L. P. Maguire, "An experimental evaluation of novelty detection methods," *Neurocomputing*, vol. 135, pp. 313–327, Jul. 2014, doi: 10.1016/J.NEUCOM.2013.12.002.

[22] D. Xu and Y. Tian, "A Comprehensive Survey of Clustering Algorithms," *Ann. Data Sci.* 2015 22, vol. 2, no. 2, pp. 165–193, Aug. 2015, doi: 10.1007/S40745-015-0040-1.

[23] M. Usama *et al.*, "Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges," *IEEE Access*, vol. 7, pp. 65579–65615, 2019, doi: 10.1109/ACCESS.2019.2916648.

[24] S. Ramaswamy, R. Rastogi, and K. Shim KAIST, "Efficient Algorithms for Mining Outliers from Large Data Sets," 2000, Accessed: Oct. 03, 2022. [Online]. Available: www.bell-labs.com/projects/serendip.

[25] V. Hautamäki, I. Kärkkäinen, and P. Fränti, "Outlier detection using k-nearest neighbour graph," *Proc. - Int. Conf. Pattern Recognit.*, vol. 3, pp. 430–433, 2004, doi: 10.1109/ICPR.2004.1334558.

[26] T. Zoppi, A. ceccarelli, T. Capecchi, and A. Bondavalli, "Unsupervised Anomaly Detectors to Detect Intrusions in the Current Threat Landscape," *ACM/IMS Trans. Data Sci.*, vol. 2, no. 2, pp. 1–26, Dec. 2020, doi: 10.48550/arxiv.2012.11354.

[27] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying Density-Based Local Outliers," *SIGMOD 2000 - Proc. 2000 ACM SIGMOD Int. Conf. Manag. Data*, pp. 93–104, 2000, doi: 10.1145/342009.335388.

[28] "Algorithm selection for Anomaly Detection | by Sahil Garg | Analytics Vidhya | Medium." <https://medium.com/analytics-vidhya/algorithm-selection-for-anomaly-detection-ef193fd0d6d1>

[29] J. Tang, Z. Chen, A. W. C. Fu, and D. W. Cheung, "Enhancing effectiveness of Outlier detections for low Density Patterns," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2336, pp. 535–548, 2002, doi: 10.1007/3-540-47887-6_53/COVER.

[30] E. Schubert, A. Koos, T. Emrich, A. Züfle, K. A. Schmid, and A. Zimek, "A framework for clustering uncertain data," *Proc. VLDB Endow.*, vol. 8, no. 12, pp. 1976–1979, Aug. 2015, doi: 10.14778/2824032.2824115.

[31] C. Graziano Gentili and T. Zoppi Tutor Co-Tutor Andrea Bondavalli Dott Andrea Ceccarelli Coordinatori Graziano Gentili Cristina Pinotti, "Towards Effective Anomaly Detection in Complex Dynamic Systems," 2019, Accessed: Nov. 02, 2022. [Online]. Available: <https://flore.unifi.it/handle/2158/1119284>.

[32] Y. G. Jung, M. S. Kang, and J. Heo, "Clustering performance comparison using K-means and expectation maximization algorithms," *Biotechnol. Biotechnol. Equip.*, vol. 28, pp. S44–S48, 2014, doi: 10.1080/13102818.2014.949045.

[33] M. Çelik, F. Dadaşer-Çelik, and A. Ş. Dokuz, "Anomaly detection in temperature data using DBSCAN algorithm," *INISTA 2011 - 2011 Int. Symp. Innov. Intell. Syst. Appl.*, pp. 91–95, 2011, doi: 10.1109/INISTA.2011.5946052.

[34] M. C. R. Murça, R. John Hansman, H. Balakrishnan, R. Delaura, R. Jordan, and T. Reynolds, "Trajectory clustering and classification for characterization of air traffic flows," *16th AIAA Aviat. Technol. Integr. Oper. Conf.*, 2016, doi: 10.2514/6.2016-3760.

[35] J. Li and D. G. Mani, "Machine Learning Application on Prediction of Male Breast Cancer with PLCO Dataset," *J. Student Res.*, vol. 10, no. 3, Nov. 2021, doi: 10.47611/jsrhs.v10i3.2199.

