Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9974270, 35 pages
https://doi.org/10.1155/2021/9974270

Research Article

Design and Development of an Efficient Network Intrusion Detection System Using Machine Learning Techniques

Thomas Rincy N[1] **and Roopam Gupta**[2]

[1]*Department of Computer Science and Engineering, University Institute of Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, M.P, India*
[2]*Department of Information Technology, University Institute of Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, M.P, India*

Correspondence should be addressed to Thomas Rincy N; rinc_thomas@rediffmail.com

Received 17 March 2021; Revised 16 April 2021; Accepted 14 May 2021; Published 28 June 2021

Academic Editor: Philippe Fournier-Viger

Today's internets are made up of nearly half a million different networks. In any network connection, identifying the attacks by their types is a difficult task as different attacks may have various connections, and their number may vary from a few to hundreds of network connections. To solve this problem, a novel hybrid network IDS called NID-Shield is proposed in the manuscript that classifies the dataset according to different attack types. Furthermore, the attack names found in attack types are classified individually helping considerably in predicting the vulnerability of individual attacks in various networks. The hybrid NID-Shield NIDS applies the efficient feature subset selection technique called CAPPER and distinct machine learning methods. The UNSW-NB15 and NSL-KDD datasets are utilized for the evaluation of metrics. Machine learning algorithms are applied for training the reduced accurate and highly merit feature subsets obtained from CAPPER and then assessed by the cross-validation method for the reduced attributes. Various performance metrics show that the hybrid NID-Shield NIDS applied with the CAPPER approach achieves a good accuracy rate and low FPR on the UNSW-NB15 and NSL-KDD datasets and shows good performance results when analyzed with various approaches found in existing literature studies.

1. Introduction

Research in network security is a vastly emerging topic in the domain of computer networking due to the ever-increasing density of advanced cyberattacks. The intrusion detection systems (IDSs) are designed to avert the intrusions and to protect the programs, data, and illegitimate access of the computer systems. The IDSs can classify the intrinsic and extrinsic intrusions in the computer networks of an organization and instigate the alarm if security infringement is comprised in an organization network [1]. One of the notable definitions for intrusion is that it produces malignant, outwardly activated functional violations. The primary goal of intrusion detection systems is to recognize a broad variety of intrusions, heretofore identified and unidentified attacks; to discover and adapt to unfamiliar attacks; and to detect and recognize intrusions in a prompt pattern [2]. The preliminary work on IDSs was researched by Anderson [3] who recommended means of examining data. Subsequent to Anderson's work, the previous work was aimed at developing the algorithms and procedures for online automated systems. The Sytek project [4] started producing audit trails having enhanced security and considered different approaches for analyzing automated systems. These observations contributed to the first empirical evidence that the end users can be recognized from each other through user action of using the computer [5]. The proof of SRI and Sytek studies [6] was the foundation of real-time IDS. The behavior of the users, whether it is normal or suspected, is continuously monitored by these systems. The real-time IDS relies on two techniques: (1) intrusions whether normal or suspected can be tracked by the flagged departure from the factual patterns of respective users and (2) perceived system susceptibilities and various infractions of the system-aimed security

protocols are best tracked from rule-based expert systems. The stability of precision and detection is primarily two measures applied mainly to assess the IDSs [7], and in recent years, many IDS research surveys have been accomplished to enhance these measures [8]. In the inception stages, many of the research studies mainly focus on the rule-based expert system and statistical approach. However, the various performance results show that these approaches when applied to large datasets are not accurate and precise [9].

To get the better of the above-mentioned problem, data mining approaches [10, 11] and machine learning techniques were introduced [12]. Some machine learning paradigms containing Graph-based methods [13], Linear Genetic Programming [14], Bayesian Network [15], *k*-NN [16], *K*-means clustering [17], Hidden Markov Model [18], Self-organizing map [19], etc. have been explored for the architecture of IDSs. Machine learning [20] can detect the correlation between features and classes found in training data and identify relevant subsets of attributes by feature selection and dimensionality reduction, then use the data to build a model for classifying data to perform predictions. The data dimensionality related to data mining and machine learning has doubled in the last decade that leads to several questions to current learning approaches [21]. Due to the presence of excessive cardinal features, the model that tends to learn gets overfitted, resulting in the performance degradation of the model.

To solve the problem of data dimensionality in machine learning and data mining, various dimensionality reduction approaches have been accessed which is considered as an essential step in the area of machine learning and data mining. Feature selection is an extensively employed and efficient technique applied for dimensionality reduction. The main aim of feature selection is to select the limited feature subsets from primary features conferring to relevancy appraisal standard that manages the training model to accomplish greater performance outcomes and reduced execution time and achieve higher model predictability. Most of the classification problem needs the supervised learning where the class-conditional possibilities and cardinal class are not familiar and the class labels and its instances are associated with each other [22]. There is a scarcity of knowledge in real-world applications related to relevant features. Endless feature candidates are acquired to generate the more coherent domain, which results in the existence of irrelevant and redundant features to the target approach or objective function. For the target approach, the relevant or significant features are not irrelevant or redundant; neither the redundant feature is spontaneously correlated with the target approach or objective function but impacts the learning approach. The new events are not added by the redundant features to the target approach or objective function. In the majority of the classification problems, it is a composite to learn even if the classifier is competent due to the presence of an enormous number of data, till the redundant features are excluded from the objective function. For the classification problem, the features are once generated then instead of processing with full data; the feature selection will bring about the feature subsets from the initial features and then process with the feature subsets to a learning algorithm. The nominally sized feature subsets for the classification problem are selected by the feature selection approach conferring to the following criterion:

(i) Normally, the classifier accuracy does not decline considerably

(ii) Among all the likely features, the initial distribution of the class shall be approximately close to the proceeding distribution of the class whenever the values are likely towards the features selected

To obtain the high merit feature subsets from 2^m subsets, the feature subset selection approaches search feature subsets conferring to a few significant appraisal criteria. However, this approach is intensive for the conclusion of the best subset and to select the intermediate-sized feature subsets with the volume (m); the strategy is expensive and restrictive. Various approaches like heuristic and random search lower the computational intricacy by a trade-off. To prohibit the feature subsets from exhaustively searching, a stopping criterion is required. A feature selection approach [23] does the job by subset generation, subset evaluation, stopping criterion, and the result from the validation. With the likely search approach, the chosen feature subsets are sent for subset evaluators with significant evaluation criteria. After the stopping criterion is performed, the feature subset that is competent enough to fit in the evaluation strategy is preferred, and then, finally, the finest feature subset is selected and gets authenticated by employing the domain knowledge or validation.

The detection methods of intrusion detection systems are classified into three major types: anomaly-based, signature-based, and hybrid-based. The signature-based IDS and anomaly-based IDS were the most favored methods in an organization until numerous shortcomings were observed, which leads to the development of hybrid intrusion detection systems. In the designing of IDS, classifying the datasets according to attack types and selecting the good feature subsets are a hard problem. The classifying of datasets according to attack types aids in predicting the vulnerability of individual attacks in various networks. Moreover, relevant features should not be irrelevant or redundant so that accurate and highly merit feature subsets are obtained. To address this issue, a new hybrid network intrusion detection system called NID-Shield is designed that classifies the dataset according to attack types. Furthermore, the hybrid CAPPER approach is applied as a feature subset selection approach. Screening is applied to those features by the CAPPER approach which is redundant having a high-class correlation. Moreover, machine learning algorithms are applied for selecting high merit and accurate feature subsets.

The major contributions of this manuscript are as follows:

(i) An efficient hybrid NID-Shield NIDS is proposed in this manuscript that classifies the UNSW-NB15 and NSL-KDD datasets according to the attack types and attack names

(ii) An effective hybrid feature subset selection method called CAPPER is applied as a feature subset selection that combines the CFS and Wrapper approaches for obtaining the reduced accurate and high merit feature subsets

(iii) The reduced accurate and high merit datasets obtained from CAPPER are trained by the machine learning approaches and assessed by a 10-fold cross-validation method

(iv) The hybrid NID-Shield network intrusion detection system shows overall good improvement results on the different approaches found in the existing literature studies

The remaining article is coordinated accordingly. Section 2 focuses on related work. Section 3 proposes the architecture of the hybrid NID-Shield NIDS. Section 4 relates to the characteristics of UNSW-NB15 and NSL-KDD datasets. Section 5 discusses the performance evaluation of the hybrid NID-Shield NIDS approach with various existing approaches on the UNSW-NB15 and NSL-KDD dataset, and Section 6 then concludes the work.

2. Related Work

This section introduces the existing literature studies on the hybrid network intrusion detection system. Moreover, this section discusses the advantage of a hybrid intrusion detection system over a traditional intrusion detection system. Furthermore, distinct machine learning approaches are acquainted and discuss the usefulness of selecting specific machine learning techniques.

2.1. State-of-the-Art Network IDSs. The research in the manuscript is focused on studying the appropriateness of intrusion detection approaches to recognize network-level intrusions, as the network structures generate resources more susceptible to intrusions than autonomous machines. Three facets of network structures generate resources more exposed to attack by an autonomous machine: (1) networks generally provide additional resources than autonomous machines; (2) networks are usually formed to aid resource sharing; and (3) the global security policies that are applied to the IDS are limited [24]. Moreover, the hybrid methods are suggested over the signature and anomaly-based IDS, as the integration of multiple approaches into a distinct hybrid system retains the advantages of multiple techniques, while reducing many of the deficiencies [25].

Acharya and Singh [26] conclude that for obtaining the best possible detection and accuracy rate, the hybrid learning approaches can be a good choice and proposed intelligent water drop (IWD) algorithm, introduced by Shah–Hosseini [27]. This approach applies the support vector machine (SVM) as a classification algorithm and IWD approach as a feature selection technique that is inspired by the nature. IWD approach selects the best feature subsets, and the evaluation of the subsets is executed by the SVM classifier. The proposed model lowers the forty-five features from the applied dataset to the lowest of ten features. KDD-Cup '99 dataset is used for the appraisal of metrics. The proposed approach attains an accuracy, detection, and precision of 99%. The disadvantage of applying the elemental IWD algorithm is the likelihood of choosing the adjacent node for a water drop to stream.

Arif et al. [28] introduced the hybrid approach for IDSs. In this approach, pruning of the node is performed by PSO and pruned decision tree is applied for the classification purpose in a NIDS. The proposed approach applies the single and multiple-objective particle swarm optimization (PSO) algorithms. The KDD-Cup '99 dataset is used as an experimental evaluation approach. From the 10% KDD-Cup '99 training and testing dataset, thirty arbitrary samples are chosen for evaluation purposes. The statistical records in every training and testing dataset are 12,000 and 24,000 accordingly for the appraisal of the metrics. The precision of 99.95% and accuracy of 93.5% are achieved using the above approaches. But there are some primary problems involved with traditional PSO when adopted as a feature selection approach. The most significant problem submits the following question: in a random initialization, from the initial population, how far is it to reach an optimal solution. If the optimum answer tells that the predicted prediction is far distant, then it may not be possible to obtain the global optimal solution within the allocated time. The second problem involves the conventional upgrading mechanism of global best and personal best of the PSO approach, as these mechanisms may result in losing some valuable features.

Ahmed et al. [29] applied a triple strategy to build a hybrid IDS in which the Naive Bayes feature subset selector (NBFS) technique has been applied for dimensionality reduction. For the outlier rejection, optimized support vector machines (OSVM) are applied, whereas prioritized k-nearest neighbors (PKNN) are applied as a classifier. The NSL-KDD, KDD-Cup '99, and Kyoto 2006+ datasets are used for evaluation purposes. 18 efficient features are preferred from the KDD-Cup '99 dataset with a detection ratio of 90.28%. 24 features are selected from the Kyoto 2006+ dataset having a detection ratio of 91.60%. The author has compared with previous work and has the best overall detection ratio of 93.28%. The major disadvantage with the Naive Bayes is that it presumes prediction of the features that are mutually independent to one another. The features with mutual independence are consistently hard to get in real-world problems.

Dash et al. [30] reports two new hybrid intrusion detection methods that are GS and sequence of GSPSO which is the combination of gravitational search and the particle swarm optimization algorithms. It involves search agents who relate to each other having heavy masses from the gravitational force, and their performance is assessed by their mass. The combination approach has been carried out to train ANN with models such as GS-ANN and GSPSO-ANN. The random selection of 10% features is selected for training purposes, while 15% is used for testing purposes and is applied successfully for intrusion detection purposes. The author does not apply any feature selection technique. The KDD-Cup '99 dataset was applied as a metric for

Table 1: Taxonomy of latest hybrid intrusion detection methods.

Hybrid-based intrusion detection techniques with feature selection techniques							
Year	Research papers	Algorithms	Techniques	Dataset	Evaluation criteria	Feature selection	Results
2017	[26]	SVM, IWD	SVM is applied as a classifier. Feature reduction applying IWD (intelligent water drop) method	KDD-Cup '99 dataset	Detection rate, precision rate, accuracy rate, false alarm rate	IWD	Achieves a detection rate of 99.40%, precision rate of 99.10%, false alarm of 1.40%, accuracy rate of 99.05%
2017	[28]	Particle swarm optimization (PSO)	Particle swarm optimization (PSO) algorithm is applied for pruning the node of DT, and the pruned DT is applied for the network IDS classification	KDD-Cup '99 dataset	Accuracy rate, precision rate, FPR., IDR, time	PSO	Accuracy of 96.65%, a precision of 99.98%, FPR of 0.136, IDR of 92.71%, and execution time of 383.58 sec. is obtained
2017	[29]	Prioritized KNN algorithm, optimized SVM algorithm, Naïve Bayes feature selection approach	PKNN is used for detecting input attacks, hybrid HIDS strategy (based on Naïve Bayes feature selection); OSVM is applied for outlier rejection. Naïve Bayes is applied as the feature selector approach	Kyoto 2006+ dataset, KDD-Cup '99 dataset, and NSL-KDD dataset	Specificity, sensitivity, detection rate, precision	NBFS	An overall sensitivity rate of 53.24%, detection rate of 94.6%, precision of 56.62%, specificity of 98.21% are obtained on all datasets
2017	[30]	Artificial neural network	Particle swarm optimization (GSPSO) is employed to train ANN, gravitational search (GS), and combination of GS	NSL-KDD dataset	MSE, detection rate, time	Not applied	MSE of 0.4527%, a detection ratio of 95.26%, and execution time of 103.70 seconds are obtained
2017	[31]	Hybrid multilevel data mining algorithm	Flexible mutual information-based feature selection (FIMS) is employed as feature selector, MH-ML (multilevel hybrid machine learning), MH-DE (multilevel hybrid data engineering), MEM (micro expert module) for training the KDD-Cup '99 dataset	KDD-Cup '99 dataset	Detection rate, recall, accuracy rate, F-value, precision rate	FIMS	A detection rate of 66.69%, accuracy of 96.70%, recall of 96.70%, precision of 96.55%, and F-value of 96.60% are achieved
2018	[32]	Support vector machine (SVM)	Chisqselector employing the SVM classifier for reduction of features	KDD-Cup '99 dataset	AUPR, AUROC, time	Chisqselector	AUPR of 96.24%, AUROC of 99.55%, and execution time of 10.79 seconds are obtained

Table 1: Continued.

Hybrid-based intrusion detection techniques with feature selection techniques							
Year	Research papers	Algorithms	Techniques	Dataset	Evaluation criteria	Feature selection	Results
2018	[33]	Vector-based genetic algorithm	Three feature selection methods are employed, linear correlation-based feature selection (LCFS), modified mutual information-based feature selection (MMIFS), and forward feature selection algorithm (FFSA), chromosomes as vector and training data as metrics	KDD-Cup '99 dataset and CTU-13 dataset	FPR, accuracy rate	LCFS, FFSA, MMIFS	FPR of 0.17% is achieved, and accuracy rate for the DoS is 99.8%
2018	[34]	Neural network with resilient back propagation algorithm, CART	Neural network with resilient back propagation algorithm to update the weights; feature reduction is performed by CART	ISCX & ISOT dataset	Detection rate, accuracy rate, FPR	CART	An accuracy rate of 99.20%, detection rate of 99.08%, and FPR of 0.75% are obtained
2018	[35]	Symmetrical uncertainty and genetic algorithm (SU-GA) is used as classification algorithm	Genetic algorithm is used on selected features; symmetric uncertainty is applied to find best features	UCI dataset	Accuracy rate	GA	An accuracy of 83.83% is obtained, and an execution time of 0.23 seconds is achieved on all approaches
2018	[36]	Genetic algorithm	Neurofuzzy inference system, neural fuzzy genetic, fuzzy logic controller, multilayer perception for attack classification	KDD-Cup '99 dataset	Accuracy rate	Fuzzy rule	A true attack detection and false alarm detection accuracy up to 99% rate of 1%.
2019	[37]	Random forest, Naive Bayes, J-48, *k*-nearest neighbor algorithm	WrapperSubsetEval and CfsSubsetEval are applied as two feature selection techniques, while random forest, *k* -NN algorithm, Naive Bayes, and J-48 are applied as the classifiers	NSL-KDD dataset	Detection rate, accuracy rate, *F*–measure, TP rate, FP rate, MCC, and time	Wrapper and filter	Overall accuracy rate of 99.86%, overall FPR of 0.00035%, overall detection ratio of 0.9828%, *F*-measure of 0.706%, overall TPR of 0.929%, overall MCC of 0.955%, and total execution time of 10.625 seconds (executed on NSL-KDD dataset with 25 attributes on all attack types)
2019	[38]	*K*-means clustering, DBSCAN, SMO	*K*-means is applied for data grouping, DBSCAN is employed to eliminate noise from data, and SMO is applied for intrusion detection	KDD-Cup '99 dataset	Detection rate, accuracy rate	DBSCAN	An approx detection rate of 70% and an approx accuracy of 98.1% are obtained

Table 1: Continued.

