Review on Network Intrusion Detection using Recurrent Neural Network Algorithm

Mr. Gunjal Somnath P[1], Prof. Aher. S.M[2]

[1]*Student of Computer Engineering, VACOE' College of Engg. Ahmednagar, India*
[2]*HOD, Dept. of Computer Engineering, VACOE' College of Engg. Ahmednagar, India*

---***---

Abstract - *Internet is a widely used platform nowadays by people across the word. This has led to the advancement in science and technology. Many surveys conclude that network intrusion has registered a consistent increase and lead to personal privacy theft and has become a major platform for attack in the recent years. Network intrusion is unauthorized activity on a computer network. Hence there is a need to develop an effective intrusion detection system. In proposed system acquaint an intrusion detection system that uses improved recurrent neural network (RNN) to detect the type of intrusion. In proposed system also shows a comparison between an intrusion detection system that uses other machine learning algorithm while using smaller subset of kdd-99 dataset with thousand instances and the KDD-99 dataset.*

***Key Words*: Intrusion detection, Feature selection, linear correlation coe_cient, deep learning, RNN.**

1. INTRODUCTION

The deep integration of the Internet and society is increasing day by day, the Internet is changing the way in which people live, study and work, but the various security threats that we face are becoming more and more serious. How to identify different types of network attacks, especially unforeseen attacks, is an unavoidable key technical issue. An Intrusion Detection System (IDS), a significant research achievement in the information security field, can identify an attack, which could be an ongoing attack or an intrusion that has already occurred.

There are two types IDS: 1) Host-based IDS (HIDS) and 2) Network- based IDS (NIDS). The first one, HIDS, watches the host system operation or states and detects system events such as unauthorized installation or access. It also checks the state of ram or file system whether there is an expected data or not, but it cannot analyze behaviors related to the network. The second one, NIDS is placed on choke point of the network edge which observes a real-time network traffic and analyzes it for detecting unauthorized intrusions or the malicious attacks. The detection can be a behavior-based intrusion detection called anomaly detection or knowledge based intrusion detection called misuse detection. Behavior-based intrusion detection catches attacks by comparing an abnormal behavior to a normal behavior. Knowledge based intrusion detection detects the attacks based on the known knowledge.

Existing machine learning methodologies have been widely used in identifying various types of attacks, and a machine learning approach helps the network administrator take the preventive measures for intrusions. However, most of the traditional machine learning methodologies belong to shallow learning and often emphasize feature engineering and selection; they cannot effectively solve the massive intrusion data classification problem that arrive in the face of a real network application environment. With the dynamic growth of data sets, multiple classification tasks will lead to decreased accuracy. In addition, shallow learning is unsuited to intelligent analysis and the forecasting requirements of high-dimensional learning with huge data. In contrast, deep learners have the potential to extract better representations from the data to create much better models. As a result, intrusion detection technology has experienced fast development after falling into a relatively slow period.

Filter algorithms utilize an independent measure (i.e., information measures, distance measures, or consistency measures) as a criterion for estimating the relation of a set of features, while wrapper algorithms make use of particular learning algorithms to evaluate the value of features. In comparison with filter methods, wrapper methods are often much more computationally expensive when dealing with high-dimensional data. In this study hence, we focus on filter methods for IDS. Due to the continuous growth of data dimensionality, feature selection as a pre-processing step is becoming an important part in building intrusion detection systems [3].

2. RELATED WORK

Due to the increasing the importance of cyber security, researches about Intrusion Detection System (IDS) have been actively studying. Nathan Shone [1] proposed a network intrusion detection system (NIDS) using non-symmetric deep autoencoder (NDAE) for unsupervised feature learning. He constructed the proposed model using stacked NDAEs. The model is a combination of deep and shallow learning, capable of correctly analysing a broad-range of network traffic. He combined the power of stacking proposed Non- symmetric Deep Auto-Encoder (NDAE) (deep learning) and the accuracy and speed of Random Forest (shallow learning). The classifier implemented in graphics processing unit enabled Tensor Flow and evaluated using the benchmark KDD Cup '99 and NSL-KDD datasets.

Sung and Mukkamala [4] proposed a novel feature selection algorithm to minimize the feature space of KDD Cup 99 dataset from 41 dimensions to 6 dimensions and evaluated the 6 selected features using an intrusion detection system based on SVM. The results show that the classification accuracy increases by 1 when using the selected features.

