INTRUSION DETECTION USING DEEP LEARNING TECHNIQUE: A REVIEW

Olatunde Iyaniwura [1], Festus Ayetiran [2], Alaba T. Owoseni [1]

[1] Department of Mathematical Sciences, Kings University, Odeomu, Nigeria
[2] Department of Computer Science, Elizade University, Ilara-Mokin, Nigeria

Corresponding Author: Olatunde Iyaniwura, niwura_51@yahoo.com

ABSTRACT: This work is a survey paper that represents the review of the current research in deep learning. Developing a flexible and efficient NIDS for unforeseen and unpredictable attacks, deep learning based provides more efficient result. Shallow learners mostly depend on the features used for creating the prediction model. On the other hand, deep learners have the potential to extract better representations from the raw data to create much better models. The self-taught learning algorithms work best when they are allowed to learn rich models using large amounts of unlabeled data (millions of examples). The STL model could be implemented in an environment where data is unclean. Though, an important consideration is to recognize that the best practice for applying any model, would be to ensure that the data is clean. Deep learning give room for the elimination of manual work done by system administrator.

KEYWORDS: Intrusion Detection, shallow learning deep learning, Self-Taught Learning Encoder.

1. INTRODUCTION

The effectiveness of Intrusion Detection System [IDS] depends on its “configurability” (ability to define and add new specifications attack), robustness (fault tolerance) and the small amount of false positive and negative [FA12]. This is not so because of the IDS which often required to evaluate events on the network in real time. This requirement cannot be met because of the large volume of events on the network, high false error rate (and associate cost), difficulty in obtaining reliable training data, longevity of training data and behavioral dynamics of the system [S+17].

For the above, mostly shallow machine is being used for building the anomaly detection models. Shallow learners mostly depend on the features used for creating the prediction models. The current situation will reach a point whereby reliance on such techniques leads to inaccurate and ineffective detection.

Because of the above challenges, there is need to create an effective and widely accepted anomaly detection technique capable of overcoming limitation induced by the ongoing changes occurring in modern networks the following are some of the machine learning technique being used, Artificial Neural Network, Support Vector Machine, Decision Tree etc.

The application of these techniques has offered improvement in detection accuracy but the limitations with their usage are as follows: -

Comparatively high level of human expert interaction is required. Expert knowledge needed to process data e.g. identifying useful data and patterns. This is not only labour intensive and expensive but also error prone [N+15]. Similarly, a large quantity of training data is required for operation (with associated time overheads) which can become challenging in a heterogenous and dynamic environment.

The use of deep learning, which is a sub-set of machine learning can overcome some of the above limitations. Deep learning-based algorithms provide better result than the existing machine learning in this area. Intrusion Detection involves monitoring network traffic, detecting attempts to gain unauthorized access to a system or resources and alerting the appropriate person so that countermeasures can be taken [PB13].

Traffic monitoring is in general the responsibility of Intrusion Detection System. It captures packets at particular time frame, namely every millisecond from those captured packets, IDs extracts more detail information such as packet size, the origin of IP address, the attacked port number and also its packet type like, ICMP, TCP, UDP as well.

IDS can be classified according to its detection system, the source of data and behavior. The detection system is made up of Misuse (Knowledge/Signature) and Anomaly (Heuristic). The Misuse detection is normally used for detecting known attacks. It requires that all known threats will be defined first; and the information regarding this threat be submitted to the IDS. The misuse detection is as good as its database. It has a relatively low rate of false alarms, and it cannot detect a new attack for which a signature is not yet installed.

The Anomaly detection is based on defining the network behavior. There are two main approaches used in anomaly detection: self-learning approach or programmed anomaly detection.

In the self-learning approach, the anomaly detection system will begin to automatically monitor events such as, live network traffic on the environment it has been implemented on and attempt to build information on what is considered normal. This is online learning,

The programmed approach or offline learning, anomaly-based IDS must manually learn what is considered normal behavior by having a user or some form of function teaching the system through input of information [PB13].

Anomaly detection can also detect malformed packets. Anomaly system detects anomalous behavior. It must just the trained to recognize normal system activity. The two phases of anomaly detection system consist of the training phase and testing phase. The training phase consists of building a profile of normal behaviors and testing phase has the existing traffic being compared with the profile created in the training phase. [Van17].

The anomaly detection is computational expensive because every metrics are often maintained that need to be updated every system activity and due to insufficient data, they may be trained incorrectly to recognize an intrusive behavior as normal due to insufficient data.

2. RELATED WORK

[JR13]; due to the limitation above, SVM cannot be used for IDS domain without a variant in SVM framework to address the mentioned limitations.

For solving the features selections of SVM, [E+10] made use of Principal Component Analysis (PCA) with SVM as an approach to select optimum subset.

Intrusion detection based on SVM optimized with Swarm intelligence. [EP14], stated in their paper that SVM performance depends on selection of the appropriate parameters. The IDS model based on Information Gain for features selection combined with the SVM classifiers. The parameters for SVM are selected by Swarm optimization or artificial bee colony, using the NSL-KDD, dataset. The model achieved higher detection rate and lower false alarm rate than regular SVM.