[36] R. Abdulhammed, M. Faezipour, H. Musafer, and A. Abuzneid, "Efficient network intrusion detection using PCA-based dimensionality reduction of features," *2019 Int. Symp. Networks, Comput. Commun. ISNCC 2019*, Jun. 2019, doi: 10.1109/ISNCC.2019.8909140.

[37] G. Sidhu, N. Asgarian, R. Greiner, and M. R. G. Brown, "Kernel principal component analysis for dimensionality reduction in fMRI-based diagnosis of ADHD," *Front. Syst. Neurosci.*, vol. 0, no. OCTOBER 2012, pp. 1–17, Oct. 2012, doi: 10.3389/FNSYS.2012.00074/BIBTEX.

[38] L. J. Cao, K. S. Chua, W. K. Chong, H. P. Lee, and Q. M. Gu, "A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine," *Neurocomputing*, vol. 55, no. 1–2, pp. 321–336, Sep. 2003, doi: 10.1016/S0925-2312(03)00433-8.

[39] C. C. Aggarwal, "Data Mining," 2015, doi: 10.1007/978-3-319-14142-8.

[40] M. Goldstein and A. Dengel, "Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm," Accessed: Oct. 05, 2022. [Online]. Available: <http://madm.dfki.de/rapidminer/anomalydetection>.

