International Journal of Advanced Science and Technology
Vol. 29, No. 9s, (2020), pp. 2324-2331

Identification of Malicious Activity for Network Packet using Deep Learning

Monika D.Rokade[1] , Dr. Yogesh Kumar Sharma [2]
Research Scholar, Computer Department Shri J.J.T. University, Rajasthan
Associate Professor (HOD/Research-Coordinator), Department of Computer Science and Engineering,
Shri J.J.T. University, Rajasthan

Abstract

Data and application security is most essential in today environment due to the advancement as well as exchange of information and communication techniques that generating new value added services by different network threats. As a result, they developed diverse online services. However, cyber security threats are also growing as the contact points to the Internet are increasing. A significant security issue today is the intrusion detection system (IDS). A Network Intrusion Detection System (NIDS) helps system administrators detect violations of network security within their operations. However, many problems arise when a robust and efficient NIDS is developed for unexpected and unforeseeable attacks. In this work, a deep learning based approach is to implement such an effective and flexible NIDS. Through the performance test, it is confirmed that the deep neural network is effective for NIDS. In this work, A deep learning based approach to implement such an effective and flexible Intrusion Detection System on cloud environment. System uses Recurrent Neural Network (RNN) which is supervised learning algorithm to detect known and unknown attacks respectively. Initially, the The data is pre-processed using Data Balancing and standardization for input to the RNN model. The RNN algorithm was applied to the refined data to create a learning model by preprocessing, and the whole KDD Cup 99 was used to check that. When everything's said and done, the false alarm rate, accuracy and detection rate were calculated to ascertain the detection efficiency of the RNN model. Additionally, We are evaluating and comparing different deep learning algorithms, namely. RNN, CNN, DNN and PNN algorithm on cloud environment to detect intrusion in the network.
Keyword: *Recurrent neural network, KDD, WSN Trace dataset, Deep learning, Intrusion detection system, Long short term memory,*

1 INTRODUCTION

The wide augment utilization of computer systems in today's general public, especially the sudden surge in hugeness of e-business to the world riches, has made computer system asylum a global priority. As it is not in fact practicable to fabricate a plan without any vulnerabilities, interruption recognition has occur for an essential range of analyze. For the most part a gatecrasher is characterized as a framework, project or person who tries to and may get to be unbeaten to split into a framework of data or execute an activity not formally permitted. We imply interruption as any arrangement of procedures that attempt to trade off the honesty, privacy, or availability of a network asset. The demonstration of identifying procedures that attempt to trade off the integrity, attentiveness, or convenience of a network asset can be implied as interruption discovery.
A disruption location framework is a device or programming application that screens system and/or framework exercises for resentful exercises or approach infringement and produces data to an administration position. Interruption identification is the procedure of observing the activities happening in a network framework or organizes and breaking down them for indications of likely occurrence, which are infringement or hones of normal security or dangers of infringement of arrangements of network security, adequate use strategies, or normal security hones

2 LITARATURE SURVEY

2.1 RELATED WORD

IDS once we have firewall because the networks having firewall were not designed to detect attack at network layer and application layer such as worms, viruses, Denial of services (DoS), distributed denial of services (DDoS) and Trojans.

1) Data Preprocessing and Normalization:

Information preprocessing by Weka system.

This is offline system. Preprocessing data includes three main tasks: a) transforming non-numeric NSL-KDD data set functions into numeric values.

B) Transferring attack types to the integer values at the end.

C) In the end, the right dataset is prepared.

2) Feature Selection:

Normalization done in this phase. Min-Max normalization used to normalizing the features. Information Gain (IG) is use to reduce the features. Information Gain(IG) is apply on Feature selection phase. Information Gain is nothing but attribute selection mechanism in both training and testing dataset.

3) Deep Learning Model:

Building the Deep Learning model consisting number of classifiers. Mapping Test_DB with Rule set and apply on deep learning method. Developing a model that exhibits the best performance and accuracy. Compare the accuracy of each classifier and select best model.