[MW14], in a paper proposed intrusion detection system using data mining techniques. The SVM and PSO (particle swarm optimization) parameter optimization was first performed by the PSO using SVM in optimizing the value of cost and gammar parameter. This was followed by the PSO performing feature optimization as to obtain optimized feature. The SVM was finally used to optimize the parameters and features to obtain higher degree of accuracy.

In 2014, [Ift15] made use genetic algorithm to search the genetic principal components that offer a subset of features with optimal sensitivity and the highest discriminatory power. The SVM was used as the classifier. The results show that the proposed method enhanced SVM performance in intrusion detection.

[U+12] mentioned the two main challenges when generating the training data needed for IDS modeling. First, network traffic is very complex and unpredictable and model in subject to change over time since anomalies are continuously evolving.

As attack techniques and patterns change, previously gained information about how to tell them apart from normal traffic may no longer valid. To overcome this, machine learning technique was adopted to implement semi-supervised anomaly detection system where the classifier was trained with 'normal' traffic data only, so that knowledge about anomalous behaviour can be constructed and evolve in a dynamic way. Using the machine learning, Discriminative Restricted Boltzmann machine with expressive power of generative models with good classification accuracy capability to infer part of its knowledge from incomplete training data.

[WP15] observed that a single artificial neural network produces over fitting on intrusion detection system. Their work used two of ANN namely Lavenberg- Marquaralt and Quaxi-Newton to overcome the issue. Both algorithms are used to detect computer networks from attack. In addition, they use Possibilistic Fuzz C- means (PFCM) before going into the neural network ensemble Naïve Bayesian Classification method in used. The outcome shows that the neural network ensembles method produces a better average accuracy than previous researches.

[QPM14] implemented a fuzzy logic-risk analysis technique for analyzing the alarms generated. This is because of serious concern in information security false alarm which in having severe impact through the distribution of information availability

[S+11] in their work on new forms of attacks on the computer system. They discovered that these attacks gave room for high rate of false positive alarms. They tried to reduce the false alarms using alert clustering mechanism and system hibernation capabilities Alert clustering and authentication algorithms were used

Detecting new attacks in real time is a big problem. Alrawashden et. al. approach the problem by using deep learning method anomaly detection using a Restricted Boltzmann machine (RBM) and deep belief network. This approach was able to perform self-learning to extract features from unlabeled data. With this, self-learning in deep learning is essential in the design of online intrusion detection system. With this, human involvement is eliminated.

[AA18], discussed about attackers always changing their tools techniques in IDS on a challenge talk. Making use of various machine learning classifiers based on KDD intrusion dataset, several experiments were performed and tested to evaluate the efficiency and the performance of J48, Random forest, Random time, Decision table, MLP, Naïve Bayes and Bayes network. The experiment resulted in that no simple machine learning algorithm can handle efficiently all types of attacks. Furthermore, to save the availability and confidentiality of the networks resources, the true positive and average accuracy rate are not sufficient to detect the intrusion false negative and false positive rates are also needed to be taken into consideration.

Many challenges arrive while developing a flexible and effective NIDS. For unforeseen and unpredictable attacks, Ponkarthika et.al made use of deep learning-based approach to implement long short-term memory (LSTM) architecture applied to recurrent neural network (RNN) and train the IDS model using KDD cup 99 dataset. The proposed method is to detect the network behavior whether it is normal or affected based on the past observations. After the experiments by comparing it to IDS classifiers the LSTM-RNN is a better classifier.

[AJ15] brought in the use of particle swarm optimization (PSO) (another optimization approach based on the behavioral study of animals/birds) for the development of reliable and intelligent intrusion detection system.

The main advantage of PSO is that it is easy to implement and only a few input parameters are needed to be adjusted and is in non-linear optimization problem. In their research, the following factors affected the performance of IDS. First is the selection and extraction of relevant feature. If all features are evaluated, then it degrades the IDS performance Hybridization of different supervised machine better clarification.

Hybridization of PSO with Roughset, ANN, SVM were made use of for better result with this, a scalable solution for detecting network-based anomaly was obtained. Based on the behavioral study of animals/birds.

[Ift15] worked on the maximization of features selection. Optimal feature subset is worked upon as to improve the classifier performance, principal component Analysis is used to obtain the subset of the features. There is possibility to miss several important features and include irrelevant features in the subset during the process. This process selected those features which had highest eigen values (most significant) and ignored those features which had lower eigen values. A method of features selection in intrusion detection of wireless sensor network proposed based on PSO which selects optimal subset of features from the principal space or the pert space

3. INTRUSION DETECTION USING DEEP LEARNING

Machine learning is used to build anomaly detection models and there are two approaches. Shallow learning and Deep learning. Shallow learners mostly depend on the features used for creating the prediction model. On the other hand, deep learners have the potential to extract better representations from the raw data to create much better models.

3.1. DEEP LEARNING

Deep learners can learn better because they are composed of multiple hidden layers. At each layer the model can extract a better representation from the features set when composed to shallow listeners who don't have hidden layers [G+18].

