International Journal of Research in Engineering, Science and Management
Volume-3, Issue-8, August-2020
journals.resaim.com/ijresm | ISSN (Online): 2581-5792 | RESAIM Publishing

Studying the Efficiency of Deep Neural Networks as Intrusion Detection Systems

Monica R. Mundada[1], Shreyes Purushottam Bhat[2*], B. N. Varun[3], Suraj S. Jarali[4], Sudarshan[5]
[1]*Professor, Department of Computer Science and Engineering, M. S. Ramaiah Institute of Technology, Bangalore, India*
[2,3,4,5]*UG Student, Department of Computer Science and Engineering, M. S. Ramaiah Institute of Technology, Bangalore, India*
*Corresponding author: shreyespb@gmail.com

***Abstract*: Firewalls and antivirus software are insufficient protection against intruders in a network. Without a good detection system, a computer network can be accessed by an unauthorized individual and their malicious activities can go undetected. Such malicious users endanger the confidentiality of the data of other users in the system. Absence of a good detection system also increases the threat of a Denial of Service (DoS) attack, since the system is vulnerable to be targeted from the inside. There is thus a need for a powerful detection system that validates whether a user in the system is an authorized user or indeed a misfeasor. This can be achieved using machine learning and deep learning techniques, which were implemented in the project to build efficient and scalable models for intrusion detection. For comparison purposes, the training was done with several other classical machine learning algorithms along with DNNs for prediction and classification of attacks on a Network Intrusion Detection System (N-IDS). The dataset used for training and testing was the KDD-'99' dataset, and the evaluation was done against classical Machine Learning algorithms that are effective classifiers. The findings have been analyzed to understand if Deep Learning can be used to improve existing solutions being used in the industry. The compared results have shown that the data has an affinity for 3-layer neural networks for performance over all the machine learning algorithms.**

***Keywords*: Deep Neural Network, Intrusion detection system, Logistic regression, Naive-Bayes, Random Forest.**

1. Introduction

Intrusion Detection Systems (IDSs) have become a necessity in a world which looks at newer, more robust methods for Cyber-Security. There are several reasons for this, including increased complexity of the type of cyber-attacks. We resort to usage of Neural Networks as a result to combat this with an intelligence metric that can act as a deterrent and a gateway for such systems.

In this paper, we use DNNs for prediction and classification of attacks on a Network Intrusion Detection System (N-IDS). The dataset used is the KDD-'99' dataset, and the evaluation is done against classical Machine Learning Algorithms like Logistic Regression, Naive-Bayes, Decision Tree and Random Forest that are effective classifiers. The DNNs used are with one, two, three, four and five hidden layers respectively and we are comparing the accuracy among them to detect intrusions effectively.

2. Literature Survey

[1] Othman et al. in 2018 stated that high rate of generation and high variety of data in the network have made the process of detecting intrusions difficult. They introduced Big Data analytics concepts such as Spark-Chi-SVM model as their core IDS algorithm, used ChiSqSelector for feature selection, and built an intrusion detection model by using support vector machine (SVM) classifier on Apache Spark Big Data platform operating on the KDD-'99' dataset. The results showed that Spark-Chi-SVM model has competent performance, tolerable training time and is efficient for Big Data.

[2] Akbar, Rao and Hussain proposed a hybrid scheme based on Big Data Analytics tools to create a real time Intrusion Prevention System (IPS). Heterogeneous data was picked from the KDD Cup dataset and divided into two phases, learning phase and detection phase. In the learning phase, known attacks were identified and similarly, the detection phase also considered the same for testing the performance of the system. The proposed system could generate a set of rules and depicted high detection rates of DoS, R2L, U2R, and Probe based intrusions.

[3] P. Kushwaha, H. Buckchash and B. Raman in 2017 proposed that, the many problems encountered by the Network Users are DoS, testing, phishing, website and defaults. It caused the attackers to expose the network resources. They proposed an algorithm which distinguishes the anomalous from the normal link. The solution of a DoS problem allows the user to access network services more rapidly. This reinforces the current IDS systems by quickly classifying the attacks. The key contribution of the analysis is to define the most suitable algorithm for selecting the correct characteristics from 41 relation vector attributes. In classification of instances of KDD-CUP 99, this paper used statistical techniques. The findings are evaluated by comparing its exactness, detection distance, FAR etc. for different models. The analytical examination validates

the algorithm's superiority over other state-of-the-art approaches.

