Machine Learning and Deep Learning Approaches for Intrusion Detection: A Comparative Study

G. Prethija and Jeevaa Katiravan

Abstract Intrusion detection is a major challenge for security experts in the cyber world. Traditional IDS failed to detect complex and unknown cyber-attacks. Machine learning has become a vibrant technology for cybersecurity. There exists several machine learning algorithms to detect intrusion. Most classifiers are well suited to detect the attacks. However, improving accuracy and detecting unknown attacks in existing IDSs is a great challenge. Therefore, the detailed comparative study of various machine learning approaches such as artificial neural networks, support vector machine, decision tree, and hybrid classifiers used by researchers for intrusion detection are done. Deep learning is an emerging approach which suits well for large data. Deep learning techniques find optimal feature set and classify low-frequency attacks better than other techniques. This study also summarizes literatures in deep learning approaches such as deep auto-encoder, Boltzmann machine, recurrent neural networks, convolutional neural networks, and deep neural networks. Moreover, the datasets used in various literatures and the analysis of deep learning approaches based on the performance metrics are also done. Future directions to detect intrusion are also provided. This study in fact will be helpful to develop IDS based on artificial intelligence approaches such as machine learning and deep learning.

Keywords Machine learning · Intrusion detection · Feature selection · Deep learning · Cyber security · Classifier

1 Introduction

Cyber security threats are increasing day by day. So, there is a high demand for intrusion detection. Recently, more cyber threats are reported. Even though the technology grows, the hackers are still increasing, and it is a big challenge for cyber

G. Prethija (✉)
Department of Information Technology, Velammal Engineering College, Chennai, India

J. Katiravan
Department of Computer Science and Engineering, Velammal Engineering College, Chennai, India

G. Ranganathan et al. (eds.), *Inventive Communication and Computational Technologies*,
Lecture Notes in Networks and Systems 311,
https://doi.org/10.1007/978-981-16-5529-6_7

security experts. The National Cyber Security Center (NCSC) is investigating a large-scale domain name system (DNS) hijacking campaign that has targeted Middle East, Europe, and the US countries which affected government and commercial organizations worldwide in January 2019 [1]. Data breaches, crypto jacking, and Ransomware attacks and threats to connected devices are few cyber security dangers reported. Norton's Cyber security facts and statistics reports that WannaCry Ransomware attack affected nearly tens of thousands of computers across the world. Machine learning plays a predominant role in cyber security field for detecting real threats in an enterprise by security analysts.

An intrusion detection system (IDS) checks for malicious activity among all incoming and outgoing packets. The firewall has major shortcomings such as inability to detect interior attacks, providing reliable security strategy, and it has a single bottleneck spot and invalid spot, etc. An IDS assesses a suspected intrusion and warns the administrator. An IDS also monitors the interior attacks. Host intrusion detection systems (HIDS) check the inward and outward packets only from the devices and warn the administrator or user if any malicious activity is discovered. HIDS cannot monitor the entire network. Network intrusion detection systems (NIDS) monitor all inbound and outbound traffic by placing an IDS within the network. It alerts the administrator once a malicious activity is identified.

In a misuse or signature-based detection approach, current behavior of network is matched against predefined patterns of attacks detected. They are not efficient to recognize unknown attacks. One of the major drawbacks of signature-based IDS is signature database must be frequently updated and preserved. Anomaly-based detection determines the normal behavior of the system and uses it as baseline for detecting anomalies. It can detect unknown attacks. An IDS can be successfully developed using machine learning algorithms.

Machine learning is a complex computation process which infers a learning model from input samples automatically. Learning models use some statistical functions or rules for describing data dependencies. Machine learning algorithms is categorized into unsupervised learning, supervised learning, and semi-supervised learning. In supervised machine learning, all data are labeled. The pair of input and target output is fed to train the given function, and thus, the entire learning model is trained. If an algorithm is used to learn the mapping function $Y = f(X)$ from the input X to output Y, then it is supervised learning. The aim is approximating the mapping function so that the algorithms learn to estimate the output from the input data. Regression and classification problems are the major grouping of supervised learning problems. If the output variable is a categorical value, then it is classification problem. If the output variable is a real value, then it is a regression problem. In unsupervised learning, all data are unlabeled, and no label is provided in sample data. If only input data is available without corresponding output variables, then it is unsupervised learning. Clustering and association problems come under the category of unsupervised learning problems. If inherent grouping in data is done, then it is called clustering problem. If rules that describe large portions of data are discovered, then it is association rule. In semi-supervised machine learning, only some data are labeled, and most of the