Hybrid-based intrusion detection techniques with feature selection techniques							
Year	Research papers	Algorithms	Techniques	Dataset	Evaluation criteria	Feature selection	Results
2019	[39]	Intelligent flawless feature selection algorithm (IFLFSA), entropy-based weighted outlier rejection (EWOD), intelligent layered classification algorithm	EWOD is used to detect outliers in data, IFLFSA is used as feature selection, and intelligent layered classification algorithm is applied to classify the data	KDD-Cup '99 dataset	Accuracy rate	IFLFSA	Overall accuracy of 99.45% is achieved
2019	[40]	ID3, *k*-nearest neighbor, isolation forest	*k*-nearest neighbor is used to apply a class to unknown data point, ID3 is used as feature selector, and isolation forest is employed to segregate normal data from anomaly	NSL-KDD & KDD-Cup '99 dataset	Detection rate, accuracy rate, false alarm rate	*k*-NN	The performance with KDD-Cup '99 dataset has a detection rate of 97.20%, accuracy of 96.92%, and FPR of 7.49%. Performance on NSL-KDD dataset has a detection rate of 95.5%, accuracy of 93.95%, and a FPR of 10.34%
2019	[41]	Best first search and Naïve Bayes (BFS-NB) algorithm	Best search is applied as attribute optimization approach, and Naïve Bayes is employed as classifier	KDD datasets from the US Air Force	Accuracy, sensitivity, specificity	Naive Bayes	Sensitivity analysis of 97%, accuracy of 92.12%, and specificity of 97.5% are obtained
2020	[42]	Deep neural network (DNN), classical AutoEncoder (CAE)	Deep neural network (DNN) is applied as classification, and classical AutoEncoder (CAE) is applied as a feature selector approach	UNSW-NB15 dataset	(DNN)	Classical AutoEncoder (CAE)	Precision of 92.08%, *F* -measure of 91.35%, accuracy of 91.29%, recall of 90.64%, and FPR of 0.805
2020	[43]	*k*-nearest neighbor (*k* -NN), extreme learning machine (ELM), hierarchical extreme learning machine (H-ELM), SDN controller	Hierarchical extreme learning machine (H-ELM), extreme learning machine (ELM), and *k*-nearest neighbor (*k*-NN) are applied for classification, and SDN controller is employed as a feature selection approach	NSL-KDD dataset	(*k*-NN), (ELM), (H-ELM)	SDN controller	An accuracy of 84.29%, FPR of 6.3%, precision of 94.18%, recall of 77.18%, *F* -measure of 84.83%
2021	[44]	ANN is applied as a classifier	An integration technique (CFS + ANN) is employed to improve the classification accuracy	NSL-KDD dataset and UNSW-NB15 dataset	(CFS + ANN)	Correlation-based feature selection technique	An accuracy of 98.45%, specificity of 94.38%, sensitivity of 92.94%, and execution time of 500 seconds are obtained on the NSL-KDD dataset. For the UNSW-NB15 dataset, an accuracy of 96.44%, specificity of 98.4%, a sensitivity of 50.4%,

Table 1: Continued.

Hybrid-based intrusion detection techniques with feature selection techniques							
Year	Research papers	Algorithms	Techniques	Dataset	Evaluation criteria	Feature selection	Results
							and an execution time of 1023 seconds are achieved
2021	[45]	SVM, modified binary gray wolf algorithm	SVM is used as a classifier and, modified binary gray wolf algorithm is applied as feature selection approach	NSL-KDD dataset	SVM	Modified binary gray wolf algorithm	An accuracy of 96%, FPR of 0.03, detection rate of 0.96, and execution time of 69.6 h
2021	[46]	Multiclassifier, deep neural network, kernel density	Random forest differential evaluation with kernel density for predicting unusual activities. For input classification, a multiclassifier is applied, while a deep neural network is employed as the learning and training of the data. Kernel density is used for clustering and prediction of data.	HHAR dataset	Random forest differential evaluation with kernel density, multiclassifier, deep neural network, kernel density	Basic sort-merge tree	An accuracy rate of 98.4%, a sensitivity of 96.02%, and a specificity of 99.8%

calculation. Normalization of the dataset was done for uniform distribution by MATLAB. An average detection ratio of 95.26% was achieved. The gradual shift of the search agent encourages the relevant solution of the algorithm, but the major weakness is its speed of convergence that slows down in subsequent stages and has the tendency to get trapped in the local optimum solution.

Yao et al. [31] introduced a hybrid framework for IDS. *K*-means algorithm is employed for clustering purposes. In the classification phase, many machine learning algorithms (SVM, ANN, DT, and RF) which are all supervised learning algorithms are compared on different parameters. The supervised learning algorithm has various parameters for different kinds of attacks (DoS, U2R, Probe, and R2L). FIMS is applied as a feature selection technique. The proposed approach has obtained an accuracy rate reaching 96.70% with the KDD-Cup '99 dataset. The drawback with the FIMS approach is that the correlation between the candidate features and their class is not considered.

Suad and Fadl [32] introduced an IDS model applying the machine learning algorithm to the big data environment. This paper employs a Spark-Chi-SVM model. ChisqSelector is applied as a feature selection method, and an IDS model is constructed by applying the SVM as a classifier. The comparison is done with the Spark-Chi-SVM classifier and Chi-logistic-regression classifier. The KDD-Cup '99 dataset is used for the metrics of the evaluation process. The result shows that the Spark-Chi-SVM model shows good performance having an AUROC of 99.55% and an AUPR of 96.24%. The disadvantage of ChisqSelector is having a larger sensitiveness towards the sample size. However, when the sample size increases, the total differences become smaller than the predicted value.

Ijaz et al. [33] introduce a genetic algorithm, which is based on vectors. In this technique, vector chromosomes are applied. The uniqueness of this algorithm is that it shows the chromosomes as a vector and training data as metrics. It grants multiple pathways to have a fitness function. Three feature selection techniques are chosen: forward feature selector algorithm (FFSA), linear correlation feature selector (LCFS), and modified mutual information feature selector (MMIFS). The novel algorithm is tested in two datasets (CDU-13 and KDD-Cup '99). Performance metrics demonstrate that the vector genetic algorithm has a high detection ratio of 99.8% and a low false positive rate of 0.17% on the denial of service (DoS) attack. However, the authors do not evaluate the U2R, Probe, and R2L attacks which are considered important metrics in the IDS.

Alauthaman et al. [34] proposed an approach of peer-to-peer bot detection build on a feed-forward neural network in assistance with the DT. CART is then applied as a feature selection approach to obtain the significant features. Network traffic reduction techniques were applied by using six rules to pick the most relevant features. Twenty-nine features are selected from six rules. The proposed approach obtained an accuracy of 99.20% and a detection ratio of 99.08%, respectively. The disadvantage of utilizing a CART is that the decision tree may not be stable and the CART splits the variables one by one.

Venkataraman and Selvaraj [35] report an efficient hybrid feature selection structure for the classification of

the data. For classification purposes, symmetrical uncertainty is applied to find the relevant features. Moreover, GA is applied to search for the merit subset with higher accuracy. The author combined SU-GA as a hybrid feature selection approach. MATLAB and Weka tools are applied for evaluation purposes. Different classification algorithms (KStar, J48, NB, SMO, DT, JRIP, Multilayer Perceptron, and Random forest) are used to classify different attacks. The average learning accuracy with Multi Perpn and SU-GA is the highest having 86.0%. The major drawback of a genetic algorithm is that it may be computationally expensive, as the training of the model is required for the appraisal of each candidate. GA is stochastic, so it may require a longer time to converge.

Kumar and Kumar [36] introduce an intelligent-based hybrid NIDS model. This model then integrates the multilayer perception, fuzzy logic controller, adaptive neurofuzzy interference system, and a neurofuzzy genetic. The author applied fuzzy logic as a feature selection method. The proposed system has three key elements: analyzer, collector, and predictor modules, for gathering and filtering network traffic to classify the data and prepare the final decision in assuming knowledge on the accurate attack. The experiment is assessed on the KDD-Cup '99 dataset that achieves an improvement of true attack detection and false alarm detection accuracy upto 99% rate of 1% using MATLAB. The disadvantage of fuzzy logic is that the results are observed based on assumptions, and due to this reason, accuracy is sometimes incorrect.

Cavusoglu et al.[37] applied the hybrid approach for IDS using machine learning techniques. k-nearest neighbor and Naive Bayes algorithms are used for classification purposes, while the random forest algorithm is used as a classifier. The author applied two feature selection techniques called the CfsSubsetEval and WrapperSubsetEval approach. J48 algorithm is applied in conjunction with WrapperSubsetEval for selecting accurate attributes. For the evaluation of metrics, the NSL-KDD dataset is applied. The overall accuracy of 99.86% is obtained on all types of attacks.

Saxena et al. [38] implemented a DBSCAN-based hybrid technique for obtaining the high-quality feature subsets for IDS. DBSCAN is employed as a method for eliminating noise from data. For grouping data, K-means clustering is proposed. The SMO classifier is applied for classification purposes. The KDD-Cup '99 dataset is applied for evaluation purposes with reduced attributes. The proposed approach, DBKSMO, achieved an accuracy of about 98%. Weka and MATLAB tools are applied for the execution of the results. However, the major disadvantage of DBSCAN is that whenever there is a cluster having variations in density or the clusters having similar variation, its performance declines, the major reason being the setting of ε (distance threshold), and minimum points for determining the neighborhood points will change from clusters to clusters, whenever density changes. This problem exists for high-dimensional data, as the ε (distance threshold) becomes difficult to examine.

Kambattan and Rajkumar [39] introduced effective IDS, which employs a feature selection technique named IFLFSA to select the finest reduced features that are effective for analyzing the attacks. To identify the outliers from the dataset, the EWOD approach is utilized. An intelligent layered technique is employed for efficient classification. For experimental purposes, the KDD-Cup '99 dataset is applied. The comprehensive detection rate wraps the detection rate on four types of attacks, namely, Probe, DoS, U2R, and R2L. The detection rate of the proposed system is achieved at a rate of 99.45%. The weakness of using intelligent agents is that whenever the global constraints are applied, the intelligent agent fails to deliver appropriately. Each agent is more effective in dealing individually with the main or central controller. The agents make the decisions based on locally acquired knowledge; whenever there is global knowledge available, the agents are missing the major available knowledge globally.

Kar et al. [40] utilize the decision tree algorithm called ID3 which is applied for the classification of the data into its corresponding classes. To designate the class labels to its unexplored data point on its class labels to the k-nearest point, the k-NN approach is applied. Isolation forest is introduced to isolate the anomaly against normal instances. The suggested approach HFA has applied to the NSL-KDD and KDD-Cup '99 dataset. The metrics on the KDD-Cup '99 dataset obtained the ACC of 96.92%, DR of 97.20%, and FPR of 7.49%. The proposed algorithm performance with the NSL-KDD dataset has an ACC of 93.95%, DR of 95.5%, and FPR of 10.34%. However, the main drawback of applying the k-NN is that whenever the size of the variables increases, the k-NN finds it difficult in predicting the output of the new data positions. On the other side, the k-NN does well with the variables having smaller numbers.

Mishra et al. [41] applied the BFS-NB hybrid structure in IDS. This paper proposes the best first search technique for dimensionality reduction which was employed for the attribute selection technique. For the classification of data, Naïve Bayes classifier is applied for a classification purpose and to maximize the accuracy of detecting intrusion. The BFS-NB algorithm is analyzed with the KDD dataset gathered from the US Air Force. The classification accuracy of BS-NFB is 93% while the sensitivity analysis of 97% is achieved. The major disadvantage with the Naive Bayes is that it presumes prediction of the features that are mutually independent to one another.

Dutta et al. [42] introduced a hybrid model for improving the classification metrics in a NIDS. The literature applies a deep neural network for enhancing classification accuracy. Furthermore, classical autoencoder is used as a feature subset selection technique. The efficiency of a proposed technique is evaluated with the UNSW-NB15 dataset. A precision rate of 92.08%, a recall of 90.64%, an accuracy of 91.29%, and F-measure of 91.35%, and an FPR of 0.805 are obtained from the proposed architecture. The deep neural network has activation functions and multiple layers that produce nonconvex shapes. The drawback of a deep neural network probably introduces the complex error space, leading to the substantially tuning of hyperparameters to be able to get into a small error space so that the model can be beneficial. Moreover, the training is very slow due to the tuning of many hyperparameters.

Latah and Toker [43] introduce an efficient flow-based multilevel hybrid intrusion detection system. The author

applies the k-NN, H-ELM, and ELM which are used for classification purposes, and the SDN controller is used as a feature selection method. An accuracy of 84.29%, FPR of 6.3%, a precision of 94.18%, a recall of 77.18%, and F-measure of 84.83% are obtained from the proposed approach. However, the disadvantages of k-NN are that it is not able to handle well with large and high-dimensional datasets. Furthermore, the k-NN is sensitive to the noise in the dataset.

Sumaiya Thaseen et al. [44] applied the integrated techniques CFS + ANN to improve the classification accuracy. CFS is applied as a feature selection approach for selecting the best feature subsets, while the ANN is employed as a classifier. UNSW-NB15 and NSL-KDD datasets are used for evaluating purposes. An accuracy of 98.45%, a sensitivity of 92.94%, a specificity of 94.38%, and an execution time of 500 seconds are obtained on the NSL-KDD dataset. For the UNSW-NB15 dataset, an accuracy of 96.44%, a sensitivity of 50.4%, specificity of 98.4%, and an execution time of 1023 seconds are achieved. The major disadvantage of ANN is that it takes a longer time for training the data.

Safaldin et al. [45] applied the improved binary gray wolf optimizer as a feature selection method and support vector machine for classification in an IDS in a wireless sensor network. The proposed approach attains an accuracy of 96%, FPR of 0.03, a detection rate of 0.96, and an execution time of 69.6 h. The choosing of a good kernel function is hard which is the major disadvantage of the SVM classifier. Moreover, SVM takes a longer time in training the large datasets, and to store all the support vectors, the memory consumption is extensive.

Vallathan et al. [46] introduce the skeptical action detection system that is based on the deep learning approach in IoT surroundings. Unexpected activities obtained from the footage of the N/W surveillance devices are predicted with the help of deep learning approaches and RFKD. For classification purposes, the multiclassifier approach is used, while DNN is used for training and learning the data. Moreover, for prediction and clustering of data, the kernel density approach is applied. The proposed approach uses the basic merge-sort tree as a feature subset selection approach. For evaluation purposes, HHAR datasets are used. The proposed approach obtained an accuracy rate of 98.4%, specificity of 99.8%, and a sensitivity of 96.02%, on the HHAR dataset. However, the main drawback of the neural network is that the training is very slow due to the tuning of many hyperparameters.

Table 1 depicts the taxonomy of the latest hybrid IDS techniques with its various feature selection approaches. When the literature studies are analyzed, most of them do not classify the dataset according to attack types and attack names thus preventing the assessment of individual attacks on the various networks. Distinct attacks may have peculiar connections as some of the attacks such as R2L and U2R may have very few N/W connections, while other attacks such as Probe and DoS may have a large number of N/W connections or can be a combination of any of them. The attack names found in the attack types help in predicting the vulnerability of individual attacks in various networks. Moreover, a feature selection approach that utilizes highly merit and accurate feature subsets which apply machine learning techniques is not utilized. Furthermore, performance metrics such as precision, MCC, ROC area, PRC area, kappa statistic, MAE, RAE, RMSE, and RRSE which are considered important metrics in model predictability are not utilized in the existing works of literature.

Due to the reviewed problem in the literature studies, a novel hybrid network IDS named NID-Shield has been introduced that employs a distinct machine learning and efficient hybrid feature subset selection approach called CAPPER that is the sequence of the CFS and Wrapper method. Moreover, the hybrid NID-Shield NIDS classifies the dataset according to the various attack names and their types found in the dataset.

2.2. Advantages of Hybrid NIDSs. This section introduces the problem of the existing approaches of IDSs based on anomaly and signature IDSs and explains the advantages of hybrid network intrusion detection systems.

Cybersecurity ventures [47] in the report estimate that the damages arising due to cybercrime in 2025 will increase to $10.5 trillion annually as compared to $3 trillion in 2015. Furthermore, there is a prediction of nearly 7.5 billion active internet users by the end of 2030 worldwide and spending on cybersecurity aggregately surpasses $1 trillion approximately in the coming five years globally.

Despite having enormous financing in the field of IDSs, the losses brought by the intrusions are soaring at an alarming rate leading to enormous debt revenues to the organizations. Considering the efficiency of the IDS, there should be an analytical and stringent proceeding to be acclimated so that network susceptibilities can be classified in a precise and accurate fashion. In past decades, the IDS has been the blocking source for ever-growing intrusion violations and it is utilized as a primary prevention method against computer attacks, safeguarding networks and computer systems. IDS employs statistical techniques, logical operation, and machine learning approaches to analyze distinct kinds of network behaviors [48]. Although present-day IDSs are certainly effective and pursue upgrades, they still develop numerous false alarm rates and fail to analyze the unidentified attacks. Utmost IDSs rely upon inappropriate and redundant inferior level network data to observe cyber intrusions [49]. At two layers of supervision, the existing intrusion detection approaches work to counter the cyberattacks, the host, and the network level. NIDS audits the details of N/W connections to identify the cyberattacks. Contrarily, HIDS scans the workstations' stature and internals of the computing structure utilizing definitive IDS techniques so that at the host level, the potential intrusions can be detected. NIDS is the operating system and platform-independent that does not require any modification when NIDS operates. This makes NIDS more scalable and robust compared with HIDS.

Machine learning analysts classify IDS within three extensive categories: anomaly-based, signature-based, and hybrid-based [50]. The anomaly-based IDS employs the new action profiles which are created every time to distinguish the deviation of outliers from the new profiles. Anomaly-based IDS depends on analytical methods to

constitute an attack predictor model. The attack that does not have predefined signatures is recognized by the anomaly-based IDS as its main strength. However, a major weakness lies in the difficulty in creating new action profiles every time. Moreover, the deviations of outliers from the new profiles always are not an attack. Failing to analyze the perimeters of new actions leads to the false prediction of new actions as an attack, possibly ending in a high false-positive rate. The signature-based intrusion detection systems evaluate resemblance among occurrences under scrutiny and the familiar attack patterns. If the patterns formerly established are recognized, then alarms are triggered. For signature-based IDS, e.g., the SNORT [51] is among the utmost preferable, consistently adapted technique. SNORT carries out content seeking, content resembling, and real-time traffic investigation to recognize attacks by employing the predefined precise signatures. Although these systems are definite in analyzing the identified attack, they are incapable to perceive the unidentified attack.

