Open Access Article

PERFORMANCE ANALYSIS OF MACHINE LEARNING MODELS FOR THREATS AND ATTACKS IN NETWORK SECURITY TRAFFIC MODEL

P.Prasanya Devi
Research Scholar, Department of Computer Application, School of Information Technology, Madurai Kamaraj University, Madurai, prasanyamsc@gmail.com

Dr.S.Kannan
Professor, Department of Computer Application, School of Information Technology, Madurai Kamaraj University, Madurai, skannanmku@gmail.com

Abstract—In Internet Engineering Task Force 97 (IETF97), the challenge is introduced as networks suffer from the lack of a unified theory that can be applied to all networks. It means that the behavior of networks is heterogeneous based on their different topologies, equipment, scale, applications, etc. It causes an important problem that ML techniques should be trained for each network separately. The accuracy of Machine Learning(ML) techniques that are trained by public datasets can be reduced in different networks. There are some efforts to provide representative datasets, but it seems that the challenge increases the need for ML techniques that can label the data and re-train frequently for each network separately. Therefore, rather than public datasets, the Deep Learning(DL) techniques should be trained by exclusive datasets gathered from the target network and labeled with high accuracy. To solve this challenge, DL techniques should be learned for each network separately, but as mentioned above, retraining the DL models for each network is a time and resource consuming task.
Keywords: Machine Learning, Deep Learning, Algorithms in ML and DL, Deep Learning Threats and Attacks, Traffic Analysis, Traffic Prediction, Performance Analysis.

摘要：在 Internet 工程任务组 97 (IETF97) 中，由于网络缺乏可应用于所有网络的统一理论，因此引入了挑战。这意味着网络的行为基于其不同的拓扑、设备、规模、应用等是异构的。这导致了一个重要的问题，即应该为每个网络分别训练 ML 技术。由公共数据集训练的机器学习 (ML) 技术的准确性可以在不同的网络中降低。有一些努力提供具有代表性的数据集，但似乎挑战增加了对 ML 技术的需求，这些技术可以分别为每个网络标记数据并频繁地重新训练。因此，深度学习（DL）技术应该通过从目标网络收集并以高精度标记的专有数据集进行训练，而不是公共数据集。为了解决这一挑战，应该为每个网络分别学习 DL 技术，但如上所述，为每个网络重新训练 DL 模型是一项耗时且耗费资源的任务。

Received: October 05, 2021 / Revised: October 31, 2021 / Accepted: November 30, 2021 / Published: December 31, 2021
About the authors：P.Prasanya Devi
Corresponding author- *Email: prasanyamsc@gmail.com

关键词：机器学习、深度学习、机器学习和深度学习中的算法、深度学习威胁和攻击、流量分析、流量预测、性能分析。

I.INTRODUCTION

The appeal and pervasiveness of machine learning (ML) is growing. Existing methods are being improved, and their ability to understand and answer real issues is highly appreciated. These achievements have led to the adoption of machine learning in several domains, such as computer vision, medical analysis, gaming and social media marketing [1]. In some scenarios, machine learning techniques represent the best choice over traditional rule-based algorithms and even human operators [2]. This trend is also affecting the cyber security field where some detection systems are being upgraded with ML components [3]. Although devising a completely automated cyber defence system is yet a distant objective, first level operators in Network and Security Operation Centres (NOC and SOC) may benefit from detection and analysis tools based on machine learning.

Our study is based on an extensive review of the literature and on original experiments performed on real, large enterprises and network traffic. Other academic papers compare ML solutions for cyber security by considering one specific application (e.g.: [4], [3], [5]) and are typically oriented to Artificial Intelligence (AI) experts rather than to security operators. In the evaluation, we exclude the commercial products based on machine learning (or on the abused AI term) because vendors do not reveal their algorithms and tend to overlook issues and limitations. First, we present an original taxonomy of machine learning cyber security approaches. Then, we map the identified classes of algorithms to three problems where machine learning is currently applied: intrusion detection, malware analysis, spam and phishing detection. Finally, we analyse the main limitations of existing approaches. Our study highlights pros and cons of different methods, especially in terms of false positive or false negative alarms. Moreover, we point out a general underestimation of the complexity of managing ML data for training, and by the time required for fine-tuning operations in a domain characterized by continuous change. We also consider recent results emphasizing the effectiveness of adversarial attacks [6] [5] in evading ML detectors. The evidenced drawbacks pave the way to future improvements that ML components require before being fully adopted in cyber defense platforms.

Unlike other applications of ML, networking suffers from high dynamicity and consequently it needs to retrain the models to be adapted with the new situations in a network. In a network, models should be retrained frequently because of the following events:

- Daily training
- To be adapted with the new changes upon network manager’s requests
- Triggered by detecting some events, e.g. security breaches,network behavior changes, or starting new packet streams.

Generally, DL models suffer from high complexity in the training phase as they consume plenty of resources and time. Most of the networking applications are time-consuming, thereby the time complexity of DL models to be retrained can be considered as a great challenge as DL models should be optimized in terms of time complexity and resource

consumption.Figure 1 depicts the Perfomance comparison of ML and DL.