- [41] T. Zoppi, A. Ceccarelli, L. Salani, and A. Bondavalli, "On the educated selection of unsupervised algorithms via attacks and anomaly classes," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, doi: 10.1016/J.JISA.2020.102474.
- [42] M. Zhang, B. Xu, and J. Gong, "An Anomaly Detection Model Based on One-Class SVM to Detect Network Intrusions," *Proc. - 11th Int. Conf. Mob. Ad-Hoc Sens. Networks, MSN 2015*, pp. 102–107, Feb. 2016, doi: 10.1109/MSN.2015.40.
- [43] Z. Ding and M. Fei, "An Anomaly Detection Approach Based on Isolation Forest Algorithm for Streaming Data using Sliding Window," *IFAC Proc. Vol.*, vol. 46, no. 20, pp. 12–17, Jan. 2013, doi: 10.3182/20130902-3-CN-3020.00044.
- [44] "Isolation Forest | Anomaly Detection with Isolation Forest." <https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/>
- [45] R. Sun, S. Zhang, C. Yin, J. Wang, and S. Min, "Strategies for data stream mining method applied in anomaly detection," *Cluster Comput.*, vol. 22, no. 2, pp. 399–408, Jun. 2019, doi: 10.1007/S10586-018-2835-2.
- [46] K. Leung and C. Leckie, "Unsupervised Anomaly Detection in Network Intrusion Detection Using Clusters," doi: 10.5555/1082161.
- [47] W. Chimphlee, A. H. Abdullah, M. N. M. Sap, S. Srinoy, and S. Chimphlee, "Anomaly-based intrusion detection using fuzzy rough clustering," *Proc. - 2006 Int. Conf. Hybrid Inf. Technol. ICHIT 2006*, vol. 1, pp. 329–334, 2006, doi: 10.1109/ICHIT.2006.253508.
- [48] S. Zhong, T. M. Khoshgoftaar, and N. Seliya, "CLUSTERING-BASED NETWORK INTRUSION DETECTION," <http://dx.doi.org/10.1142/S0218539307002568>, vol. 14, no. 2, pp. 169–187, Nov. 2011, doi: 10.1142/S0218539307002568.
- [49] M. Jianliang, S. Haikun, and B. Ling, "The application on intrusion detection based on K-means cluster algorithm," *Proc. - 2009 Int. Forum Inf. Technol. Appl. IFITA 2009*, vol. 1, pp. 150–152, 2009, doi: 10.1109/IFITA.2009.34.
- [50] A. P. Muniyandi, R. Rajeswari, and R. Rajaram, "Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm," *Procedia Eng.*, vol. 30, pp. 174–182, Jan. 2012, doi: 10.1016/J.PROENG.2012.01.849.
- [51] S. Gujral, E. Ortiz, and V. L. Syrmos, "An unsupervised method for intrusion detection using spectral clustering," *2009 IEEE Symp. Comput. Intell. Cyber Secur. CICS 2009 - Proc.*, 2009, doi: 10.1109/CICYBS.2009.4925096.
- [52] X. Ni, D. He, S. Chan, and F. Ahmad, "Network anomaly detection using unsupervised feature selection and density peak clustering," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9696, pp. 212–227, 2016, doi: 10.1007/978-3-319-39555-5_12.
- [53] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "NADO: Network Anomaly Detection using Outlier approach," *ACM Int. Conf. Proceeding Ser.*, pp. 531–536, 2011, doi: 10.1145/1947940.1948050.
- [54] I. Syarif, A. Prugel-Bennett, and G. Wills, "Unsupervised Clustering Approach for Network Anomaly Detection," pp. 135–145, 2012, doi: 10.1007/978-3-642-30507-8_13.
- [55] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "An effective unsupervised network anomaly detection method," *ACM Int. Conf. Proceeding Ser.*, pp. 533–539, 2012, doi: 10.1145/2345396.2345484.
- [56] M. Prasad, S. Tripathi, and K. Dahal, "Unsupervised feature selection and cluster center initialization based arbitrary shaped clusters for intrusion detection," *Comput. Secur.*, vol. 99, Dec. 2020, doi: 10.1016/J.COSE.2020.102062.
- [57] J. Dromard, G. Roudière, and P. Owezarski, "Unsupervised network anomaly detection in real-time on big data," *Commun. Comput. Inf. Sci.*, vol. 539, pp. 197–206, 2015, doi: 10.1007/978-3-319-23201-0_22.
- [58] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, pp. 134–147, Nov. 2017, doi: 10.1016/J.NEUCOM.2017.04.070.
- [59] B. C. Rhodes, J. A. Mahaffey, and J. D. Cannady, "Multiple Self-Organizing Maps for Intrusion Detection."
- [60] H. Güneş Kayacık, A. Nur Zincir-Heywood, and M. I. Heywood, "On the Capability of an SOM based Intrusion Detection System," *Proc. Int. Jt. Conf. Neural Networks*, vol. 3, pp. 1808–1813, 2003, doi: 10.1109/IJCNN.2003.1223682.
- [61] A. D. Landress, "A hybrid approach to reducing the false positive rate in unsupervised machine learning intrusion detection," *Conf. Proc. - IEEE SOUTHEASTCON*, vol. 2016-July, Jul. 2016, doi: 10.1109/SECON.2016.7506773.
- [62] S. Y. Huang and Y. N. Huang, "Network traffic anomaly detection based on growing hierarchical SOM," *Proc. Int. Conf. Dependable Syst. Networks*, 2013, doi: 10.1109/DSN.2013.6575338.
- [63] N. Marir, H. Wang, G. Feng, B. Li, and M. Jia, "Distributed abnormal behavior detection approach based on deep belief network and ensemble SVM using spark," *IEEE Access*, vol. 6, pp. 59657–59671, 2018, doi: 10.1109/ACCESS.2018.2875045.
- [64] K. Keerthi Vasan and B. Surendiran, "Dimensionality reduction using Principal Component Analysis for network intrusion detection," *Perspect.*

Sci., vol. 8, pp. 510–512, Sep. 2016, doi: 10.1016/J.PISC.2016.05.010.

[65] D. Yang and H. Qi, “A network intrusion detection method using independent component analysis,” *Proc. - Int. Conf. Pattern Recognit.*, 2008, doi: 10.1109/ICPR.2008.4761087.

[66] T. M. Pattewar and H. A. Sonawane, “Neural network based intrusion detection using Bayesian with PCA and KPCA feature extraction,” *2015 IEEE Int. Conf. Comput. Graph. Vis. Inf. Secur. CGVIS 2015*, pp. 83–88, Apr. 2016, doi: 10.1109/CGVIS.2015.7449898.

[67] F. Kuang, W. Xu, and S. Zhang, “A novel hybrid KPCA and SVM with GA model for intrusion detection,” *Appl. Soft Comput. J.*, vol. 18, pp. 178–184, May 2014, doi: 10.1016/j.asoc.2014.01.028.

