Asesorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.
José María Pino Suárez 400-2 esq a Lerdo de Tejada. Toluca, Estado de México. 7223898475
RFC: ATI120618V12

Revista Dilemas Contemporáneos: Educación, Política y Valores.
http://www.dilemascontemporaneoseducacionpoliticayvalores.com/
Año: VI Número:3 Artículo no.:58 Período: 1ro de mayo al 31 de agosto del 2019.

TÍTULO: Un enfoque de análisis futurista de la red neuronal para el sistema de detección de intrusos.

AUTORES:

1. Máster. Rahim Khan.

2. Dr. Mohmmed Al-shehri.

RESUMEN: La seguridad es tema crucial para los sistemas informáticos, por lo que los Sistemas de Detección de Intrusos comenzaron a existir como el siguiente nivel de seguridad. Las técnicas de inteligencia artificial proporcionan una solución concebible a la detección de intrusos. En este documento, se propone un enfoque cualitativo de redes neuronales para detectar intrusiones basadas en análisis de características relevantes. Los resultados muestran que las redes neuronales separadas basadas en el análisis de características relevantes para cada tipo de ataques detectan ataques más rápido que el clasificador de redes neuronales únicas con el conjunto de datos que hemos usado de los datos de la Copa KDD. El enfoque de red neuronal reduce el tiempo de entrenamiento y los ataques pueden detectarse en el tiempo.

PALABRAS CLAVES: Detección de intrusión, datos de la Copa KDD, redes neuronales, análisis relevante de características.

TITLE: A Futuristic Analysis Approach of Neural Network for Intrusion Detection System.

AUTHORS:

1. Master. Rahim Khan.

2. Dr. Mohmmed Al-shehri.

ABSTRACT: Security is a crucial issue for computer systems; so, Intrusion Detection Systems began to exist as the next level of security. Artificial intelligence techniques provide a conceivable solution to intrusion detection. In this document, a qualitative approach of neural networks is proposed to detect intrusions based on analysis of relevant characteristics. The results show that separate neural networks based on the analysis of relevant characteristics for each type of attack detect attacks faster than the unique neural network classifier with the data set we have used from the KDD Cup data. The neural network approach reduces training time and attacks can be detected over time.

KEY WORDS: Intrusion Detection, KDD Cup data, Neural Networks, Feature relevant analysis.

INTRODUCTION.

Firewalls are providing first level of security to companies and organizations. Some attackers are passing through the firewalls by finding vulnerabilities in the firewall technology. Intrusion Detection Systems (IDS) are providing second level of security to the companies and organizations. At present, almost in all companies IDS has become a needful complement to the firewall technology that has been implemented by the companies connected to the Internet.

The IDS, according to the environment they are operating, can be classified into: Host-based IDS and Network-based IDS. Host-based IDS is installed on a host and the IDS process analyzes information generated by the applications and the operating system. It is usually based on revision of logs as application logs, event logs, and kernel logs. Network-based IDS analyzes information from the network traffic and can be considered as a network analyzer with an attack signature module.

DEVELOPMENT.

There are two primary types of Intrusion Detection and the categorization is done based on data analyzing method they use to detect (Khorramabad, Iran. 2015). Misuse detection model refers to the attacks that are widely known and the attack pattern can be recognized due to an attack signature. Anomaly detection model is based on recognition of the abnormal activity and this technique is often based on statistical measures, rule-based measures and threshold detection.

There are fundamental differences in efficiency of these two approaches. Since the misuse-based IDS detect attacks according to the signature, this process can be considered reliable with a very low volume of false positive. False alarm is an event that has been detected as an attack though it was a regular event. The most important drawback of the misuse-based IDS is impossibility to detect a novel attack since the appropriate signature has not been enabled.

On the other hand, we have the anomaly-based IDS that can detect a novel attack since it has been triggered because of the abnormal behavior. One drawback of the anomaly-based IDS is high volume of false alarms since the least variation of the normal behavior is considered as an attack state. However, due to the capability to detect a novel attack as well as its capability to meet the basic origin of intrusion detection problem inherited from a dynamic nature of systems and networks, it appears that the anomaly-based intrusion detection is an approach that has the future.