Figure 1 .Perfomance comparison of ML and DL

II.LITERATURE SURVEY

The author [1] has studied the network security issues and conducted the experiment using Naive Bayes, Random Forest Support Vector Machine and K-means ML algorithms to detect four types of attacks like DOS, PROBE, U2R and R2L. They concluded that the Random Forest Classifier (RFC) surpasses the other methods and also stated that hierarchical clustering method can be used to improve the performance of the system.

In paper [2] the author has done a comparison using supervised machine learning classifiers namely, Random Forest, Support Vector Machine, Gaussian Naive Bayes, and Logistic Regression are compared for an intrusion detection in network. Effective classifying algorithm is identified based on performance matrix namely F1-Score, accuracy, precision, and recall. Based on the observed results they have concluded that the Random forest classifier outperforms other classifiers for the considered data-set and parameters. A light weight IDS method is proposed here [3] mainly concerned on pre-processing of the data, so that they can use important features of online data. The main step is to remove the redundant data from dataset to standardize the data. This helps the machine learning algorithms to give the unbiased and accurate result. In paper [4] the author proposed intrusion detection system (IDS) using supervised machine learning techniques to detect the online network data as normal or anomaly. The proposed method only identifies the Denial of Service (DOS) and probe attacks, but the other attacks are not taken into consideration. The author proposed Intrusion detection method using Support Vector Machine (SVM) [5]. They also used feature removal method to improve the efficiency of the algorithm. Using the proposed feature removal method, they selected best nineteen features from the KDD-CUP99 data-set. The authors have proposed [6] anomaly intrusion detection using improved Self Adaptive Bayesian Algorithm to process the large amount of data. In this paper [7] authors proposed a novel idea to reduce the dimensionality of the data by using triangle-based K-NN approach. An Intrusion Detection system using fuzzy logic is tested [8]. This technique uses a set of fuzzy rules which are obtained from the definite rules using frequent items. The classification accuracy of this approach is above 90% for all types of attacks.

G. Meera Gandhi et al., [9] examined the performance measure of four supervised machine learning algorithms in detecting the four types of attack such as DoS, R2L, Probe, and U2R. The result shows that the C4.5 decision tree classifier performs best in prediction accuracy compared to Naıve Bayes. The authors [10] have compared the performance of the three machine learning algorithm namely Neural Network, Support Vector Machine and Decision Tree. The algorithms were measured based on false alarm rate, accuracy and detection rate of four categories of attacks classes. From these experiments they found that the Decision tree (J48)

algorithm outperformed the other two algorithms. The author [11] suggested using a collective of, Support Vector Machines (SVMs), Multivariate Adaptive Regression Splines (MARS), and Artificial Neural Networks (ANNs). Whereas [12] proposed a hybrid approach in which they have used a Support Vector Machine (SVM) and Radial Basis Function (RBF). The sequential search strategy for feature selection or feature extraction through determining the importance of a given attribute by simply removing it and recording the performance [13]. If performance of the algorithm is increased, then the feature is unimportant and thus shall be removed. The author [14] suggested that every attribute in the dataset is not much important and it will not give the accurate result as expected. It is very important to reduce the no of features using feature selection technique and also they concluded that simple Cart algorithms gives accurate result than other five algorithm J4.8 Naive Bayes, NBTree, Multi- Layer Perceptron, and SVM.

III. METHODOLOGY

A. Shallow Learning

1) Supervised SL algorithms

- Naïve Bayes (NB). These algorithms are probabilistic classifiers which make the a-priori assumption that the features of the input dataset are independent from each other. They are scalable and do not require huge training datasets to produce appreciable results.
- Logistic Regression (LR). These are categorical classifiers that adopt a discriminative model. Like NB algorithms, LR methods make the a-priori independency assumption of the input features. Their performance is highly dependent on the size of the training data.
- Support Vector Machines (SVM). These are non-probabilistic classifiers that map data samples in a feature space with the goal of maximizing the distance between each category of samples. They do not make any assumption on the input features, but they perform poorly in multi-class classifications. Hence, they should be used as binary classifiers. Their limited scalability might lead to long processing times.
- Random Forest (RF). A random forest is a set of decision trees, and considers the output of each tree before providing a unified final response. Each decision tree is a conditional classifier: the tree is visited from the top and, at each node, a given condition is checked against one or more features of the analysed data. These methods are efficient for large datasets and excel at multiclass problems, but deeper trees might lead to overfitting.
- Hidden Markov Models (HMM). These model the system as a set of states producing outputs with different probabilities; the goal is to determine the sequence of states that produced the observed outputs. HMM are effective for understanding the temporal behaviour of the observations, and for calculating the likelihood of a given sequence of events. Although HMM can be trained on labelled or unlabelled datasets, in cyber security they have mostly been used with labelled datasets.
- K-Nearest Neighbour (KNN). KNN are used for classification and can be used for multi-class problems. However, both their training and test phase are computationally demanding as to classify each test sample, they compare it against all the training samples.
- Shallow Neural Network (SNN). These algorithms are based on neural networks, which

consist in a set of processing elements (that is, neurons) organized in two or more communicating layers. SNN include all those types of neural networks with a limited number of neurons and layers. Despite the existence of unsupervised SNN, in cyber security they have mostly been used for classification tasks.