Chebrolu et al. [5] investigated the performance in the use of a Markov blanket model and decision tree analysis for feature selection, which showed its capability of reducing the 41 to 12 features in KDD Cup '99

Chen et al. [6] proposed an IDS based on Flexible Neural Tree (FNT). The model applied a preprocessing feature selection phase to improve the detection performance. Using the KDD Cup 99, FNT model achieved 99.19 detection accuracy with 4 features.

Recently, Amiri [2] proposed a forward feature selection algorithm using the mutual information method to measure the relation among features. The maximum feature set was then used to train the LS-SVM classifier and build the IDS. They were evaluate their CSV-ISVM-based IDS on the Kyoto 2006+ [11] dataset. Experimental results showed that their IDS produced promising results in terms of false alarm rate and detection rate. The IDS was claimed to perform realtime network intrusion detection (NID). Thus, in this work, to make a fair comparison with those detection systems, we evaluate our proposed model on the aforementioned datasets.

Horng et al. [7] proposed an SVM-based IDS, which combines a hierarchical clustering and the SVM. The hierarchical clustering algorithm was used to provide the classifier with fewer and higher quality training data to reduce the average training and testing time and improve the classification performance of the classifier. Experimented on the corrected labels KDD Cup 99 dataset, which includes some new attacks, the SVM-based IDS scored an overall accuracy of 95.75 with a false positive rate of 0.7. All of the aforementioned detection techniques were evaluated on the KDD Cup 99 dataset. However, because there are some limitations in this dataset.

Some other detection methods [8], [9] evaluated using other intrusion detection datasets, such as NSL-KDD and Kyoto 2006+. A dimensionality reduction method proposed in [11] was to find the most essential features involved in building a naive Bayesian classifier for intrusion detection. Experiments were conduct on the NSL-KDD dataset produced encouraging results. Chitrakar et al. [10] there was proposed a Candidate Support Vector based Incremental SVM algorithm (CSV-ISVM in short). The algorithm was applied to network intrusion detection.

3. PROPOSED WORK

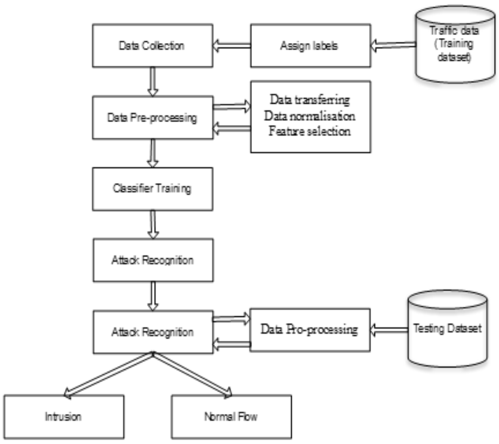
The framework of the proposed intrusion detection system is depicted in Figure. The detection framework is make of four main phases:

1. Data collection, where sequences of network packets are collected,

2. Data preprocessing, where training and test data are preprocessed and important features that can distinguish one class from the others are selected,

3. Classifier training, where the model for classification is trained using recurrent neural network.

4. Attack recognition, where the trained classifier is used to detect intrusions on the test data



3.1Data Collection

Data collection is the first and a critical step to intrusion detection. The type of data source and the location where data is collected from are two determinate factors in the design and the effectiveness of an IDS. To provide the best suited protection for the targeted host or networks, this study proposes a network-based IDS to test our proposed approaches. The proposed intrusion detection system (IDS) runs on the nearest router to the victim(s) and monitors the inbound network traffic. During the training stage, the collected data samples are categorized with respect to the transport/Internet layer protocols and are labeled against the domain knowledge. However, the data were collected in the test stage are categorized according to the protocol types only.

3.2 Data Preprocessing

The data obtained during the phase of data collection are first processed to generate the basic features such as the ones in KDD Cup 99 dataset. This phase contains three main stages shown as follows.

3.2.1 Data transferring

The trained classifier requires each record in the input data to be represented as a vector of real number. Thus, every symbolic feature in a dataset is first converted into a numerical value. For example, the KDD CUP 99 dataset contains symbolic as well as numerical features. These symbolic features include the type of protocol (i.e., TCP, UDP and ICMP), service type (e.g., HTTP, FTP, Telnet and so on) and TCP status ag (e.g., SF, REJ and so on). The method simply replaces the values of the categorical attributes with numeric values.