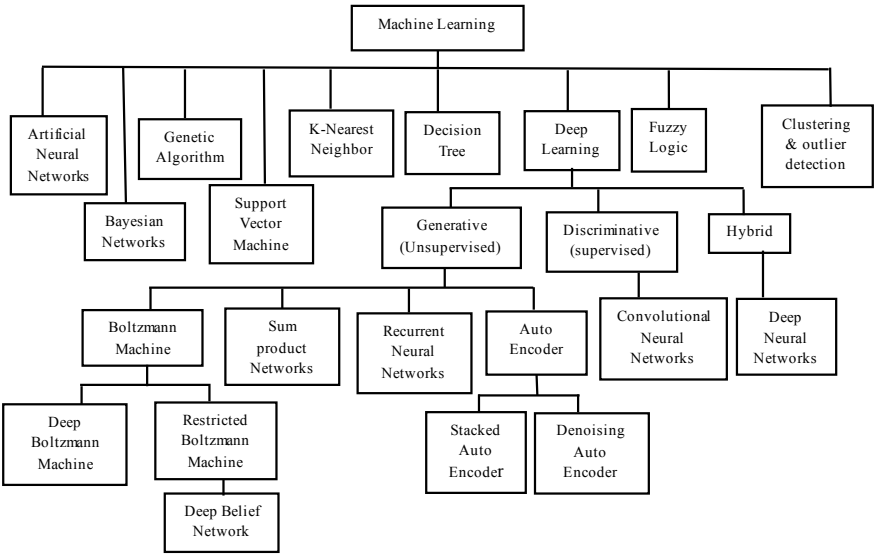

Fig. 1 Machine learning techniques used for IDS

data are unlabeled. It is an amalgamation of supervised and unsupervised techniques. Figure 1 depicts the various machine learning techniques used for IDS.

The paper is organized into sections. In Sect. 2, the survey papers related to machine learning for intrusion detection is discussed. The performance metrics that can be used to evaluate intrusion detection is discussed in Sect. 3. In Sect. 4, the datasets used for intrusion detection using machine learning are explained. In Sects. 5 and 6, recent literatures in machine and deep learning for intrusion detection are focused. In Sect. 7, observations and future directions in intrusion detection using machine learning algorithms are discussed.

2 Related Works

Tsai et al. [2] provided a survey of various machine learning algorithms. They distributed the research articles year-wise based on type classifier design, datasets used, and feature selection algorithm used. However, they did not compare the performance metrics of any machine learning algorithms. Their review is done with research papers published during the period 2000 and 2007.

Buczak et al. [3] surveyed machine learning approaches and data mining techniques that are used for intrusion detection. They categorized the papers based on the machine learning approaches. As well, they have categorized the research papers based on detection methodology either misuse or anomaly. They insisted the

importance of datasets. The time complexity of machine learning algorithms is also discussed. However, their discussion is done till 2014.

Mishra et al. [4] provided detailed information about classification of attacks, machine learning approaches, and feature selection algorithm. They compared the performance of machine learning algorithms based on classifier type. The detailed analysis is done on types of attacks for different types of classifier. They carried out performance analysis based on detection rate of various machine learning approaches for all attack types. The tools used for machine learning are also discussed. They focused mostly on low-frequency attacks.

3 Metrics Used to Evaluate Intrusion Detection System

The performance evaluation of any intrusion detection system can be done by the metrics such as: accuracy (ACC), recall (REC), precision (PRE), true negative rate (TNR), false alarm rate (FAR), false negative rate (FNR), F-measure, Mathews correlation coefficient (MCC), ROC graph, and Kappa statistics. The metrics required for evaluation are computed from confusion matrix. A matrix that describes the performance of a given classification model (or "classifier") is called confusion matrix. It denotes true and false classification results. The ways in which confusion is made when a prediction is done by the classification model is depicted by confusion matrix. True positive (TP): It is the number of correctly identified anomaly records. False positive (FP): It represents the number of incorrectly identified usual records that are detected as anomaly. True Negative (TN): It represents the number of correctly detected records. False Negative (FN): It shows the number of incorrectly detected anomaly records.

4 Datasets Used for Intrusion Detection Research

Most researchers used the datasets DARPA, knowledge discovery and data mining (KDD) Cup, and network security laboratory-KDD (NSL-KDD), UNSW-NB15, Kyoto, and AWID for intrusion detection. Figure 2 illustrates the relation between DARPA, KDD, and NSL-KDD datasets.

The datasets used for intrusion detection by researchers have both training data and testing data. The first standard corpus for the evaluation of intrusion detection

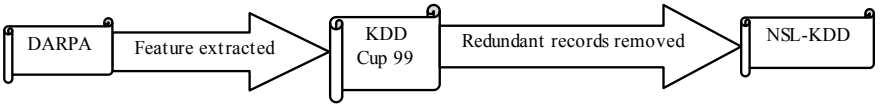

Fig. 2 Relation between DARPA, KDD, and NSL-KDD datasets

Table 1 Data size comparison for different datasets

Dataset	Training size	Testing size
DARPA 99	6.2 GB	3.67 GB
KDD99	4898431bytes	311,029 bytes
NSL-KDD	125,973 bytes	22,444 bytes
UNSW-NB15	175,341 bytes	82,332 bytes
AWID	1,795,575 bytes	575,643 bytes

system was created by MIT Lincoln Laboratory's in 1998 under the sponsorship of DARPA (Kendell 1999).

Tavallaee et al. [5] have analyzed KDD dataset in detail. KDD'99 features can be classified into three groups, namely basic, traffic, and content features. The major problem in this dataset is the enormousness of duplicate records. Tavallaee et al. [5] published the NSL-KDD dataset which eliminates duplicate records in training set thereby overcoming the drawback of classifiers gets biased toward more frequent records. Due to absence of modern attack styles and traffic situations in KDD dataset, a new dataset (UNSW-NB15) was developed by ACCS—an American Cyber security Center. This dataset has a 49-feature set and a total of 2,540,044 records [6]. Kyoto dataset (2009) is created from real environment traffic data collected from honey pot over 3 years. AWID is a dataset that is generated from a wireless network traffic [7]. The traces were produced from a wireless local area network (WLAN) and were secured by the wired equivalent protocol (WEP). Iman Sharafaldin et al. [8] introduced a reliable and real-world dataset, namely CICIDS2017. It contains benign and seven common attack network flows, namely brute force attack, heartbleed attack, botnet, DoS attack, DDoS attack, web attack, and infiltration attack with 80 features.