A deep network can be thought of as a program in which the functions computed by the lower layered neuron can be thought of as subroutine. These routines are revised many times in the computational of the final program. Whereas, using a shallow network is similar to the writing a program without the ability of calling subroutine. Without this ability, at any place we could otherwise call the subroutines There is need to explicitly write the code for the subroutine. In terms of number lines of codes, the program for shallow network is therefore longer than a deep network. Worse, the execution time is also longer because the computation of subroutine is not properly used [Le15].

Deep learning is a complex version of machine learning with multiple levels of abstractions of data at multiple processing layers. Deep learning can learn intricate structures in the data set through back propagation and indicate how machine changes the internal parameters at each layer. Deep learning exploits many layers of non-linear information processing for supervised or unsupervised feature extracts and transformation [DY13].

Deep neural network is now an integral part of network security because of its ability to work on large volume of data generated on the network today. Deep neural network has the potential to ensure exhaustive and conclusive evaluation of the network.

Deep learning Architectures are made up of multiple layers of non-linear operation similar to neural networks with many hidden layers.

The different between multi-layer perception and Deep network is their training procedure. Deep learning is a class of machine learning techniques whose classification is conducted by training data with many layers [H+15].

The key difference between machine learning and deep learning is the change in the performance as the scale of the data increases. Deep learning algorithms requires a large amount of data to find the patterns in the network while machine learning requires the less data.

The deep learning algorithm are able to perform self-learning by extracting features from unlabeled data example of such algorithms are the SVM and Auto-Encoder algorithms. Both algorithms can extract important features from unlabeled data. The difference between Autoencoder and Multi-layer Perceptron is that they may have the same number of inputs and outputs, instead of predicting y, encoders try to reconstruct x [L+12].

The weights are initialized properly. This is a great advantage since most of the data in real life are unlabeled and are always with larger volume of features, and the data are always in large volume. Feature selection helps in the elimination of the possibility of incorrect features and noises [N+16].

3.2. SELF-TAUGHT LEARNING

SLT is a deep learning-based technique. Challenges arising while developing an effective and flexible NIDS for unknown attacks are, first, proper feature selection from the network traffic dataset for anomaly detection is difficult. As attack scenarios are continuously changing and evolving. The features selected for one class of attack may not be well for others classes of attacks.

Secondly, unavailability of labeled traffic dataset from the real networks for developing an NIDS. Great efforts are required to produce such a labeled dataset from the raw network traffic traces collected over a period or in the real-time and this serves as the reason behind the second challenge [N+15].

The STL consists of two stages for classification:

First, a good feature representation is learnt from a large collection of unlabeled data, xu termed as Unsupervised Feature Learning (UFL). Secondly, this learnt representation is applied to label data xL and used for classification task.

Sparse auto-encoder used for UFL. A sparse auto encoder is a neural network consist of an input, a hidden and out layer. The input and output layers contain N nodes and the hidden layer contain K nodes.

The target values in the output layer set to the inputs value i.e:

$$\bar{x} = x_i$$

Where, $\bar{x}$ is the target.

Deep learning method based on neural network are now very popular of recent because deep learning algorithm are able to perform self- learning by extracting features from unlabeled data. Auto-encoder is one of such algorithms. The algorithm can extract important features from unlabeled data.. Self-taught of the deep learning is an example of unsupervised learning. Human intervention is eliminated from the process. Extraction of important features are done through self-taught learning This process is good for online intrusion detection system. Semi-supervised layer of back propagation is to finetune the result obtained so as to obtain higher detection rate and the SVM as the classifier.

3.3. DEEP NETWORK

The deep network is presented in Figure 1.

3.3.1. Generative Architecture

A Generative Model is a powerful way of learning any kind of data distribution using unsupervised learning and it has achieved tremendous success in just few years. All types of generative models aim at learning the true data distribution of the training set so as to generate new data points with some variations.

Generative models are associated with supervised learning since their training does not depend on the labels of the data. For classification purposes the models go through a pre-training stage (unsupervised learning). During this process, each of the lower layers are trained separately from the other layers which allows the other layers to be trained in a greedily layer by layer from bottom to up. All other layers are trained after pre-training [H+15].

3.4. RECURRENT NEURAL NETWORK

They are networks with loops in them, allowing information to persist. It is the first algorithm that remembers its input, due to an internal memory, which makes it perfectly suited for Machine Learning problems that involve sequential data. It is one of the algorithms behind the scenes of the amazing achievements of Deep Learning in the past few years.

A multi-layer perceptron is built with an input layer, a hidden layer with certain activations and finally we receive an output.