In this paper we have focused on anomaly-based intrusion detection. We have tested our proposed approach with the anomalous data that we prepared manually from KDD Cup data.

This paper is organized as follows. Section I introduces Intrusion Detection System and motivation for this research work. Section II covers literature survey regarding KDD Cup data set and analysis of previous approaches in the field of intrusion detection. In section III, the use of neural networks for intrusion detection is explained. In section IV, proposed approach and implementation details are explained. Section V describes about results of this research work and their discussions. In final section, conclusion and future scope of the work are discussed.

Analysis of previous methods.

A. KDD Data Set.

The KDD CUP 99 data set was built using TCP packets collected during DARPA 1998 intrusion detection evaluation program (Idris, Shanmugam, 2005). Data packets that form a complete session are gathered in a single feature vector or connection record. It contains 41 features. There are four kinds of features present in KDD data set, namely, basic, content, time-based, and host-based features. Basic or intrinsic features are common network connection features, like duration of connection, service requested, and bytes transferred between source and destination machine addresses.

Content features were collected using domain knowledge of U2R (User to Root) and Remote to Local (R2L) attacks e.g. logged in flag, number of failed logins, number of root commands, number of compromised conditions, and hot indicators.

Time-based features were collected by observing various connections within a two-second time window with respect to current connection e.g. SYN error rates, rejection rates, and number of different services requested. Host-based features include the many slow probing attacks that require several minutes to execute. For such attacks, host-based features were collected that were based on the past one hundred connections. In table I, KDD Cup data details are shown.

Table I. KDD Cup Data Details.

Dataset	DoS*	Probe	U2R	R2L	Normal
10% KDD.	391458	4107	52	1126	97277
Corrected.	229853	4166	70	16347	60593
Whole.	3883370	41102	52	1126	972780

**DoS: Denial of Service.*

In this work, we have used corrected data and focused on DoS attacks particularly Smurf and Neptune attacks.

B. Analysis of previous methods.

The researchers have implemented different methods to detect intrusions in host and network. Early systems like intrusion detection expert system (IDES) and next-generation IDES (NIDES) were built around the concept of a statistical anomaly detector (KDD CUP Data, 1999). These systems were confounded by two difficulties, first one is that systems had a fairly loose threshold for tolerance of anomalous behavior, and were designed to learn new nominal statistics as they worked.

The solution to the limitations of statistical anomaly detectors led to the second difficulty: intruders could work below the threshold of tolerance and “teach” the systems to recognize increasingly abnormal patterns as normal. In the next generations a new paradigm for intrusion detection was introduced i.e. signature recognition. The performance of these systems is limited by the signature database they work from (KDD CUP Data, 1999).

Idris and Shanmugam (Idris, Shanmugam, 2005) proposed a dynamic model Intelligent Intrusion Detection System, based on specific AI approach for intrusion detection. The techniques that are being investigated include neural networks and fuzzy logic with network profiling, these use simple data mining techniques to process the network data. The proposed system is a hybrid system that combines anomaly, misuse and host-based detection. Simple Fuzzy rules, allow constructing if-then rules that reflect common ways of describing security attacks. For host-based intrusion detection they used neural networks along with self-organizing maps. Both network traffic and system audit data are used as inputs for both.

Problem with this method is that it has to automate the process for making fuzzy rules otherwise it is time consuming process and the total method they explained was in the initial stage and test is based on offline data. Real time issues were not considered.

Susan and David (Susan, David, 2001) paper describes an experiment with an IDS composed of a hierarchy of Neural Networks (NN) that functions as a true anomaly detector. This result is achieved by monitoring selected areas of network behavior, such as protocols, that are predictable in advance. While this does not cover the entire attack space, a considerable number of attacks are carried out by violating the expectations of the protocol/operating system designer. Within this focus, the NNs are trained using data that spans the entire normal space. These detectors are able to recognize attacks that were not specifically presented during training. They showed that using small detectors in a hierarchy gives a better result than a single large detector.