The hybrid-based IDS integrates the anomaly and signature detection approaches to detect attacks. However, the computational expense of utilizing the anomaly and signature IDS that examines the N/W connections is the major drawback of hybrid approaches. The anomaly and signature IDS were the most preferred methods in an organization until various weaknesses were observed leading to the development of hybrid intrusion detection systems. Furthermore, when Table 1 is observed related to hybrid network IDS, most of the literature studies do not classify the dataset according to attack types and their names leading to the difficulty in predicting the attacks individually on different networks. To solve this problem, a novel hybrid NIDS called NID-Shield is proposed in the manuscript that classifies a dataset according to different attack types.

2.3. Machine Learning Algorithms Used in This Study. Distinct machine learning algorithms such as neural network [52], decision trees [53], *k*-nearest neighbor [54], and support vector machine [55] are introduced by the researchers to attain learning on the datasets. Under the contrasting structure of the datasets, the particular algorithms apply distinct methods for achieving higher performance from the datasets. The relevant approach may be applied according to the divergent form of the datasets [56]. Machine learning algorithms such as Naive Bayes, random forest, and J48 (C 4.5) are applied in this study for analyzing the outcome of feature selection and training of the classifier. These algorithms are known to be prominent in the area of machine learning and have proven appropriate in the process.

2.3.1. Random Forest. Random forest [57] is the sequel of tree predictors, and every tree corresponds to the profit of random vector that is sampled independently, and there is an identical distribution of entire trees in the forest. In a forest, as the tree grows larger, the generalization error coincides to a greater extent. The generalization error from a forest relies on the individual strengths of a tree in the forest and correlation between each other in a forest of classifier trees. The random forest performs sequences of inputs or the inputs that are randomly selected at every node so that the accuracy can be increased. By applying this method, the correlation is decreased and simultaneously yields efficacy to forests. The random forest constructs the random features at every node by dividing the limited number from the input variables and electing the features randomly. In the random forest, the tree is grown with the same procedure as the CART [58] approach and the branch that is to be developed is determined by the Gini index. Random forest applies bagging [59] besides the selection of random features. From the standard training dataset, a contemporary training dataset is performed with substitution, and then, on the contemporary training dataset with the help of random feature selection, the tree is grown. Pruning of the tree is not performed on the random forest; rather, the trees are grown in this approach.

Employing bagging has mainly two benefits. Firstly, the accuracy is increased each time the random features are enforced. Secondly, estimation of the generalization error containing the ensemble tree combination and the correlations and its intensity appraisal is provided by the bagging. The assessment is carried out-of-bag [60]. The main approach behind the out-of-bag estimation is the incorporation of nearly one-third of classifiers from the continuing prevailing sequence. Whenever the statistic of the sequence is incremented, the rate of error declines. Therefore, the contemporary error rate can be augmented by out-of-bag estimation; hence, it is necessary to pass on from the area where the merging of the error occurs. In the cross-validation, there is a high probability of the existence of bias; also, the degree of extent of the bias is unfamiliar, whereas the out- of-bag estimation is free from bias. The random forest applies two-thirds of the data and for testing one-third of the data from training data, to grow the tree. Out-of-bag data is simply the one-third data from the training data. Pruning is not performed by the random forest which thus aids in fast and high performance. Moreover, having the multiple tree construction, the random forest performs reasonably well with additional tree framework and it achieves a higher performance rather than any other decision tree method.

2.3.2. Naïve Bayes. Naive Bayes [61] is the classifier having the probabilistic nature, having the relationship relevant to Bayes belief with the strong expectation, and having naive independence between its features. With the kind of probabilistic analysis, the Naive Bayes represent the knowledge. In mathematical terms, the Naive Bayes can be defined as

$$P\left(\frac{R}{S}\right) = P(R)\frac{(S/R)}{P(S)}, \tag{1}$$

where R and S are the events and $P(R)$ and $P(S)$ are the events.

$P(R/S)$ is the posterior probability, having the probability of observation of the event R, given that the S is true. $P(R)$ and $P(S)$ are called the prior probabilities of R and S. $P(S/R)$ is called likelihood, the probability of observation of an event S, given that R is true. The Naive Bayes version that is applied in this study is the implementation by [62]. The nominal feature probabilities are approximated from the

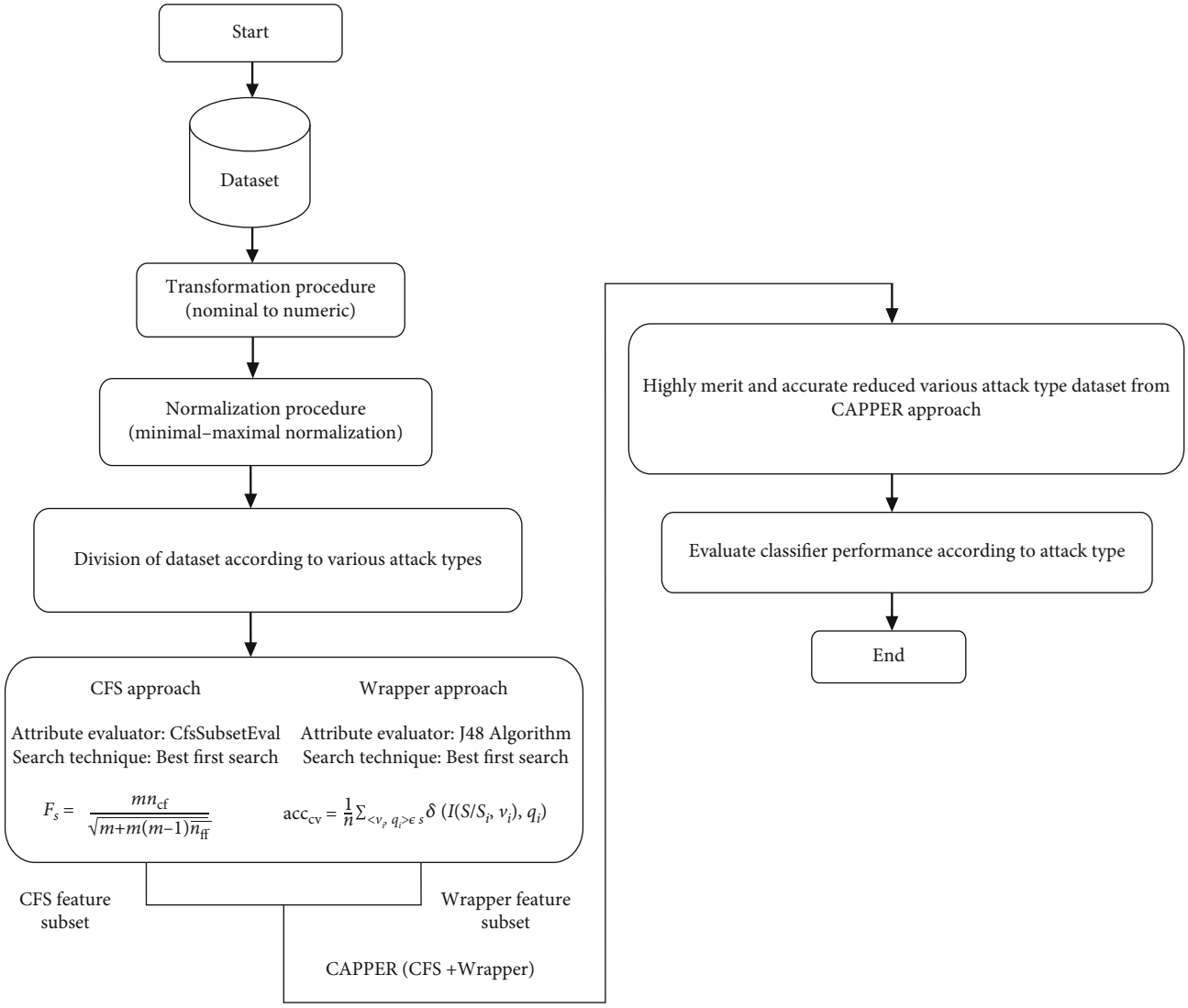

Figure 1: The simplified block diagram of hybrid NID-Shield NIDS according to various attack types.

given data and the Gaussian distribution. The highly apparent class for the given instance based on the entire data distribution is predicted by the Bayes classifier or Bayes rule. Whenever the log probabilities are applied, the Naive Bayes is easy to understand. There are added scoring objectives and natural expression capabilities found in the log probabilities. High accuracy can be obtained from the Bayes classifier. Whenever the redundant features have been eliminated, the performance of the Naive Bayes improves considerably, as discussed by Langley and Sage [63]. Moreover, when modest dependencies prevail in the data, the Naive Bayes performs exceptionally, as discussed by Domingos and Pazzani [64]. A minimal execution time is needed from Naive Bayes to train the data.

2.3.3. J48(C4.5) Decision Tree Generator. C4.5 [65] decision tree is applied in this study. C4.5 is a descendent of an ID3 algorithm. C4.5 is commonly known as J48 in the Weka library. C4.5 constructs the decision trees, and the pruning is performed on the decision trees with the help of the top-down method. The construction of the trees is performed by C4.5 by finding the feature sets having distinct best characteristics so that on the root node of a tree, the testing of the features can be performed. The nodes of the tree relate to its features and branches that relate to its values. The leaf of the tree is reciprocal to the classes, and to classify the new instance, one needs to analyze the features that are tested at the nodes of the tree and pursue the branch corresponding to the values noticed in an instance. The process gets terminated, whenever it arrives at the leaf and also the nomination of the class to its instance.

The greedy approach is used by C4.5 to construct the decision trees which applies the information-theoretic estimates. For obtaining the attribute for tree root, this algorithm splits the instances of the training into subsets which coincides with the attributes corresponding amount. If there is insignificant entropy among the labels of the class in a subset corresponding to labels of the class in the entire training dataset, gaining the information is done by dividing the attribute. The gain ratio principle is enforced by C4.5 for the

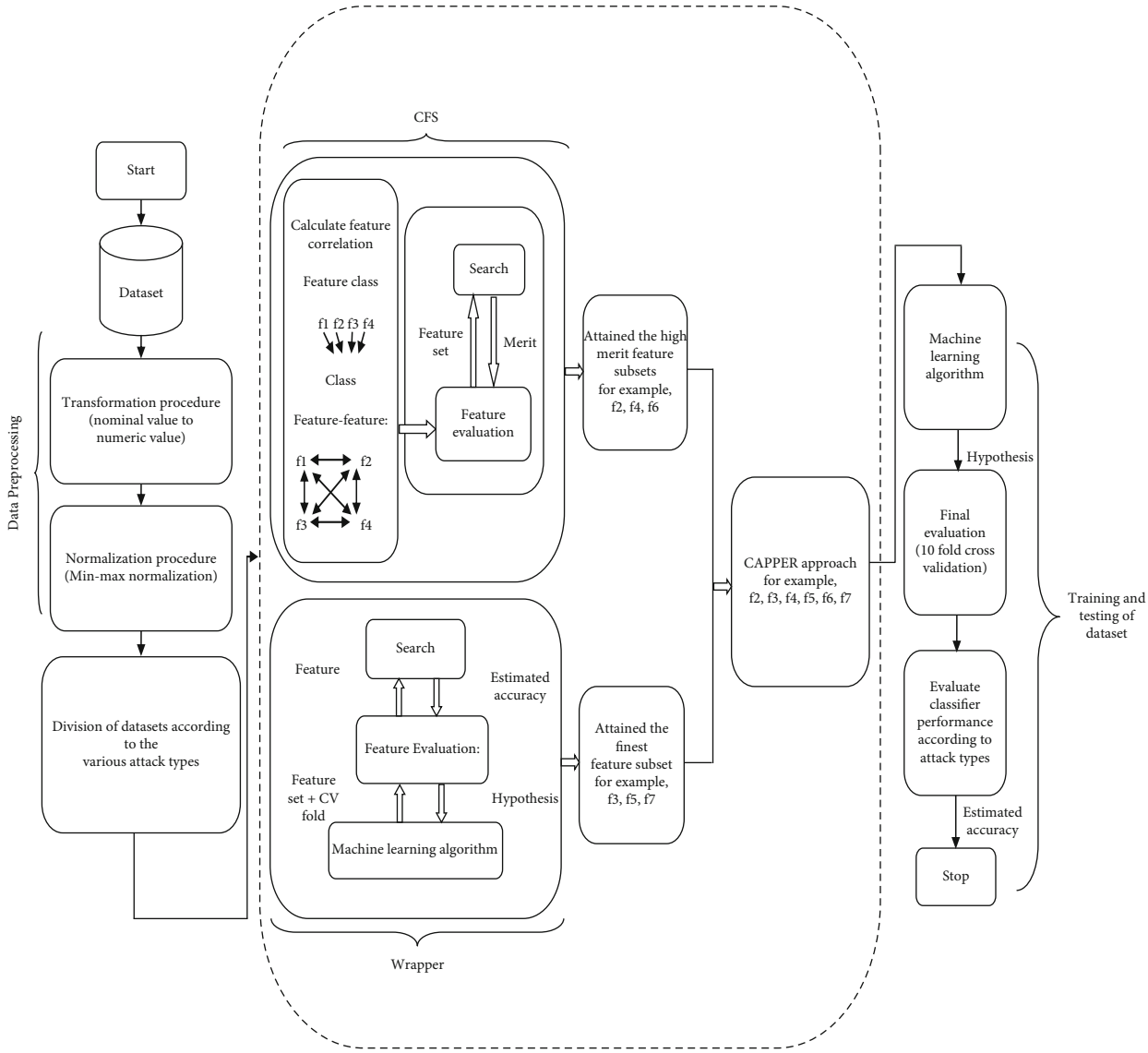

Figure 2: A proposed architecture of hybrid NID-Shield network intrusion detection system

Table 2: Four categories of attack.

Attack category	Name of attack
Denial of service (DoS)	teardrop, smurf, neptune, back, land, pod
Probe	satan, nmap, ipsweep, portsweep
User to root (U2R)	loadmodule, buffer_overflow, rootkit
Remote to local (R2L)	multihop, phf, ftp_write, warezclient, imap, guess_passwd, warezmaster

Table 3: Number of instances in NSL-KDD and NSL-KDD 20% training on normal and attack type.

NSL-KDD dataset	Normal	Probe	DoS	U2R	R2L	Total instances
NSL-KDD training	67343	11656	45927	52	995	125973
NSL-KDD 20% training	13449	2289	9234	11	209	25192

Table 4: Features of NSL-KDD 20% dataset.

Index	Feature name	Type	Missing	Distinct	Unique	Mean	Std. Dev.
1	duration	Numeric	0	758	682	0.007	0.063
2	protocol_type	Nominal	0	3	0	0.125	0.283
3	service	Nominal	0	66	1	0.157	0.198
4	flag	Nominal	0	11	0	0.062	0.103
5	src_bytes	Numeric	0	1665	864	0	0.006
6	dst_bytes	Numeric	0	3922	2377	0.001	0.017
7	land	Nominal	0	2	0	0	0.009
8	wrong_fragment	Numeric	0	3	0	0.008	0.087
9	urgent	Numeric	0	2	1	0	0.006
10	hot	Numeric	0	22	7	0.003	0.028
11	num_failed_logins	Numeric	0	5	2	0	0.011
12	logged_in	Numeric	0	2	0	0.395	0.489
13	num_compromised	Numeric	0	28	18	0	0.012
14	root_shell	Numeric	0	2	0	0.002	0.039
15	su_attempted	Numeric	0	3	0	0.001	0.024
15	num_root	Numeric	0	28	20	0	0.012
17	num_file_creations	Numeric	0	20	13	0	0.013
18	num_shells	Numeric	0	2	0	0	0.019
19	num_access_files	Numeric	0	7	2	0.001	0.012
20	num_outbound_cmds	Numeric	0	1	0	0	0
21	is_host_login	Nominal	0	1	0	0	0
22	is_guest_login	Numeric	0	2	0	0.009	0.095
23	count	Numeric	0	466	70	0.164	0.225
24	srv_count	Numeric	0	414	69	0.052	0.142
25	serror_rate	Numeric	0	70	9	0.286	0.447
26	srv_serror_rate	Numeric	0	56	25	0.284	0.448
27	rerror_rate	Numeric	0	72	9	0.119	0.319
28	srv_rerror_rate	Numeric	0	42	10	0.12	0.322
29	same_srv_rate	Numeric	0	97	7	0.661	0.44
30	diff_srv_rate	Numeric	0	79	14	0.062	0.179
31	srv_diff_host_rate	Numeric	0	57	4	0.096	0.257
32	dst_host_count	Numeric	0	256	1	0.716	0.388
33	dst_host_srv_count	Numeric	0	256	1	0.451	0.434
34	dst_host_same_srv_rate	Numeric	0	101	0	0.52	0.449
35	dst_host_diff_srv_rate	Numeric	0	101	0	0.083	0.187
36	dst_host_same_src_port_rate	Numeric	0	101	0	0.147	0.308
37	dst_host_srv_diff_host_rate	Numeric	0	63	8	0.032	0.111
38	dst_host_serror_rate	Numeric	0	100	5	0.286	0.445
39	dst_host_srv_serror_rate	Numeric	0	88	19	0.28	0.446
40	dst_host_rerror_rate	Numeric	0	101	0	0.118	0.306
41	dst_host_srv_rerror_rate	Numeric	0	100	7	0.119	0.317

selection tree of the root attribute. The gain ratio principle elects those attributes which have an average or better gain between its distinct attributes.

By employing the algorithm iteratively, subtrees are constructed in this algorithm. Furthermore, the algorithm terminates upon finding the likely subset that contains a distinct class. The main distinction between C4.5 and ID3 is that pruning is performed on decision trees by C4.5; hence, by applying the pruning, the simplification is done on the decision trees and has the high chance of reducing the overfitting on a training data. C4.5 performs pruning by employing the confidence interval upper bound on the resubstitution error. The succession of the node is preceded by the best leaf, whenever the error of the estimation of the leaf is situated within a single standard deviation from the predicted error of a node. C4.5 is considered as an efficient algorithm, whenever the

efficacy regarding the machine learning algorithm is assessed; also, it is fast, robust, and accurate whenever the knowledge is brought in. Moreover, it performs well with feature subset selection on the relevant and redundant data, thus aiding in increasing the accuracy.