[68] Z. Elkhadir, K. Chougali, and M. Benattou, “Intrusion detection system using PCA and kernel PCA methods,” *Lect. Notes Electr. Eng.*, vol. 381, pp. 489–497, 2016, doi: 10.1007/978-3-319-30298-0_50.

[69] M. Amer, M. Goldstein, and S. Abdennadher, “Enhancing one-class Support Vector Machines for unsupervised anomaly detection,” *Proc. ACM SIGKDD Work. Outlier Detect. Descr. ODD 2013*, pp. 8–15, 2013, doi: 10.1145/2500853.2500857.

[70] Q. T. Nguyen, T. Thu Huong, K. P. Tran, M. K. Nguyen, P. Castagliola, and S. Lardjane, “Nested one-class support vector machines for network intrusion detection,” *2018 IEEE 7th Int. Conf. Commun. Electron. ICCE 2018*, pp. 7–12, Sep. 2018, doi: 10.1109/CCE.2018.8465718.

[71] M. Verkerken, L. D’hooge, T. Wauters, B. Volckaert, and F. De Turck, “Towards Model Generalization for Intrusion Detection: Unsupervised Machine Learning Techniques,” *J. Netw. Syst. Manag.*, vol. 30, no. 1, pp. 1–25, Jan. 2022, doi: 10.1007/S10922-021-09615-7/FIGURES/6.

[72] N. Paulauskas and A. Baskys, “Application of Histogram-Based Outlier Scores to Detect Computer Network Anomalies,” *Electron. 2019, Vol. 8, Page 1251*, vol. 8, no. 11, p. 1251, Nov. 2019, doi: 10.3390/ELECTRONICS8111251.

[73] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001, doi: 10.1162/089976601750264965.

[74] R. Aliakbarisani, A. Ghasemi, and S. Felix Wu, “A data-driven metric learning-based scheme for unsupervised network anomaly detection,” *Comput. Electr. Eng.*, vol. 73, pp. 71–83, Jan. 2019, doi: 10.1016/J.COMPELECENG.2018.11.003.

[75] J. Auskalis, N. Paulauskas, and A. Baskys, “Application of Local Outlier Factor Algorithm to

Detect Anomalies in Computer Network,” *Elektron. ir Elektrotechnika*, vol. 24, no. 3, pp. 96–99, Jun. 2018, doi: 10.5755/J01.EIE.24.3.20972.

[76] T. Ding, M. Zhang, and D. He, “A Network Intrusion Detection Algorithm Based on Outlier Mining,” doi: 10.1007/978-981-10-6571-2_147.

[77] M. Manulis, A.-R. Sadeghi, and S. Schneider, Eds., “Applied Cryptography and Network Security,” vol. 9696, 2016, doi: 10.1007/978-3-319-39555-5.

[78] M. A. Kabir and X. Luo, “Unsupervised Learning for Network Flow Based Anomaly Detection in the Era of Deep Learning,” *Proc. - 2020 IEEE 6th Int. Conf. Big Data Comput. Serv. Appl. BigDataService 2020*, pp. 165–168, Aug. 2020, doi: 10.1109/BIGDATASERVICE49289.2020.00032.

[79] T. Truong-Huu *et al.*, “An Empirical Study on Unsupervised Network Anomaly Detection using Generative Adversarial Networks,” *SPAI 2020 - Proc. 1st ACM Work. Secur. Priv. Artif. Intelligent, Co-located with AsiaCCS 2020*, pp. 20–29, Oct. 2020, doi: 10.1145/3385003.3410924.

[80] D. Sovilj, P. Budnarain, S. Sanner, G. Salmon, and M. Rao, “A comparative evaluation of unsupervised deep architectures for intrusion detection in sequential data streams,” *Expert Syst. Appl.*, vol. 159, Nov. 2020, doi: 10.1016/J.ESWA.2020.113577.

[81] F. Carrera, V. Dentamaro, S. Galantucci, A. Iannacone, D. Impedovo, and G. Pirlo, “Combining Unsupervised Approaches for Near Real-Time Network Traffic Anomaly Detection,” *Appl. Sci. 2022, Vol. 12, Page 1759*, vol. 12, no. 3, p. 1759, Feb. 2022, doi: 10.3390/APP12031759.