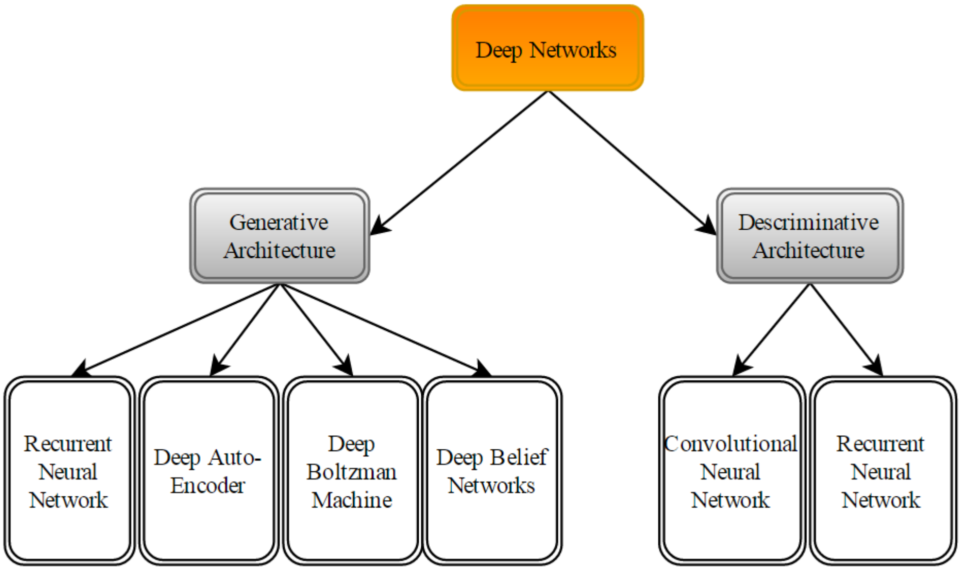

Figure 1. The deep method

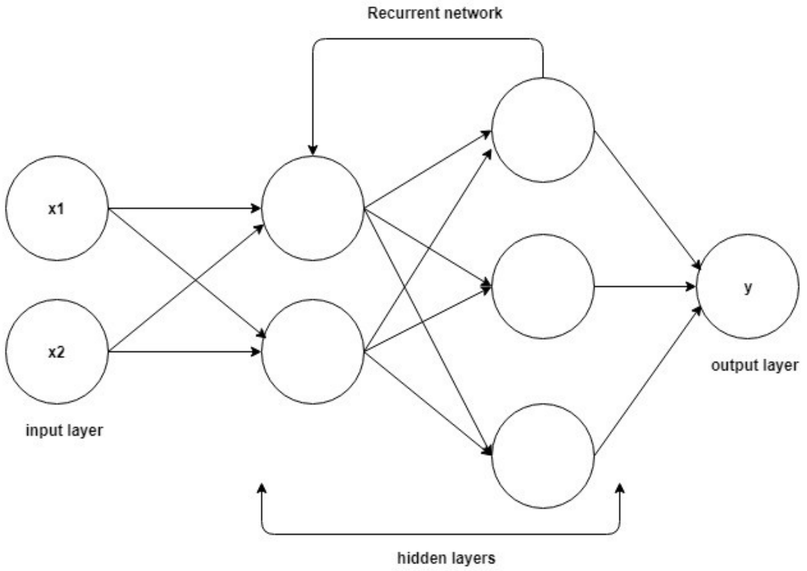

Figure 2. Recurrent Network

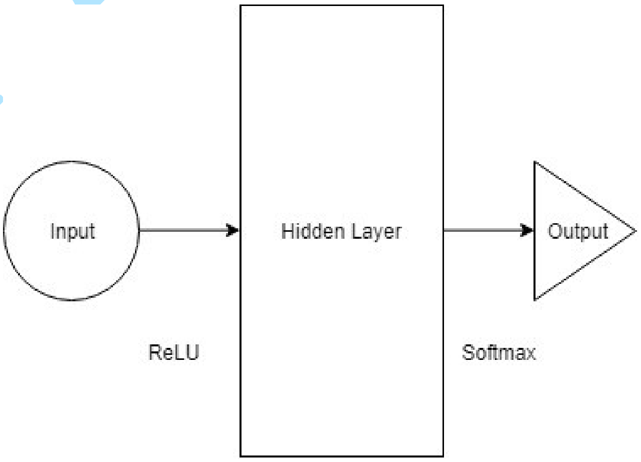

Figure 3. Recurrent Neural Network Architecture

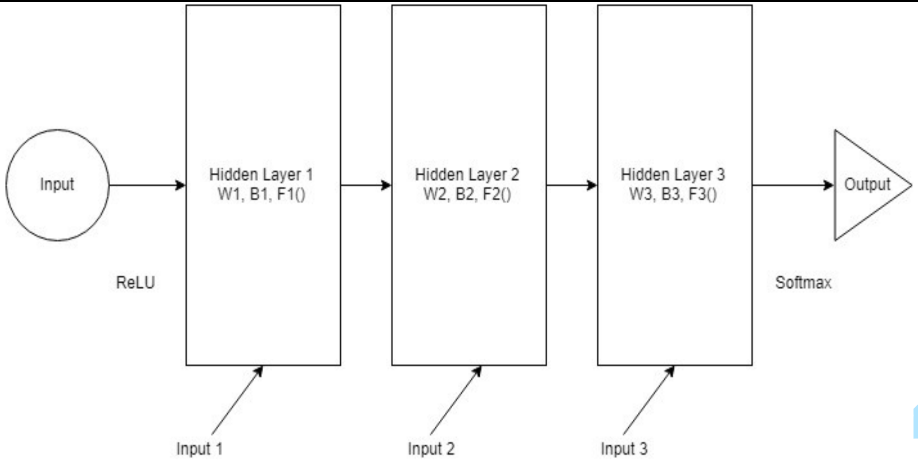

Figure 4. Increasing number of Layers

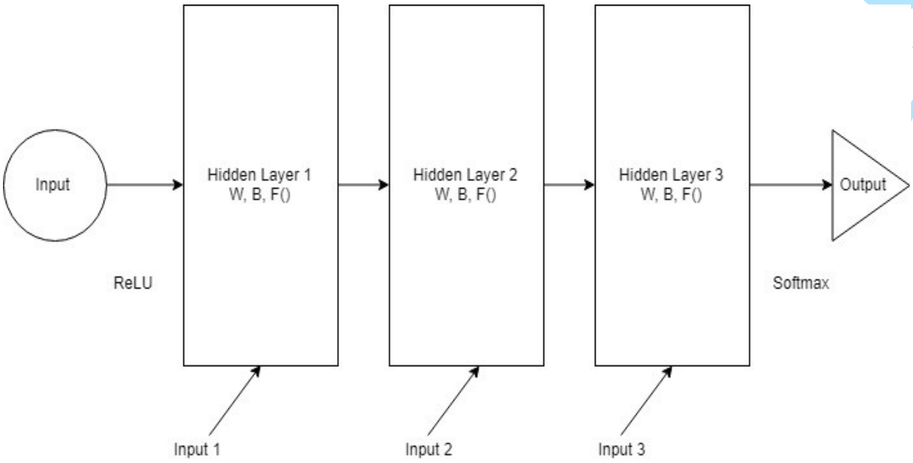

Figure 5. Attaching weights

Increasing the number of layers in the above example, input layer takes the input. Then the first hidden layer does the activation passing onto the next hidden layers and so on. Finally, it reaches the output layer which gives the output. Each hidden layer has its own weights and biases (see Figure 4).

Each layer has its own weight (W), biases (B), Activation Functions (F). These layers behave differently and technically would be challenging to merge together. To be able to merge them, lets replace all the layers with the same weights and biases. It will look something like Figure 5.

Merging all the layers together, the hidden layers can be combined into a single recurrent layer to obtain a structure as Figure 6.

Input is provided to the hidden layer at each step. A recurrent neuron now stores all the previous step input and merges that information with the current step input. Thus, it also captures some information regarding the correlation between current data step and the previous steps. The decision at a time step t-1 affects the decision taken at time t [DEB18].

