IOSR Journal of Engineering (IOSR JEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 37-45

Machine Learning And Deep Learning Method For Computer Data Prevention

[1]Ms. Diksha Shankarrao Burde, [2]Mr. Dhiraj Rane
[1]Department of Computer Science, MCA-sem5, RTMNU, Nagpur, Maharashtra.
[2]Department of Computer Science, MCA, RTMNU, Nagpur, Maharashtra.

Abstract: *Machine learning and deep learning is adopted in a wide range of domains where it shows its superiority over traditional rule-based algorithms. These methods are being accommodatedfor computer data prevention systems with the goal of supporting or even replacing the first level of security analysts. Although the complete automation of detection and analysis is an enticing goal, the efficacy of machine learning and deep learning in computer data prevention must be evaluated with the due intensity. We present an analysis, addressed to security specialists, of machine learning techniques applied to the detection of intrusion, malware, and spam. The goal is twofold: to assess the current maturity of these solutions and to identify their main limitations that prevent an immediate adoption of machine learning and deep learning data prevention schemes. Our conclusions are based on an extensive review of the literature as well as on experiments performed on real enterprise systems.*

Keywords: *Machine learning, deep learning, Computer data prevention*

I. Introduction

With the increasingly in-depth assimilation of the Internet and social issue, the Internet is changing how the people learn and work, but it also exposes us to increasingly serious data threat. How to identify various network attacks, particularly not previously seen attacks, is a key issue to be solved urgently.Computer data prevention is the collection of polices,techniques, technologies and processes that work together to protect the confidentiality, integrity, and availability of computing resources, networks, software programs, and data from attack. Cyber defences mechanisms exist at the application, network, host, and data level. Prevention breaches include exterior interference and interior interference. There are three main types of network analysis for Intrusion Detection System: misuse-based, also known as signature-based, anomaly-based, and hybrid. Misuse-based detection techniques aim to detect known attacks by using the signatures of these attacks .They are used for known types of attacks without generating a large number of false alarms. However, administrators often must manually update the database rules and signatures. There are such a plethora of tools—such as firewalls, antivirus software and intrusion protection systems (IPSs)—that work in safe to prevent attacks and detect security breaches.

Exception-based techniques study the normal network and system behavior and identify anomalies as deviations from normal behavior. They are appealing because of their capacity to detect zero-day attacks. Another advantage is that the profiles of normal activity are customized for every system, application, or network, therefore making it difficult for attackers to know which activities they can perform undetected. Additionally, the data on which exception-based techniques alert can be used to define the signatures for misuse detectors. The main disadvantage of exception-based techniques is the potential for high false alarm rates because previously unseen system behaviors can be categorized as anomalies. Hybrid detection combines misuse and anomaly detection. It is used to increase the detection rate of known intrusions and to reduce the false positive rate of unknown attacks. Most Machine Learning/Deep Learning methods are amalgam.

This paper presents a literature review of machine learning (ML) and deep learning (DL) methods for computer data prevention. ML/DL methods and some applications of each method in network intrusion detection are described. It focuses on ML and DL technologies for network security, ML/DL methods and their descriptions. Our research aims on standards-compliant publications that use ‘‘machine learning’’, ‘‘deep learning’’ and data prevention as keywords to search on Google Scholar. In particular, the new hot papers are used because they describe the popular techniques. The purpose of this paper is for those who want to study network intrusion detection in ML/DL.Thus great emphasis is placed on a thorough description of the ML/DL.

Moreover, we point out a general underestimation of the complexity of managing Machine Learning and Deep Learning architectures in computer data prevention caused by the lack of publicly available and labeled data for training, and by the time required for fine-tuning operations in a domain characterized by continuous change. The evidenced drawbacks gravel the way to future improvements that ML and DL components require before being fully adopted in standalone defense platforms.