[82] X. Sáez-de-Cámara, J. L. Flores, C. Arellano, A. Urbieto, and U. Zurutuza, “Clustered federated learning architecture for network anomaly detection in large scale heterogeneous IoT networks,” *Comput. Secur.*, vol. 131, p. 103299, Aug. 2023, doi: 10.1016/J.COSE.2023.103299.

[83] P. Casas Hernandez, J. Mazel, P. Owezarski, P. O. Unsupervised, and P. Casas, “Network Intrusion Detection Systems: Detecting the Unknown without Knowledge,” *Comput. Commun.*, vol. 35, no. 7, pp. 772–783, 2012, Accessed: May 23, 2022. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-00736278>.

[84] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, “An Effective Unsupervised Network Anomaly Detection Method,” 2012.

[85] P. Casas, J. Mazel, and P. Owezarski, “Knowledge-independent traffic monitoring: Unsupervised detection of network attacks,” *IEEE Netw.*, vol. 26, no. 1, pp. 13–21, Jan. 2012, doi: 10.1109/MNET.2012.6135851.

- [86] I. Aljarah and S. A. Ludwig, "MapReduce intrusion detection system based on a particle swarm optimization clustering algorithm," *2013 IEEE Congr. Evol. Comput. CEC 2013*, pp. 955–962, 2013, doi: 10.1109/CEC.2013.6557670.
- [87] J. Song, H. Takakura, Y. Okabe, and K. Nakao, "Toward a more practical unsupervised anomaly detection system," *Inf. Sci. (Ny)*, vol. 231, pp. 4–14, May 2013, doi: 10.1016/J.INS.2011.08.011.
- [88] A. M. Chandrasekhar and K. Raghuveer, "Intrusion detection technique by using k-means, fuzzy neural network and SVM classifiers," *2013 Int. Conf. Comput. Commun. Informatics, ICCCI 2013*, 2013, doi: 10.1109/ICCCI.2013.6466310.
- [89] "(PDF) Artificial Immune System Based Intrusion Detection: Innate Immunity using an Unsupervised Learning Approach." https://www.researchgate.net/publication/270276153_Artificial_Immune_System_Based_Intrusion_Detection_Innate_Immunity_using_an_Unsupervised_Learning_Approach.
- [90] K. A. P. Costa, L. A. M. Pereira, R. Y. M. Nakamura, C. R. Pereira, J. P. Papa, and A. Xavier Falcão, "A nature-inspired approach to speed up optimum-path forest clustering and its application to intrusion detection in computer networks," *Inf. Sci. (Ny)*, vol. 294, pp. 95–108, Feb. 2015, doi: 10.1016/J.INS.2014.09.025.
- [91] P. V. Amoli, T. Hamalainen, G. David, M. Zolotukhin, and M. Mirzamohammad, "Unsupervised Network Intrusion Detection Systems for Zero-Day Fast-Spreading Attacks and Botnets Distributed Intrusion Detection Systems Based on Artificial Immune System View project Unsupervised Network Intrusion Detection Systems for Zero-Day Fast-Spreading Attacks and Botnets," *Artic. Int. J. Digit. Content Technol. its Appl.*, 2016, Accessed: Jun. 16, 2022. [Online]. Available: <https://www.researchgate.net/publication/301549262>.
- [92] H. Bostani and M. Sheikhan, "Modification of supervised OPF-based intrusion detection systems using unsupervised learning and social network concept," *Pattern Recognit.*, vol. 62, pp. 56–72, Feb. 2017, doi: 10.1016/J.PATCOG.2016.08.027.
- [93] M. Almi'Ani, A. A. Ghazleh, A. Al-Rahayfeh, and A. Razaque, "Intelligent intrusion detection system using clustered self organized map," *2018 5th Int. Conf. Softw. Defin. Syst. SDS 2018*, pp. 138–144, May 2018, doi: 10.1109/SDS.2018.8370435.