3. The Proposed Hybrid NID-Shield Network Intrusion Detection System

This section introduces the various techniques applied by the hybrid NID-Shield NIDS. The data preprocessing steps are performed by applying the transformation and normalization operations on datasets, and then, an effective hybrid feature subset selection technique called CAPPER is applied for obtaining the accurate and highest merit feature subsets. Finally, the hybrid NID-Shield NIDS is suggested as a whole exclusively.

3.1. Data Preprocessing. In data preprocessing, the transformation and normalization operation is performed on NSL-KDD 20% dataset. It can help to better expose the underlying structure of the data to the learning algorithm and, in turn, may result in better predictive performance.

3.1.1. Data Transformation. In the transformation operation, the nominal values are converted to numeric values. The IDSs are considered as the classification issue and some classification approaches are not able to handle the nominal features [66]. In the NSL-KDD 20% dataset, the attributes such as protocol_type, service, and flag are transformed from nominal to numeric values and the final NSL-KDD 20% dataset contains the entire numeric values for the classification process.

3.1.2. Data Normalization. Data normalization is an essential paradigm, specifically in the area of classification. The instances are observed as a multidimensional area in the linear classification approaches. Without normalization, few objective functions do not work accordingly due to the wide variations of raw data. For example, if the particular feature has wide value ranges, then the range within the points is controlled by the distinct feature. Thus, normalization of the numeric features needs to be done so that every feature provides nearly proportional to the eventual distance. Therefore, by applying the normalization, there is a significant improvement in accuracy and speed. For this study, minimal-maximal normalization approach is applied to the dataset. The minimal-maximal normalization is given as

$$z = \frac{x - \text{minimal}(x)}{[\text{maximal}(x) - \text{minimal}\,(x)]}. \tag{2}$$

The minimal-maximal normalization technique linearly scales each feature to the interval of [0, 1]. Resizing of the interval [0, 1] is performed by altering every feature value such that the minimum value is 0, and then, division is performed by the current maximum value. The current maximum value is the change among the initial maximum value and minimum value which is obtained from equation (2).

Table 5: Total instances in UNSW-NB 15 training and testing dataset.

S.no.	Total instances	Training dataset (UNSW-NB15)	Testing dataset (UNSW-NB15)
1	Normal	56000	37000
2	DoS	12264	4089
3	Fuzzers	18184	6062
4	Analysis	2000	677
5	Worms	130	44
6	Exploits	33393	11132
7	Shellcode	1133	378
8	Generic	40000	18871
9	Reconnaissance	10491	3496
10	Backdoor	1746	583

3.2. Feature Selection Approaches. A hybrid feature subset selection approach named CAPPER [67] is employed for feature subset selection that combines the feature subsets from the CFS and Wrappers for the feature subset selection method. This section introduces the CAPPER approach.

3.2.1. Correlation-Based Feature Subset Approach. CFS is the filter method that utilizes correlation-based searching for the appraisal of the feature subsets. The feature subset ranking is accomplished by conferring to correlation-based searching. The bias is accessed to those subsets which are greatly correlated to its class and uncorrelated among them. This approach ignores the features that are irrelevant and having fewer correlations among its class. The screening is applied to the features which are redundant and hugely correlated among its class. The acceptance of the features is done by the CFS when the residual features do not predict the predicted class in the instant space.

$$F_S = \frac{m\overline{n_{cf}}}{\sqrt{m + m(m-1)\overline{n_{ff}}}}, \tag{3}$$

where F_S is the heuristic merit of the feature subset S having the m features, $\overline{n_{\text{ff}}}$ is the average feature-feature intercorrelation, and $\overline{n_{\text{cf}}}$ are the feature mean class correlation. The searching of the space is performed with the help of a best-first approach. The high-quality subset of the features is obtained by equation (3), which aids in reducing the dimensional reduction of testing and training data. Moreover, the numerator of equation (3) illustrates that how remarkably the class predictability is with feature sets and the denominator denotes the redundancy between the features.

3.2.2. Wrapper Subset Selection Approach. In the Wrapper approach, the feature subset selector is performed with the help of an induction approach. The searching of the feature subset space is performed with the help of backward elimination and forward selection methods. The backward elimination begins with complete feature sets and removing those features that degrade the performance. The forward selection

Table 6: Features of UNSW-NB 15 dataset.

Index	Feature name	Type	Missing	Distinct	Unique	Mean	Std. Dev.
1	id	Numeric	0	82332	82332	0.5	0.289
2	dur	Numeric	0	39888	35946	0.017	0.079
3	proto	Nominal	0	131	0	0	0.002
4	service	Nominal	0	13	0	0.26	0.438
5	state	Nominal	0	7	2	0.047	0.5
6	spkts	Numeric	0	420	174	0.002	0.013
7	dpkts	Numeric	0	436	205	0.002	0.01
8	sbytes	Numeric	0	4489	2570	0.001	0.012
9	dbytes	Numeric	0	4034	2446	0.001	0.01
10	rate	Numeric	0	40616	32279	0.082	0.149
11	sttl	Numeric	0	11	1	0.71	0.398
12	dttl	Numeric	0	8	0	0.378	0.461
13	sload	Numeric	0	42873	38993	0.012	0.034
14	dload	Numeric	0	40614	37491	0.03	0.115
15	sloss	Numeric	0	253	101	0.001	0.012
15	dloss	Numeric	0	311	124	0.001	0.01
17	sinpkt	Numeric	0	39970	36718	0.013	0.103
18	dinpkt	Numeric	0	37617	34993	0.002	0.022
19	sjit	Numeric	0	39944	37503	0.004	0.038
20	djit	Numeric	0	38381	36358	0.001	0.008
21	swin	Nominal	0	11	9	0.523	0.499
22	stcpb	Numeric	0	39219	37322	0.253	0.324
23	dtcpb	Numeric	0	39108	37295	0.25	0.322
24	dwin	Numeric	0	14	11	0.503	0.5
25	tcprtt	Numeric	0	26130	22613	0.015	0.03
26	synack	Numeric	0	24934	20749	0.009	0.022
27	ackdat	Numeric	0	24020	19622	0.009	0.019
28	smean	Numeric	0	1282	178	0.078	0.141
29	dmean	Numeric	0	1222	236	0.078	0.163
30	trans_depth	Numeric	0	8	4	0.001	0.004
31	response_body_len	Numeric	0	1190	809	0	0.007
32	ct_srv_src	Numeric	0	57	0	0.138	0.179
33	ct_state_ttl	Numeric	0	7	1	0.228	0.178
34	ct_dt_ltm	Numeric	0	50	1	0.082	0.145
35	ct_src_dport_ltm	Numeric	0	50	1	0.068	0.145
36	ct_dst_sport_ltm	Numeric	0	33	1	0.072	0.16
37	ct_dst_src_ltm	Numeric	0	57	0	0.104	0.184
38	is_ftp_login	Numeric	0	3	0	0.004	0.046
39	ct_ftp_cmd	Numeric	0	3	0	0.004	0.046
40	ct_ftw_http_mthd	Numeric	0	8	0	0.008	0.04
41	ct_src_ltm	Numeric	0	50	1	0.093	0.145
42	ct_srv_dst	Numeric	0	57	0	0.134	0.182
43	is_sm_ips_ports	Numeric	0	2	0	0.011	0.105
44	attack_cat	Nominal	0	10	0	0.074	0.261
45	label	Numeric	0	2	0	0.551	0.497

begins with empty feature sets and starts adding the good features. The goal of this approach is to obtain the state with the highest appraisal by applying the heuristic function. For the appraisal function, the five fold cross-validation approach is performed, and it is replicated numerous times by examining the accuracy estimation and its standard deviation. The k

-fold cross-validation is also called an out-of-sample test or rotation estimation. S is the original sample, which splits into folds of S_1, S_2, S_3,...,S_n relatively identical size every time $t \in \{1, 2, \cdots, k\}$ trained on $S \setminus S_t$ and tested on S_t. The induction approach is tested and trained k times. The estimation of the cross-validation accuracy is the comprehensive figure of accurate classifications divided from the total instances from the dataset. Let S_i is the testing set that contains the instances $p_i = <v_i, q_i>$; then, the estimation of cross-validation accuracy is obtained as

$$\mathrm{acc}_{cv} = \frac{1}{n} \sum_{<v_i, q_i> \in S} \delta(I(S \setminus S_i, v_i), q_i). \quad (4)$$

The best-first approach is applied as the search technique. Upon arriving at the goal, the best-first search usually terminates. The accuracy estimation is obtained from equation (4). By combining the feature subsets from Wrapper and CFS approaches, CAPPER attains the accurate and high-quality feature subsets.

3.3. Ensemble Learning. Ensemble learning [68] was initially evolved in automated decision-making systems to lessen the variance and thus increase the accuracy. The problems in machine learning domains such as error correction, estimation confidence, missing features, and cumulative learning are strongly addressed by ensemble learning techniques. Ensemble learning is widely used in the area of pattern recognition, artificial intelligence, machine learning, data mining, and neural networks. Ensemble learning has proved its efficiency and functionality in an extensive area of real-world problems.

The ensemble learning combines various base learners or weak learners and integrates them to make a strong learner. The superiority of ensemble learning is that it increases the accuracy of the weak learning system so that the comprehensive accuracy of the classifier on the training datasets is increased as compared to the single base learning algorithms.

3.3.1. Stacking. In stacking [69], the cardinal classifier obtains a new dataset from the original datasets. If the same instances are generated from the original dataset by the cardinal classifier, then there is high speculation that the data gets overfitted, which is the primary reason the datasets with contemplating nature need to be obtained for discarding the overfitting of the data. There is a suggestion to use the cross-validation approach for the new instances of the cardinal classifier; also, the group of features has to be considered for the contemporary training dataset and the different categories of the learning algorithms on the Meta-learner. Distinct learning algorithms are applied for obtaining the cardinal learner. Then, the new datasets are used with Meta-learner to train the data. Stacking is the induction of numerous machine learning approaches.

3.4. k-Fold Cross-Validation. Cross-validation techniques are frequently mentioned as test/train holdout approach by the researchers. In the k-fold cross-validation [70], the repetition on the dataset is performed k times. At every round, the dataset is split into k parts; one part is applied for the validation and the residual k-1 parts of the datasets are combined into a training subset for appraisal of the model. In k-fold cross-validation, a complete set of testing and training data is used, and the main idea of this technique is to lessen the fatalistic bias by applying the major number of training data while keeping the large testing datasets separately. The folds of the test data do not overlap each other. In k-fold cross-validation, each of the samples is applied for validation. Sometimes, it is necessary to choose the exact value of k to avoid the high bias in the model. Usually, the value of $k = 10$ is chosen mostly, as the various experimental results show that the model has small bias and low variance whenever this value is applied. The results from this approach are then combined or averaged to generate the distinct estimation.

Table 7: Confusion matrix.

Class	Predicted negative class	Predicted positive class
Actual negative class	FP	TN
Actual positive class	TP	FN

3.5. The Proposed Hybrid NID-Shield Network Intrusion Detection System Using Hybrid Feature Selector. The preliminary design approach behind the hybrid NID-Shield is the classification of datasets according to different attack types. The advantage of classification of the dataset according to attack types is that it can find a set of arbitrary features. Moreover, the attack names found in the attack types help in predicting the vulnerability of individual attacks in various networks. Distinct machine learning algorithms are analyzed as per the individual attack types. The machine learning algorithms having high accuracy; low FPR are selected for different attack types and applied in the designing of the hybrid NID-Shield NIDS. The hybrid NID-Shield NIDS applies the hybrid approach called CAPPER for selecting the optimal feature subsets. The hybrid CAPPER approach for feature selection combines the optimal feature subsets from the CFS and Wrappers for the feature subset selection method. From the CFS approach, a prominently superior feature subset is obtained which is independent of irrelevant and redundant features. The wrapper method uses induction learning algorithms to attain a highly accurate feature subset. By combining the filter and wrapper approaches, high merit and accurate feature subsets are obtained which is then applied for training and testing purposes.

For designing the hybrid NID-Shield NIDS, single and ensemble learning algorithms are used together so that a high-performance rate and lower FPR can be achieved. Testing is performed with single and ensemble learning algorithms; it has been found that ensemble learning achieved high-performance results, where the NSL–KDD 20% having fewer samples in some of its attack types. The high-performance classifier is determined for different attack types, and for the classifier performance, the k-fold cross-

Table 8: DOS attack evaluated with hybrid NID-Shield NIDS approach.

(a)

Total instances	22,683
Correctly classified instances	22,679 (99.98%)
Incorrectly classified instances	4 (0.00176%)
Execution time	6.77 seconds
Kappa measures	0.9997
MAE	0.0003
RMSE	0.0081
RAE	0.2207%
RRSE	2.9778%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	normal
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	back
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	land
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	neptune
94.7%		0.947	0.000	0.973	0.947	0.960	0.960	1.000	0.997	pod
99.6%		0.996	0.000	0.998	0.996	0.997	0.997	1.000	1.000	smurf
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	teardrop
Weighted Avg.	100%	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	

(c)

Confusion matrix						
a	*b*	*c*	*d*	*e*	*f*	*g*
13,449	0	0	0	0	0	0
0	196	0	0	0	0	0
0	0	4	0	0	0	0
0	0	0	8279	0	0	0
0	0	1	0	36	1	0
0	0	0	0	2	527	0
0	0	0	0	0	0	188

a—classified as normal, *b*—classified as back, *c*—classified as land, *d*—classified as neptune, *e*—classified as pod, *f*—classified as smurf, *g*—classified as teardrop.

validation approach is applied. The advantage of this method is that all observations are used for training and validation, and each observation is used for validation exactly once. For the classification problem, a cross-validation technique with 10-fold is applied. The folds are selected in a manner such that every fold consists of the approximately identical distribution of the class. To test the network data according to attack type, the various attack type data are passed to different layers of the hybrid NID-Shield NIDS; the high-performance classifier determines the data as normal or attack type. Figure 1 depicts the simple block diagram of the hybrid NID-Shield network intrusion detection system and Figure 2 displays the architecture of the hybrid NID-Shield NIDS.

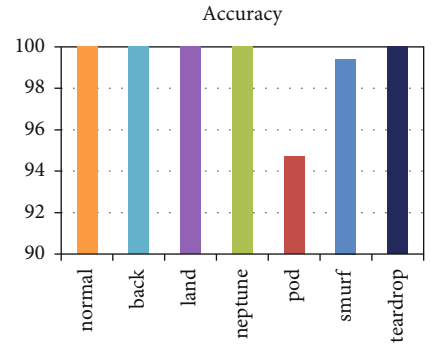

Figure 3: Accuracy of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

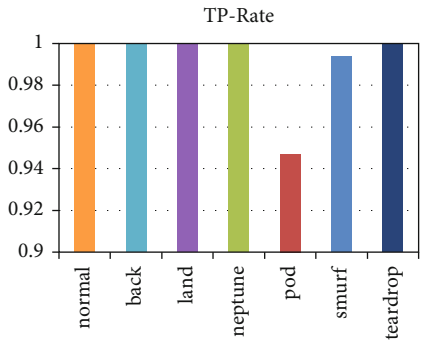

Figure 4: TP rate of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

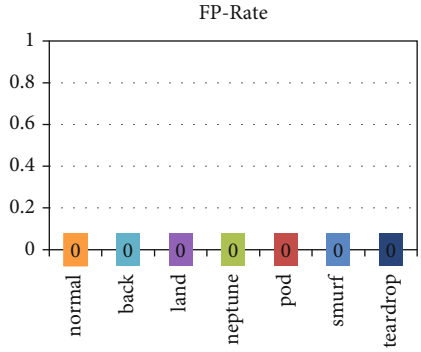

Figure 5: FP rate of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

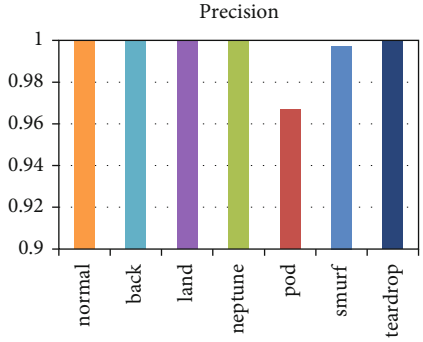

Figure 6: Precision of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

4. Dataset Characteristics

For the performance of the hybrid NID-Shield NIDS, two contemporary UNSWNB-15 and NSL-KDD 20% datasets are utilized for evaluation purposes. These datasets are related to cybersecurity and are high-dimensional and class imbalanced datasets [71]. For the NSL-KDD dataset, the statistical prevalence of around 36% was found in denial of service (DoS), while for other attack types like Root to Local (R2L) and User to Root (U2R), the prevalence is lesser than 1%. This shows that NSL-KDD is a highly imbalanced data-

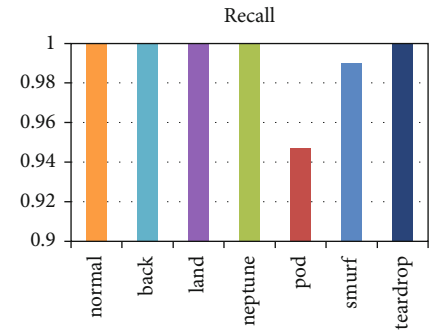

Figure 7: Recall of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

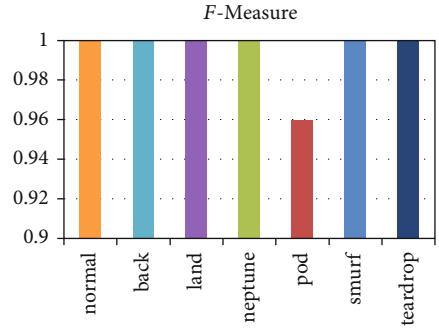

Figure 8: *F*-measure of the normal and attack types evaluated by the NID-Shield NIDS on DoS attack.

set. For the UNSW-NB15 dataset, the normal class frequency is about 32%, while attack type frequency is very few and differs highly. For example, Worms and Exploits attack patterns vary around 257 times. This reflects that UNSW-NB15 is a highly imbalanced dataset.