II. Similarities And Differnce In Ml And Dl

There are many puzzles about the relationship among Machine Learning, Deep Learning .It is a branch of computer science that seeks to understand the essence of intelligence and to produce a new type of intelligent machine that responds in a manner similar to human intelligence. Research in this area includes computer vision, nature language processing and expert systems. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. ML is occasionally conflated with data mining, but the latter subfield focuses more on exploratory data analysis and is known as unsupervised learning. ML can also be unsupervised and be used to learn and establish baseline behavioral profiles for various entities and then used to find meaningful anomalies. ML as a ‘‘field of study that gives computers the ability to learn without being explicitly programmed.’’ ML primarily focuses on classification and regression based on known features previously learned from the training data. DL is a new field in machine-learning research. Its motivation lies in the establishment of a neural network that simulates the human brain for analytical learning. It mimics the human brain mechanism to interpret data such as images, sounds and texts. The concept of DL was proposed by Hinton based on the deep belief network (DBN), in which an unsupervised greedy layer-by-layer training algorithm is proposed that provides hope for solving the optimization problem of deep structure. Then the deep structure of a multi-layer automatic encoder is proposed. DL is a machine-learning method based on characterization of data learning. An observation, such as an image, can be expressed in a variety of ways, such as a vector of each pixel intensity value, or more abstractly as a series of edges, a region of a particular shape, or the like. Using specific representations makes it easier to learn tasks from instances. Similarly to ML methods, DL methods also have supervised learning and unsupervised learning. Learning models built under different learning frameworks are quite different. The benefit of DL is the use of unsupervised or semi-supervised feature learning and hierarchical feature extraction to efficiently replace features manually.

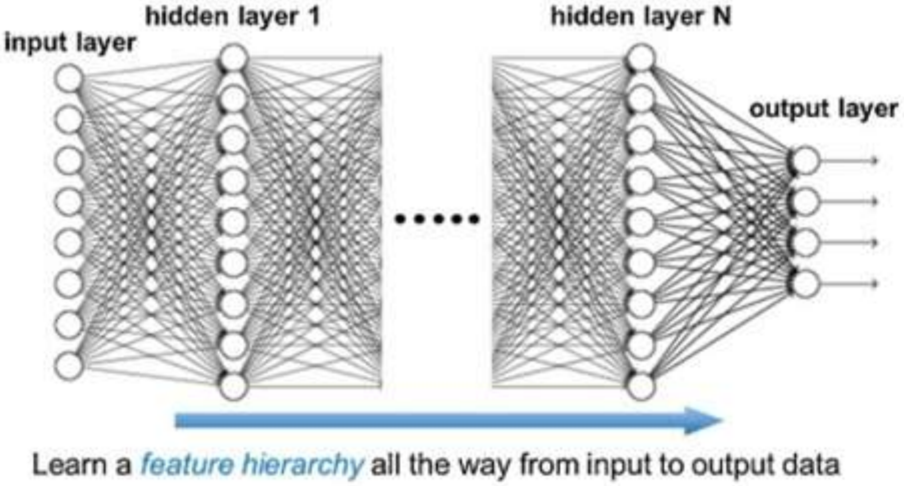
hidden layer 1
input layer
hidden layer N
output layer
Learn a feature hierarchy all the way from input to output data

Figure 1: ***Deep Belief Network***

The differences between ML and DL include the following:

a. Data Dependencies

Performance is the main key difference between both algorithms. Although, when the data is small, Deep Learning algorithms don’t perform well. This is the only reason DL algorithms need a large amount of data to understand it perfectly.

b. Hardware Dependencies

Generally, Deep Learning depends on high-end machines while machine learning depends on low-end machines. Thus, Deep Learning requirement includes GPUs. That is an integral part of its working. They also do a large amount of matrix multiplication operations.

c. Feature Engineering

It’s a general process. In this, domain knowledge is put into the creation of feature extractors to reduce the complexity of the data and make patterns more visible to learn the algorithm working. Although, it’s very difficult to process. Hence, it’s time consuming and expertise.

d. Problem Solving Approach

Generally, we use the machine algorithm to solve problems. However, it needs to break a problem into different parts to solve them individually. To get a result, combine them all.

For Example:
Let us suppose you have a task of multiple object detection. In this task, we have to identify what the object is and where is it present in the image. In a Machine Learning approach, we have to divide the problem into two steps:

- object detection
- object recognition

First, we use the grab cut algorithm to skim through the image and find all the possible objects. Then, of all the recognized objects, you would use an object recognition algorithm like SVM with HOG to recognize relevant objects.

e. Execution Time
Usually, Deep Learning takes more time to train as compared to Machine Learning. The main reason is that there are so many parameters in a Deep Learning algorithm. Whereas Machine Learning takes much less time to train, ranging from a few seconds to a few hours.

f. Interpretability
We have interpretability as a factor for comparison of both learning techniques. Although, Deep Learning is still thought 10 times before its use in industry.