3.2.2 Data normalization

An essential step of data preprocessing after transferring all symbolic attributes into numerical values is normalization. Data normalization is a process of scaling the value of each attribute into a well-proportioned range, so that the bias in favor of features with greater values is eliminated from the dataset. Data used in Section 5 are standardized. Every feature from each record is normalized by the respective maximum value and falls into the same range of [0-1]. The transferring and normalization process will also be applied to test data. For KDD Cup 99 and to make a comparison with those systems that have been evaluated on different types of attacks we construct five classes. One of these classes contains pure normal records and the other four hold different types of attacks (i.e., DoS, Probe, U2R, R2L), respectively.

3.2.3 Feature selection

Even though every connection in a dataset is represented by various features, not all of these features are needed to build an intrusion detection system (IDS) Therefore, it is important to identify the most informative features of traffic data to achieve higher performance. In the previous section using Algorithm 1, for the problem of feature selection.

However, the proposed feature selection algorithms can only rank features in terms of their relevance but they cannot reveal the best number of features that are needed to train a classifier. Therefore, this study applies the same technique proposed in to determine the optimal number of required features. To do so, the technique first utilizes the proposed feature selection algorithm to rank all features based on their importance to the classification processes. Then, incrementally the technique adds features to the classifier one by one. The final decision of the optimal number of features in each method is taken once the highest classification accuracy in the training dataset is achieved. The selected features for all datasets, where each row lists the number and the indexes of the selected features with respect to the corresponding feature selection algorithm. In addition, for KDDCup 99, the proposed feature selection algorithm is applied for the aforementioned classes.

3.2.3.1 Module 1: Input Dataset

The input dataset is NSL-KDD dataset. It contains Normal, Probe, U2R, R2L and DoS attacks. Since the NSL-KDD dataset was retrieved unlabeled data, one of the _rst important step to add columns headers to it. The total 41 columns headers are added that contain information such as duration, protocol type, service, src bytes, dst bytes, ag, land, wrong fragment, etc. The classification of attacks are given as:

- Denial of Service Attacks: In a Denial of Service Attacks (DoS), the attacker tries to render a resource or system feature unusable by legitimate users by making it too busy with false requests. There are different types of Denial of Service (DOS) Attacks. Some attacks try to exploit bugs in network software and protocol stack by sending malformed packets. The remote access is sufficient to perform Denial of Service Attacks. The examples are back, ping of death, smurf, Neptune, teardrop etc.

- Probes: The probes do not cause any damage by themselves but they provide valuable which can be used later to launch an attack. The attacker tries to search for valid IP addresses, services running on each machine or for known vulnerabilities. The examples of probes and probing tools are ipsweep, mscan, nmap, saint, Satan etc.

- Remote to user: In remote to user attack, the attacker has remote access to the system but not local access. The attacker tries to exploit some vulnerability in the system to gain local access. The vulnerabilities include buffer overflows in network server software, weakly configured and misconfigured systems. The examples of remote to user attacks are dictionary attacks, guest login, ftpwrite, sshtrojan, httptunnel etc.

- User to root: In user to root, the attacker has local access to the system. The attacker tries to exploit some vulnerability in the system to gain superuser access. The common vulnerability is the buffer overflow and other vulnerabilities are bugs in management of temporary _les and race conditions. The examples are eject, loadmodule, casesen, anypw, yaga etc.

3.2.3.2 Module 2: Data Preprocessing

The data should be preprocessed to increase the efficiency of the system. Instead of giving direct input data, the raw data is preprocesses to avoid some issues i.e. detection rate ratio, false alarm, training overhead. For example, consider a one single vector from dataset, (0,tcp,ftpdata,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, 2,2,0.00, 0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00

,0.00,0.00,0.05,0.00,normal,20)At the time of preprocessing, the presence of comma ',' and other symbolic characters (tcp, ftp data and SF etc.) are removed. The last word gives the information about the class i.e. normal or anomaly.

Data normalization is a process of scaling the value of each attribute into a well-proportioned range, so that the bias in favor of features with greater values is eliminated from the dataset. To identify the most informative features of traffic data to achieve higher performance. The proposed feature selection algorithms can only rank features in terms of their relevance but they cannot reveal the best number of features that are needed to train a classifier.