2)Unsupervised SL algorithms

- Clustering. These group data points that present similar characteristics. Well known approaches include k-means and hierarchical clustering. Clustering methods have a limited scalability, but they represent a flexible solution that is typically used as a preliminary phase before adopting a supervised algorithm or for anomaly detection purposes.
- Association. They aim to identify unknown patterns between data, making them suitable for prediction purposes. However, they tend to produce an excessive output of not necessarily valid rules, hence they must be combined with accurate inspections by a human expert.

B. Deep Learning

All DL algorithms are based on Deep Neural Networks (DNN), which are large neural networks organized in many layers capable of autonomous representation learning.

1) Supervised DL algorithms

- Fully-connected Feedforward Deep Neural Networks (FNN). They are a variant of DNN where every neuron is connected to all the neurons in the previous layer. FNN do not make any assumption on the input data and provide a flexible and general-purpose solution for classification, at the expense of high computational costs.
- Convolutional Feedforward Deep Neural Networks (CNN). They are a variant of DNN where each neuron receives its input only from a subset of neurons of the previous layer. This characteristic makes CNN effective at analysing spatial data, but their performance decreases when applied to nonspatial data. CNN have a lower computation cost than FNN.
- Recurrent Deep Neural Networks (RNN). A variant of DNN whose neurons can send their output also to previous layers; this design makes them harder to train than FNN. They excel as sequence generators, especially their recent variant, the long short-term memory.

2) Unsupervised DL algorithms

- Deep Belief Networks (DBN). They are modelled through a composition of Restricted Boltzmann Machines (RBM), a class of neural networks with no output layer. DBN can be successfully used for pre-training tasks because they excel in the function of feature extraction. They require a training phase, but with unlabelled datasets.

- Stacked Autoencoders (SAE). They are composed by multiple Autoencoders, a class of neural networks where the number of input and output neurons is the same. SAE excel at pre-training tasks similarly to DBN, and achieve better results on small datasets.

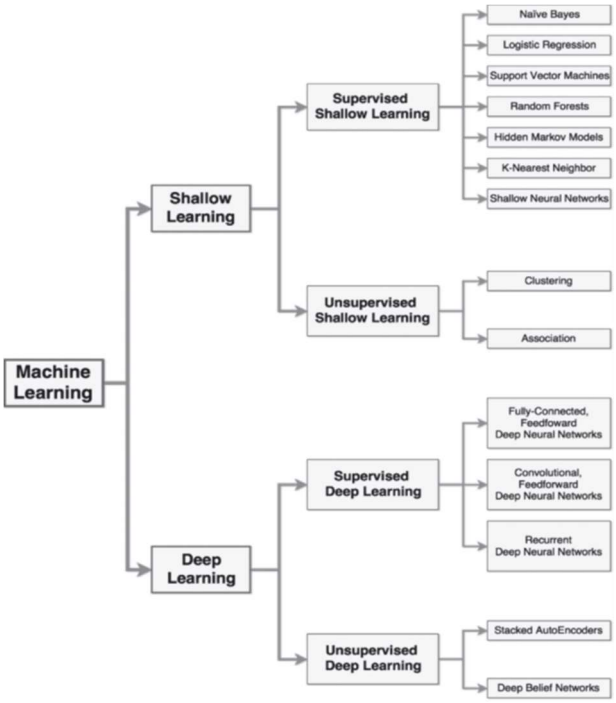

Figure 2. CLASSIFICATION OF ML ALGORITHMS FOR NETWORK SECURITY APPLICATIONS.

IV.Deep Learning Methods for Attack Detection

Considering the current deep learning methods for attack detection [15] and following the categorization of the previous works [16], we roughly divide them into three categories as well, that is, unsupervised (e.g., autoencoder (AE), deep belief network (DBN), and generative adversarial network (GAN)), supervised (e.g., deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN)), and other hybrid methods; we show the details of categorization in Figure 2. Essentially, there exist other classification criterions. For example, Berman et al. review the related deep learning methods according to attacks type and focus on how deep learning is used for various attacks. Moreover, Al-Garadi et al. offer a comprehensive view of deep learning methods based on the applications of cybersecurity. Adopting different kinds of deep learning algorithms could bring variant advantages for attack detection methods. Supervised learning based methods often result in high accuracy, due to quantity of information provided by manually labeled samples. Without sufficient knowledge from labeled data, unsupervised learning based methods are generally low in performance. However, manually labeling is a time-consuming task, especially for complex attacks. ere even exist cases that cannot be described by a simple label, due to the inherent complexity of real-world network attacks.)Therefore, unsupervised learning based methods could perform well without prior knowledge of attacks, which is an obvious advantage. Hybrid methods decrease the number of training samples and maintain a relatively high performance, which is suitable to deal with variant attack situations.In Figure 2 algorithms of Network Security Applications are classified above. However, it is generally complex in structure and high in computing time, which prevents its wide usage

A.Deep Learning Threats and Attacks

Deep learning faces various types of threats and attacks, and all famous threats and attacks are listed below.