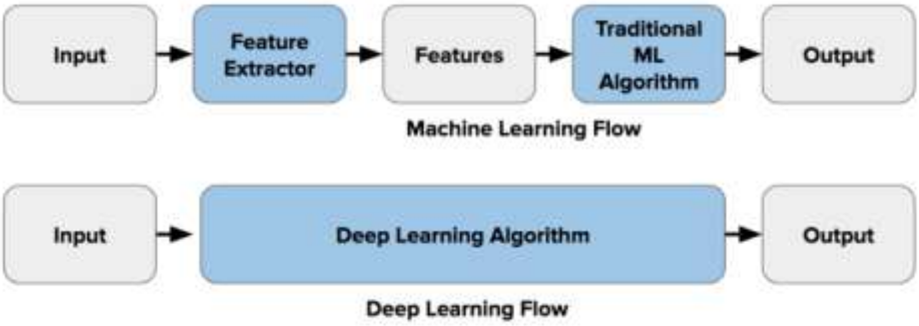

Figure 2: ***Differencebetween ML and DL***

III. Network Security Data Set

Data constitute the basis of computer network security research. The correct choice and reasonable use of data are the prerequisites for conducting relevant security research. The size of the dataset also affects the training effects of the ML and DL models. Computer network security data can usually be obtained in two ways: 1) directly and 2) using an existing public dataset. This approach is highly targeted and suitable for collecting short-term or small amounts of data, but for long-term or large amounts of data, acquisition time and storage costs will escalate. The use of existing network security datasets can save data collection time and increase the efficiency of research by quickly obtaining the various data required for research.

A. Darpa Intrusion Detection Data Sets

DARPA Intrusion Detection Data Sets [21], which are under the direction of DARPA and AFRL/SNHS, are collected and published by The Cyber Systems and Technology of MIT Lincoln Laboratory for evaluating computer network intrusion detection systems. The first standard dataset provides a large amount of background traffic data and attack data. It can be downloaded directly from the website. Currently, the dataset primarily includes the following three data subsets:

· 1998 DARPA Intrusion Detection Assessment Dataset: Includes 7 weeks of training data and 2 weeks of test data.

· 1999 DARPA Intrusion Detection Assessment Dataset: Includes 3 weeks of training data and 2 weeks of test data.

· 2000 DARPA Intrusion Detection Scenario-Specific Dataset: Includes LLDOS 1.0 Attack Scenario Data, LLDOS 2.0.2 Attack scenario data, Windows NT attack data

B. KDD CUP 99 DATASET

KDD Cup’99 dataset used for benchmarking intrusion detection problem is used in our experiment. These are generated by processing the tcpdump segment of DARPA 1998 evaluation data set. This data set consists of 41 feature and separate feature (42nd feature) that labels the connection as ‘normal’ or a type of attack [11]. The data set contains a total of 23 attack, these are grouped into 4 major categories:

- Denial-of-Service (DoS)
- Probing or Surveillance
- User-to-Root (U2R)
- Remote-to-Local (R2L)

C. NSL-KDD DATASET

NSL-KDD is a data set suggested to solve some of the inherent problems of the KDD'99 data set which are mentioned in. Although, this new version of the KDD data set still suffers from some of the problems discussed by McHugh and may not be a perfect representative of existing real networks, because of the lack of public data sets for network-based IDSs, we believe it still can be applied as an effective benchmark data set to help researchers compare different intrusion detection methods.

- **KDDTrain+.ARFF**: The full NSL-KDD train set with binary labels in ARFF format
- **KDDTrain+.TXT**: The full NSL-KDD train set including attack-type labels and difficulty level in CSV format
- **KDDTrain+_20Percent.ARFF**: A 20% subset of the KDDTrain+.arff file
- **KDDTrain+_20Percent.TXT**: A 20% subset of the KDDTrain+.txt file
- **KDDTest+.ARFF**: The full NSL-KDD test set with binary labels in ARFF format
- **KDDTest+.TXT**: The full NSL-KDD test set including attack-type labels and difficulty level in CSV format
- **KDDTest-21.ARFF**: A subset of the KDDTest+.arff file which does not include records with difficulty level of 21 out of 21
- **KDDTest-21.TXT**: A subset of the KDDTest+.txt file which does not include records with difficulty level of 21 out of 21

IV. Compariosn With Algorithm

This section is divided into two parts. The first part introduces the application of traditional machine-learning algorithms in network security. The second part introduces the application of deep learning in the field of computer data prevention. It not only describes the research results but also compares similar studies.