4.1. NSL-KDD Dataset. From DARPA 98 intrusion detection system appraisal programs, the KDD-Cup '99 dataset is obtained and widely applied dataset in the domain of IDS, but the main disadvantage of the KDD-Cup '99 datasets has various duplicate and redundant records. The duplicate records have a total of 75%. The redundant record has a total of 78%. Due to this duplication and redundant information hinders from categorizing the additional records [72]. A new NSL-KDD dataset was suggested [73] that does not contain the duplicate and redundant records in testing and training data [74], which aided in removing the duplicate and redundant issues which is an implicit issue in KDD-Cup '99 dataset. The arrangements of elected records from every adversity class level are inversely proportional to the percent of records available in the standard KDD datasets. With these results, the classifying rates of apparent machine learning approaches differ in an extensive range that makes it more efficient to obtain a precise appraisal of distinct learning approaches. The statistical records in the training and testing sets are feasible that causes it to be reasonable to conduct the experiments on an entire set, thus preventing the unnecessary need to randomly select the limited part. Therefore,

Table 9: Probe attack evaluated with hybrid NID-Shield NIDS approach without stacking.

(a)

Total instances	15,738
Correctly classified instances	15,685
Incorrectly classified instances	53
Execution time	4.23 seconds
Kappa measures	0.9872
MAE	0.0034
RMSE	0.0343
RAE	3.1818%
RRSE	14.9173%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
99.9%		0.999	0.016	0.997	0.999	0.998	0.988	1.000	1.000	normal
99.3%		0.993	0.000	1.000	0.993	0.997	0.996	0.999	0.998	portsweep
96.8%		0.968	0.000	0.999	0.968	0.983	0.982	0.999	0.996	satan
99.3%		0.993	0.001	0.989	0.993	0.991	0.990	1.000	0.996	ipsweep
95.3%		0.953	0.000	0.976	0.953	0.965	0.964	0.998	0.992	nmap
Weighted Avg.	99.7%	0.997	0.014	0.997	0.997	0.997	0.988	1.000	0.999	

(c)

Confusion matrix				
a	*b*	*c*	*d*	*e*
13441	0	1	0	3
2	593	0	0	0
4	0	679	1	2
2	0	0	718	2
0	0	0	3	287

a—classified as normal, *b*—classified as portsweep, *c*—classified as satan, *d*—classified as ipsweep, and *e*—classified as nmap.

the classified results will be persistent and proportionate. There are entirely 37 attacks in a testing dataset out of which 21 different attacks are of training dataset and the remaining attacks are available only for testing the data. Table 2 shows the four categories of attack. Table 3 depicts the total number of instances on the distinct attack types and normal and on the NSL-KDD and 20% of the NSL-KDD training dataset. The attack classes are categorized into Probe, DoS, U2R, and R2L categories, and Table 4 shows the features of the NSL-KDD 20% dataset.

(i) *Denial of service (DoS)*. These kinds of attack result in the unavailability of computing resources to legitimate users. The intruder overloads the resources, by accomplishing the resources of the computer active, so that authentic users are unable to utilize the full resources of the computer. In DoS, there are 13449 normal instances and 9234 attack instances with six attack names, namely, neptune, smurf, back, teardrop, pod, and land

(ii) *Probe*. The intruder gathers the knowledge from the networks or hosts and scans the whole networks or hosts that are prone to attacks. An intruder then exploits the system vulnerabilities by looking at the known security breaches so that the whole system is compromised for malicious purposes. In Probe, there are 13449 normal instances and 2289 attack instances with four attack names, namely, nmap, ipsweep, satan, and portsweep

(iii) *User to root (U2R)*. An intruder tries to acquire accessing the system roots or the administrator privileges by sniffing the passwords. The attacker then looks for the vulnerabilities in the system, to acquire the gain of the administrator authorization. In U2R, there are 13449 normal instances and 11 attack instances with three attack names, namely, loadmodule, buffer_overflow, and rootkit

(iv) *Root to local (R2L)*. The intruder attempts by gaining a connection to the remote machine, which does not

Table 10: Probe attack evaluated with hybrid NID-Shield NIDS approach with stacking.

(a)

Total instances	15,738
Correctly classified instances	15,690
Incorrectly classified instances	48
Execution time	42.99
Kappa measures	0.9884
MAE	0.002
RMSE	0.0318
RAE	1.8696%
RRSE	13.82%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
99.9%		0.999	0.010	0.998	0.999	0.999	0.990	0.999	1.000	normal
99.7%		0.997	0.000	1.000	0.997	0.998	0.998	1.000	1.000	portsweep
97.7%		0.977	0.000	0.993	0.977	0.985	0.984	0.995	0.990	satan
99.3%		0.993	0.001	0.987	0.993	0.990	0.990	0.999	0.994	ipsweep
96.3%		0.963	0.001	0.967	0.963	0.965	0.964	0.997	0.986	nmap
Weighted Avg.	99.7%	0.997	0.009	0.997	0.997	0.997	0.990	0.999	0.999	

(c)

Confusion matrix				
a	*b*	*c*	*d*	*e*
13435	0	5	5	4
2	585	0	0	0
8	2	675	1	2
1	2	0	705	4
3	2	4	3	290

a—classified as normal, *b*—classified as portsweep, *c*—classified as satan, *d*—classified as ipsweep, and *e*—classified as nmap.

have the necessary and legal privilege to access that machine. The attacker then exploits the susceptibility of the remote system and tries gaining access rights to the remote machine. There are 13449 normal instances and 209 attack instances in this dataset. There are eight attack names in this dataset, namely, ftp_write, guess_passwd, multihop, phf, imap, warezclient, spy, and warezmaster

4.2. UNSW-NB15 Dataset. The UNSW-NB15 [75] dataset was generated at the cyber range lab by the IXIA PerfectStorm tool at the Center for cybersecurity, Australia. There are 2,540,044 records in the dataset. The part of the dataset is further divided into train and test sets. There are 82,332 records in the testing set and 1,75,341 records in the training set, having normal and attack instances. There are 45 features in this dataset obtained in immaculate format, including class and label. Moreover, there are nine attack types in a UNSW-NB15 dataset: DoS, Analysis, Backdoor, Exploit, Fuzzers, Generic, Worm, Shellcode, and Reconnaissance and a Normal instance. Table 5 shows the total instances in UNSW-NB 15 training and testing dataset, and Table 6 depicts the UNSW-NB 15 dataset and its features.

For the evaluation of the proposed approach, the machine learning workbench tool, Weka 3.8 [76], is used. In Weka, the Wrapper approach, the CFS approach, and the classifier algorithms like J48, Naïve Bayes, and Random forest are implemented in Java, and evaluation of code is accomplished on Intel i3 8100 processor with 2.20 GHz having 4.00 GB RAM and carried out on NetBeans 8.0.2.

5. Performance Metrics

For validation of the results, this section presents various performance evaluation metrics. The researchers apply false negative (FN), true negative (TN), true positive (TP), false positive (FP), etc. [77] for the justification of the results.

Definition 1 (confusion matrix). Also called error metric, which allows the interaction among actual and predicted classes. It is significant for calculating precision, recall, accuracy, specificity, AUC, and ROC curve. On the testing

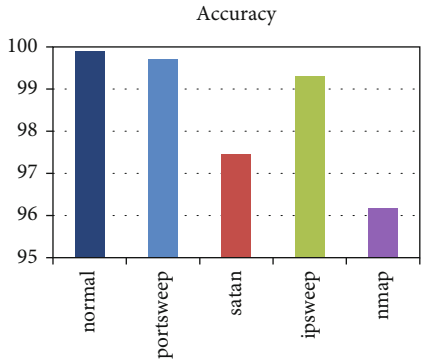

Figure 9: Accuracy of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

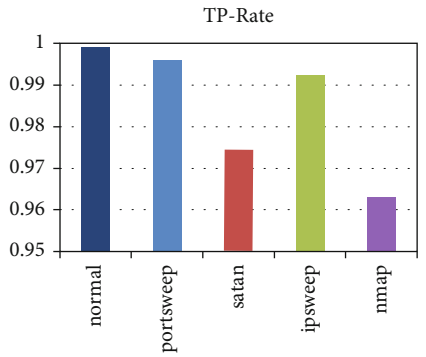

Figure 10: TP-Rate of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

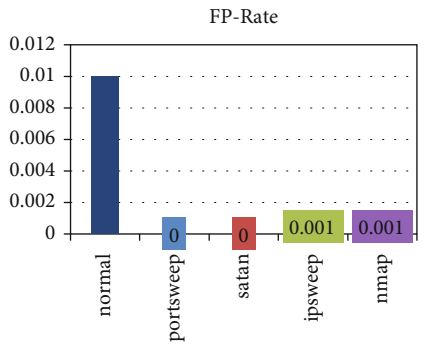

Figure 11: FP-Rate of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

dataset, the confusion matrix allows visualizing the algorithms' efficiency and is usually adapted to describe the classifier performance. Table 7 shows the confusion matrix.

Definition 2 (accuracy). The proportion of correct predictions of calculating the classification instances precisely is obtained from

$$\text{acc} = \frac{\text{TP} + \text{TN}}{(\text{TN} + \text{FN} + \text{TP} + \text{FP})}. \quad (5)$$

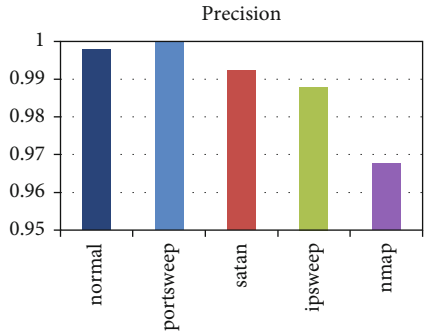

Figure 12: Precision of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

Definition 3 (error rate). The proportion of whole predictions done that are classified falsely: it is given by

$$\text{ERR} = 1 - \text{acc}. \quad (6)$$

Definition 4 (true positive). The intrusions are accurately classified as an attack by the intrusion detection systems. It is also called sensitivity, recall, or detection rate. It is obtained from

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (7)$$

Definition 5 (false positive). The usual patterns which are misclassified as attacks and calculated as

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \quad (8)$$

Definition 6 (true negative). The usual patterns that are precisely analyzed as normal and obtained from

$$\text{TNR} = 1 - \text{FPR}. \quad (9)$$

Definition 7 (false negative). The intrusions misclassified as normal and obtained from

$$\text{FNR} = 1 - \text{TPR}. \quad (10)$$

Definition 8 (precision). The behaviors that are exactly arrayed as attacks and given by

$$\text{Precision} = \frac{\text{TP}}{\text{FP} + \text{TP}}. \quad (11)$$

Definition 9 (F-measure). It is interpreted as the harmonic mean of recall and precision. Also known as F-score or F-value and calculated as

$$\text{FM} = 2 \times \frac{\text{precision x recall}}{\text{precision} + \text{recall}}. \quad (12)$$

Definition 10 (Matthews's correlation coefficient). Applied only in the binary intrusion detection system in which it

computes the observed and predicted values of binary classification. It is calculated by

$$\mathrm{MCC} = \frac{(\mathrm{TPxTN}) - (\mathrm{FPxFN})}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}. \tag{13}$$

Definition 11 (kappa statistic). It is applied to calculate the concurrence between observed and predicted values of the datasets, while the concurrence is corrected that occurs unexpectedly. It is calculated by

$$k = \frac{p_0 - p_e}{1 - p_e}, \tag{14}$$

where p_0 is the comparative noticed concurrence between the estimates and p_e is the assumed likelihood of possible concurrence.

Definition 12 (mean absolute error). It is the averaging of the magnitude of the distinctive error and the computing the standard of absolute errors. It is calculated as

$$\mathrm{MAE} = \frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{n}, \tag{15}$$

where p_1 is the value predicted on the test instances and a_1 is the actual value.

Definition 13 (root mean-squared error). The RMSE calculates the dissimilarities among observed values and predicted values of a model. It is given by

$$\mathrm{RMSE} = \sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}}, \tag{16}$$

where p_1 is the value predicted on the test instances and a_1 is the actual value.

Definition 14 (relative absolute error). The errors are normalized from the errors of simple predictors in which the average value is predicted. It is calculated as

$$\mathrm{RAE} = \frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{|a_1 - \bar{a}| + \cdots + |a_n - \bar{a}|}, \tag{17}$$

where p_1 is the value predicted on the test instances and a_1 is the actual value.

Definition 15 (root relative squared error). It normalizes the total squared error by division of the total squared error from the simple predictor. It is obtained from

$$\mathrm{RRSE} = \sqrt{\frac{(p^1 - a^1)^2 + \cdots + (p_n - a_n)^2}{(a^1 - \bar{a})^2 + \cdots + (a_n - \bar{a})^2}}, \tag{18}$$

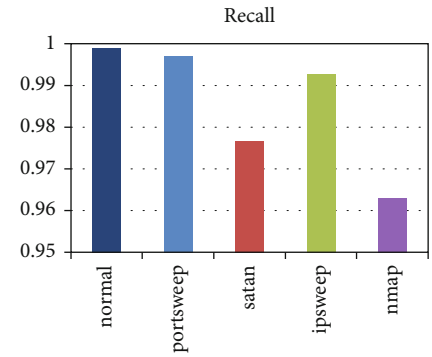

Figure 13: Recall of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

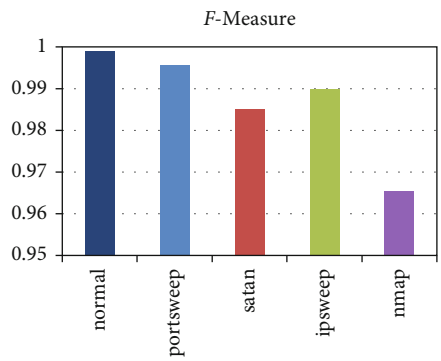

Figure 14: *F*-measure of the normal and attack types evaluated by the NID-Shield NIDS on Probe attack.

where p_1 is the value predicted on the test instances and a_1 is the actual value.

Definition 16 (AUC and ROC). ROC explains detection ratio changes in contrast to its internal verge to develop a high or low FPR. The larger the AUC values, the better the performance of the classifier.

5.1. Performance Evaluation with NSL-KDD 20% according to Attack Types. This section evaluates the DOS, Probe, U2R, and R2L, the types of attack of the NSL-KDD 20% dataset. The NID-Shield NIDS is assessed with J48 as an attribute selection approach, and finally, the selected attributes are appraised with a machine learning algorithm as a classifier.

5.1.1. Evaluation of DoS Attack with Normal and Attack Instances on Hybrid NID-Shield NIDS

(1) DoS Attack Evaluated with Hybrid NID-Shield NIDS. The following algorithms were applied for evaluation of feature subsets: attribute evaluator: CAPPER, attribute evaluator algorithm: J48, search method: best first, classifier evaluator: random forest.

The CAPPER evaluated subsets are as follows: 3, 4, 5, 6, 7, 8, 10, 12, 23, 24, 25, 29, 30, 36, 38, and 41.

In this section, the DoS attack is evaluated by the hybrid NID-Shield NIDS on the DoS attack dataset. The CAPPER

Table 11: U2R attack evaluated with hybrid NID-Shield NIDS approach.

(a)

Total instances	13,460
Correctly classified instances	13,460
Incorrectly classified instances	0
Execution time	1.86 seconds
Kappa measures	1
MAE	0.0003
RMSE	0.0066
RAE	11.2411%
RRSE	19.9452%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	normal
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	buffer_overflow
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	loadmodule
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	rootkit
Weighted Avg.	100%	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	

(c)

Confusion matrix			
a	*b*	*c*	*d*
13423	0	0	0
0	7	0	0
0	0	13	0
0	0	0	17

a—classified as normal, *b*—classified as buffer_overflow, *c*—classified as loadmodule, and *d*—classified as rootkit.

feature selector obtains the highest merit and accurate feature subsets from the combination of CFS and Wrapper approaches. Table 8 depicts the metrics of the DoS attack with its attack names classified individually. In the DoS, there are six attacks, namely, neptune, back, land, smurf, pod, and teardrop, and the normal instances. Figure 3 shows that the NID-Shield NIDS achieved an accuracy of 100% on the normal instances and 100% accuracy on the attack names such as land, back, teardrop, and neptune, while on the names of the attack such as pod and smurf, the NID-Shield NIDS achieves an accuracy of 94.7% and 99.6%, respectively. Overall, the weighted average of the accuracy of the normal and attack names is calculated; the NID-Shield NIDS achieves 100% accuracy on normal and all the attack types. Figure 4 shows the NID-Shield NIDS achieved a TP rate of 1.000 on the normal instances and a TP rate of 1.000 on attack names such as land, back, teardrop, and neptune, while on the attack names such as pod and smurf, the NID-Shield NIDS achieves a TP rate of 0.947 and 0.996, respectively. Overall, the weighted average of the TP rate is measured on normal and all attack names; the NID-Shield NIDS achieves 100% TP rate on normal and all attack names. Figure 5 depicts the FP rate evaluated by the NID-Shield NIDS on normal and attacks names, the NID-Shield NIDS achieves a 0.000 false-positive rate on all attack names, and an FP rate of 0.000 is achieved on the normal instance.