The dataset size comparison of training and test data for different datasets is shown in Table 1. The datasets used for intrusion detection by researchers have both training data and testing data. The dataset size comparison of training and test data for different datasets is shown in Table 1.

5 Literatures in Machine Learning for Intrusion Detection

Nowadays, researchers used machine learning approaches for intrusion detection. The datasets mostly used for evaluation of the algorithms are KDD Cup 99, NSL-KDD, Kyoto, UNSW-NB15, and AWID. The machine learning approaches are categorized as either single or hybrid based on classifier type used. Feature selection algorithm is also used by few researchers. Table 2 shows the comparison of different machine learning algorithms with classifier type, classification technique, feature selection technique, datasets used, performance metrics, and techniques used for comparison. The performance metric comparison for different machine learning approaches discussed in Table 2 is tabulated in Table 3.

Table 2 Comparison of different machine learning algorithms

Classifier type/baseline	Author	Classification technique	Feature selection algorithm	No. of features and feature no	Dataset used	Merits	Demerits
Single/ANN	Wang et al. [9]	Constrained-Extreme learning machines, Adaptively incremental learning strategy	–	–	KDD-DoS, 10% KDD, UNSW-NB15, NSL-KDD	Better detection rate, Learning speed high	Concept drift
Single/ANN	Chiba et al. [10]	Back Propagation Learning Algorithm	Information Gain Feature Selection Algorithm Modified Kolmogorov–Smirnov Correlation-based Filter Algorithm	12 features (3, 5, 6, 12, 23, 24, 27, 28, 31, 32, 33, 35) 17 features (2, 3, 4, 5, 6, 7, 8, 10, 11, 14, 22, 23, 24, 28, 30, 36, 39) 34 features (1, 5, 6, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41)	KDD CUP 99	higher detection rate and lower false positive rate	Learning Rate and Momentum term is to be added as optimal parameters

(continued)

Table 2 (continued)

Classifier type/baseline	Author	Classification technique	Feature selection algorithm	No. of features and feature no	Dataset used	Merits	Demerits
Single/SVM	Zhao et al. [11]	Multiclass SVM	Redundant Penalty Between Features Mutual Information Algorithm (RPFMI)	DOS 23 features (7, 2, 13, 4, 19, 15, 16, 17, 18, 14, 28, 20, 23, 31, 29, 11, 26, 27, 40, 42, 3, 38, 1) U2R 16 features (8, 9, 6, 21, 22, 23, 10, 4, 27, 39, 15, 11, 14, 18, 12, 19) R2L 5 features (8, 9, 10, 12, 15) Kyoto 6 features (16, 17, 4, 14, 19, 2)	KDD Cup 99 Kyoto 2006+	Feature selection algorithm is optimal Well applied to large and small samples	Anomaly detection with Byzantine fault tolerance is to be done
Single/SVM	Thaseen et al. [12]	SVM	Chi-square feature	31 features (1, 2, , 4, 5, 6, 10, 11, 12, 13, 14, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41)	NSL-KDD	Detects unknown pattern, Parameter tuning is done	Kernel methods can be used for parameter optimization
Single/SVM	Safaldin et al. [13]	SVM	Modified gray wolf optimization	12 features	NSL-KDD	Find optimal feature	Other classifiers can be used

(continued)

Table 2 (continued)

Classifier type/baseline	Author	Classification technique	Feature selection algorithm	No. of features and feature no	Dataset used	Merits	Demerits
Single/Decision tree	Eesa et al. [14]	Decision tree	Cuttlefish optimization algorithm and decision tree	41, 35, 30, 25, 29, 15, 10, 5 features	KDD Cup 99	Produce the optimal subset of features	CFA as a rule generator
Single/Clustering	Zhong et al. [15]	ICLN, SICLN	–		KDD Cup99	Tolerant to missing or delay labels deals with both labeled and unlabeled data	Better estimation method is required to improve efficiency
Hybrid/SVM + K-Means	Al-Yaseen et al. [16]	Multilevel hybrid SVM + ELM modified K-means algorithm	–	–	10% KDD dataset	Detect known and unknown attacks	Cannot classify new attacks efficiently
Hybrid/SVM + Naïve Bayes	Gu et al. [17]	SVM + naïve Bayes	–	–	UNSW-NB15 dataset NSL-KDD Kyoto 2006 +	High-quality data is achieved	Not applicable for multiclass classification
Hybrid	Hajisalem et al. [18]	ABC and AFS algorithms, Fuzzy C-means clustering (FCM)	Correlation-based feature selection (CFS)	NSL-KDD 6 features (6, 4, 5, 12, 29, 27) UNSW-NB15 6 features (46, 45, 23, 24, 47, 43)	NSL-KDD UNSW-NB15	Optimizing rapidity and accuracy	

Table 3 Performance metrics comparison of different machine learning approaches for intrusion detection