[4] Rathore et. al. suggested a high-speed intrusion detection system using the C4.5 model based on the decision tree, with a lesser flow rate. Of the 41-intrusion data set KDD99 using FSR and BER technology, the nine best characteristics were selected. The accuracy of the IDS being proposed is assessed as positive, false positive and time efficient. The greater accuracy and reliability help the machine to operate in a high-speed setting in real time.

[5] B. Subba, S. Biswas and S. Karmakar proposed two different statistical methods and developed anomaly-based intrusion detection models, employing the Linear Discriminant Analysis (LDA) and logistical regression (LR) algorithms. The results of these NSL-KDD data sets are then evaluated and analyzed against other Naive Bayes, C4.5 and Support Vectors (SVM) based models. They are based on the benchmarks NSL-KDD. Experimental tests demonstrate that both

the LDA and the LR dependent models work at an acceptable standard and are, in some cases, much higher than other IDS models, with precision and detection speeds. The proposed models were found to be slightly more efficient, increasing their suitability for real world deployment.

[6] Nahla Ben Amor, Salem Benferhat and Zied Elouedi proposed the idea of using Naive Bayes classifier in the intrusion detection. This paper shows that Naive Bayes provides results with a competitive accuracy, despite having a very simple structure. The test is done on the KDD'99 dataset. By considering three levels of granularity depending on whether whole attacks are addressed or grouped into four principal categories, or whether normal and abnormal behaviors. in all experiments it compares the performance of Naive Bayes with a decision-making tree. In addition, compare the Bayes networks' strong performance with established best results on KDD'99.

3. Implementation

A. *Architecture Design*

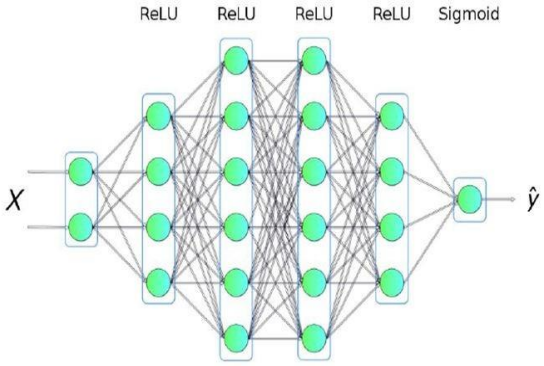

Fig. 1. Architecture of DNN implemented

where:

- X refers to the features of the dataset. Features are normalized.
- All hidden layers use ReLU as activations (non-linear activation)
- Final layer, the output layer, has a Sigmoid activation
- Y refers to the classification from the network. The answers are either 0 or 1, attributed to no intrusion or otherwise, respectively.

B. *Pseudocode*

```
for i=1 to n do
    Compute the output oi  =  ai.
end for
for i with i = n+1 to MNN do
    Get connections by using individual x1,i.
    Get connections vector z for neuron i from Wi.
    Get synaptic weights s for neuron i from Wi.
    Get the bias b for neuron i from Wi.
    Get the transfer function index t for neuron i from Wi.
    Compute the output of neuron i as oi =  ft(∑j=1 sj*zj*oj + b).
end for
for i = MNN-m to MNN do
    Compute the ANN output with yi-(MNN-m+1)+1 = oi
end for
```

C. *Implementation*

The system comprised implementations of four machine learning models, namely Logistic Regression, Gaussian Naive Bayes, Decision tree learning and Ensemble Random Forest learning. The dataset was split into a training set of 494021 tuples and a testing set of 311029 tuples. Prior to this, Exploratory Data Analysis on the data helped understand and isolate the features that contributed most towards the end classification. We theorized from analyzing the correlation matrices that 23 out of 42 attributes were contributing most towards the classification, and that the data showed good linear separability. This was confirmed by the relatively high accuracy that was obtained from the logistical classification, which would otherwise perform poorly for data that isn't linearly separable.

This system also implemented 5 DNNs with increasing depth, first having single hidden layer with [1024] nodes, second with two hidden layers having [1024, 768] nodes, third with three hidden layers having [1024, 768, 512] nodes, fourth having four hidden layers with [1024, 768, 512, 256] nodes and fifth with five hidden layers having [1024, 768, 512, 256, 128] nodes respectively with only 10 epochs sufficing to reach the degree of accuracy we desired.