- [94] D. He, S. Chan, X. Ni, and M. Guizani, "Software-Defined-Networking-Enabled Traffic Anomaly Detection and Mitigation," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1890–1898, Dec. 2017, doi: 10.1109/JIOT.2017.2694702.
- [95] E. Ariafar and R. Kiani, "Intrusion detection system using an optimized framework based on datamining techniques," *2017 IEEE 4th Int. Conf. Knowledge-Based Eng. Innov. KBEI 2017*, vol. 2018-January, pp. 0785–0791, Mar. 2018, doi: 10.1109/KBEI.2017.8324903.
- [96] E. Bigdeli, M. Mohammadi, B. Raahemi, and S. Matwin, "Incremental anomaly detection using two-layer cluster-based structure," *Inf. Sci. (Ny)*, vol. 429, pp. 315–331, Mar. 2018, doi: 10.1016/J.INS.2017.11.023.
- [97] M. Almi'Ani, A. A. Ghazleh, A. Al-Rahayfeh, and A. Razaque, "Intelligent intrusion detection system using clustered self organized map," in *2018 5th International Conference on Software Defined Systems, SDS 2018*, May 2018, pp. 138–144, doi: 10.1109/SDS.2018.8370435.
- [98] Y. Zhou, L. Yu, M. Liu, Y. Zhang, and H. Li, "Network intrusion detection based on kernel principal component analysis and extreme learning machine," *Int. Conf. Commun. Technol. Proceedings, ICCT*, vol. 2019-October, pp. 860–864, Jan. 2019, doi: 10.1109/ICCT.2018.8600104.
- [99] H. Choi, M. Kim, G. Lee, and W. Kim, "Unsupervised learning approach for network intrusion detection system using autoencoders," *J. Supercomput.*, vol. 75, no. 9, pp. 5597–5621, Sep. 2019, doi: 10.1007/S11227-019-02805-W/TABLES/10.
- [100] R. H. Hwang, M. C. Peng, C. W. Huang, P. C. Lin, and V. L. Nguyen, "An Unsupervised Deep Learning Model for Early Network Traffic Anomaly Detection," *IEEE Access*, vol. 8, pp. 30387–30399, 2020, doi: 10.1109/ACCESS.2020.2973023.
- [101] S. Zavrak and M. Iskefiyeli, "Anomaly-Based Intrusion Detection from Network Flow Features Using Variational Autoencoder," *IEEE Access*, vol. 8, pp. 108346–108358, 2020, doi: 10.1109/ACCESS.2020.3001350.
- [102] A. A. Megantara and T. Ahmad, "A hybrid machine learning method for increasing the performance of network intrusion detection systems," *J. Big Data*, vol. 8, no. 1, pp. 1–19, Dec. 2021, doi: 10.1186/S40537-021-00531-W/TABLES/8.
- [103] J. Liao, S. G. Teo, P. Pratim Kundu, and T. Truong-Huu, "ENAD: An ensemble framework for unsupervised network anomaly detection," *Proc. 2021 IEEE Int. Conf. Cyber Secur. Resilience, CSR 2021*, pp. 81–88, Jul. 2021, doi: 10.1109/CSR51186.2021.9527982.
- [104] A. Singh and J. Jang-Jaccard, "Autoencoder-based Unsupervised Intrusion Detection using Multi-Scale Convolutional Recurrent Networks," Apr. 2022, Accessed: Jun. 05, 2023. [Online]. Available: <https://arxiv.org/abs/2204.03779v1>.
- [105] Y. Wang, G. Sun, X. Cao, and J. Yang, "An Intrusion Detection System for the Internet of Things

Based on the Ensemble of Unsupervised Techniques,” *Wirel. Commun. Mob. Comput.*, vol. 2022, 2022, doi: 10.1155/2022/8614903.

[106] G. de C. Bertoli, L. A. P. Junior, A. L. dos Santos, and O. Saotome, “Generalizing intrusion detection for heterogeneous networks: A stacked-unsupervised federated learning approach,” *Comput. Secur.*, vol. 127, Sep. 2022, doi: 10.1016/j.cose.2023.103106.

[107] M. E. Eren *et al.*, “General-purpose Unsupervised Cyber Anomaly Detection via Non-negative Tensor Factorization,” *Digit. Threat. Res. Pract.*, vol. 4, no. 1, pp. 1–28, Mar. 2023, doi: 10.1145/3519602.