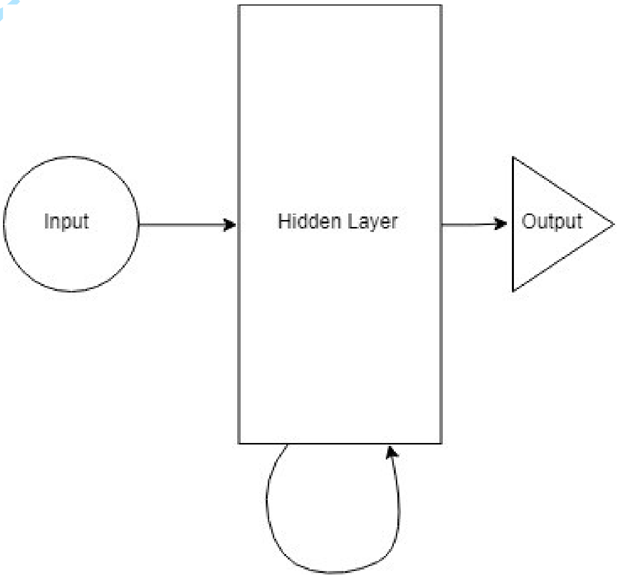

Figure 6. Merging Layers

Deep Auto Encoder

Autoencoder is an unsupervised learning neural net. Autoencoders are deep neural networks used to reproduce the input at the output layer i.e. the number of neurons in the output layer is exactly the same as the number of neurons in the input layer.

3.4.1. Auto-Encoder

This is a very popular technique used in deep learning research. It is an unsupervised neutral network-based feature extraction algorithm, which learns the best parameters required to reconstruct its output as close as to the input as possible. One of the its desirable characteristics is the capability to provide a more powerful and non-linear generalization than the Principle Component Analysis (PCA).

This is done by the use of backpropagation and setting the target value to be equal to the input, trying to learn an approximation to the identity function. An Auto-encoder typically has an input layer, output layer (the same dimension as the input layer) and a hidden layer. The hidden layer normally has a smaller dimension than that of an input (known as undercomplete or sparse auto-encoder).