4) Result Generation or analysis:

Finally It produces results whether the obtained packet is normal or anomaly. If it is anomaly so subclasses of that anomaly are also identified.

2.3 EXISTING METHODOLOGY

Salo et al.(2018), According to [1] a research gap in Establishing the utility of classifiers to identify existing network traffic intrusions when they are equipped with outdated databases. Our analysis highlights the need for more analytical research to tackle Big Data approaches in real time against contemporary attacks. An SLR to check IDS DM Techniques. Our emphasis was on the related empirical studies that had been published between the target time in the journals and conferences. We manually searched for some 873 separate documents, which were returned via initial search. In all, after applying our selection criteria, 95 related papers were picked. Establishment of malicious networks Intrusions have been subject to inquiry for decades. However, as data scientists can understand, when the size of a problem increases by an order of magnitude, current solutions are often no longer effective; the problem is sufficiently different to the one it requires a new solution.

Vinayak Kumar et. al. (2019), According to [2] Deep nerve Network (DNN), A kind of deep learning model that builds versatile and efficient IDs and categorizes unexpected and unpredictable cyber attacks. Network behavior and continuous changes Due to the rapid evolution of the attacks it was important to examine the numerous databases that were built on it through static and dynamic approaches. This kind of research promotes selection of the best algorithms that can operate to detect possible cyber-attacks effectively. A thorough evaluation of the DNN experiments and other classical machine-learning classificatory is shown on various publicly available Malware dataset samples. Optimal Network conditions and The following hyper parameter selection methods are chosen for the DNN network topologies with the KDDUP99 dataset.

Weiwei Chen et Al.(2017), According to [3] It said clustering and KDD would excel in explaining a new phenomenon named NEC. An unsupervised anomaly is used to produce high detection rate and less falsified passive results. It's an easy way to solve the problem and locate the phenomenon that doesn't include a list of information numbered. For 2009, the program should be tested via the NSL-KDD dataset.

ISSN: 2005-4238 IJAST

The preprocessing model converts both roles into the actual number and the standardized measures of data collection at the end of the test section into a true predicate outcome result.

Zhang et al. (2018), According to [4] a network intrusion Detection architecture based on a distributed random forest which could handle high-speed traffic data. This framework is composed of three parts: a part of NetFlow-based data capture, a part of preprocessing data, and a part of classification-based intrusion detection. In this paper we apply the random forest classification algorithm to the distributed processing system Apache Spark and adapt it for real-time detection. Verifying the success of the System, we Implement the software and conduct many empirical studies. The results show that the device has a sufficient performance and accuracy compared with existing systems and is therefore very good for detecting infiltration of the network in real time, with a high capacity and speed.

Zaman, Marzia, and Chung-Horng Lung (2018), According to [5] In this field, early work and commercially available intrusion detection systems (IDS) are basically based on signatures. The drawback with the signature-based approach is that when new attack signatures are usable the database signature needs to be updated and is therefore not ideal for detecting anomaly in the network in real time. The recent trend in detecting anomalies is based on the machine learning classification techniques. Use of the network changes at a very rapid rate. Network traffic volumes are also steadily on the rise. Network traffic management and intrusion protection is not a new concept, because there are also other types of attacks the virus and malware.

Zhou et. al. (2018), According to [6] Called a DFEL window to monitor IoT environment infringement over Internet. The authors note that DFEL not only allows the measurement of classifier accuracy through experimental results by cyber-attack, but also significantly reduces search time. What's more, the DFEL scales in search efficiency and pace. Reinforcing the identification of cyber threats by implementing prompt countermeasures to block potential risks is critical to countering cyber attacks in the modern IoT environment.