Author	Approaches	Dataset used	Performance metrics					
			Accuracy	False alarm/positive rate	True positive/recall/detection rate	Precision	True –ve rate	False –ve rate
Wang et al. [9]	CAI	10% KDD	99.91 ± 0.02	0.10 ± 0.03	± 0.08	99.9 ± 0.01		
	CAI	10% KDD-DoS attack	99.97 ± 0.01	0.10 ± 0.03	99.99 ± 0.00			
	CAI	NSL-KDD	98.92 ± 0.10	0.92 ± 0.11	98.63 ± 0.29	98.33 ± 0.20	1.37 ± 0.29	
	CAI	2% KDD binary	99.88 ± 0.03	0.27 ± 0.14	99.92 ± 0.03	99.93 ± 0.04	99.73 ± 0.14	0.08 ± 0.03
	CAI	UNSW-NB15 binary	82.74	36.46	98.41	76.78	63.54	1.59
Chiba et al. [10]	A12_MinMax_Hrule1_ActS	KDD Cup 99	98.66	1.13	98.59	99.62	98.87	1.41
	A34_Zscore_Hrule1_ActS		99.10	1.60	99.33	99.47	98.41	0.67
Zhao et al. [11]	SVM-RPFMI	DoS (KDD)	99.772	0	99.99	99.55		
		USR (KDD)	96.19	0.37	65.233	95.139		
		R2L (KDD)	91.077	9.907	10.835	99.403		
		Kyoto 2006+	97.749	1.788	97.285	98.196		
Thaseen et al. [12]	SVM	NSL-KDD	98	0.13				

(continued)

Table 3 (continued)

Author	Approaches	Dataset used	Performance metrics					
			Accuracy	False alarm/positive rate	True positive/recall/detection rate	Precision	True –ve rate	False –ve rate
	multiclass SVM	NSL-KDD	96.01 (normal) 95.8 (probe) 99.87 (DoS) 96.37 (R2L) 76.92(U2R)					
Safaldin et al. [13]	SVM	NSL-KDD	96	0.03	96			
Eesa et al. [14]	CFA (no. of features = 10)	KDD Cup 99	92.837	3.9	92.051			
Zhong et al. [15]	ICLN	KDD 99	99.58		99.59	98.59		
	SICLN		99.66		99.60	98.92		
Al-Yaseen et al. [16]	hybrid SVM + ELM	10% KDD	95.75	1.87	95.17			
Gu et al. [17]	SVM + Naïve Bayes	UNSW-NB15	93.75	7.33	94.73			
		CICIDS2017	98.92	3.00	99.46			
		NSL-KDD	99.36	0.54	99.25			
		Kyoto 2006+	98.58	2.62	99.73			
Hajisalem et al. [18]	ABC-AFS method	NSL-KDD	99	0.01	99			
	ABC-AFS method	UNSW-NB15	98.9	0.13	98.6			

5.1 Single Classifier

Wang et al. [9] applied equality constrained optimization-based ELM (C-ELM), an approach proposed by Huang et al. that detects intrusion. They also proposed an adaptively incremental learning strategy, namely construction with adaptive increments (CAI) which derives the finest count of hidden neurons. Their approach eliminates the computation of weights from the scratch when the numbers of neurons are increased as suggested in C-ELM approach. Their approach overcomes the drawback of C-ELM which caused wastage of time during computation. They conducted their experimental work on KDD-DoS dataset 10% KDD dataset, NSL-KDD dataset, and UNSW-NB15. They used the traditional method of converting categorical values into numeric for preprocessing. They have done comparison of their algorithm with few approaches such as Tan's, Lin's, Hu's, Singh's, SVM, Xu's, and MLP and shown improvement in accuracy, recall, false alarm rate, precision, time, false negative rate, and specificity. The authors suggested that concept drift can be used as a future work.

Chiba et al. [10] introduced optimal anomaly network intrusion detection system (ANIDS) approach based on BPNN. They adapted a learning algorithm, namely back propagation to develop a new architecture. They utilized modified Kolmogorov–Smirnov correlation-based filter (CBF) algorithm and information gain algorithm for dimensionality reduction. The authors build 48 IDSs by combining the classifiers. Their proposed ANIDS have four modules, namely feature selection, data preprocessing, normalization, and detection. They considered performance metrics such as false alarm rate, detection rate, F-measure, ability to avoid false classification (AUC) to choose the best two IDSs. They employed the dataset KDD CUP 99 for their experimental study. The comparative analysis of their proposed IDS was done with several techniques. Their approach showed performance improvement with regard to detection rate, F-score, accuracy, score, and lower false alarm rate. As their future work, they may improve the performance of IDS using an optimization algorithm that uses momentum term and learning rate as parameters.

Zhao et al. [11] developed a novel algorithm that utilizes FB feature selection based on MI called the RPFMI algorithm. In their proposed algorithm, they considered three factors, namely redundancy among features, the relationship among candidate features and classes, and the impact among selected features and classes in order to increase relevancy, and reduce redundancy among features. They used Kyoto 2006+ and KDD Cup 99 datasets in their experiment. The accuracy rate on the DoS data is 99.772%, USR data is 96.19%, and R2L data is 91.077% which is better than all other compared algorithms. The Kyoto 2006+ dataset achieves the highest accuracy of 97.749% when compared to other algorithms. As a future work, the authors suggested to use the proposed RPFMI algorithm with Byzantine fault tolerance to detect anomaly.

Thaseen et al. [12] proposed a feature selection (chi-square) and SVM (multiclass) model for intrusion detection by adapting over fitting constant (C) and gamma (γ) as parameters to optimize the RBF kernel. They used the dataset NSL-KDD for their

experimental works. Their algorithm showed high true positive rate and low false positive rates when compared with other traditional approaches.

Safaldin et al. [13] developed an enhanced intrusion detection system (IDS) which used modified binary gray wolf optimizer for feature selection and SVM classifier for classification. They varied the number of wolves to find the exact number of wolves. They used NSL-KDD dataset as benchmark to compute accuracy, detection rate, and processing time. The seven wolves GWOSVM-IDS outperformed existing algorithms.