4.1. Security Attack Taxonomy.

Ji et al. proposed classification of security threats for Deep Learning in 3 different angles, which influence classifieds, security breaches, and privacy of attacks. In the view of impact, security risks and threats of Deep Learning are characterized into two categories.

4.1.1. Causative Attack. In the causative attack used to decrease the performance and reliability of the training processes, the machine learning algorithm provided incorrect training data after

modification in the labels of the samples that are not covered under the decision limit. Many researchers performed causative attacks on the images and revealed that it expressively decreases the performance of the training phase.,is means that the opponents have the ability to change
the input of training data, which becomes the cause of changes in the parameters of the learning models during recycling, resulting in a substantial reduction in the presentation of jobs in succeeding taxonomy tasks.

4.1.2. Exploratory Attack. exploratory attacks basically do not influence on a training dataset. ,e key objective of the exploratory attacks is to get knowledge with respect to the learning algorithm as much as it can about the basic system. Model invasion attack, model extraction, and
membership inference are the examples of the exploratory attacks. In a security break viewpoint, threats to Deep Learning may be characterized into 3 groups:

(1) Integrity Attack. integrity attack occurs and then the Deep Learning models failed to trace the negative cases when categorizing harmful samples. , output of the system will clearly show that the integrity of the learning machine has been compromised. Suppose, we used spam filter to stop unwanted/harm messages, and if the attacker sends a message that has unwanted/harm words then, the filter does not get it. ,e integrity attack is tested through exploratory testing.

(2) Availability Attack. availability attack is the opposite of an integrity attack in which the Deep Learning models filtered out the legitimate cases during the categorization of the unwanted/harmful samples. ,e output of the system will clearly show that the availability of the learning machine has been compromised and it is no more available and hacked. , DoS attack is one of the examples of availability wherein legitimate cases failed to cross the filters and ultimately the system becomes compromised.

(3) Privacy Violation Attack. In the privacy violation attack, the attacker becomes successful to get the sensitive/confidential information of the system from both training and learning models. In terms of attack privacy, security threats for Deep Learning have further 02 categories.

4.1.3. Targeted Attack. It is highly dangerous, and it directly decreases the performance of the classifier in a single specific sample or set of one of the samples.

4.1.4. Indiscriminate Attack. An indiscriminate attack is the subtype of the poisoning attack. ,e attacker's key goals are to increase the general classification error. Further, the indiscriminate
attack always chooses a random value from the training sample. It randomly fails the classifier.

4.2. Deep Learning Attack Types.

Although Deep Learning becomes successful to get draw the attention of the industry its security and privacy challenges, unfortunately, it could not get full attention as it should have. Here, we discuss the attack surface of the machine learning and discuss the weaknesses in the implementation of Deep Learning. During the research, numerous types of attacks targeting DL applications and containing DoS attacks, evasion attacks, and organic termination attacks are revealed. ,Though all these attacks are different in its nature and in terms of their offensive objectives, the attacker's attack sources in Deep Learning applications are essentially from the following three angles.

4.2.1. Deep Learning Attack Surface Type-I. Deep learning application after trained mostly works on input data of the user for its

classification. ,e attacker planned a malformed input attack on the input files or sometimes the network [17]. ,is type of attack applies to image recognition application which uses files on input and also applied to the applications that use sensors and cameras on the input. Due to the input type of the application, this risk can be reduced to implement risk mitigation techniques but the risk cannot be eliminated.

4.2.2. Deep Learning Attack Surface Type-II. ,is surface attack is also called a poisoning attack. ,earlier surface type attack is due to the contaminated input data type of the application. ,is type of attack is not dependent on the application flaws or software breaches. However, defects in applications can become the reason of data poisoning easier. Suppose we observed variation in the procedure of analyzing the image in the frame and in common desktop applications, ,is variation allows the contamination of confidential data without being observed by the people who monitor the training process.

4.2.3. Deep Learning Attack Surface Type-III. It is a great chance of an attack on the Deep Learning applications if the developer will opt the model developed by the experts. Even though many programmers plan and create models from the beginning, many templates of the models exist for programmers who do not sufficient knowledge of machine learning. In this scenario, the attacker has also access to the template of the models. Like attacks of data poisoning, an attacker can easily attack all those applications and can get access to the private data that uses external models without any barrier. However, implementation flaws, such as a security vulnerability in the form analysis code, help attackers hide damaged models, readers should keep in mind that there are many types of attack surfaces and differ from each other, and it depends on the particular application, but above these types of attack, surfaces cover most of the attack area

4.3. Types of Threats. During the literature review, the authors studied many types of threats that affect the functionality of Deep Learning, and these threats targets different stages of Deep Learning. Here, in this paper, we are going to present the threat caused by the malformed input with the assumption that Deep Learning applications are taking input from files or networks.

4.3.1. Deep Learning Threat Type-I. The most common weaknesses in Deep Learning frameworks are program errors that which cause software crashes, an infinite loop, or full memory depletion. ,e immediate threat of these errors is the denial of service attacks for applications running at the top of the window .