Nathan Shone et. Al. [7], proposed an new deep intrusion innovation approaches to address these issues Paper has proposed no symmetric deep auto encoder (NDAE) for the production of unsupervised apps. Also, suggest a novel model of deep learning classification focused on stacked NDAEs. The suggested classifier was introduced and evaluated using Tensor Flow based datasets of the KDD Cup's 99 and NSL-KDD in the Graphics Processing Unit (GPU). Promising results were obtained from model to point, demonstrating changes in existing methodologies and today's broad potential for use in the NIDSs. RF is fundamentally a learning strategy for a community which has the ability to label poor learners' ' into a ' strong learner.

In this paper Guangzhen Zhao et. Al. [8] Suggested intrusion detection strategy with DBN and PNN in mind. This strategy utilizes DBN to abbreviate the training and testing period for the PNN network by converting the raw data into low-dimensional details. While, the PSO algorithm is used to maximize the number of hidden-layer DBN nodes in order to improve the DBN network function speech performance. Exploratory results show that the combination of deep learning and PSO algorithm and PNN is efficient and offers some guidance to solve the problems with identification of intrusion described above. The application is a shared dataset and even the network context is more true and dynamic than the dataset itself. The next move would be to adapt the methodology to the actual network in order to improve the method through the feedback in the network.

Authors Baoan et. Al. [9] Proposes a Xgboost based on an inadequate stacked auto encoder network (SSAE-XGB) technique for knowing latent representation of the original information. Since the assignment distribution of the curriculum and the compilation of test data were convicted, the author makes use of the sparsely restricted to boost the model's generalization ability. Stacked, sparse auto encoder network is used to reduce the size of initial high-dimensional and unlabeled data for depth representation of the original data. This paper suggests a novel hybrid classifier, or in other terms the use of binary tree and ensemble process, because of the class disparity of intrusion data. Research with all NSL-KDD datasets shows that the proposed SSAE-XGB binary tree and ensemble method can achieve incredibly high F1 performance and outperform the previous work.

ISSN: 2005-4238 IJAST

Daniel E. Kim, Mikhail Gofman [10] Assert that previous studies showed that shallow neural networks are stronger for network intrusion detection than deep neural networks. Shallow networks can identify network data more precisely, and produce lower error rates than large networks.

3 PROPOSED SYSTEM DETAILS

3.1 PROBLEM STATEMENT

The proposed system works with deep learning approach. Program first collects examination data from various online and offline outlets. Once the data is collected through the program it applies pre-process and feature extraction. After this rules are created and stored into local database directory called as Background rules (BK Rules). Background rules are given as an input to the deep learning approach for the classification of sub attack. In this work, The RNN The algorithm was applied to pre-processing distilled data to create a learning model, and the entire KDD Cup 99 dataset was used for testing. In the end, the accuracy, detection rate, and false alarm rate were determined to assess the detection efficiency of the RNN model.

3.2 OBJECTIVE

The main objectives of this project are itemize as follows:

- The classification of attacks based on their characteristics is presented. Different components that make the detection of low-frequency attacks (like U2R and R2L, Worms, Shell Code etc.) hard to accomplish by machine learning strategies are examined and techniques are proposed for enhancing their detection rate.
- The discourse of different existing literature for intrusion detection is provided, featuring the key characteristics, the detection mechanism, feature selection employed, attacks detection capability.
- The critical performance analysis of different intrusion detection techniques is provided with respect to their attack detection ability. The limitations and comparison with different methodologies are additionally talked about. Various suggestions are provided for enhancement in each category of techniques.
- Future headings of Deep learning are provided for intrusion detection applications.
- To generate strong and dynamic rules depending upon the real time behavior of the packet in training phase.