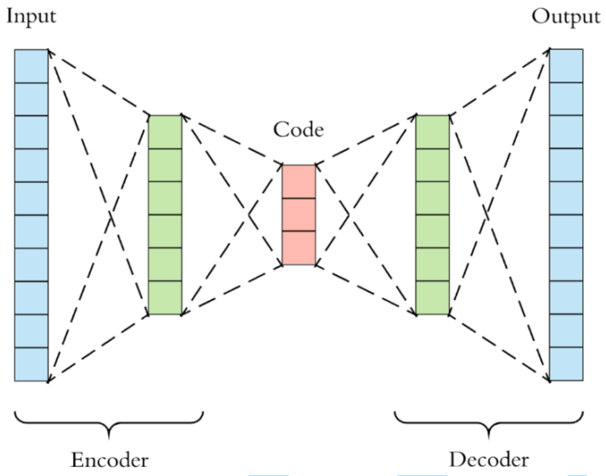

Figure 7. Auto-Encoder

To be able to discover more interesting data structure, use the autoencoder as a non-linear transformation by imposing other constrains on the network, then compare the result with those of the PCA (linear transformation). This method is based on encoder-decoder paradigm.

The input is first transformed into a typically lower dimension space (encoder) and then expanded to reproduce the initial data (decoder). Once a layer is trained, its code is fed to the next, to better model highly non-linear dependence in the input [GBV15]..

Auto-encoder are considered an unsupervised learning technique since they don't need explicit labels to train on. But to be more precise they are Self-Supervised because they generate their own labels from the training data.

In self-taught, we first trained a sparse Autoencoder (sparse: to have many of the activation exactly to zero). On unlabeled data, then, given a new example x, we used the hidden layer to extract features a.

3.4.2. Methodologies

The auto-encoder learns features from unlabeled data from the feature learnt from the main data, this trained data can now be trained in Sparse Autoencoder. This training is done with the $W^{(1)}$, $b^{(1)}$, $W^{(2)}$,$b^{(2)}$ as parameters (w=weight , b= base).

If the input is X_u (u for unlabeled), w, b, are used for train *a* to obtain the target x (a better representation of X_{II}).

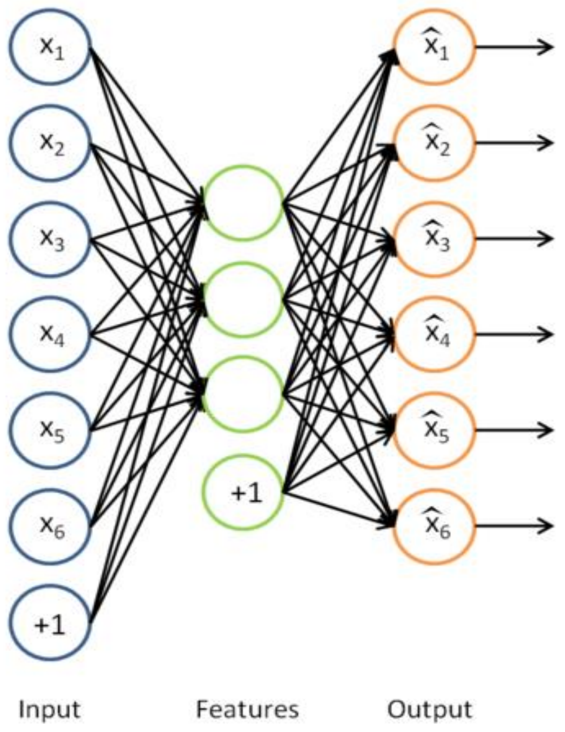

Figure 8. Encoder

After this, we have a labeled training set

$(X_L^{(I)}, y^{(I)})$, $(X_I^{(2)}, y^{(2)})$…… $(X_1^{(ml)}, y^{(ml)})$

m_l examples where l stands for labelled.

Instead of representing the first training example $x_l^{(l)}$, let the autoencoder be fed with $x_l^{(l)}$, $a_l^{(l)}$ is obtained as the corresponding vector of the activation.

Representing this example, the original feature can either be either just be replaced with the vector $a_l^{(l)}$ or alternately the two features vectors can be concatenated together, getting $(x_l^{(l)}, a_l^{(l)})$.

The training set is now for representation $(a_l^{(1)}, y^{(1)})$, $(a_l^{(2)}, y^{(2)})$……. $(a_l^{(ml)}, y^{(ml)})$ for replacement representation and use a_l for l-th training example.

For concatenation representation

$\{(x_l^{(1)}, a_l^{(1))}) y^{(1)}\}$, $\{(x_l^{(2)},a_l^{(2)}$, $y^{(2)}\}$……$\{(x_l^{(ml)},a_l^{(l)},y^{ml})\}$

Concatenation is better, but for memory or computation representation, replacement is often used.

Training and Testing the Support Vector Machine as Classifier

Support Vector Machine (SVM) is trained as the supervised learning to obtain a function that makes predictions on the y values.

Given a test example x_{test} with above examples, this is fed to auto-encoder as to get a_{test}.

Feeding either a_{test} (as in replacement) or (x_{test}, a_{test}) for the trained classifier SVM to get prediction.