4.3.2. Deep Learning Threat Type-II. Deep Neural Networks are vulnerable to attacks at the time of its testing [45–48]. For example, in image recognition, an attacker may insert little noise to test a sample so that the error is classified as a DNN [73]. An example of a noise test is called an adversarial example. ,e noise is usually so small for a human, benign is the alternate name of the adversarial example. Evasion attacks are one of the Deep Learning attacks that restrict sensitive security and protection applications, like vehicles that drive on their own. Examples of self-driving adversaries can make unwanted decisions . For example, one of the basic capabilities of autonomous cars is to automatically identify stop signals and traffic lights of the broad. Let us say, the adversary generates an adversarial stop, which means that the adversarial adds many imperceptible points

to the stop, so that the vehicle that is driving alone is not recognized as a stop. As a result, vehicles that drive on their own will not stop at the stop sign and may collide with other vehicles, which could lead to serious traffic accidents. ,ere are many memory corruption-related bug in Deep Learning framework which may be a cause of wrong output. , evasion can be achieved through exploiting bugs in the Deep Learning framework by overwriting classification and control flow. In order to develop an effective defense against evasion attack, Goodfellow et al. proposed adversarial training and adversarial example by introducing training of a DNN through augmenting training dataset. In order to train a DNN, the system generates training adversarial example through evasion attacks. learner understands both the original training example and relating adversarial examples. adversarial training is weak as compared with adversarial examples that cannot be seen during training. Papernot et al. developed a decontamination based technique to train Deep Neural Networks and Carlini and Wagner revealed that their generated attacks have maximum success for Deep Neural Networks trained with concentration. Furthermore, Carlini and Wagner determined that all measures must be assessed against the taxonomy of evasion attacks.

4.3.3. Deep Learning Threat Type-III. ,e software bugs of the systems that hosted Deep Learning applications on its operating system can be hijacked due to remote compromise and application bugs . ,is mostly happens when the system is connected with the cloud system and the Deep Learning applications are also running on that cloud-based system. All the input to the Deep Learning system is received through the network.

V.TRAFFIC ANALYSIS

The major obstacles researchers face when training NN are centered around the data itself. The limited availability of labeled data decreases the accuracy in classification and limits the choices of algorithms since DL techniques often require large amount of data for training. Finding a way to combine supervised learning with supervised learning to teach NN how to learn with fewer data is a promising area of research. Furthermore, teaching NN to accumulate its knowledge will make it more effective and efficient in learning new things, and thus, less data will be required for training.

A. Traffic Classification

Network traffic involves encrypted/encapsulated flow packets which hide the features of the flows. Classifying such traffic requires advanced DL methods that can reveal hidden patterns.

The evolution in the networking architecture has brought flexibility and extensibility. However, the decoupling of data and control planes has also made the network more prone to security issues. For example, since the network is managed by a single controller, overloading it with malicious flows creates a challenging problem. To address this problem, DL algorithms can be used more often in detecting suspicious flows and anomaly based attacks.

B. Traffic Prediction

Traffic prediction is necessary in providing high quality communication over the network. Forecasting possible congestion will enable a solution to be offered before QoS/QoE drops. RNN can be applied for prediction analysis since it will use historical data to make better decisions and therefore achieve higher accuracy. Additionally, foreseeing a possible elephant flow

occurrence at unusual times which can most probably be labeled as a flow-based intrusion, will provide a more secure network. In addition to that, the prediction of such elephant flows can also eradicate the risk of overburdening the controller in SDN. Moreover, traffic prediction will enable determining the possible congestion on the links before they lower the QoS & QoE and route the traffic to the less congested links. Exploiting DL algorithms for this manner can make the routing process more intelligent and autonomous, therefore sophisticated enough for SDN. Hence, routing optimization with DL is a significant research problem.

C. Comparisons and Analysis

5.1. Public Datasets. Many public datasets are popular to prove and compare efficiency and effectiveness among
different attack detection methods. Among them, we list two famous benchmark datasets, that is, KDDCup 99 and NSLKDD, which are widely used in the academic research to evaluate the ability to detect attacks.

5.1.1. KDDCup 99 Dataset. Despite the fact that there exist some drawbacks like containing a great deal of redundant training and testing data, KDDCup 99 dataset is famous in the field of cybersecurity. It includes both labeled training data and unlabeled test data, which correspond to seven and two weeks of data originated from DARPA′98 IDS evaluation program [18]. Five categories of labels are contained in the dataset which are normal, DoS, Probe, R2L and U2R, that is, short for DoS, Probe, R2L, and U2R, where normal refers to normal traffic instances, Dos is an attack in which the attacker tries to make the target machine stop providing service or resource access to system, Probe represents surveillance and probing, and R2L refers to the unauthorized access while there is an illegal access from the remote machine to local one and represents that there is an unauthorized access to local superuser privileges by local unprivileged user. In Table 1, we display 22 different attacks in training and test data, which could be categorized into these four attack types.