A. SUPPORT VECTOR MACHINE

The objective of the support vector machine algorithm is to find a hyper plane in an N-dimensional space(N—the number of features) that distinctly classifies the data points.To separate the two classes of data points, there are many possible hyper planes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

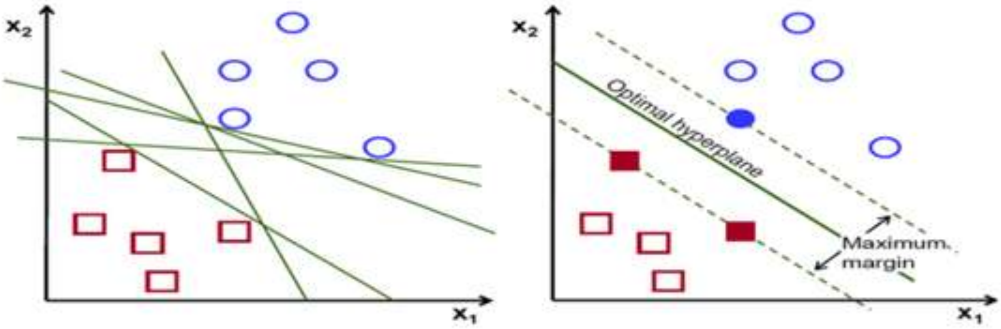

Figure 3: ***Possible Hyper planes***

There are 2 kinds of SVM classifiers:

1. Linear SVM Classifier
2. Non-Linear SVM Classifier

Svm Linear Classifier:

In the linear classifier model, we assumed that training examples plotted in space. These data points are expected to be separated by an apparent gap. It predicts a straight hyper plane dividing 2 classes. The primary focus while drawing the hyper plane is on maximizing the distance from hyper plane to the nearest data point of either class. The drawn hyper plane called as a maximum-margin hyper plane.

SVM Non-Linear Classifier:

In the real world, our dataset is generally dispersed up to some extent. To solve this problem separation of data into different classes on the basis of a straight linear hyper plane can't be considered a good choice.

Linear Support Vector Machine Classifier

In Linear Classifier, A data point considered as a p-dimensional vector (list of p-numbers) and we separate points using (p-1) dimensional hyper plane. There can be many hyper planes separating data in a linear order, but the best hyper plane is considered to be the one which maximizes the margin i.e., the distance between hyper plane and closest data point of either class.

The Maximum-margin hyper plane is determined by the data points that lie nearest to it. Since we have to maximize the distance between hyper plane and the data points. These data points which influences our hyperplane are known as support vectors.

Non-Linear Support Vector Machine Classifier

It often happens that our data points are not linearly separable in a p-dimensional (finite) space. To solve this, it was proposed to map p-dimensional space into a much higher dimensional space.Every kernel holds a non-linear kernel function.

This function helps to build a high dimensional feature space. There are many kernels that have been developed. Some standard kernels are:

1. **Polynomial (homogeneous) Kernel:** The polynomial kernel function can be represented by the above expression. Where $k(x_i, x_j)$ is a kernel function, x_i &x_j are vectors of feature space and d is the degree of polynomial function.
2. **Polynomial (non-homogeneous) Kernel:** In the non-homogeneous kernel, a constant term is also added. The constant term "c" is also known as a free parameter. It influences the combination of features. x& y are vectors of feature space.
3. **Radial Basis Function Kernel:** It is also known as RBF kernel. It is one of the most popular kernels. For distance metric squared Euclidean distance is used here. It is used to draw completely non-linear hyper planes.

Advantages of SVM Classifier:

- SVMs are effective when the number of features is quite large.
- It works effectively even if the number of features is greater than the number of samples.
- Non-Linear data can also be classified using customized hyper planes built by using kernel trick.
- It is a robust model to solve prediction problems since it maximizes margin.