Figure 6 illustrates the precision of the NID-Shield NIDS which is assessed with normal and attack names. The NID-Shield NIDS obtained a precision of 1.000 on all normal instances and a precision of 1.000 on attack names such as neptune, back, land, and teardrop, while the precision of 0.998 and 0.973 is obtained on smurf and pod attack by the NID-Shield NIDS. Overall, a weighted average of 1.000 is obtained on precision for normal instances and attack names. Figure 7 depicts the recall appraised with NID-Shield NIDS on normal and attack names, the normal instances achieve a recall of 1.000 by the NID-Shield NIDS, and the attack names such as neptune, land, back, and teardrop achieve a recall of 1.000 by the NID-Shield NIDS, while the NID-Shield NIDS achieves a recall 0.996 and 0.947 on the attack names such as smurf and pod, respectively. Overall, the weighted average of recall is appraised for normal and all types of attack names; the NID-Shield NIDS obtained a recall of 1.000 on normal and attack names. Figure 8 depicts the *F*-measure of the NID-Shield NIDS appraised with the normal and attack names; the NID-Shield NIDS achieves an *F*

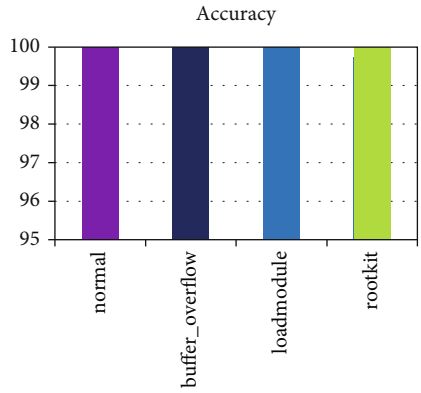

Figure 15: Accuracy of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

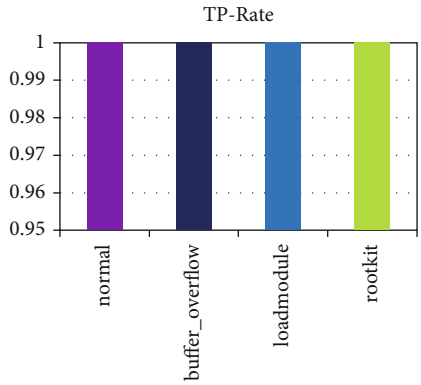

Figure 16: TP-Rate of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

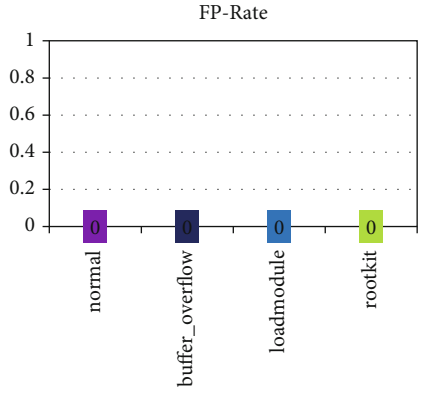

Figure 17: FP-rate of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

-measure of 1.000 on normal instances and attack names such as neptune, land, back, and teardrop; and the NID-Shield NIDS achieves an F-measure of 1.000, while for attack names such as smurf and pod, the NID-Shield NIDS obtains the F-measure of 0.997 and 0.960, respectively. Overall, the weighted average is appraised for F-measure on normal and all attack names; the NID-Shield NIDS obtained an F -measure of 1.000 on normal and attack names. For the

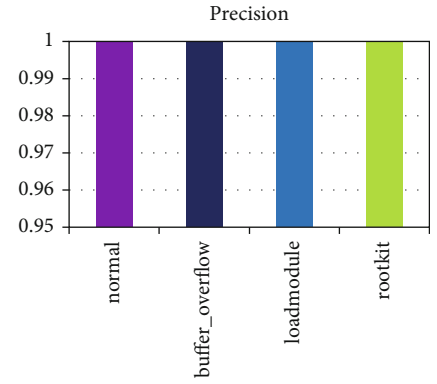

Figure 18: Precision of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

MCC, the NID-Shield NIDS is appraised with the normal and attack names; the NID-Shield NIDS achieves an MCC of 1.000 on normal instances and with attack names such as neptune, back, land, teardrop; and the NID-Shield NIDS achieves an MCC of 1, while for attack names such as smurf and pod, the NID-Shield NIDS obtains the MCC of 0.997 and 0.960, respectively. Overall, the weighted average of MCC is measured for normal and on all attack names; the NID-Shield NIDS obtained an MCC of 1.00, respectively.

For the ROC area, the NID-Shield NIDS achieves an overall 1.000 on normal and all attack names, respectively. For the PRC area, the NID-Shield NIDS obtained a 1.000 on normal instances, while for attack names such as land, back, teardrop, smurf, and neptune, the NID-Shield NIDS obtained a PRC area of 1.000 and for attack names called pod, the NID-Shield NIDS obtained a PRC area of 0.997. Overall, the weighted average is calculated for the PRC area on normal and all attack names; the NID-Shield NIDS achieved a PRC area of 1.000 on normal and all attack names.

5.1.2. Evaluation of Probe Attack with Normal and Attack Instances on Hybrid NID-Shield NIDS

(1) Probe Attack Evaluated with Hybrid NID-Shield NIDS. The following algorithms were applied for evaluation of feature subsets: attribute evaluator: CAPPER, attribute evaluator algorithm: J48, search method: best first, classifier evaluator: random forest.

The CAPPER evaluated subsets are as follows: 2, 3, 4, 12, 24, 27, 29, 31, 32, 35, 36, 37, and 40.

In this section, the Probe attack is evaluated with the hybrid NID-Shield NIDS on the Probe attack dataset. The stacking is applied for further improvement of the metrics. The stacked ensemble applies the random forest plus the Naive Bayes as a base classifier. Table 9 shows the Probe attack evaluation metrics without stacking ensemble, and Table 10 shows the evaluation of the Probe attack with a stacked ensemble. A considerable improvement in the FP rate is noticed when the NID-Shield NIDS is evaluated with a stacked ensemble. In the Probe attack, there are four attacks, namely, portsweep, satan, ipsweep, and nmap, and

the normal instances. Figure 9 shows that the NID-Shield NIDS achieved an accuracy of 99.90% on the normal instances and for the attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieved an accuracy of 99.7%, 97.7%, 99.3%, and 96.3%, respectively. Overall, the weighted average of accuracy is calculated on normal and attack names; the NID-Shield NIDS obtains 99.7% accuracy on normal and on all attack names.

Figure 10 depicts that the NID-Shield NIDS achieved a TP rate of 0.999 on the normal instances and attack names such as portsweep, satan, ipsweep, and nmap; the NID-Shield NIDS achieved an accuracy of 0.997, 0.977, 0.993, and 0.963, respectively. Overall, the weighted average of the TP rate is measured on normal and attack names; the NID-Shield NIDS achieves a TP rate of 0.997 on normal and all attack names. Figure 11 depicts the FP rate evaluated by the NID-Shield NIDS on normal and attacks names; the NID-Shield NIDS achieves a 0.000 false-positive rate on attack names such as portsweep and satan; and for other attack names like ipsweep and nmap, the NID-Shield NIDS obtains an FP rate of 0.001, respectively. For the normal instance, an FPR of 0.010 is achieved by the proposed NIDS. Figure 12 depicts that the precision of the NID-Shield NIDS is assessed with normal and attack names. The NID-Shield NIDS achieves a precision of 0.998 on normal instances, and for attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieved a precision of 1.000, 0.993, 0.987, and 0.967, respectively.

Figure 13 depicts the recall appraised with NID-Shield NIDS on normal and attack names; the normal instances achieve a recall of 0.999, while for the attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieves a recall of 0.997, 0.977, 0.993, and 0.963, respectively. Overall, a weighted average of the recall is appraised for normal and on all types of attack names; the NID-Shield NIDS obtains a recall of 0.997. Figure 14 illustrates the F-measure of the NID-Shield NIDS assessed with the normal and attack names, the NID-Shield NIDS achieves an F-measure of 0.999 on normal instances, and on attack name types such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieves an F-measure of 0.998, 0.985, 0.990, and 0.965, respectively. Overall, the weighted average is appraised for F-measure on normal and on all types of attack names; the NID-Shield NIDS obtained an F -measure of 0.997. For the MCC, the NID-Shield NIDS is appraised with the normal and attack names, the NID-Shield NIDS achieves an MCC of 0.990 on normal instances, and with attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieves an MCC of 0.998, 0.984, 0.990, and 0.964, respectively.

Overall, the weighted average of MCC is calculated for normal and on all attack names; the NID-Shield NIDS obtained an MCC of 0.990, respectively. The NID-Shield NIDS obtained a ROC of 0.999 on normal instances, and with attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieves a ROC area of 1.000, 0.995, 0.999, and 0.997, respectively. Overall, the weighted average of 0.999 is obtained by the NID-Shield NIDS in the ROC area. For the PRC area, the NID-Shield NIDS achieves

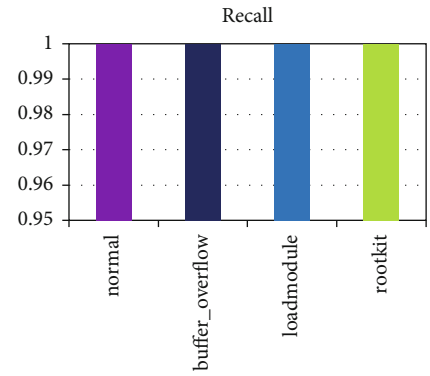

Figure 19: Recall of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

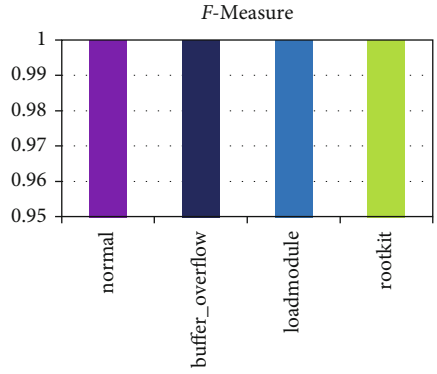

Figure 20: F-measure of the normal and attack types evaluated by the NID-Shield NIDS on U2R attack.

1.000 on normal instances, and with attack names such as portsweep, satan, ipsweep, and nmap, the NID-Shield NIDS achieves a PRC area of 1.000, 0.990, 0.994, and 0.986, respectively. Overall, a weighted average is appraised for the PRC area; the NID-Shield NIDS achieves a PRC area of 0.999, respectively.

5.1.3. Evaluation of U2R Attack with Normal and Attack Instances on Hybrid NID-Shield NIDS

(1) U2R Attack Evaluated with Hybrid NID-Shield NIDS. The following algorithms were applied for evaluation of feature subsets: attribute evaluator: CAPPER, attribute evaluator algorithm: J48, search method: best first, classifier evaluator: random forest.

The CAPPER evaluated subsets are as follows: 3, 4, 6, 9, 10, 13, 14, 17, 18, 33, and 36.

In this section, the U2R attack is evaluated by the hybrid NID-Shield NIDS on the U2R attack dataset. Table 11 shows the metrics of the U2R attack with the three attack names in the U2R attack, namely, buffer_overflow, loadmodule, and rootkit. Figure 15 shows that the NID-Shield NIDS achieved an accuracy of 100% on the normal instances and all attack types. Figure 16 shows the NID-Shield NIDS achieved a TP rate of 1.000 on the normal instances and all attack names. Figure 17 depicts the FP rate evaluated by the NID-Shield

Table 12: R2L attack evaluated with hybrid NID-Shield NIDS approach.

(a)

Total instances	13,658
Correctly classified instances	13,648
Incorrectly classified instances	10
Execution time	1.92 seconds
Kappa measures	0.9758
MAE	0.0005
RMSE	0.0118
RAE	7.3124%
RRSE	20.4253%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
100%		1.000	0.019	1.000	1.000	1.000	0.978	1.000	1.000	normal
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	ftp_write
100%		1.000	0.000	0.875	1.000	0.933	0.935	1.000	0.982	imap
100%		1.000	0.000	0.900	1.000	0.947	0.949	1.000	1.000	phf
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	multihop
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	warezmaster
97.4%		0.974	0.000	0.974	0.974	0.974	0.974	1.000	0.999	warezclient
91.7%		0.917	0.000	1.000	0.917	0.957	0.957	1.000	0.969	spy
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	gess_passwd
Weighted Avg.	99.99%	0.999	0.019	0.999	0.999	0.999	0.978	1.000	1.000	

(c)

Confusion matrix								
a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*
13444	0	0	0	0	0	0	0	0
0	5	0	0	0	0	0	0	0
0	0	7	0	0	0	0	0	0
0	0	0	9	0	0	0	0	0
0	0	0	0	6	0	0	0	0
0	0	0	0	0	12	0	0	0
0	0	0	0	0	0	150	0	0
0	0	1	0	2	0	0	11	0
0	0	0	1	0	2	4	0	4

a—classified as normal, *b*—classified as ftp_write, *c*—classified as imap, *d*—classified as phf, *e*—classified as multihop, *f*—classified as warezmaster, *g*—classified as warezclient, *h*—classified as spy, *i*—classified as guess_passwd.

NIDS on normal and attacks names; the NID-Shield NIDS achieves a 0.000 false-positive rate on all attack names and normal instances. Figure 18 depicts the precision of the NID-Shield NIDS assessed with normal and attack names. The NID-Shield NIDS achieves a precision of 1.000 on all normal instances and attack names. Figure 19 depicts the recall appraised with NID-Shield NIDS on normal and attack names the normal instances and attack names achieve a recall of 1.000. Figure 20 illustrates the *F*-measure with NID-Shield NIDS evaluated with the normal instances and attack names; the NID-Shield NIDS achieves an *F*-measure of 1.000 on normal instances and attack names.

For the MCC, the NID-Shield NIDS is appraised with the normal and attack names; the NID-Shield NIDS achieves an MCC of 1.000 on normal instances and attack names. For the ROC area and PRC area, the NID-Shield NIDS achieves an overall 1.000 on normal and all attack names, respectively.

5.1.4. Evaluation of R2L Attack with Normal and Attack Instances on Hybrid NID-Shield NIDS

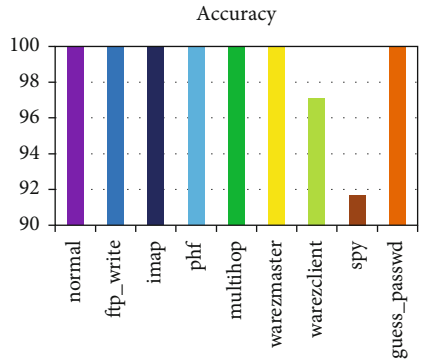

Figure 21: Accuracy of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

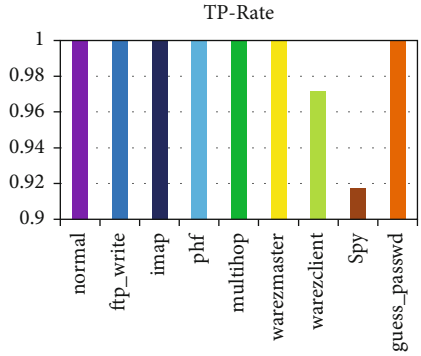

Figure 22: TP-rate of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

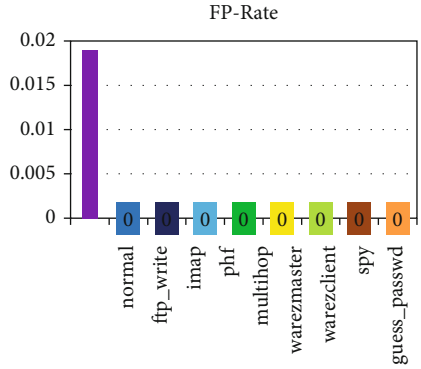

Figure 23: FP-rate of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

(1) R2L Attack Evaluated with Hybrid NID-Shield NIDS. The following algorithms were applied for evaluation of feature subsets: attribute evaluator: CAPPER, attribute evaluator algorithm: J48, search method: best first, classifier evaluator: random forest.

The CAPPER evaluated subsets are as follows: 4, 5, 6, 10, 11, 17, 22, 31, 32, 33, 36, and 38.

In this section, the R2L attack is evaluated by the hybrid NID-Shield NIDS approach on the R2L attack dataset. Table 12 shows the evaluation metrics of the R2L attack. In

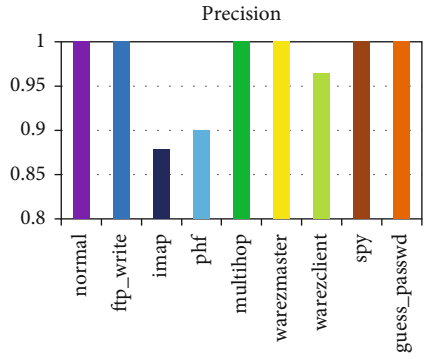

Figure 24: Precision of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

the R2L attack, there are eight attack names, namely, ftp_write, guess_passwd, phf, imap, warezmaster, multihop, warezclient, and spy, and normal instance. Figure 21 shows that the NID-Shield NIDS achieved an accuracy of 100% on the normal instances and for attack names such as ftp_write, guess_passwd, phf, imap, warezmaster, and multihop, the NID-Shield NIDS achieved an accuracy of 100%, respectively, while for the attack names such as warezclient and spy, the NID-Shield NIDS achieves an accuracy of 97.4% and 91.7%, respectively. Overall, the weighted average in terms of accuracy is appraised for the normal and attack names; the NID-Shield NIDS achieves 99.99% accuracy on normal and all attack names. Figure 22 depicts that the NID-Shield NIDS achieved a TP rate of 1.000 on the normal instances and for the attack names such as ftp_write, guess_passwd, phf, imap, warezmaster, and multihop, the NID-Shield NIDS achieved a TP rate of 1.000, respectively, while the attack names such as warezclient and spy, the NID-Shield NIDS achieved a TP rate of 0.974 and 0.917, respectively. Overall, the weighted average of the TP rate is measured on normal and an attack name; the NID-Shield NIDS achieves a TP rate of 0.999 on normal and all attack names. Figure 23 depicts the FP rate evaluated by the NID-Shield NIDS on normal and attacks names; the NID-Shield NIDS achieves a 0.000 false-positive rate on all attack names. For the normal instance, an FPR of 0.019 is achieved. Overall, a weighted average FP rate of 0.019 is obtained on normal and attack names. Figure 24 shows that the precision of the NID-Shield NIDS is evaluated with normal and attack names. The NID-Shield NIDS achieved a precision of 1.000 on normal instances, and for attack names such as guess_passwd, ftp_write, multihop, warezmaster, and spy, the NID-Shield NIDS achieved a precision of 1.000, respectively, while for attack names such as imap, phf, and warezclient, the NID-Shield NIDS obtained a precision of 0.875, 0.900, and 0.974, respectively. Overall, a weighted average precision of 0.999 is achieved on normal and attack names. Figure 25 depicts the recall appraised with NID-Shield NIDS on normal and attack names, the normal instances achieve a recall of 1.000, and for the attack names such as guess_passwd, ftp_write, imap, phf, multihop, and warezmaster, the NID-Shield NIDS achieves a recall of 1.000, respectively, while for attack names such as warezclient and spy, a recall of

0.974 and 0.917 is achieved by the proposed NIDS. Overall, the weighted average of the recall is appraised for normal and all types of attack names; the NID-Shield NIDS obtained a recall of 0.999, respectively. Figure 26 depicts the *F*-measure with the NID-Shield NIDS assessed with the normal and attack names, the NID-Shield NIDS achieves an *F* -measure of 1.000 on normal instances, and with attack names such as guess_passwd, ftp_write, multihop, and warezmaster, the NID-Shield NIDS achieves an *F*-measure of 1.000, respectively, while for attack names such as warezclient, spy, phf, and imap, the NID-Shield NIDS achieves an *F*-measure of 0.974, 0.957, 0.947, and 0.933, respectively.