All 5 neural networks built were tested against the testing set of 311029 tuples at the end of training in addition to being trained at each step using a validation step to improve accuracy.

Since KDD-99 data is very linear, we theorized that 2 or 3 layers of a network would suffice to find the deciding factors. Anything greater, variance in the model would increase (which

causes the model to lose performance since the network becomes more selective in feature weights). We were able to maintain a competent accuracy and reduced training time considerably by achieving this accuracy within just 10 epochs, due to using Model Checkpoints and feature extraction. The project was able to achieve this while maintaining high precision of over 99%, which signifies almost negligible number of false positives.

4. Results

Table 1
Model accuracies

Model	Accuracy
Logistic Regression	0.8480
Naive Bias	0.9100
Decision Tree	0.9200
Random Forest	0.9290
DNN (1 hidden layer)	0.9291
DNN (2 hidden layers)	0.9295
DNN (3 hidden layers)	0.9301
DNN (4 hidden layers)	0.9287
DNN (5 hidden layers)	0.9283

5. Data Flow Implementation

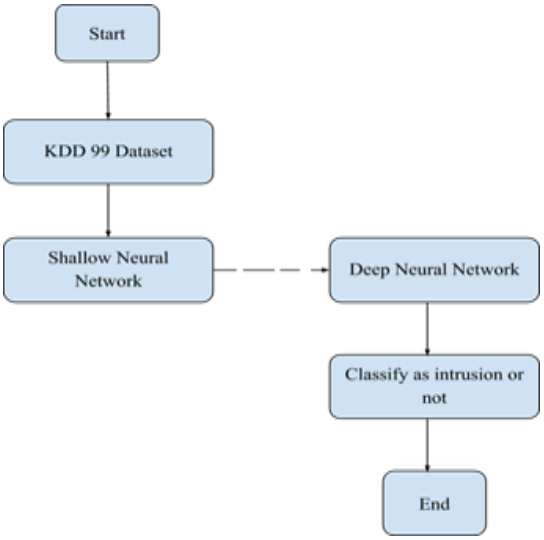

Fig. 2. Data Flow diagram for IDS

6. Conclusion

The designed DNN was able to reduce the model size (compared to classical ML models) and training resources considerably while maintaining a competent performance metric. Also, with the usage of Keras Model Checkpoint, there was considerable reduction in the overall number of weights stored per epoch for a DNN model and this allows us to deploy the IDS on most (if not all) mobile and routing devices, which was not previously possible due to the bulky nature of the models. Model size was reduced and optimized by running the same training procedure with networks of varying depths (1-5 hidden layers) in combination with model check pointing to make sure each iteration only gives out the best fit model, no matter the number of epochs. Since real-time IDS are deployed on networks that process massive number of requests, IDS has to be small, efficient and scalable, and DNNs provide the kind of scalability that is essential for these systems to work efficiently.

References

[1] Othman, S.M., Ba-Alwi, F.M., Alsohybe, N.T. et al. Intrusion detection model using machine learning algorithm on Big Data environment. J Big Data 5, 34, 2018.
[2] Akbar S, Rao TS, Hussain MA. A hybrid scheme based on Big Data analytics using intrusion detection system. Indian J Sci Technol. 2016.
[3] P. Kushwaha, H. Buckchash and B. Raman, "Anomaly based intrusion detection using filter based feature selection on KDD-CUP 99," TENCON 2017, 2017 IEEE Region 10 Conference, Penang, 2017, pp. 839-844.
[4] Rathore, Muhammad Mazhar & Saeed, Faisal & Rehman, Abdul & Paul, Anand & Daniel, Alfred. (2018). Intrusion Detection using Decision Tree Model in High-Speed Environment. 1-4.
[5] B. Subba, S. Biswas and S. Karmakar, "Intrusion Detection Systems using Linear Discriminant Analysis and Logistic Regression," 2015 Annual IEEE India Conference (INDICON), New Delhi, 2015, pp. 1-6.
[6] Nahla Ben Amor, Salem Benferhat and Zied Elouedi, “Naive Bayes vs decision trees in intrusion detection systems” Conference: Proceedings of the ACM Symposium on Applied Computing (SAC), Nicosia, Cyprus.