[108] J. Lan, X. Liu, B. Li, and J. Zhao, “A novel hierarchical attention-based triplet network with unsupervised domain adaptation for network intrusion detection,” *Appl. Intell.*, vol. 53, no. 10, pp. 11705–11726, May 2022, doi: 10.1007/S10489-022-04076-0/FIGURES/8.

[109] T. K. Boppana and P. Bagade, “GAN-AE: An unsupervised intrusion detection system for MQTT networks,” *Eng. Appl. Artif. Intell.*, vol. 119, p. 105805, Mar. 2023, doi: 10.1016/J.ENGAPPAI.2022.105805.

[110] “MIT Lincoln Laboratory: DARPA Intrusion Detection Evaluation.” <https://archive.ll.mit.edu/ideval/data/2000data.html> (accessed May 25, 2022).

[111] “KDD Cup 1999 Data.” <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> (accessed May 25, 2022).

[112] “DEF CON® Hacking Conference - Call for CTF Organizers.” <https://defcon.org/html/links/dc-ctf-cfo.html> (accessed Feb. 10, 2024).

[113] “LBNL/ICSI Enterprise Tracing Project - Traces Project Papers.” <https://www.icir.org/enterprise-tracing/papers.html> (accessed May 25, 2022).

[114] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, and K. Nakao, “Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation,” *Proc. 1st Work. Build. Anal. Datasets Gather. Exp. Returns Secur. BADGERS 2011*, pp. 29–36, 2011, doi: 10.1145/1978672.1978676.

[115] “NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB.” <https://www.unb.ca/cic/datasets/nsl.html> (accessed May 25, 2022).

[116] “Cyber Research Center - Data Sets | United States Military Academy West Point.” <https://www.usma.edu/centers-and-research/cyber-research-center/data-sets> (accessed Feb. 10, 2024).

[117] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, “Toward developing a systematic approach to generate benchmark datasets for intrusion detection,” *Comput. Secur.*, vol. 31, no. 3, pp. 357–374, May 2012, doi: 10.1016/J.COSE.2011.12.012.

[118] M. Jonker, A. King, J. Krupp, C. Rossow, A. Sperotto, and A. Dainotti, “Millions of targets under

attack: A macroscopic characterization of the DoS ecosystem,” *Proc. ACM SIGCOMM Internet Meas. Conf. IMC*, vol. Part F131937, pp. 100–113, Nov. 2017, doi: 10.1145/3131365.3131383.

[119] N. Moustafa and J. Slay, “UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” *2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc.*, Dec. 2015, doi: 10.1109/MILCIS.2015.7348942.

[120] M. Ring, S. Wunderlich, D. Grödl, D. Landes, and A. Hotho, “Flow-based benchmark data sets for intrusion detection,” 2017.

[121] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization,” *ICISSP 2018 - Proc. 4th Int. Conf. Inf. Syst. Secur. Priv.*, vol. 2018-January, pp. 108–116, 2018, doi: 10.5220/0006639801080116.

[122] M. P. Bharati and S. Tamane, “NIDS-Network Intrusion Detection System Based on Deep and Machine Learning Frameworks with CICIDS2018 using Cloud Computing,” *Proc. 2020 Int. Conf. Smart Innov. Des. Environ. Manag. Plan. Comput. ICSIDEMPC 2020*, pp. 27–30, Oct. 2020, doi: 10.1109/ICSIDEMPC49020.2020.9299584.

[123] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,” *Proc. - Int. Camahan Conf. Secur. Technol.*, vol. 2019-October, Oct. 2019, doi: 10.1109/CCST.2019.8888419.

[124] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, “Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset,” *Futur. Gener. Comput. Syst.*, vol. 100, pp. 779–796, Nov. 2019, doi: 10.1016/J.FUTURE.2019.05.041.

[125] “IoT-23 Dataset: A labeled dataset of Malware and Benign IoT Traffic. — Stratosphere IPS.” <https://www.stratosphereips.org/datasets-iot23> (accessed Jul. 08, 2023).

[126] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” *IEEE Symp. Comput. Intell. Secur. Def. Appl. CISDA 2009*, Dec. 2009, doi: 10.1109/CISDA.2009.5356528.