NSL-KDD was proposed to overcome the limitation of KDD Cup dataset.

A sparse autoencoder is a neural network consisting of an input, a hidden, and an output layer. The input and output layers contain N nodes and the hidden layer contains K nodes. The target values in the output layer set to the input values, i.e.,

$$\hat{x}_i = x_i$$

The sparse auto-encoder network finds the optimal values for weight matrices,$W \in \mathbb{R}^{K\times N}$ and $V \in \mathbb{R}^{N\times K}$ and bias vectors, $b_1 \in \mathbb{R}^{K\times 1}$ and $b_2 \in \mathbb{R}^{KN\times 1}$ using back-propagation algorithm while trying to learn the approximation of the identity function, i.e., output $\hat{x}$ similar to x.

Sigmoid function, $g(z) = \frac{1}{1+e^{-z}}$ is used for the activation, $h_{w,b}$ of the nodes in the hidden and output layers:

$$h_{W,b}(x) = g(W_x + b) \qquad (1)$$

$$J = \frac{1}{2m}\sum_{i=1}^{m}\|x_i - \hat{x}_i\|^2 + \frac{\lambda}{2}\left(\sum_{k,n} W^2 + \sum_{n,k} V^2 + \sum_k b_1^2 + \sum_n b_2^2\right) + \beta\sum_{j=1}^{K} KL(\rho\|\hat{\rho}_j) \qquad (2)$$

The cost function to be minimized in sparse auto-encoder using back-propagation is represented by Eqn. 2.

The Support Vector Machine is finally used as to fine tune the output of the result of the regression.

3.4.3. Deep Boltzmann Machine

DBM is a unidirectional graphical model. Currently there exist no connection between units on the same layer but between the input units and the hidden units. DBM when trained with a large supply of unlabeled data and fine-tuned with labelled data acts as a good classifier [F+17].

A reduction in the number of hidden layers of a DBM to one form is a Restricted Boltzmann Machine (RBM).

In detecting new attack in real time for anomaly detection, Restricted Boltzmann Machine and Deep Belief Network (DBN), one-hidden layer of RBM was used to perform unsupervised feature reduction. Resultant weight from this RBM are passed to another RBM producing DBN. The pre-trained weights are passed into a fine-tuning layer consisting of a Logistic Regression (LR) classifier with multi-class Soft-Max.

3.4.4. Deep Belief Network

Deep Belief Network integrates unsupervised learning into its network training. Deep Belief Network can be viewed as a composition of unsupervised Restricted Boltzmann Machines, where each Restricted Boltzmann Machine's hidden layer serves as the visible layer for the next. This leads to a layer-by-layer unsupervised training procedure and supervised learning is only applied at the end to fine-tune the network parameters and convert the learned representation into probability predictions (for example, softmax function is used in the last layer to convert the outputs from hidden layers to probability predictions) [F+17]. Therefore, Deep Belief Network has less dependence on initial labels and we expect it to perform better for our problem with limited labels.

3.4.5. Convolutional Neural Network

[D+05] propose an IDS platform based on convolutional neural network (CNN) called IDS-CNN to detect DoS attack. Experimental results show that CNN based DoS detection obtains high accuracy at most 99.87%. Moreover, comparisons with other machine learning techniques including KNN, SVM, and Naïve Bayes demonstrate that the proposed method outperforms traditional ones.

4. CONCLUSION

The advent of big data, the speed at which they come, has great effect on the efficiency of making use of shallow network in the intrusion detection. The noisy data that distorts the performance of the IDS when shallow machine language is made use of is no problem for the deep network. This paper gives an over view of what is happening in the development of Intrusion Detection System with Deep Network in recent time. The use of each deep network mentioned is carefully explained and mention is made of their characteristics.

The elimination of human intervention is of vital importance. The self-taught learning in deep network encoder, using Restricted Boltzmann

Machine, Deep Belief Network and similar algorithm have made intrusion detection performance more efficient.

REFERENCES

[AA18] **Almseidin M., Alkasassbh M.** - *Machine Learning Methods for Network Intrusion Detection*, 2018.

[AJ15] **Asgharzadeh P., Jamal S. -** *A Survey on Intrusion Detection System Based Support Vector Machine Algorithm*, International Journal of Research in Computer Application and Robotics, Vol. 3, Issue 12, Pp. 42-50, www.ijrcar.com, 2015.

[Deb18] **Debarko D.** - *What is a Recurrent Neural Network or RNN, how it works*, https://hackernoon.com/rnn-or-recurent-neural-network-for-noobsa9afbb00e860, 2018.

[DY13] **Deng L., Yu D.** - *Deep Learning: Methods and Applications*. Foundations and Trends in Signal Processing, vol. 7, nos. 3–4, pp. 197–387, 2013.

[D+05] **Dongseong K., Hanam N., Syngyup O., Jongsou P.** - *Fusions of GA and SVM for Anomaly Detection in Intrusion Detection System*, Advances in Neural Networks - ISNN 2005, Second International Symposium on Neural Networks, Chongqing, China, May 30 - June 1, 2005, Proceedings, Part III, 2005.