Table1. Category of 22 different attacks contained by KDDCup 99

Class Label	Attack Name
Dos	**back,land,neptune,pod,smurf,rear drop**
Probe	**ipsweep,nmap,portsweep,satan**
R2l	**ftp_write,guess_paawd,,imap,multihop,pfp,spy,warezclient,warezmaster**
U2R	**buffer_overflow,loadmodule,perl,rootkit**

.

5.1.2. NSL-KDD Dataset. NSL-KDD is famous as a new development of KDDCup 99 dataset, which comes out to reduce shortcomings of the previous dataset. Specifically, it not only removes redundant data from the training and test data to achieve more accurate detection rate but also officially sets the number of records in both training and test data. Moreover, different difficulty level group has different number of records, which is inversely proportional to the percentage of that in the primary KDD dataset. Hence, evaluations and comparisons among different learning
technologies become more effective and obvious NSL-KDD and KDDCup 99 dataset are similar in structure, where both of them are divided into four attack types as mentioned before. NSL-KDD dataset is divided into two parts: KDDTrain+ and KDDTest+, where we show the

specific numbers corresponding to each attack type in Table 2. It is noted that there are 17 attack types in KDDTest+, which do not appear in KDDTrain+.)is interesting setting makes NSL-KDD more challenging than KDDCup 99 dataset, which imitates real-life network environment with unknown attacks. We believe only these learning methods built on realistic theoretical basis, that is, analyzing inherent characteristics of attack behaviors, would achieve promising results on NSL-KDD

Table 2. Records distribution in training and test data[20]

class	KDD Train+	KDD Test+
Dos	**45927**	**74588**
Probe	**11656**	**2421**
R2l	**995**	**2754**
U2R	**52**	**200**

VI.PERFORMANCE METRICS

A. Accuracy

It is the number of correctly predicted data points out of all the data points and it is represented in terms of percentage. The accuracy is calculated as shown in Equation 1.

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN} \quad \text{EQ-----(1)}$$

Where

False positive (FP): It defined as the number of detected attacks which are actually normal, and False negative (FN): means the wrong prediction i.e. it detects the instances as normal but in actual i ts an attack.

True positive (TP): is an instances that are correctly predicted as normal, and

True negative (TN): is an instances that are correctly classified or detected as attack.

B. Precision

It is a measure which estimates the probability that a positive prediction is correct, and the formula is as shown in Equation 2.

$$Precision=\frac{TP}{TP+FP} \quad \text{EQ----(2)}$$

C. Recall

It is the proportion of instances belonging to the positive class that are correctly predicted as positive, and the formula is as shown in Equation 3.

$$Recall=\frac{TP}{TP+FN} \quad \text{EQ----(3)}$$

D. Kappa

Its value ranges from 0 to 1. 0 means totally disagreement and full agreement. It checks the reliability of classifying algorithm on dataset.

E. Receiver Operating Characteristics (ROC)

It is used to design the curve between true positive rate and false positive rate, and the Area Under Curve (AUC) gives the value of ROC. More the area under curve and more will be the value of ROC.

F. F1 Score

The F1 score or F-measure is a measure of the test's accuracy. It considers both the precision P and the recall R of the test to compute the score.

VII. RESULTS

The experiment was conducted on an HP Pavilion 14-AL143TX loaded with the Windows 10 Operating System with the following processor: Intel(R) Core(TM) i5-7200U CPU @ 2.5–3.1 GHz. The building, training, and evaluation of the Machine Learning model were performed by Pandas, NumPy, Scikit-Learn

(sklearn), etc. Machine Learning libraries in the python environment of Jupyter Notebook, an open-source tool.

A.PERFORMANCE ANALYSIS

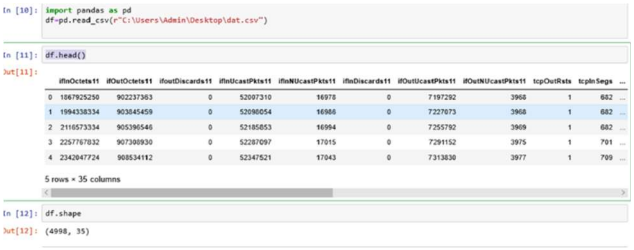

Figure 3.Read .csv file

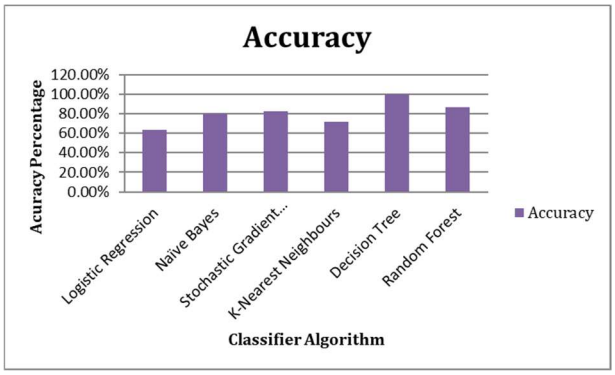

Figure 4. Accuracy chart for Classification Algorithm for KDD Cup dataset

Classification Algorithms	Accuracy
Logistic Regression	63.82%
Naïve Bayes	80.11%
Stochastic Gradient Descent	82.20%
K-Nearest Neighbours	72.10%
Decision Tree	100.00%
Random Forest	86.70%