Disadvantages of SVM Classifier:

- The biggest limitation of Support Vector Machine is the choice of the kernel. The wrong choice of the kernel can lead to an increase in error percentage.
- With a greater number of samples, it starts giving poor performances.
- SVMs have good generalization performance but they can be extremely slow in the test phase.
- SVMs have high algorithmic complexity and extensive memory requirements due to the use of quadratic programming.

B. K-NEARESTNEIGHBOR

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

It is widely disposable in real-life scenarios since it is non-parametric, meaning; it does not make any underlying assumptions about the distribution of data.

Algorithm

Letm be the number of training data samples. Let p be an unknown point.

1. Store the training samples in an array of data points arr[]. This means each element of this array represents a tuple (x, y).
2. for i=0 to m:
3. Calculate Euclidean distance d(arr[i], p).
4. Make set S of K smallest distances obtained. Each of these distances correspond to an already classified data point.

5. Return the majority label among S.

The kNN classifier is based on a distance function that measures the difference or similarity between two instances. The standard Euclidean distance d(x, y) between two instances x and y is defined as:
$d(x,y)=\surd\sum_{k=1}^{n}(xk-yk)\ 2$
where, xk is the kth featured element of instance x, yk is the k th featured element of the instance y and n is the total number of features in the dataset.

C. Decision Tree

Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The tree can be explained by two entities, namely decision nodes and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.

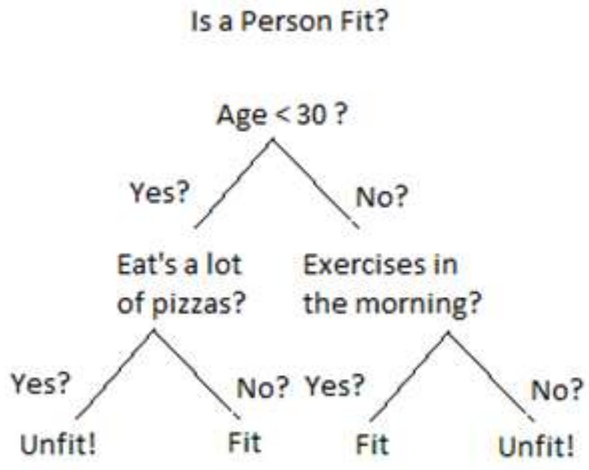

Figure 3: ***Example of Decision Tree***

An example of a decision tree can be explained using above binary tree. Let’s say you want to predict whether a person is fit given their information like age, eating habit, and physical activity, etc. The decision nodes here are questions like ‘What’s the age?’, ‘Does he exercise?’, ‘Does he eat a lot of pizzas’? And the leaves, which are outcomes like either ‘fit’, or ‘unfit’. In this case this was a binary classification problem (a yes no type problem).

Disadvantages of Decision Tree.

1. There is a high probability of **over fitting in** Decision Tree.
2. Generally, it gives low prediction accuracy for a dataset as compared to other machine learning algorithms.
3. Information gain in a decision tree with categorical variables gives a biased response for attributes with greater no. of categories.
4. Calculations can become complex when there are many **class labels**.

Advantages of Decision Tree.

1. Decision Trees are easy to explain. It results in a set of rules.
2. It follows the same approach as humans generally follow while making decisions.
3. Interpretation of a complex Decision Tree model can be simplified by its visualizations. Even a naive person can understand logic.
4. The Number of hyper-parameters to be tuned is almost null.

D. Deep Belief Network

DBN is a Unsupervised Probabilistic Deep learning algorithm.DBN id composed of multi layer of stochastic latent variables. Latent variables are binary, also called as feature detectors or hidden unitsDBN is a generative hybrid graphical model. Top two layers are undirected. Lower layers have directed connections from layers above.