Eesa et al. [14] developed a feature selection model based on the cuttlefish optimization algorithm (CFA) and decision tree classifier. CFS is used to produce the best feature subsets, and decision tree is used to improve the quality of the created feature subsets. The proposed model is evaluated using KDD Cup 99 dataset. This algorithm yields better true positive rate, and accuracy, and lower false positive rate when a maximum of 20 features are chosen. They suggested using CFA as a rule generator for IDS.

Zhong et al. [15] proposed two new clustering algorithms for network intrusion detection. One of the algorithms is unsupervised algorithm, namely the improved competitive learning network (ICLN), and the other is supervised improved competitive learning network (SICLN) to detect network intrusion. The authors have done comparative analysis of performance of the proposed algorithms with both SOM and K-means. The datasets used for their experimental work are the KDD 99, vesta transaction data, and iris data. Their experimental results showed that ICLN achieved similar accuracy when compared with other unsupervised clustering algorithms. But, SICLN performs better than other algorithms in solving classification problems using clustering approaches.

5.2 Hybrid Classifier

Al-Yaseen et al. [16] introduced a multilevel hybrid model that uses both SVM and ELM. They also introduced a modified K-means algorithm that builds high-quality training datasets. Their approach showed improved performance than multilevel SVM and multilevel ELM. They used 10% KDD dataset for their work. In their proposed work, they used the equivalent numerical attributes for the symbolic ones, then they normalized data to [0, 1], and the instances of 10% KDD training dataset are separated into five categories such as normal, probe, DoS, R2L, and U2R.Then, they applied modified K-means on each separated category and trained both SVM and ELM with these those training datasets. Finally, testing is done with these datasets. They achieved an overall accuracy of 95.75%, true positive rate of 95.17%, and false positive rate of 1.87. As a future extension, the authors recommended to construct an efficient model to classify new attacks with better performance.

Gu et al. [17] proposed a hybrid classifier based on SVM and naive Bayes feature embedding. They utilized naive Bayes technique for feature enhancement and SVM for classification. The experiments are done using the datasets, namely UNSW-NB15,

NSL-KDD, Kyoto, and CICIDS2017. Their experiments have shown an accuracy of 93.75% on UNSW-NB15 dataset, 98.92% accuracy on CICIDS2017 dataset, 99.35% accuracy on NSL-KDD dataset, and 98.58% accuracy on Kyoto 2006+ dataset.

Hajisalem et al. [18] proposed a novel hybrid classification approach by combining both artificial bee colony (ABC) and artificial fish swarm (AFS) algorithms. They split the training dataset and eliminated the irrelevant features by applying fuzzy *C*-means clustering (FCM) and CFS techniques. They used CART technique to generate If–Then rules which distinguished the normal and anomaly records for the selected features. The authors trained the proposed hybrid method through the generated rules. They used the datasets UNSW-NB15 and NSL-KDD for their experimental work. They achieved false alarm rate of 0.01% and detection rate of 99%. In their proposed method, they have computed the computational complexity and time.

6 Literatures in Deep Learning for Intrusion Detection

Based on learning techniques, ML algorithms can be classified as shallow learning and deep learning. Algorithms with few layers are known as shallow learning which is better suited for less complex datasets. The emerging technique which uses more layers of neural network is referred as deep learning which is used for complex target function and larger datasets. Table 4 shows the comparison of different deep learning algorithms with classifier type, datasets used, performance metrics, and techniques used for comparison. The performance metric comparison for different deep learning approaches discussed in Table 4 is tabulated in Table 5.

6.1 Deep Auto-encoders

Farahnakian et al. [19] developed a deep auto-encoder method to improve the performance of IDS. Their DAE model extracts important features from training data by utilizing a nonlinear activation function, namely sigmoid function. To avoid over fitting and local optima, they pre-trained their model using a greedy layer-wise unsupervised learning algorithm. A softmax classifier is used to denote the preferred outputs (normal or attack type). They used the dataset 10% of KDD Cup 99 their experimental work. The results are done in two scenarios, namely binary classification and multiclassification. In binary classification scenario, the detection rate is 95.65%, false alarm is 0.35, and accuracy is evaluated as 96.53%. In multiclassification scenario, the detection rate is 94.53%, false alarm is 0.42, and accuracy is evaluated as 94.71%. They suggested sparse deep auto-encoders as an approach to enhance the detection efficiency.

Shone et al. [20] introduced a technique for unsupervised feature learning, namely non-symmetric deep auto-encoder (NDAE). They also proposed stacked NDAEs as a classification model that does not use a decoder. The benchmark dataset used for their

Table 4 Comparison of different deep learning algorithms for intrusion detection

Baseline	Author	Technique	Datasets used	Techniques compared	Merits	Demerits
DAE	Farahnakian et al. [19]	Deep auto-encoder (DAE)	KDD CUP'99	DBN, auto-encoder + DBN	Avoid overfitting and local optima	Sparsity constraints not imposed
	Shone et al. [20]	Non-symmetric deep auto-encoder (NDAE)	KDD CUP'99 and NSL-KDD	DBN	Training time less	Do not handle zero-day attack
BM	Zhang et al. [21]	RBM + SVM, RBM + DBN	KDD Cup 99	–	Unsupervised learning is used for feature extraction training time less	
RNN	Yin et al. [22]	RNN-IDS	NSL-KDD	NB tree, random tree, J48, naïve Bayes, random forest, MLP, RNN, SVM	Applicable for binary and multiclass classification	Training time is not reduced
	Kim et al. [23]	LSTM-RNN	KDD Cup 99	GRNN, PNN, RBNN, K-NN, SVM, and Bayesian	Detection rate high	Do not detect U2R, and FAR is to be improved
	Su et al. [24]	BLSTM	NSL-KDD	RNN, DNN, DBN, LSTM, CNN, BLSTM, and BAT	Automatically learn the key features and efficient anomaly detection	–
CNN	Ho et al. [25]	CNN	CICIDS2017	Hierarchical, WISARD, forest PA, J48, LIBSVM, FURIA, random forest, MLP, and naive Bayes	Storage and computation overhead are less and detect innovative attacks	Automated methods can be used to solve class imbalance issues