3.3 SYSTEM ARCHITURE

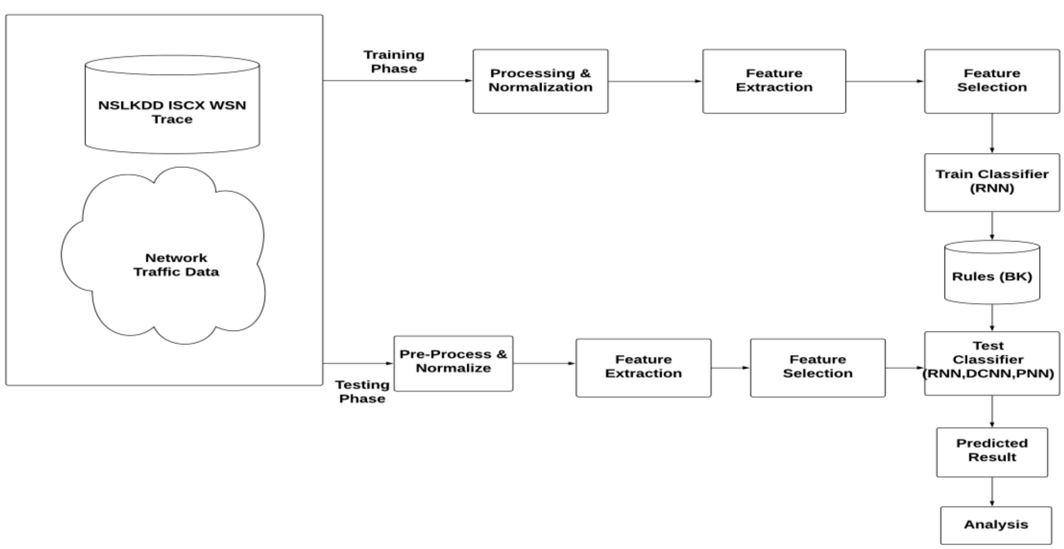

Figure 1: Proposed System architecture

Training Phase:
Step 1: To generate the rules based on supervised learning algorithm we used synthetic dataset like KDDCup99, NSLKDD, ISCX and WSN Trace etc.
Step 2: Select features for each selected instances and execute the train classifier to generate the training rules.
Step 3: The result of training modules called as training rules or policies which has stored in repository those defined as Background Knowledge (BK).

System Testing Phase:
Step 1: System accumulate the network traffic data from network audit log data or NSLKDD
Step 2: Read each input packets from network environment and apply various machine learning as well as deep learning algorithm (RNN).
Step 3: RNN has apply to generate the runtime weight for each input packet and validate with the quality threshold.
Step 4: Classify the detected packet as master attack like DoS, PROBE, U2R, R2L, Network attacks etc), and finally also shows the subtype of attack for respective class.

3.4 ALGORITHM DESIGN

Weight calculation using deep learnig Algorithm (RNN)

Input: Train dataset which already store background knowledge by train classifier TD[], test dataset includes multiple pdf's TestDb[], and desired threshold for validate the current weight.

Output: Hash_Map <class_label, sim_weight> all objects which having similarity weight larger than desired threshold.

Step 1: Read each test object using below function

$$testFeature(m) = \sum_{m=1}^{n} (.\ featureSet[A[i] \ldots \ldots .. A[n] \leftarrow \text{TestDBLits})$$

Step 2: Extract each feature as a hot vector or input neuron from $testFeature(m)$ using below equation.

$$\text{Extracted\_FeatureSetx}[\text{t} \ldots \ldots \text{n}] = \sum_{x=1}^{n} (\text{t}) \leftarrow testFeature\ (\text{m})$$

Extracted_FeatureSetx[t] contains the feature vector of respective domain

Step 3: extract each train objects using below function

$$trainFeature(m) = \sum_{m=1}^{n} (.\ featureSet[A[i] \ldots \ldots .. A[n] \leftarrow \text{TrainDBList})$$

Step 4: extract features from each test set as best features for specific document object $testFeature(m)$ using below function.

$$\text{Extracted\_FeatureSetx}[\text{t} \ldots \ldots \text{n}] = \sum_{x=1}^{n} (\text{t}) \leftarrow testFeature\ (\text{m})$$

Extracted_FeatureSetx[t] contains the feature vector of respective domain.