Architecture of DBN

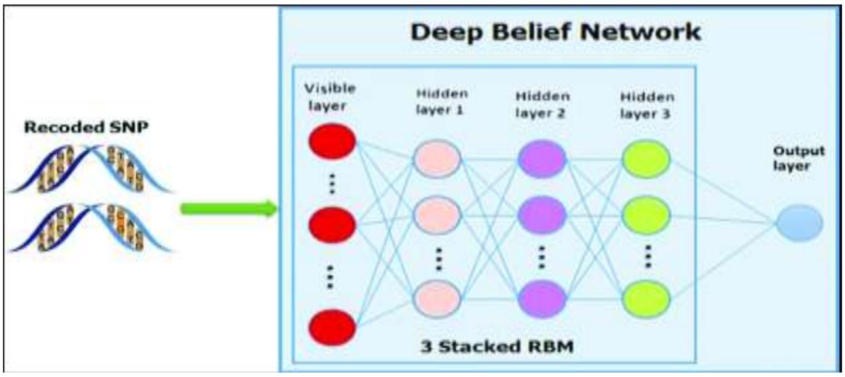

Figure 4: ***Architecture of DBN***

Deep Belief Network

- It is a stack of Restricted Boltzmann Machine (RBM) or Auto encoders.
- Top two layers of DBN are undirected, symmetric connection between them that form associative memory.
- The connections between all lower layers are directed, with the arrows pointed toward the layer that is closest to the data. Lower Layers have directed acyclic connections that convert associative memory to observed variables. The lowest layer or the visible units receives the input data. Input data can be binary or real.
- There are no intra layer connections likes RBM
- Hidden units represents features that captures the correlations present in the data
- Two layers are connected by a matrix of symmetrical weights W.
- Every unit in each layer is connected to every unit in the each neighboring layer

E. Recurrent Neural Networks

Recurrent Neural Networks (RNN) are a powerful and robust type of neural networks and belong to the most promising algorithms out there at the moment because they are the only ones with an internal memory.

RNN's are relatively old, like many other deep learning algorithms. They were initially created in the 1980's, but can only show their real potential since a few years, because of the increase in available computational power, the massive amounts of data that we have nowadays and the invention of LSTM in the 1990's.

Because of their internal memory, RNN's are able to remember important things about the input they received, which enables them to be very precise in predicting what's coming next.

This is the reason why they are the preferred algorithm for sequential data like time series, speech, text, financial data, audio, video, weather and much more because they can form a much deeper understanding of a sequence and its context, compared to other algorithms.

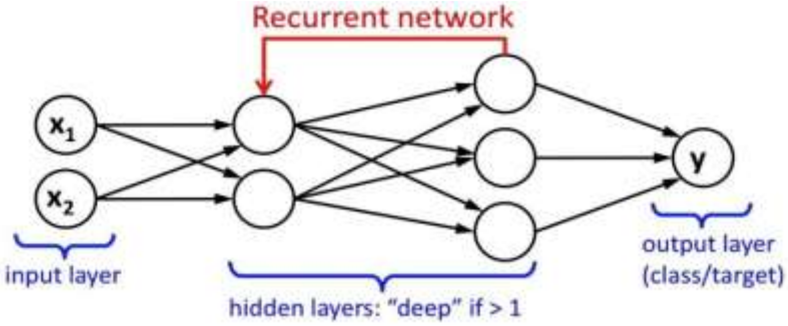

Figure 5: ***Architecture of RNN***

V. Future Scope

Our work examines a large number of academic intrusion detection studies based on machine learning and deep learning as shown in Table 5. In these studies, many imbalances appear and expose some of the problems in this area of research, largely in the following areas:

(i) The benchmark datasets are few, although the same dataset is used, and the methods of sample extraction used by each institute vary.

(ii) The evaluation metrics are not uniform, many studies only assess the accuracy of the test, and the result is one-sided. However, studies using multi-criteria evaluation often adopt different metric combinations such that the research results cannot be compared with one another.
(iii) Less consideration is given to deployment efficiency, and most of the research stays in the lab irrespective of the time complexity of the algorithm and the efficiency of detection in the actual network.

The problems and trends described above also provide a future for intrusion detection research:

A. Data Sets

Existing datasets have the defects of old data, redundant information and unbalanced numbers of categories. Although the data can be improved after processing, there is a problem of insufficient data volume. Therefore, establishing network intrusion detection datasets with large amounts of data, wide-type coverage and balanced sample numbers of attack categories becomes a top priority in the field of intrusion detection.

B. Hybrid Method

Hybrid detection methods mostly combine machine-learning methods such as those described by whereas intrusion detection with a combination of deep learning and machine-learning methods is less studied

C. Detection Speed

By reducing the detection time and improving the detection speed from the algorithm and hardware aspects, the algorithm can be used less time given the complexity of the machinelearning algorithm and deep learning algorithm. Hardware can use multiple computers for parallel computing. Combining the two approaches is also an interesting topic.