Table 3. Classification Algorithm- Accuracy analysis

B.ATTACK ANALYSIS

1) User to Root attack

Performance of the selected machine learning classifiers against the User to Root attack

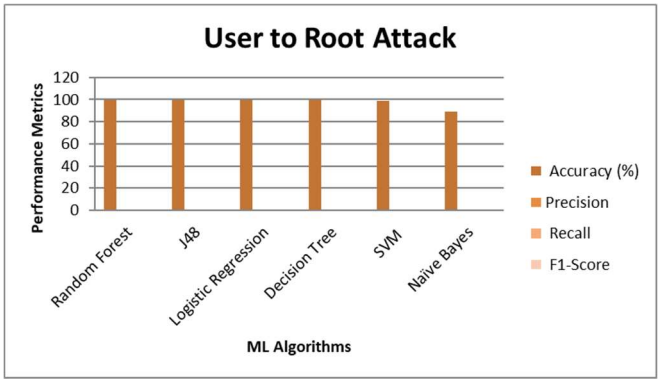

Figure 5 .User to Root attack

2) Remote to Local attack

Performance of the selected machine learning classifiers against the Remote to Local attack

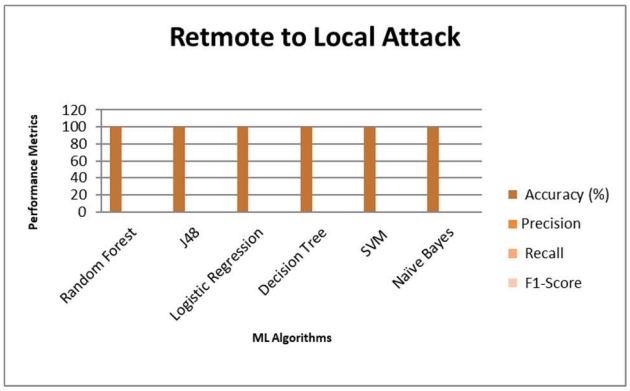

Figure 6. Remote to Local attack

3) DoS attack

Performance of the selected machine learning classifiers against the DoS attack

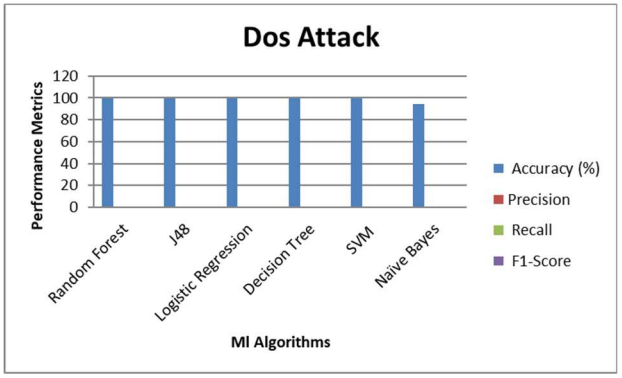

Figure 7.DoS attack

4) Probe attack

Performance of the selected machine learning classifiers against the Probe attack

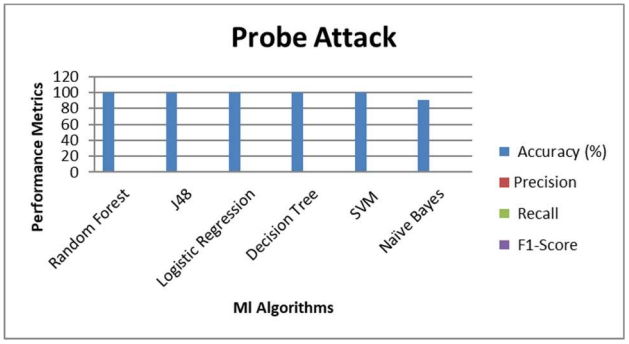

Figure 8. Probe attack

VIII.CONCLUSION

In this paper, we have conducted a comprehensive overview of machine learning algorithms for intelligent data analysis and applications. According to our goal, we have briefly discussed how various types of machine learning methods can be used for making solutions to various real-world issues. A successful machine learning model depends on both the data and the performance of the learning algorithms. The sophisticated learning algorithms then need to be trained through the collected real-world data and knowledge related to the target application before the system can assist with intelligent decision-making. We also discussed several popular application areas based on machine learning techniques to highlight their applicability in various real-world issues. Finally, we have summarized and discussed the challenges faced and the potential research opportunities and future directions in the area. Therefore, the challenges that are identified create promising research opportunities in the field which must be addressed with effective solutions in various application areas. Overall, we believe that our study on machine learning-based solutions opens up a promising direction and can be used as a reference guide for potential research and applications for both academia and industry professionals as well as for decision-makers, from a technical point of view. Over the past few years, research on how to apply deep learning methods on attack detection has made a great progress. However, many problems still exist. Firstly, it is challenging to modify deep learning methods as real-time classifiers for attack detection. In most of the previous works, they only reduce feature dimension for less computation cost during phase of feature extraction. Secondly, most of the deep learning techniques are appropriate for analysis of image and pattern recognition.)us, how to conduct the classification of network traffic reasonably with deep learning techniques will be an interesting issue.)thirdly, with more data involving the experiments, the classification results will be better . However, most of the attack detection problems are short of sufficient data According to the above analysis, we hold a belief that this overview is a benefit for those who have ideas to improve the performance of attack detection in terms of accuracy; our review will provide guidance and dictionaries for further research in this field.