Step 5: Now evaluate each test vector with entire train features and generate weight for respective instance

$$weight = calcSim\ (\text{FeatureSetx}\ ||\ \sum_{i=1}^{n} \text{FeatureSety[y]})$$

Step 6: Return object [label] [weight]

3.5 MATHEMATICAL MODEL

1) Let S be the system: Such that,
S= {Sys1,Sys2, Sys3, Sys4}
S1= Data preprocessing
S2= Feature Selection and Normalization
S3= Deep Learning Model
S4= Analysis

2) Let S1 be a data preprocessing phase:
S1= {TrainDB}

$$\text{MI(x: c)} = \sum_{k=0}^{n} (k = 0) P(X = x, C = c). \log(P(X = x, C = c))/(P(X = x)P(C = c)))$$

Where,
MI= preprocess Information
C= Class which can either be normal or anomaly
X= set of x vectors

3) Let S2 be a feature selection and normalization phase: S2= F1,F2, F3,..,Fn
F= All features in TrainDB
Policy for attribute selection:
Info = {protocol; service; duration; flag; srcbyte;dstbyte}
Where,
Info= Information feature selection

4) Let S3 be the deep learning model:
S3= {Test-Db, Packet(i), class}
Class= normal, anomaly
Packet= Network traffic packets
5) Let S4 be the analysis phase:
S4={Accuracy, Detection Rate}
Find accuracy of each classifier M.
Compare accuracy of each individual classifier with D.
Where,
D= deep learning model
Select best classifier model, i.e. M=D.
System basically consists of three phases like training phase,
testing phase and analysis phase. Here is the set dependency of the entire system.
System = {Train, Test, Analysis}
Train = {preprocess, feature extract, deep learning}
Test = {Pattern Match, Th, Weight, Subclass}

class = {Input → Bk-Rules→ Weight} {Normal; Attack} {sub attacks}
Analysis ={dos, probe, U2R, R2L, Normal, unknown}

4 RESULTS AND DISCUSSION

The proposed The current machine learning algorithm and the deep learning algorithms were used in two different ways by the Project. We have also introduced computational research in base system which can recommend algorithms with KDDCUP99 data set and power-contributing architecture incorporated with deep learning algorithms with custom network audit dataset. The program measured the consistency of the description and the time complexity in the same setting. Figure 2 above demonstrates the classification performance of data collection by KDDCUP using the density-based approach of the machine learning algorithm program Figure 3 used to classify and predict the precision of the proposed system using different methods like RNN algorithm.

A. Existing System Results

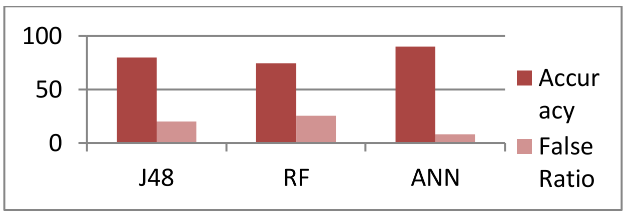

Figure 2: Detection accuracy for KDD : CUP99 dataset using machine learning

The above figure 2 Shows accuracy of kddCup 99 results classification, with five different classes. Average software output is around the algorithm for the machine learning 88.50% for all classes.

B. Proposed Result

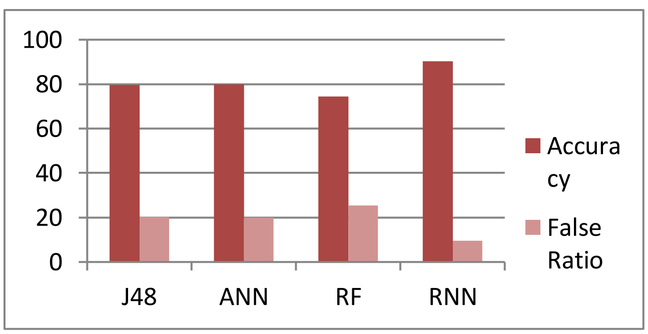

Figure 3 : Detection accuracy various network dataset using deep learning (RNN)

The above figure 3 Shows average efficiency of identification in various databases, of (n) different classes. The system's mean performance with the machine learning algorithm is around 95% for all (n) classes.