[127] N. Moustafa, G. Creech, and J. Slay, “Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models,” pp. 127–156, 2017, doi: 10.1007/978-3-319-59439-2_5.

[128] M. Ahmed, A. Naser Mahmood, and J. Hu, “A survey of network anomaly detection techniques,” *J. Netw. Comput. Appl.*, vol. 60, pp. 19–31, 2016, doi: 10.1016/j.jnca.2015.11.016.

- [129] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network Anomaly Detection: Methods, Systems and Tools," *IEEE Commun. Surv. TUTORIALS*, vol. 16, no. 1, 2014, doi: 10.1109/SURV.2013.052213.00046.
- [130] "GitHub - snayan06/Dimensionality-Reduction-Technique-PCA-LDA-ICA-SVD: In this we are going to learn about how to impliment PCA algorithm which is useful for Dimentionalitiy Reduction." <https://github.com/snayan06/Dimensionality-Reduction-Technique-PCA-LDA-ICA-SVD> (accessed Oct. 29, 2022).
- [131] "GitHub - vamsikrishnachangala/Dimensionality_reduction_KPCA." https://github.com/vamsikrishnachangala/Dimensionality_reduction_KPCA (accessed Oct. 29, 2022).
- [132] "GitHub - Habeebhassan/Anomaly_Detection_HBOS: Anomaly detection model using Histogram Based Outlier System to create labels for unsupervised dataset, by detecting outliers as a result of threshold set." https://github.com/Habeebhassan/Anomaly_Detection_HBOS (accessed Oct. 29, 2022).
- [133] "GitHub - zhongyuchen/outlier-detection: Detect outliers with 3 methods: LOF, DBSCAN and one-class SVM." <https://github.com/zhongyuchen/outlier-detection> (accessed Oct. 29, 2022).
- [134] "GitHub - yzhao062/pyod: A Comprehensive and Scalable Python Library for Outlier Detection (Anomaly Detection)." <https://github.com/yzhao062/pyod#hoffmann2007kernel> (accessed Oct. 30, 2022).
- [135] H. Bin Wang, H. L. Yang, Z. J. Xu, and Z. Yuan, "A clustering algorithm use SOM and K-Means in Intrusion Detection," *Proc. Int. Conf. E-bus. E-Government, ICEE 2010*, pp. 1281–1284, 2010, doi: 10.1109/ICEE.2010.327.
- [136] E. J. Palomo, E. Domínguez, R. M. Luque, and J. Muñoz, "Network security using growing hierarchical self-organizing maps," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5495 LNCS, pp. 130–139, 2009, doi: 10.1007/978-3-642-04921-7_14.
- [137] D. Ippoliti and X. Zhou, "A-GHSOM: An adaptive growing hierarchical self organizing map for network anomaly detection," *J. Parallel Distrib. Comput.*, vol. 72, no. 12, pp. 1576–1590, Dec. 2012, doi: 10.1016/J.JPDC.2012.09.004.
- [138] H. Gunes Kayacik, A. Nur Zincir-Heywood, and M. I. Heywood, "A hierarchical SOM-based intrusion detection system," *Eng. Appl. Artif. Intell.*, vol. 20, no. 4, pp. 439–451, Jun. 2007, doi: 10.1016/J.ENGAPPAI.2006.09.005.
- [139] M. Vasighi and H. Amini, "A directed batch growing approach to enhance the topology preservation of self-organizing map," *Appl. Soft Comput.*, vol. 55, pp. 424–435, Jun. 2017, doi: 10.1016/J.ASOC.2017.02.015.
- [140] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-October, pp. 618–626, Dec. 2017, doi: 10.1109/ICCV.2017.74.
- [141] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 3319–3327, Nov. 2017, doi: 10.1109/CVPR.2017.354.
- [142] I. Sturm, S. Lapuschkin, W. Samek, and K. R. Müller, "Interpretable Deep Neural Networks for Single-Trial EEG Classification," *J. Neurosci. Methods*, vol. 274, pp. 141–145, Apr. 2016, doi: 10.48550/arxiv.1604.08201.
- [143] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," *Proc. - IEEE Symp. Secur. Priv.*, pp. 305–316, 2010, doi: 10.1109/SP.2010.25.