[EP14] **Enache A., Patriciu V. V.** - *Intrusions detection based on Support Vector Machine optimized with swarm intelligence*, 9th International Symposium on Applied Computational Intelligence and Informatics, 2014.

[E+10] **Eid H. F., Darwish A., Hassanien A. E., Abraham A.** - *Principle Components Analysis and Support Vector Machinebased Intrusion Detection System*, 10th International Conference on Intelligent Systems Design and Applications, Cairo, November 29, 2010-December 1, 2010, pp. 363-367.

[FA12] **Farhaou Y., Asimi A.** - *Creating a Complete Model of an Intrusion Detection System Effective on the LAN*, International Journal of Advanced Computer Science and Application, Vol. 3, 2012.

[F+17] **Feng W., Wu S., Li X., Kunkle K.** - *A Deep Belief Network Based Machine Learning System for Risky Host Detection*, arXiv.org > cs > arXiv:1801.00025, 2017.

[GBC16] **Goodfellow Y., Bensio Y., Courville** - *Deep Learning*. MIT Press; https//www.deeplearningbooks.org, 2016.

[G+18] **Green C., Lee B., Amarex S., Wengals D.** - *Comparative Study of Deep Learning Models for Network Intruder Detection*, SMU. Data Science Review Vol.1, No.1 Art g, 2018.

[H+15] **Hodo E., Bellekens X., Hamilton A., Tachtatzis C., Atkinson R.** - *Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey*, 2015.

[Ift15] **Iftkhar A.** - *Feature Selection using Particle Swarm Optimization in Intrusion Detection*, International Journal of Distributed Sensor Networks 2015(2), 1-8, 2015.

[JR13] **Jha J., Ragha** L. - *Intrusion Detection System using Support Vector Machine*, International Conference & Workshop on Advanced Computing (ICWAC 2013), www.ijais.org, 2013.

[Le15] **Le Q. V.** - *A Tutorial on Deep Learning. Part1: Nonlinear Classifiers and The Backpropagation Algorithm*. Available on: http://ai.stanford.edu/~quocle/tutorial1.pdf, 2015.

[L+12] **Le Q. V., Ranzato M. A., Monga R., Devin M., Chen K., Corrado G. S., Dean J., Ng A. Y.** - *Building high level features using large scale unsupervised learning*, Proceedings of the 29th International Conference on Machine Learning, Edinburgh, Scotland, UK, 2012.

[MW14] **Manekar V., Waghmare K. -** *Intrusion Detection System using support vector machine (SVM) and particle Swarm Organization, PSO*, International Journal of Advanced Computer Research, Vol. 4, Number 3, Issue 16, 2014.

[N+15] **Niyaz Q., Sun W., Javaid A. Y., Alam M.** - *A deep learning approach for Network Intrusion Detection System*, BICT 2015, December 03-05, New York City, United States, 2015.

[N+16] **Nadeem M., Marshal O., Singh S., Fangand X., Yuan X. -** *Semi-supervised Deep Neural Network for Network Intrusion Detection*, KSU Conference on Cybersecurity Education, Research and Practice, 2016.

[PB13] **Patel K. K. Buddhadev B. V. -** *An Architecture of Hybrid Intrusion Detection System*. International Journal of Information & Network Security (IJINS), vol. 2, no.2, pp. 197-202, 2013.

[QPM14] **Qassim Q., Palid A., Mohozun A. -** *Strategy to reduce False alarms in Intrusion Detection and Prevention System*, International Arab Journal of Information Technology, Vol. II, Issue 5, 2014.

[S+11] **Salama M. A., Eid H. F., Ramadan R. A.., Darwish A., Hassanien A. E. -** *Deep Belief for Clustering and classification of continuous data*, International Journal of Advanced Research in Computer Science, Vol. 4, No. 6, 978-1-4244-9991-5/11 IEEE, 2011.

[S+17] **Shone N., Nguyen Ngoc T., Phai V. D., Shi Q.** - *A Deep Learning Approach to Network Intrusion Detection*, IEEE Transactions On Emerging Topics In Computational Intelligence, vol. 2, issue 1, 2017.

[U+12] **Ugo F., Palmieri F., Castiglione A., De Santi A. -** *Network anomaly detection with restricts Boltzmann Machine*, www.relsevier.com/locate/neucom, 2012.

[Van17] **Vani R.** - *Toward Efficient Intrusion Detect using deep learning technique: A Review*, International Journal of Advance Research in Computer and Communication Engineering 180. 3297, Vol.6 Issue 10, 2017.

[WJ17] **Wang L., John R.** - *Big Data Annalistic for N.I. Detection: A Survey*. International Journal of Networks and Communication, P-ISSN: 2168-4936, E-ISSN: 2168-4944 7(I): 24-31, 2017.

[WP15] **Wiharto A. A., Permana U. -** *Improvement of performance intrusion Detection System (TDS) Using Artificial Neuval Network Ensemble*, Journal of Theoretical and Applied Information Technology, Vol. 80, No. 2, 2015.