REFERENCES

[1] X.-Y. Yang, J. Liu, M.-Q. Zhang, and K. Niu, “A new multi-class svm algorithm based on one-class svm,” in International Conference on Computational Science. Springer, 2007, pp. 677–684.

[2] L. G´omez-Chova, G. Camps-Valls, J. Munoz-Mari, and J. Calpe, “Semisupervised image classification with laplacian support vector machines,” IEEE Geoscience and Remote Sensing Letters, vol. 5, no. 3, pp. 336–340, 2008.

[3] B. Raghavan, M. Casado, T. Koponen, S. Ratnasamy, A. Ghodsi, and S. Shenker, "Software-defined internet architecture: decoupling architecture from infrastructure," in Proceedings of the 11th ACM Workshop on Hot Topics in Networks. ACM, 2012, pp. 43–48.

[4] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and
J. Wilcox, "Intelligent design enables architectural evolution," in Proceedings of the 10th ACM Workshop on Hot Topics in Networks. ACM, 2011, p. 3.

[5] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama et al., "Onix: A distributed control platform for large-scale production networks." in OSDI, vol. 10, 2010, pp. 1–6.

[6] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. henker, and J. Turner, "Openflow: enabling innovation in campus networks," ACM SIGCOMM omputer Communication Review, vol. 38, no. 2, pp. 69–74, 2008.

[7] Dimitrios Papamartzivanos, Félix Gómez Mármol, Georgios Kambourakis, Introducing
deep learning self-adaptive misuse network intrusion detection systems, IEEE Access 7 (2019) 13546–13560.

[8] Peng Jiang, Hongyi Wu, Cong Wang, Chunsheng Xin, Virtual MAC spoofing detection through deep learning, in: 2018 IEEE International Conference on Communications (ICC), IEEE, 2018, pp. 1–6.

[9] Sheraz Naseer, Yasir Saleem, Shehzad Khalid, Muhammad Khawar Bashir, Jihun Han, Muhammad Munwar Iqbal, Kijun Han, Enhanced network anomaly detection based on deep neural networks, IEEE Access 6 (2018) 48231–48246.

[10] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, Ali A Ghorbani, A detailed analysis of the KDD cup 99 data set, in: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, IEEE, 2009, pp. 1–6.

[11] Ritesh K Malaiya, Donghwoon Kwon, Jinoh Kim, Sang C Suh, Hyunjoo Kim, Ikkyun Kim, An empirical evaluation of deep learning for network anomaly detection, in: 2018 International Conference on Computing, Networking and Communications (ICNC), IEEE, 2018, pp. 893–898.

[12] Sahil Garg, Kuljeet Kaur, Neeraj Kumar, Georges Kaddoum, Albert Y Zomaya, Rajiv Ranjan, A hybrid deep learning-based model for anomaly detection in cloud datacenter networks, IEEE Trans. Netw. Serv. Manag. 16 (3) (2019) 924–935.

[13] Mahmood Yousefi-Azar, Vijay Varadharajan, Len Hamey, Uday Tupakula, Autoencoder-based feature learning for cyber security applications, in: 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 3854–3861.

[14] Vrizlynn L.L. Thing, IEEE 802.11 network anomaly detection and attack classification:
A deep learning approach, in: 2017 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2017, pp. 1–6.

[15] Yuanfang Chen, Yan Zhang, Sabita Maharjan, Muhammad Alam, Ting Wu, Deep learning for secure mobile edge computing in cyber-physical transportation systems, IEEE Netw. 33 (4) (2019) 36–41.

[16] Khoi Khac Nguyen, Dinh Thai Hoang, Dusit Niyato, Ping Wang, Diep Nguyen, Eryk Dutkiewicz, Cyberattack detection in mobile cloud computing: A deep learning approach, in: 2018 IEEE Wireless Communications and

Networking Conference (WCNC), IEEE, 2018, pp.1–6.
[17] Jiangang Shu, Lei Zhou, Weizhe Zhang, Xiaojiang Du, Mohsen Guizani, Collaborative intrusion detection for VANETs: A deep learning-based distributed SDN approach, IEEE Trans. Intell. Transp. Syst. (2020).
[18] Mahmoud Abbasi, Amin Shahraki, Hamid R Barzegar, Claus Pahl, Synchronization Techniques in ‘‘Device to Device-and Vehicle to Vehicle-Enabled’’ Cellular Networks: A survey, Comput. Electr. Eng., 90 106955.
[19] Wei Zhong, Feng Gu, A multi-level deep learning system for malware detection, Expert Syst. Appl. 133 (2019) 151–162.
[20] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the kdd cup 99 data set,” in Proceedings of 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, IEEE, Ottawa, Canada, pp. 1–6, July 2009