Overall, the weighted average is calculated for *F*-measure on normal and all types of attack names; the NID-Shield NIDS obtained an *F*-measure of 0.999, respectively, on normal and attack names. For the MCC, the NID-Shield NIDS is appraised with the normal and attack names, the NID-Shield NIDS achieves an MCC of 0.978 on normal instances, and with attack names such as guess_passwd, ftp_write, multihop, and warezmaster, the NID-Shield NIDS achieves an MCC of 1.000, respectively, and on attack names such as imap, phf, warezclient, and spy, the NID-Shield NIDS obtained an MCC of 0.935, 0.949, 0.974, and 0.957, respectively. Overall, a weighted average is appraised for MCC; the NID-Shield NIDS achieves an MCC of 0.978 for normal and attacks names. For the ROC area, the NID-Shield NIDS achieves a 1.000 on normal instances and attack names. For the PRC area, the proposed NID-Shield NIDS achieves a 1.000 on normal instances, and with attack instances such as guess_passwd, ftp_write, phf, multihop, and warezmaster, the NID-Shield NIDS achieves a PRC area of 1.000, and for attack names such as warezclient, imap, and spy, the PRC area obtained is 0.999, 0.982, and 0.969, respectively. Overall, a weighted average PRC area of 1.000 is obtained by the NID-Shield NIDS for all normal instances and attack names.

5.1.5. Evaluation of UNSW-NB15 Dataset with Normal and Attack Instances on Hybrid NID-Shield NIDS. The following algorithms were applied for the evaluation of feature subsets: attribute evaluator: CAPPER, attribute evaluator algorithm: J48, search method: best first, classifier evaluator: random forest.

The CAPPER evaluated subsets for Reconnaissance attack are as follows: 2, 3, 7, 12, 27, 31, 36, 40, and 41.

The CAPPER evaluated subsets for Backdoor attack are as follows: 2, 3, 7, 10, 16, 27, 28, 39, and 40.

The CAPPER evaluated subsets for DoS attack are as follows: 3, 7, 8, 9, 10, 16, 31, 36, 40, and 41.

The CAPPER evaluated subsets for Exploits attack are as follows: 2, 3, 7, 8, 9, 10, 15, 17, 31, 36, and 40.

The CAPPER evaluated subsets for Analysis attack are as follows: 2, 3, 7, 8, 10, 12, 17, 28, 36, 40, and 41.

The CAPPER evaluated subsets for Fuzzers attack are as follows: 3, 7, 8, 9, 10, 12, 17, 32, and 33.

The CAPPER evaluated subsets for Worms attack are as follows: 3, 7, 8, 9, 10, 12, 17, 32, and 33.

The CAPPER evaluated subsets for Shellcode attack are as follows: 2, 3, 7, 8, 12, 27, 33, and 36.

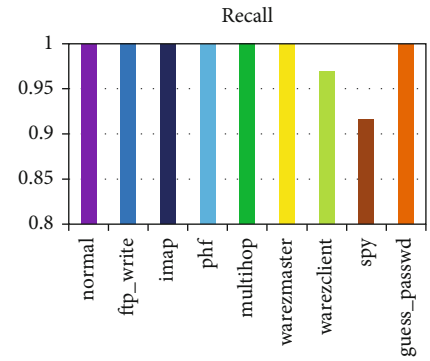

Figure 25: Recall of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

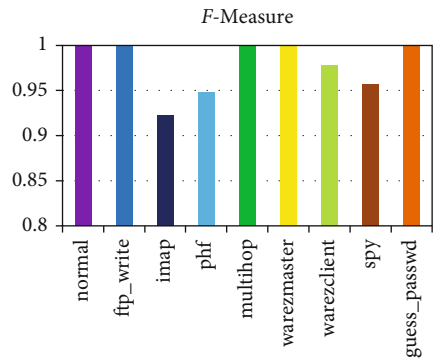

Figure 26: *F*-measure of the normal and attack types evaluated by the NID-Shield NIDS on R2L attack.

The CAPPER evaluated subsets for Generic attack are as follows: 2, 3, 7, 8, 9, 25, 31, 39, and 40.

In this section, the UNSW-NB15 dataset attack is evaluated by the hybrid NID-Shield NIDS approach on the UNSW-NB15 testing dataset. Table 13 illustrates the evaluation metrics of the UNSW-NB15 normal and attack instances. In the UNSW-NB15 dataset attack, there are nine attack names, namely, Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Worms, Generic, and Shellcode, and normal instances. Figure 27 shows that the NID-Shield NIDS achieved an accuracy of 100% on the normal instances and Worms attack while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved an accuracy of 99.71%, 99.45%, 98.70%, 99.10%, 90.14%, 99.20%, 99.70%, and 99.61%, respectively. Overall, the weighted average in terms of accuracy is appraised for the normal and an attack name; the NID-Shield NIDS achieves 99.89% accuracy on normal and all attack names. Figure 28 shows that the NID-Shield NIDS achieved a TP rate of 1 on the normal instances and Worms attack while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved a TP rate of 0.997, 0.994, 0.987, 0.991, 0.901, 0.992, 0.997, and 0.996, respectively. Overall, the weighted average in terms of TP rate is appraised for the normal and attack names; the NID-Shield NIDS achieved an accuracy of 0.998

Table 13: UNSW-NB15 dataset evaluated with hybrid NID-Shield NIDS approach.

(a)

Total instances	1,75,341
Correctly classified instances	1, 75,183 (99.91%)
Incorrectly classified instances	158
Execution time	318.15 seconds
Kappa measures	0.9835
MAE	0.0007
RMSE	0.0121
RAE	6.3124%
RRSE	18.4253%

(b)

Accuracy		TP rate	FP rate	Precision	Recall	*F*-measure	MCC	ROC area	PRC area	Class
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Normal
99.45%		0.994	0.007	0.996	0.998	0.997	0.995	0.999	0.999	Reconnaissance
99.71%		0.997	0.006	0.998	0.999	0.999	0.999	1.000	1.000	Backdoor
99.10%		0.991	0.007	0.995	0.991	0.997	0.997	1.000	1.000	DoS
98.70%		0.987	0.008	0.993	0.982	0.982	0.993	0.994	0.993	Exploits
99.20%		0.992	0.007	0.989	0.993	0.996	0.998	1.000	1.000	Analysis
90.14%		0.901	0.012	0.917	0.941	0.962	0.972	0.971	0.978	Fuzzers
100%		1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Worms
99.61%		0.996	0.006	0.997	0.999	0.997	0.997	1.000	1.000	Shellcode
99.70%		0.997	0.004	0.998	0.998	0.999	0.997	1.000	1.000	Generic
Weighted Avg.	99.89%	0.998	0.006	0.999	0.998	0.997	0.992	1.000	1.000	

(c)

Confusion matrix									
a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*
56000	0	0	0	0	0	0	0	0	0
0	10488	0	0	0	0	0	0	0	0
0	0	1740	0	0	0	0	0	0	2
0	0	0	12260	0	0	0	0	0	0
0	2	0	0	33383	0	0	0	0	3
0	0	0	0	0	1993	0	0	0	0
0	0	3	3	7	0	18177	0	4	8
0	0	0	0	0	3	0	130	0	0
0	1	0	0	3	0	7	0	1129	0
0	0	3	1	0	4	0	0	0	39987

a—classified as Normal, *b*—classified as Reconnaissance, *c*—classified as Backdoor, *d*—classified as DoS, *e*—classified as Exploits, *f*—classified as Analysis, *g*—classified as Fuzzers, *h*—classified as Worms, *i*—classified as Shellcode, and *j*—classified as Generic.

on normal and all attack names. Figure 29 shows that the NID-Shield NIDS achieved an FP rate of 0.000 on the normal instances and Worms attack while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved an FP rate of 0.006, 0.007, 0.008, 0.007, 0.012, 0.007, 0.004, and 0.006, respectively. Overall, the weighted average in terms of FP rate is appraised for the normal and attack names; the NID-Shield NIDS achieved an FP rate of 0.006 on normal and all attack names. Figure 30 shows that the precision of the NID-Shield NIDS is evaluated with normal instances and attack names. The NID-Shield NIDS achieved a precision of 1.000 on the normal instances and Worms attack while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved a precision of 0.998, 0.996, 0.993, 0.995, 0.917, 0.989, 0.998, and 0.997, respectively. Overall, the weighted average in terms of precision is

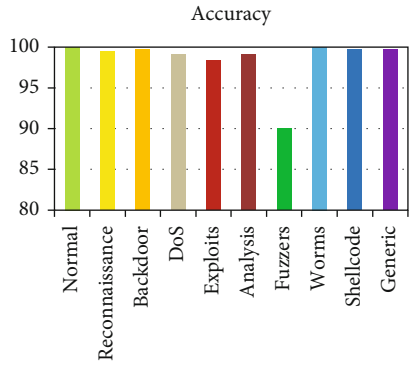

Figure 27: Accuracy of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

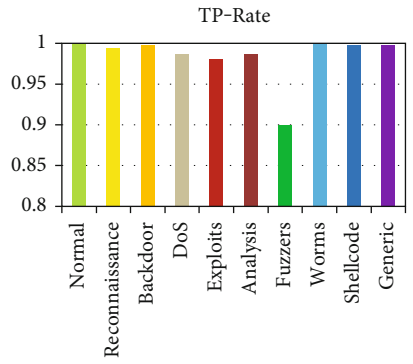

Figure 28: TP-Rate of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

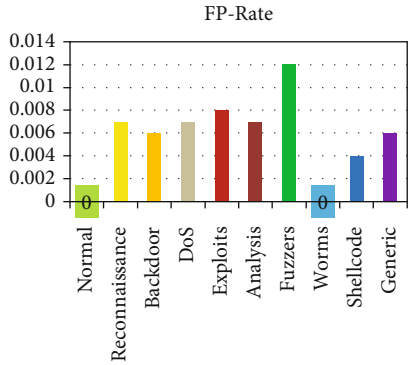

Figure 29: FP-rate of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

appraised for the normal and attack names; the NID-Shield NIDS achieved a precision of 0.999 on normal and all attack names. Figure 31 depicts the recall appraised with NID-Shield NIDS on normal and attack names, the NID-Shield NIDS achieved a recall of 1.000 on the normal instances and Worms attack, while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved a recall of 0.999, 0.998, 0.982, 0.991, 0.941, 0.993, 0.998, and 0.999,

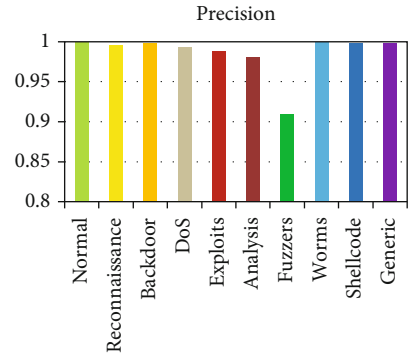

Figure 30: Precision of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

respectively. Overall, the weighted average in terms of recall is appraised for the normal and attack names; the NID-Shield NIDS achieved a recall of 0.998 on normal and all attack names. Figure 32 shows the F-measure of the NID-Shield NIDS evaluated with the normal and attack names; the NID-Shield NIDS achieved an F-measure of 1.000 on the normal instances and Worms attack, while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved an F-measure of 0.999, 0.997, 0.982, 0.997, 0.962, 0.996, 0.999, and 0.997, respectively. Overall, the weighted average in terms of F-measure is appraised for the normal and attack names; the NID-Shield NIDS achieved an F-measure of 0.997 on normal and all attack names.

For the MCC, the NID-Shield NIDS is appraised with the normal and attack names; the NID-Shield NIDS achieved an MCC of 1.000 on the normal instances and Worms attack, while for other attacks such as Backdoor, Reconnaissance, Exploits, DoS, Fuzzers, Analysis, Generic, and Shellcode, the NID-Shield NIDS achieved an MCC of 0.999, 0.995, 0.993, 0.997, 0.972, 0.998, 0.997, and 0.997, respectively. Overall, the weighted average in terms of MCC is appraised for the normal and attack names; the NID-Shield NIDS achieved an MCC of 0.992 on normal and all attack names. The NID-Shield NIDS achieves a ROC and PRC area of 1.000 on normal and attack instances.

Table 14 shows the hybrid NID-Shield NIDS with the existing approaches in this literature. The details of the existing approaches are shown in Table 1. For the evaluation of the hybrid NID-Shield NIDS approach, the proposed hybrid NID-Shield NIDS evaluates the attack names on the UNSW-NB15 dataset, and overall performance metrics are considered such as Probe, DoS, R2L, and U2R, and attack names on the NSL-KDD 20% dataset. The NID-Shield NIDS achieves a 99.89% on the UNSW-NB15 dataset and overall accuracy of 99.90% on the NSL-KDD dataset, which is the highest among all other approaches. When the TP rate is calculated, overall, the NID-Shield NIDS obtained a TPR of 0.999 on the NSL-KDD 20% dataset and 0.9998 on the UNSW-NB15 dataset which is the best among all other approaches. When FPR is comprehensively evaluated, the literature proposed by Cavusoglu achieves an overall best FPR of 0.000035 and the NID-Shield NIDS

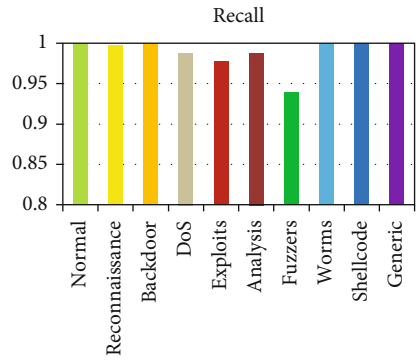

Figure 31: Recall of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

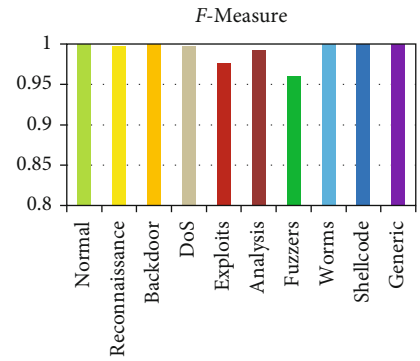

Figure 32: F-measure of normal and attack types evaluated by the NID-Shield NIDS on UNSW-NB15 dataset.

achieves a second-best FPR of 0.007 and 0.006 on NSL-KDD 20% and UNSW-NB15 datasets. The TNR is evaluated globally; the NID-Shield NIDS achieved the true negative rate of 0.993 on both datasets which are the highest among all other approaches.

The literature proposed by Arif et al. achieves the highest precision of 0.9998, and the NID-Shield NIDS achieves the second-best precision of 0.9990 on NSL-KDD 20% dataset. The NID-Shield NIDS achieves a recall of 0.999 and 0.989 on NSL-KDD 20% and UNSW-NB 15 datasets which is the highest among all other approaches. The F-measure is evaluated comprehensively; the NID-Shield NIDS achieves the highest F-measure of 0.997 and 0.999 on UNSW-NB15 and NSL-KDD 20% datasets. When MCC is appraised globally, the NID-Shield NIDS achieves the best MCC of 0.992 on both datasets which are overall best among all approaches. The ROC and PRC area of the NID-Shield NIDS is evaluated comprehensively; the NID-Shield NIDS achieves the best ROC and PRC area on both datasets which are the best among all other approaches. When the execution time is evaluated, the literature proposed by Venkataraman and Selvaraj achieves the lowest execution time of 0.23 seconds, followed by an execution time of 10.79 seconds by the literature proposed by Suad and Fadl and execution time of 10.62 seconds by the literature proposed by Cavusoglu. The NID-Shield NIDS achieves an execution time of 318.15 and 13.785 seconds on the UNSW-NB15 and NSL-KDD 20% datasets. Overall, the NID-Shield NIDS achieves the highest measures in terms of accuracy, TP rate, TNR, F-measure, MCC, recall, PRC, and ROC area on both the datasets.

For the insight of the discussion of the results, CAPPER and the random forest is the primary speculation for obtaining high metrics on both datasets. CAPPER is an effective feature subset selection technique that obtains accurate and high merit feature subsets from CFS and Wrapper methods. CFS searches the space of the feature subset by employing the best first search method and calculates the feature-class correlations and feature-feature correlations by applying the approaches based on conditional entropy. The high merit subset is measured by equation (3), which greatly aids in dimensionality reduction of both the testing and training data. In Wrapper, the feature subset search is executed by the best first search approach. The best first search at each iteration creates its successors having a node with maximal estimation accuracy. The induction algorithm is employed as a feature subset selection approach. The induction algorithm is run k times, and the training set uses the $k-1$ partitions, while the test set employs other partitions. Five fold cross-validation techniques are applied as the subset evaluation approach. The estimation of the accuracy is obtained by equation (4). To obtain the accurate and finest feature subsets, the machine learning approaches are applied by the Wrapper approach. The accurate and high merit feature subsets obtained by CFS and Wrapper are then combined to obtain the reduced dataset.