3.2.3.3 Module 3: Classifier Training

Once the optimal subset of features is selected, this subset is then taken into the classifier training phase where LS-SVM is employed. Since SVMs can only handle binary classification problems and because for NSL KDD Dataset five optimal feature subsets are selected for all classes, five LS-SVM classifiers need to be employed. Each classifier distinguishes one class of records from the others. For example the classifier of Normal class distinguishes Normal data from non-Normal (All types of attacks). The DoS class distinguishes DoS traffic from non-DoS data (including Normal, Probe, R2L and U2R instances) and so on. The five LS-SVM classifiers are combined to build the intrusion detection model to distinguish all different classes.

3.2.3.4Module 4: Attack Recognition

The classifier is trained using the optimal subset of features which includes the most correlated and important features, the normal and intrusion traffics can be identified by using the saved trained classifier. The test data is then directed to the saved trained model for intrusions detection. Matching records to the normal class are considered as normal data, and the other records are reported as attacks. If the classifier model confirms that the record is abnormal, the subclass of the abnormal record (type of attacks) can be used to determine the record's type. Output as normal or anomaly (detection accuracy, false positive rate, reduce detector generation time).

4. CONCLUSION

In this paper, mentioned the problems confronted by previous intrusion detection techniques. In response to this proposed the novel approach for feature learning. After then built upon this by proposing a novel classification model constructed from recurrent neural network classification algorithm. The result shows that given approach offers high levels of accuracy, precision and recall together with reduced training time. The proposed NIDS system is improved only 5% accuracy. So, there is need to further improvement of accuracy. And also further work on real-time network traffic and to handle zero-day attacks.

ACKNOWLEDGEMENT

I would like to thank my esteemed guide Prof. N.S. Shaikh whose interest and guidance helped me to complete the work successfully. Also the valuable help of Prof S.M.Aher (HOD Comp. Dept.) who provided facilities to explore the subject with more enthusiasm.

REFERENCES

[1] Nathan Shone, Tran Nguyen Ngoc, Vu Dinh Phai, and Qi Shi, "A Deep Learning Approach to Network Intrusion Detection", vol. 2, pp. 41-50 no. 1, feb 2018

[2] F. Amiri, M. Rezaei Youse, C. Lucas, A. Shakery, N. Yazdani, Mutual information-based feature selection for intrusion detection systems, Journal of Network and Computer Applications 34 (4) (2011) 11841199.

[3] A. Abraham, R. Jain, J. Thomas, S. Y. Han, D-scids: Distributed soft computing intrusion detection system, Journal of Network and Computer Applications 30 (1) (2007) 81-98.

[4] S. Mukkamala, A. H. Sung, Signi_cant feature selection using computational intelligent technology for intrusion detection, in: Advanced Methods for Knowledge Discovery from Complex Data, Springer, 2005, pp. 285-306.

[5] S. Chebrolu, A. Abraham, J. P. Thomas, Feature deduction and ensemble design of intrusion detection systems, Computers Security 24 (4) (2005) 295-307.

[6] Y. Chen, A. Abraham, B. Yang, Feature selection and classi_cation exible neural tree Neurocomputing 70(1) (2006) 305-313.

[7] S.-J. Horng, M.-Y. Su, Y.-H. Chen, T.-W. Kao, R.-J. Chen, J.-L. Lai, C. D. Perkasa, A novel intrusion detection system based on hierarchical clustering and support vector machines, Expert systems with Applications 38(1) (2011) 306-313.

[8] G. Kim, S. Lee, S. Kim, A novel hybrid intrusion detection method integrating anomaly detection with misuse detection, Expert Systems with Applications41 (4) (2014) 1690-1700.

[9] P. Gogoi, M. H. Bhuyan, D. Bhattacharyya, J. K. Kalita, Packet and ow based network intrusion dataset, in: Contemporary Computing, Vol. 306, Springer, 2012, pp. 322-334.

[10] R. Chitrakar, C. Huang, Selection of candidate support vectors in incremental svm for network

intrusion detection, Computers Security 45 (2014) 231-241.

[11] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, K. Nakao, Statistical analysis of honeypot data and building of kyoto 2006+ dataset for nids evaluation, in: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, ACM, 2011 pp. 29-36.