5 CONCLUSIONS

In this work, proposed a deep learning based RNN-IDS method to proposed effective IDs system. We utilized the synthetic based intrusion dataset - NSL-KDD to evaluate anomaly detection accuracy. In future, we plan to implement an IDS using deep learning technique on cloud environment. Additionally, We Evaluate and compare different deep learning technique, namely. RNN, DNN, CNN and PNN on NSL-KDD dataset to detect intrusions in the network. The system basically works like machine learning as well as reinforcement algorithm to evaluate the unknown instances during the data testing. The effective rule system provides better classification and detection accuracy for classes. Various datasets has used for experiment analysis to evaluate the algorithm performance with multiple test and conclude we get result on satisfactory level.

6 FUTURE WORKS

After upon completion of this analysis, we can conclude that it is possible to use different techniques for detection, some soft computing and some approaches to classification to detect various attacks. Some system has worked with the application of various rules to identify baseline signature anomalies. For training and testing purposes, the KDD cup data set was used. The device essentially shows the highest detection accuracy for attacks, but none of them focuses on inconsistent detection or misuse of attacks detection.

REFERENCES

[1] Salo, Fadi, et al. "Data Mining Techniques in Intrusion Detection Systems: A Systematic Literature Review." IEEE Access 6 (2018): 56046-56058.

[2] Vinayak kumar, R., et al. "Deep Learning Approach for Intelligent Intrusion Detection System." IEEE Access 7 (2019): 41525-41550.

[3] Weiwei Chen, Fangang Kong, Feng Mei, GuiginYuan, Bo Li, “a novel unsupervised Anamoly detection Approach for Intrusion Detection System”, 2017 IEEE 3rd International Conference on big data security on cloud, May 16-18,2017, Zhejiang, China.

[4] Zhang, Hao, et al. "Real-time Distributed-Random-Forest-Based Network Intrusion Detection System Using Apache Spark." 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC). IEEE, 2018.

[5] Zaman, Marzia, and Chung-Horng Lung. "Evaluation of machine learning techniques for network intrusion detection." NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium. IEEE, 2018.

[6] Zhou, Yiyun, et al. "Deep learning approach for cyberattack detection." IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2018.

[7] Nathan Shone , Tran Nguyen Ngoc, Vu Dinh Phai , and Qi Shi. Deep Learning Approach to Network Intrusion Detection”. IEEE Transactions on emerging topics in computational intelligence. VOL. 2, NO. 1, FEBRUARY 2018.

[8] Rangaswamy, K., & Rajabhushanam, C. (2019). The performance analysis of traffic control for load balancing in software-defined wireless networks. International Journal of Control and Automation, 12(6), 697-704.

[9] Guangzhen Zhao, Cuixiao Zhang* and Lijuan Zheng.” Intrusion Detection using Deep Belief Network and Probabilistic Neural Networks”. IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC).2017.

[10] Ali, K. A. (2019). State of communication security topology of networks. Test Engineering and Management, 2019, 1-9

[11] Baoan Zhang, Yanhua Yu, Jie Li.” Network Intrusion Detection Based on Stacked Sparse Autoencoder and Binary Tree Ensemble Method” IEEE International Conference on Communications Workshops.2018

[12] Daniel E. Kim,MikhailGofman.”Comparison of Shallow and Deep Neural Networks for Network Intrusion Detection.” IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC).2018.

ISSN: 2005-4238 IJAST