The random forest is considered as the most efficient classifier as compared to other classifiers. The foremost reason for obtaining the high accuracy is applying the bagging by the random forest. Employing bagging has mainly two benefits. Firstly, the accuracy is increased each time the random features are enforced. Secondly, estimation of the generalization error containing the ensemble tree combination and the correlations and its intensity appraisal is provided by the bagging. The assessment is carried out-of-bag. The main approach behind the out-of-bag estimation is the incorporation of nearly one-third of classifiers from the continuing prevailing sequence. Whenever the statistic of the sequence is incremented, the rate of error declines. Therefore, the contemporary error rate can be augmented by out-of-bag estimation; hence, it is necessary to pass on from the area where the merging of the error occurs. In the cross-validation, there is a high probability of the existence of bias; also, the degree of extent of the bias is unfamiliar, whereas the out-of-bag estimation is free from bias. The random forest applies two-thirds of the data and for testing one-third of the data from training data, to grow the tree. Out-of-bag data is simply the one-third data from the training data. Pruning is not performed by the random forest and thus aids in fast and high performance. Moreover, having the multiple-tree construction, the random forest performs reasonably well with an additional tree framework and it achieves a higher performance rather than any other decision tree method.

Table 14: Comparison of the hybrid NID-Shield NIDS with existing approaches in this study.

	Accuracy	TPR	FPR	TNR	Precision	Recall	F -measure	MCC	ROC	PRC	Time (seconds)
Proposed approach with NSL-KDD 20% dataset	99.90	0.9990	0.007	0.993	0.999	0.999	0.999	0.992	1.000	1.000	13.785
Proposed approach with UNSW-NB15 dataset	99.89	0.9989	0.006	0.993	0.999	0.989	0.997	0.992	1.000	1.000	318.15
Neha et al. [26]	99.05%	0.994	0.014	_	0.991	_	_	_	_	_	_
Arif et al. [28]	96.65%	0.9271	0.136	_	0.9998	_	_	_	_	_	_
Ahmed et al. [29]	_	0.9577	_	0.975	0.5662	_	_	_	_	_	3112.87
Tirtharaj [30]	_	0.9526	_	_	_	_	_	_	_	_	103.70
Yao et al. [31]	99.20%	0.6699	_	_	0.9655	0.967	_	_	_	_	_
Suad et al. [32]	_	_	_	_	_	_	_	_	0.995	0.962	10.79
Ijaz et al. [33]	99.8% (DoS)	_	0.17 (DoS)	_	_	_	_	_	_	_	_
Alauthaman et al. [34]	99.20%	0.9908	0.75	_	_	_	_	_	_	_	_
Venkataraman and Selvaraj [35]	83.83%	_	_	_	_	_	_	_	_	_	0.23
Kumar and Kumar [36]	99%	_	_	_	_	_	_	_	_	_	_
Cavusoglu [37]	99.86% (overall)	0.9292 (overall)	0.000035 (overall)	_	_	_	0.706 (overall)	0.954 (overall)	_	_	10.62 (overall)
Saxena et al. [38]	98.1%	0.7	_	_	_	_	_	_	_	_	_
Kambattan and Rajkumar [39]	99.45%	_	_	_	_	_	_	_	_	_	_
Kar et al. [40]	93.95%	0.955	0.1034	_	_	_	_	_	_	_	_
Mishra et al. [41]	92.12%	0.971	_	_	_	_	_	_	_	_	_
Dutta et al. [42]	91.29%	_	_	_	92.08%	90.64%	0.91	_	_	_	_
Latah and Toker [43]	84.29%	_	0.063	_	_	77.18%	84.83%	_	_	_	_
Sumaiya Thaseen et al. [44]	98.45%, on NSL-KDD dataset and 96.44% on UNSW-NB15 dataset	0.9294 on NSL-KDD dataset and 0.504 on UNSW-NB15 dataset	_	0.9438 on NSL-KDD dataset and 0.984 on UNSW-NB15 dataset	_	_	_	_	_	_	500 on NSL-KDD dataset and 1023 on UNSW-NB15 dataset
Safaldin et al. [45]	96%	0.96	0.03	_	_	_	_	_	_	_	69.6 h
Vallathan et al. [46]	98.4%	0.9602	_	0.998	_	_	_	_	_	_	_

6. Conclusion and Future Work

An efficient hybrid NID-Shield NIDS is proposed in this literature. Moreover, CAPPER an effective hybrid feature selection method is applied for accurate and highly merit feature subsets. The proposed hybrid NID-Shield NIDS classifies the UNSW-NB15 and NSL-KDD 20% dataset according to attack types and attack names. Distinct attacks may have peculiar connections as some of the attacks such as R2L and U2R may have very few N/W connections, while other attacks such as Probe and DoS may have a large number of N/W connections or can be a combination of any of them. Moreover, the hybrid NID-Shield NIDS calculates the performance metrics of attack names found in the NSL-KDD 20% dataset (DoS, Probe, U2R, and R2L) and UNSW-NB15 dataset individually. This approach further helps us to know the metrics of individual attack names and the vulnerability of the attack on the individual network. From the concluding results, it is noticed that the proposed hybrid NID-Shield NIDS with an effective CAPPER hybrid feature selection approach can improve various performance metrics on the network intrusions.

When Tables 8–14 are examined, the proposed hybrid NID-Shield NIDS obtains a comprehensive excellent performance in terms of various performance metrics on all attack types. The hybrid NID-Shield NIDS with its various parameters is investigated with existing literature studies; it has been found that the hybrid NID-Shield NIDS is the most efficient of all approaches found in the existing literature studies. In future work, we will consider applying the hybrid NID-Shield NIDS to fog computing.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

[1] L. Hung-Jen and C.-h. R. Lin, "Intrusion detection system a comprehensive review," *Journal of network and applications*, vol. 36, no. 1, pp. 16–24, 2013.

[2] H. L. Motoda and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, vol. 454, Springer, 1998.

[3] J. P. Anderson, *Computer Security Threat Monitoring and Surveillance*, Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, 1980.

[4] T. F. Lunt, J. van Horne, and L. Halme, "Automated analysis of computer system audit trails," in *Proceedings of the Ninth DOE Computer Security Group Conference*, Las Vegas, Nevada, 1986.

[5] H. S. Javitz, A. Valdes, D. E. Denning, and P. G. Neumann, *Analytical Techniques Development for a Statistical Intrusion Detection System (SIDS) Based on Accounting Records*, Technical report, SRI International, Menlo Park, California, 1986.

[6] D. Anderson, T. Frivold, and A. Valdes, *Next-Generation Intrusion Detection Expert System (NIDES). A Summary*, SRI International Computer Science Laboratory Technical Report SRI-CSL-95-07, 1995.

[7] L. D. S. Silva, A. C. Santos, T. D. Mancilha, J. D. Silva, and A. Montes, "Detecting attack signatures in the real network traffic with ANNIDA," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2326–2333, 2008.

[8] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: existing solutions and latest technological trends," *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, 2007.

[9] C. Manikopoulos and S. Papavassiliou, "Network intrusion and fault detection. A statistical anomaly approach," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 76–82, 2002.

[10] P. Fournier-Viger, C. W. Lin, A. Gomariz et al., "The SPMF open-source data mining library version 2," in *Joint European conference on machine learning and knowledge discovery in databases*pp. 36–40, Cham, Riva del Garda, Italy, 2016.

[11] P. Fournier-Viger, J. C.-W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas, "A survey of sequential pattern mining," *Data Science and Pattern Recognition*, vol. 1, no. 1, pp. 54–77, 2017.

[12] A. Smola and S. V. N. Vishwanathan, *Introduction to Machine Learning*, Cambridge University Press, 2008, ISBN-10: 0521825830.

[13] Z. Xiaojin, *Semi-Supervised Learning Literature Survey*, vol. 2, Computer Science, University of Wisconsin, Madison, 2008.

[14] S. Mukkamala, A. H. Sung, and A. Abraham, "Modeling intrusion detection systems using linear genetic programming approach," in *The 17th international conference on industrial & engineering applications of artificial intelligence and expert systems, innovations in applied artificial intelligence*, pp. 633–642, Berlin, Heidelberg, 2004.

[15] J. Pearl, "Bayesian networks. A model of self-activated memory for evidential reasoning," in *Proceedings of the 7th Conference of the Cognitive Science Society, University of California*, pp. 329–334, Irvine, CA, 2009.

[16] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression (PDF)," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[17] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, University of California Press, 1967.

[18] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *The annals of mathematical statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[19] T. Kohonen, "The self-organizing map," *Proceedings of IEEE*, pp. 1464–1480, 1990.

[20] M. Mohammed, M. B. Khan, and E. B. Bashier, *Machine Learning Algorithms and Applications*, CRC press Taylor and Francis Group, 2016, ISBN-10: 1498705383.

[21] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, vol. 2, Springer, 2009, ISBN 978-0-387-84858-7.

[22] M. Dash and H. Liu, "Feature selection for classification," *Intelligent data analysis*, vol. 1, no. 1-4, pp. 131–156, 1997.

[23] H. Liu and L. Yu, "Towards integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491–502, 2005.

[24] R. Heady, G. Luger, A. Maccabe, and M. Servilla, *The Architecture of Network Level Intrusion Detection System*, Technical report CS90-20, Department of computer science, University of New Mexico, 1990.

[25] E. Carter, *CCSP Self-Study: Cisco Secure Intrusion Detection System (CSIDS)*, Cisco Press, 2nd edition, 2004, ISBN-10: 9781587051449.

[26] A. Neha and S. Shailendra, "An IWD-based feature selection method for intrusion detection system," *Soft computing*, vol. 22, pp. 4407–4416, 2017.

[27] H. Shah-Hosseini, "Optimization with the nature-inspired intelligent water drops algorithm," in *Evolutionary Computation*, W. P. Dos Santos, Ed., pp. 298–320, I-Tech, Vienna, 2009, ISBN 978-953-307-008-7.

[28] J. Arif, F. Malik, and K. Aslam, "A hybrid technique using binary particle swarm optimization and decision tree pruning for network intrusion detection," *Cluster Computing*, vol. 21, pp. 667–680, 2017.

[29] I. Ahmed, L. Saleh, M. Fatma, and L. Talaat, "A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers," *Artificial Intelligence Review*, vol. 51, pp. 403–443, 2017.

[30] D. Tirtharaj, "A study on intrusion detection using neural networks trained with evolutionary algorithms," *Soft Computing*, vol. 21, pp. 2687–2700, 2017.

[31] Y. Haipeng and W. Qiyi, "An intrusion detection framework based on hybrid multi-level data mining," *International Journal of Parallel Programming*, vol. 47, pp. 740–758, 2017.

[32] M. Suad and M. Fadl, "Intrusion detection model using machine learning algorithm on Big Data environment," *Journal of big data*, vol. 5, pp. 1–12, 2018.

[33] S. Ijaz, F. A. Hashmi, S. Asghar, and M. Alam, "Vector based genetic algorithm to optimize predictive analysis in network security," *Applied intelligence*, vol. 48, no. 5, pp. 1086–1096, 2018.

[34] A. Mohammad and A. Nauman, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks," *Neural Computing & Applications*, vol. 29, pp. 991–1004, 2018.

[35] V. Sivakumar and S. Rajalakshmi, "Optimal and novel hybrid feature selection framework for effective data classification," in *Advances in Systems, Control and Automation*, pp. 499–514, Springer, Singapore, 2018.

[36] K. Neeraj and K. Upendra, "Knowledge computational intelligence in network intrusion detection systems," in *Knowledge Computing and Its Applications*, pp. 161–176, Springer, Singapore, 2018.

[37] C. Unal, "A new hybrid approach for intrusion detection using machine learning methods," *Applied Intelligence*, vol. 49, pp. 2735–2761, 2019.

[38] S. Akash and S. Khushboo, "Hybrid technique based on DBSCAN for selection of improved features for intrusion detection system," in *Emerging Trends in Expert Applications and Security*, pp. 365–377, Springer, Singapore, 2019.

[39] K. Rajesh and R. Manimegalai, "An effective intrusion detection system using flawless feature selection, outlier detection and classification," in *Progress in Advanced Computing and Intelligent Engineering*, pp. 203–213, Springer, Singapore, 2019.

[40] P. Kar, S. Banerjee, K. C. Mondal, G. Mahapatra, and S. Chattopadhyay, "A hybrid intrusion detection system for hierarchical filtration of anomalies," in *Information and Communication Technology for Intelligent Systems*, vol. 106, pp. 417–426, Springer, Singapore, 2019.

[41] S. Mishra, C. Mahanty, S. Dash, and B. K. Mishra, "Implementation of BFS-NB hybrid model in intrusion detection system, recent developments in machine learning and data analytics," in *Recent Developments in Machine Learning and Data Analytics*, vol. 740, pp. 167–175, Springer, Singapore, 2019.

[42] V. Dutta, M. Choras, R. Kozik, and M. Pawlicki, "Hybrid model for improving the classification effectiveness on network intrusion detection system," in *Conference on Complex, Intelligent, and Software Intensive Systems*, Cham, 2020.

[43] M. Latah and L. Toker, "An efficient flow-based multi-level hybrid intrusion detection system for software-defined networks," *CCF Transactions on Networking*, vol. 3, pp. 26–271, 2020.

[44] I. Sumaiya Thaseen, J. Saira Banu, K. Lavanya, M. Rukunuddin Ghalib, and K. Abhishek, "An integrated intrusion detection system using correlation-based attribute selection and artificial neural network," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 2, article e4014, 2021.

[45] M. Safaldin, M. Qtair, and L. Abualigah, "Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 1559–1576, 2021.

[46] G. Vallathan, A. John, and C. Thirumalai, "Suspicious activity detection using deep learning in secure assisted living IoT environments," *The Journal of Supercomputing*, vol. 77, pp. 3242–3260, 2021.

[47] Hackerpocalypse-cybercrime report, *In Cybersecurity Ventures*, 2016.

[48] A. AlEroud, G. Karabatis, P. Sharma, and P. He, "Context and semantics for detection of cyber attacks," *International Journal of Information and Computer Security*, vol. 6, no. 1, pp. 63–92, 2014.

[49] A. AlEroud and G. Karabatis, "Toward zero-day attack identification using linear data transformation techniques," in *IEEE 7th international conference on software security and reliability (SERE'13)*, pp. 159–168, Washington, D.C., 2013.

[50] S. Axelsson, "Intrusion detection systems: a survey and taxonomy," 2000.

[51] R. M. Snort, "Lightweight intrusion detection for networks," in *Proceedings of thirteenth USENIX conference on system administration, (LISA '99)*, pp. 229–238, Seattle, Washington, USA, 1999.

[52] J. Cannady, "Artificial neural networks for misuse detection," in *National information systems security conference*, vol. 26, pp. 368–381, Arlington, Virginia, United States, 1998.

[53] R. C. Quinlan, *4.5: Programs for Machine Learning*, Morgan Kaufmann publishers Inc, San Francisco, 1993, ISBN: 978-1-55860-238-0.

[54] T. Denoeux, "A k-nearest neighbor classification rule based on Dempster-Shafer theory," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 5, pp. 804–813, 1995.

[55] C. Cortes and V. N. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[56] E. Alpaydin, *Introduction to Machine Learning*, MIT Press, Cambridge, 2014, ISBN: 978-0-262-028189.

[57] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[58] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth & Brooks/Cole Advanced books & Software, Monterey, CA, 1984.

[59] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[60] L. Breiman, *Out-of-Bag Estimation*, Technical Report, Dept. of statistics, University of California, Berkeley, 1996.

[61] D. Mladenic and M. Grobelnik, "Feature selection for unbalanced class distribution and naive bayes," in *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, vol. 99, pp. 258–267, Bled, Slovenia, 1999.

[62] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," *Machine learning proceedings*, , pp. 121–129, 1994.

[63] P. Langley and S. Sage, "Scaling to domains with irrelevant features," in *Computational Learning Theory and Natural Learning Systems*, R. Greiner, Ed., vol. 4, MIT Press, 1994.

[64] P. Domingos and M. Pazzani, "Beyond independence: conditions for the optimality of the simple Bayesian classifier," in *Machine Learning: Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 105–112, San Francisco, CA, 1996.

[65] R. C. Quinlan, *4.5: Programs for Machine Learning*, Morgan Kaufmann publishers Inc, San Francisco, 1993.

[66] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 569–575, 2010.

[67] N. Thomas Rincy and R. Gupta, "An efficient feature subset selection approach for machine learning," *Multimedia tools and applications*, vol. 80, pp. 12737–12830, 2021.

[68] Z. H. Zhou, *Ensemble Methods Foundation and Algorithms*, CRC press: Taylor and Francis Group, 2012.

[69] P. Smyth and D. Wolpert, "Stacked density estimation," in *Advances in Neural Information Processing Systems*, pp. 668–674, MIT Press, Cambridge, MA, 1998.

[70] S. Samdani and S. Shukla, "A novel technique for converting nominal attributes to numeric attributes for intrusion detection," in *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–5, Delhi, 2017.

[71] A. Binbusayyis and T. Vaiyapuri, "Comprehensive analysis and recommendation of feature evaluation measures for intrusion detection," *Heliyon*, vol. 6, no. 7, 2020.

[72] S. Revathi and A. Malathi, "A detailed analysis on NSL-KDD dataset using various machine learning," *International Journal of Engineering Research & Technology*, vol. 2, no. 12, pp. 1848–1853, 2013.

[73] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP-'99 data set," in *Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications*, Ottawa, Canada, 2009.

[74] P. Kavitha and M. Usha, "Anomaly based intrusion detection in WLAN using discrimination algorithm combined with Naïve Bayesian classifier," *Journal of Theoretical and Applied Information Technology*, vol. 62, no. 1, pp. 77–84, 2014.

[75] N. Moustafa and J. Slay, "UNSW-NB15 a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Military Communications and Information Systems Conference (MilCIS)*, pp. 1–6, Canberra, 2015.

[76] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, San Francisco, 2nd edition, 2005.

[77] H. Hanan and B. David, "A taxonomy and survey of intrusion detection system design techniques, network threats and datasets," pp. 1–35, 2018, https://arxiv.org/abs/1806.03517.