(continued)

Table 4 (continued)

Baseline	Author	Technique	Datasets used	Techniques compared	Merits	Demerits
DNN	Roy et al. [26]	DNN	KDD Cup 99	SVM	High R2 value so more accurate	–
	Kasongo et al. [27]	FFDNN+ Wrapper method	UNSW-NB15 and AWID	Random forest, SVM, naïve Bayes, decision tree, and K-NN	Applicable for wired and wireless networks	Detection rates of individual classes is not done
	Devan et al. [28]	DNN	NSL-KDD	Logistic regression, SVM, and naive Bayes	Prevent overfitting and faster detection	Not applicable for multiclass classification

Table 5 Performance metrics comparison for different deep learning techniques for intrusion detection

Author	Approaches	Dataset used	Accuracy	Detection rate	False alarm/positive rate	True positive/recall	Precision	False negative
Farahnakian et al. [19]	DAE-IDS	KDD 99	94.71	95.65 (binary) 94.53 (multi-)	0.35 (binary) 0.42 (multi-)	94.42	–	
Shone et al. [20]	S-NDAE	KDD Cup	97.85	97.85	2.15	–	99.99	
	S-NDAE	NSL-KDD	85.42	85.42	14.58	–	100	
Zhang et al. [21]	RBM + DBN	KDD 99	97.160		0.480			3.610
	RBM + SVM		96.310		0.400			4.520
Yin et al. [22]	RNN(KDDTest+)	NSL-KDD	81.29					
	RNN(KDDTest-21)		64.67					
Kim et al. [23]	LSTM-RNN	KDD 99	96.93	98.88	10.04			
Su et al. [24]	BLSTM (KDD test+)	NSL-KDD	84.25	97.50 (normal) 87.55 (DoS) 44.25 (R2L) 20.95 (U2R) 85.76 (probe)	25.70 (normal) 1.52 (DoS) 0.91 (R2L) 0.09 (U2R) 1.15 (probe)			
	BLSTM (KDD test-21)	NSL-KDD	69.42					
Ho et al. [25]	CNN	CICIDS2017	99.78%					
Roy et al. [26]	DNN	KDD CUP 99	99.99					
Kasongo et al. [27]	DNN	UNSW-NB15	94.03 (Full) 92.38 (reduced)					
		AWID binary	98.69 (Full) 99.66 (reduced)					
Devan et al. [28]	DNN	NSL-KDD	97					

evaluation are KDD 99 and NSL-KDD. Their evaluation results show that for KDD 99 dataset, an accuracy of 97.85%, precision of 99.99%, recall 97.85, F-score of 98.15, and false alarm of 2.1 are achieved. With regard to 5-class NSL-KDD classification, an accuracy of 85.42%, precision of 100%, recall 85.42, F-score of 87.37, and false alarm of 14.58 are achieved. With regard to 13-class NSL-KDD classification, an accuracy of 89.22%, precision of 92.97%, recall 89.22, F-score of 90.76, and false alarm of 10.78 are achieved. Their result achieved better accuracy, detection rate, and precision, and reduced training time. As their future work, they have suggested to handle zero-day attack and apply their suggested model to real-world backbone network traffic.

6.2 Boltzmann Machine (BM)

Zhang et al. [21] analyzed the performance and characteristics of deep learning in two hybrid algorithms, namely RBM with SVM and RBM with DBN. They have done their experimental study using KDD cup 99 dataset. The performance metrics used for their evaluation are accuracy, testing time, false negative rate, and false alarm rate. They compared their hybrid algorithms with other traditional algorithms and found that DBN performs better in metrics accuracy and speed. RBM-SVM achieved an accuracy of 96.31%, and RBM-DBN achieved an accuracy of 97.16% when compared with the traditional PCA-BP that attained an accuracy of 92.26%.

6.3 Recurrent Neural Networks (RNN)

Yin et al. [22] introduced a deep learning method, namely RNN-IDS for intrusion detection. The performance study is done using binary classification and multiclass classification. They used NSL-KDD dataset for evaluation. The proposed approach is compared with those of J48, random forest, ANN, and SVM. They reported that RNN-IDS gives better accuracy and that its performance is better in both multiclass and binary classification.

Kim et al. [23] applied long short-term memory (LSTM) architecture for training IDS model in RNN. They normalized all instances from 0 to 1 before using the training dataset. They used input vector with 41 dimensions and output vector with 4 dimensions. LSTM architecture is applied to the hidden layer. For their experiment, they used batch size of 50, time step size of 100, and epoch of 500. They used an optimizer, namely stochastic gradient decent (SGD) and softmax at output layer. Mean squared error (MSE) is used as the loss function. In their first experiment, they analyzed hyper-parameter values and found that hidden layer size and learning rate will produce the best performance in their second experiment. The optimal hidden layer size and learning rate are 80 and 0.01, respectively. KDD Cup 1999 dataset

is used for their validation. Their approach achieves an accuracy of 96.93% and detection rate of 98.88% which is better than other compared approaches.