VI. Conclusion

This paper presents a literature review of ML and DL methods for computer data prevention. The paper, which has mostly focused on the last three years, introduces the latest applications of ML and DL in the field of intrusion detection. Each approach to implementing an intrusion detection system has its own advantages and disadvantages, a point apparent from the discussion of comparisons among the various methods. Thus, it is difficult to choose a particular method to implement an intrusion detection system over the others. Datasets for network intrusion detection are very important for training and testing systems. The ML and DL methods do not work without representative data, and obtaining such a dataset is difficult and time-consuming. However, there are many problems with the existing public dataset, such as uneven data, outdated content and the like. These problems have largely limited the development of research in this area. Network information update very fast, which brings to the DL and ML model training and use with difficulty, model needs to be retrained long-term and quickly. So incremental learning and lifelong learning will be the focus in the study of this field in the future.

References

[1]. Zhi Liu (liuzhi@sdu.edu.cn) (2018) ,"Machine Learning and Deep Learning Methods for Cybersecurity", Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8359287

[2]. Multiple Authors ,White Paper (2019),"MACHINE-LEARNING ERA IN CYBERSECURITY", Available:https://www.welivesecurity.com/wpcontent/uploads/2019/02/ESET_MACHINE_LEARNING_ERA.pdf

[3]. A. F. Agarap. (2017). ''A neural network architecture combining gated recurrent unit (GRU) and support vector machine (SVM) for intrusion detection in network traffic data.'' [Online]. Available: https://arxiv.org/abs/1709.03082

[4]. Vinayakumar R1, Barathi Ganesh HB1,2, Prabaharan Poornachandran3, Anand Kumar M4 and Soman KP1 1Centre for Computational Engineering and Networking (CEN), Amrita School of Engineering, Coimbatore 2Arnekt Solutions Pvt Ltd, Pune, Maharashtra, India

[5]. 3Center for Cyber Security Systems and Networks, Amrita School of Engineering, Amritapuri, AmritaVishwa Vidyapeetham, India 4Department of Information Technology, National Institute of Technology, Karnataka, Surathkal. Mangalore. Available: file:///D:/idm/1812.03519.pdf

[6]. R. B. Krishnan and N. R. Raajan, ''An intellectual intrusion detection system model for attacks classification using RNN,'' Int. J. Pharm. Technol., vol. 8, no. 4, pp. 23157–23164, 2016.

[7]. Andrew Ng, "Support Vector Machine", Available: file:///C:/Users/my/Downloads/Documents/cs229-notes3.pdf

[8]. F. Bisio, S. Saeli, L. Pierangelo, D. Bernardi, A. Perotti, and D. Massa, "Real-time behavioral DGA detection through machine learning," in IEEE International Carnahan Conference on Security Technology (ICCST), 2017

[9]. S. Vishwakarma, V. Sharma, and A. Tiwari, ''An intrusion detection system using KNN-ACO algorithm,'' Int. J. Comput. Appl., vol. 171, no. 10, pp. 18–23, 2017

[10]. Book Series (2018), "SpringerBriefs on Cyber Security Systems and Networks", Available: https://link.springer.com/bookseries/15797

[11]. Mohammad Asif* , Pratap M. Mohite, Prof. P. D. Satya(2019)," Machine learning algorithm for Cyber Security - A Review", Available: file:///C:/Users/my/Downloads/Documents/CSEIT1951141.pdf

[12]. Brij B. Gupta, Quan Z. Sheng,"machine learning "and "deep learning |"methods for "cyber security",

[13]. Daniel S. Berman, Anna L. Buczak *, Jeffrey S. Chavis and Cherita L. Corbett(2019)," A Survey of Deep Learning Methods for Cyber Security", Available: file:///C:/Users/my/Downloads/information-10-00122.pdf

[14]. Yuancheng Li, Rong Ma and Runhai Jiao(2015),” A Hybrid Malicious Code Detection Method based on Deep Learning”, Available:
file:///D:/idm/A%20Hybrid%20Malicious%20Code%20Detection%20Method%20based%20on%20Deep%20Learning.pdf

[15]. Alexander Polyakov (2018),”Machine Learning for Cybersecurity 101” ,

[16]. Available: https://towardsdatascience.com/machine-learning-for-cybersecurity-101-7822b802790b