Su et al. [24] proposed variation of BAT model with multiple convolution layers, namely BAT-MC. They utilized BLSTM for traffic classification and attention mechanism to retrieve the key feature data. Attention mechanism conducts feature learning on sequential data composed of data package vectors. The obtained feature information is reasonable and accurate. The benchmark dataset used is NSL-KDD dataset. The experimental results show that the performance of BAT-MC is better than the traditional methods.

6.4 Convolutional Neural Networks (CNN)

Ho et al. [25] developed an IDS with CNN classifier. The dataset used is CICIDS2017 which includes innovative attacks. Their model has shown better detection rate for 10 classes of attacks among 14 and has been used to train and validate the proposed model. The issues found in the dataset are missing value, imbalanced class, and scattered presence. They solved these issues and created a customized database, namely α-Dataset after preprocessing. Their model performed well in terms of metrics accuracy, detection rate, false alarm rate, and training overhead.

6.5 Deep Neural Network (DNN)

Roy et al. [26] accessed the functionality of the classifier (DNN) for validating several attacks that cause intrusion. KDD Cup 99 dataset is used for their validation. They compared their work with support vector machine (SVM). They used rectifier and softmax activation functions. Their experimental results of DNN showed a better performance in accuracy when compared with SVM.

Kasongo et al. [27] used extra trees algorithm for wrapper-based feature extraction and feedforward deep neural network (FFDNN) to develop wireless IDS. UNSW-NB15 and the AWID intrusion detection datasets are used for their experimental study which includes both binary and multiclass types of attacks. A feature vector of 22 attributes is used in UNSW-NB15. For binary and multiclass classification, an accuracy of 87.10 and 77.16% is achieved. A feature vector of 26 attributes is used in AWID. For binary and multiclass classification, an accuracy of 99.66 and 99.77% is achieved.

Devan et al. [28] proposed a method for network intrusion detection which used XGBoost feature selection and deep neural network (DNN) for classification. To optimize the learning rate, they used Adam optimizer and softmax classifier. The dataset used for their experiment is NSL-KDD dataset. Their method has shown improved performance in metrics accuracy, precision, recall, and F1-score.

7 Observations and Future Directions

Regarding datasets, most researchers have done their research on KDD Cup and NSL-KDD datasets. But, Brugger [29] claims that there is problem with this dataset. Therefore, researchers can use UNSW-NB 15, AWID, Kyoto, and CICIDS 2017 datasets for their research. Also, they can do performance analysis using some real traffic datasets. There is huge demand for real-time data set for intrusion detection.

The comparative chart illustrating accuracy of different machine learning algorithms based on different datasets is shown in Fig. 3.

Among all compared machine learning algorithms, hybrid classifier detects attack more accurately when compared with single classifiers. It is observed that classifier with feature selection algorithm shows better detection of attacks. Moreover, the performance of hybrid classifier is better when feature selection algorithm is used. Most of classifier's performance is not better when all features are used. Therefore, feature selection plays a major role in attack detection. Hence, researchers can think of developing the best algorithm for feature selection. Also, the performance of classifiers varies among datasets. So, a better IDS can be developed to sort out this issue.

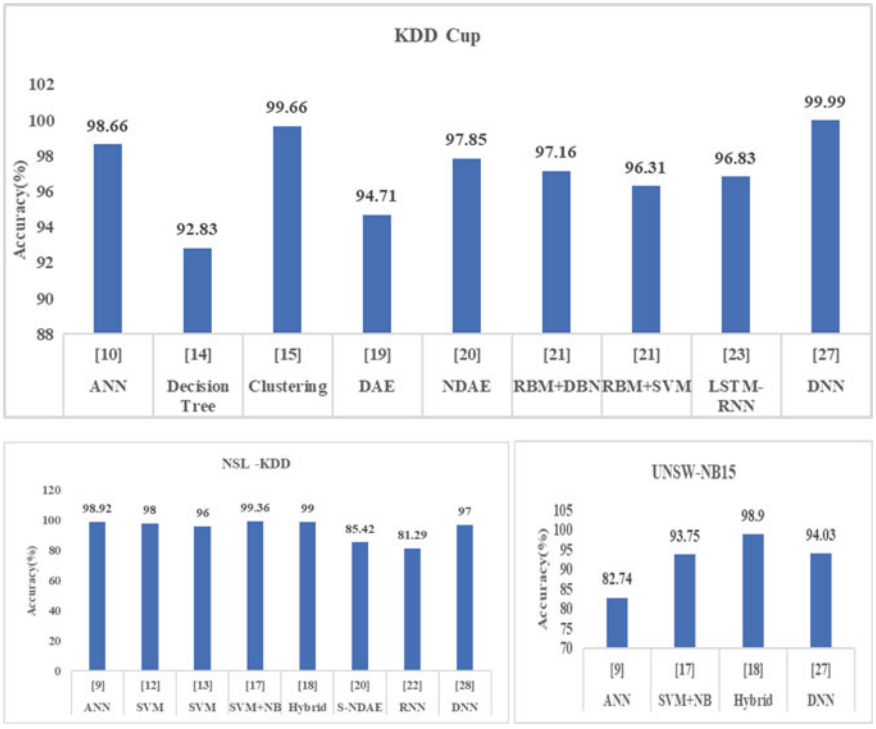

Fig. 3 Comparison of different machine learning algorithms based on accuracy

Among all deep learning algorithms discussed, deep neural networks achieved better accuracy and detection rate. But, some other deep learning approaches such as convolutional neural networks and reinforcement learning can be applied to detect attacks.

8 Conclusion

A taxonomy of different machine learning algorithms used for intrusion detection is discussed. The IDS developed based on machine learning and deep learning algorithms is analyzed. Machine learning algorithms are analyzed based on the classifier type either single or hybrid. Feature selection methods incorporated with machine learning algorithms are also discussed. Machine learning algorithms that used feature selection techniques have shown better accuracy. Deep learning models deal with huge input data. Deep learning IDS has shown better performance in terms of accuracy and running time. GPU-enabled deep learning algorithms can perform execution faster. Future directions to detect intrusion using machine learning algorithm are also discussed.

References

1. Cyber attacks, ALERT: DNS hijacking activity (2019). Online https://www.ncsc.gov.uk/alerts/alert-dns-hijacking-activity
2. C.F. Tsai, Y.F. Hsu, C.Y. Lin, W.Y. Lin, Intrusion detection by machine learning: a review. Exp. Syst. Appl. **36**(10), 11994–1200 (2009)
3. A. Buczak, E. Guven, A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutorials 99 (2015)
4. P. Mishra, V. Varadharajan, U. Tupakula, E.S. Pilli, A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Commun. Surv. Tutorials (2018)
5. M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in *IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009)* (2009), pp. 1–6
6. N. Moustafa, J. Slay, The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Inf. Secur. J. **25**(1–3), 18–31 (2016)
7. C. Kolias, G. Kambourakis, A. Stavrou, S. Gritzalis, Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset. IEEE Commun. Surv. Tutor **18**(1), 184–208 (2015)
8. I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in *ICISSP 2018—Proceedinggs of 4th International Conference on Information Systems Security and Privacy* (2018), pp. 108–116
9. C.R. Wang, R.F. Xu, S.J. Lee, C.H. Lee, Network intrusion detection using equality constrained-optimization-based extreme learning machines. Knowl. Based Syst. (2018)
10. Z. Chiba, N. Abghour, K. Moussaid, A. El, M. Rida, A novel architecture combined with optimal parameters for back propagation neural networks applied to anomaly network intrusion detection. Comput. Secur. (2018)

11. F. Zhao, Applied sciences a filter feature selection algorithm based on mutual information for intrusion detection (2018)
12. S. Thaseen, A. Kumar, Intrusion detection model using fusion of chi-square feature selection and multi class SVM. J. King Saud Univ. Comput. Inf. Sci. **29**(4), 462–472 (2017)
13. M. Safaldin, M. Otair, L. Abualigah, Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. J. Ambient Intell. Humaniz. Comput. (2020)
14. A.S. Eesa, Z. Orman, A. Mohsin, A. Brifcani, Expert systems with applications a novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems. Exp. Syst. Appl. 1–10 (2014)
15. J. Zhong, A. A. Ghorbani, Neurocomputing Improved competitive learning neural networks for network intrusion and fraud detection. Neurocomputing **75**(1), 135–145 (2012)
16. W.L. Al-Yaseen, Z.A. Othman, M.Z.A. Nazri, Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Exp. Syst. Appl. **67**, 296–303 (2017)
17. J. Gu, S. Lu, An effective intrusion detection approach using SVM with naïve Bayes feature embedding. Comput. Secur. **103** (2021)
18. V. Hajisalem, S. Babaie, A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection. Comput. Netw. **136**, 37–50 (2018)
19. F. Farahnakian, J. Heikkonen, A deep auto-encoder based approach for intrusion detection system, in *International Conference on Advanced Communications Technology* (2018), pp. 178–183
20. N. Shone, T.N. Ngoc, V.D. Phai, Q. Shi, A deep learning approach to network intrusion detection. IEEE Trans. Emerg. Top. Comput. Intell. **2**(1), 41–50 (2018)
21. X. Zhang, J. Chen, Deep learning based intelligent intrusion detection (2017)
22. C. Yin, Y. Zhu, J. Fei, X. He, A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access 21954–2196 (2017)
23. J. Kim, J. Kim, H. Le, T. Thu, H. Kim, Long short term memory recurrent neural network classifier for intrusion detection (2016)
24. T. Su, H. Sun, J. Zhu, S. Wang, Y. Li, BAT: deep learning methods on network intrusion detection using NSL-KDD dataset. IEEE Access 29575–29585 (2020)
25. S. Ho, S. Jufout, S. Al, K. Dajani, M. Mozumdar, A novel intrusion detection model for detecting known and innovative cyberattacks using convolutional neural network. IEEE Open J. Comput. Soc. **2**, 14–25 (2021)
26. S.S. Roy, A. Mallik, R. Gulati, M.S. Obaidat, P.V. Krishna, A deep learning based artificial neural network approach for intrusion detection, in *Mathematics and Computing. ICMC 2017. Communications in Computer and Information Science* ed. by D. Giri, R. Mohapatra, H. Begehr, M. Obaidat, vol 655. (Springer, Singapore, 2017)
27. S.M. Kasongo, Y. Sun, A deep learning method with wrapper based feature extraction for wireless intrusion detection system. Comput. Secur. **92** (2020)
28. P. Devan, N. Khare, An efficient XGBoost–DNN-based classification model for network intrusion detection system. Neural Comput. Appl. 12499–12514 (2020)
29. T. Brugger, KDD Cup '99 dataset (Network Intrusion) considered harmful (2007)