In project KIT-I, it adopts remote logging server (RLS) mechanism, which was used to back up the log files to the server. Taking into account security, they make use of the function of Secure Socket Layer (SSL) of Java and certificate authority (CA) based on key management. Furthermore, neural networks are applied in their project to detect the intrusion activities.

The basic problem with this method is if the computer they were using for CA has weak link then any intruder could get the keys and could establish the SSL connections. Feature selection method for neural network module was not addressed well.

Neural networks for intrusion detection.

We are using Neural Network based approach because the first advantage in the utilization of a neural network in the detection of instances of anomaly would be the flexibility that the network would provide. A neural network would be capable of analyzing the data from the network, even if the data is incomplete or distorted. Similarly, the network would possess the ability to conduct an analysis with data in a non-linear fashion. Both of these characteristics are important in a networked environment where the information which is received is subject to the random failings of the system (Cannady, 1998).

If we compare with statistical methods, they basically dependent on statistical information of system (such as CPU usage, job execution time, system calls). One cannot have any specified method to tell these are the statistics that are to be considered. Moreover these cannot handle any noisy data. Even though advantages are present by using Neural Networks, there are also some disadvantages present. In using Neural Networks, the basic problems are weight selection algorithms and evolving network topology and training time.

In this work we are focusing on the accuracy details of this neural network approach in classification of attacks and detecting attacks based on feature relevant analysis. The development of intrusion detection systems has been hampered by the lack of a common metric to gauge the performance of current systems. Evaluations have helped to solve this problem in other developing technologies and have guided research by identifying the strengths and weaknesses of alternate approaches. The desire for an evaluation in intrusion detection led to the creation of the first DARPA-sponsored Off-line Intrusion Detection Evaluation in 1998(Cannady, 1998).

Proposed approach.

A. Feature relevant analysis.

Feature relevance analysis is performed on KDD 99 training set, which is widely used by machine learning researchers. The following features are most relevant for detecting particular type of attack:

Normal:

1. Duration: continuous.

2. dst_bytes: continuous

3. logged_in: symbolic.

4. su attempted: continuous.

5. num_root: continuous.

6. Num_file_creations: continuous.

7. Num_shells: continuous.

8. Num_access_files: continuous.

9. srv_diff_host_rate: continuous.

10. dst_host_count: continuous.

11. dst_host_srv_diff_host_rate: continuous.

Smurf.

1. protocol_type: symbolic.

2. Service: symbol.

3. src_bytes: continuous.

4. Count: continuous.

5. srv_count: continuous.

6. rerror_rate: continuous.

7. srv_rerror_rate: continuous.

8. dst_host_same_src_port_rate: continuous.

9. dst_host_rerror_rate: continuous.

10.dst_host_srv_rerror_rate: continuous.

Neptune:

1. Flag: symbolic.

2. serror_rate: continuous.

3. srv_serror_rate: continuous.

4. same_srv_rate: continuous.

5. diff_srv_rate: continuous.

6. dst_host_srv_count: continuous.

7. dst_host_same_srv_rate: continuous.

8. dst_host_diff_srv_rate: continuous.

9. dst_host_serror_rate: continuous.

10. dst_host_srv_serror_rate: continuous.

B. Implementation details.

In our work, KDD Cup data is used as input data. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment. In this dataset, each event (connection) is described with 41 features (4), 22 of these features describe the connection itself and 19 of them describe the properties of connections to the same host in last two seconds.

In the input features some are not numerical. Numerical numbers are assigned to those features (e.g. tcp=0, udp=1, icmp=2...etc) manually. This numerical representation is necessary because the feature vector fed to the input of the neural network has to be numerical. If the ranges of the features were different then that will make them incomparable. Some of the features had binary values where as some others had a continuous numerical range (such as duration of connection). As a result, the features were normalized by mapping all the different values for each feature to (0, 1) range.

In our work, we have used Multi-Layer Perceptron (MLP) network for intrusion detection with supervised learning. In supervised learning algorithms we have selected back propagation algorithm for training. Multi-Layer Perceptron network is well suited for classification of nonlinear data.

MATLAB 9.0 Neural Network Toolbox was used for the implementation of the neural networks. By using this tool one can specify like number of layers, number of neurons in each layer, activation functions of neurons in different layers, and number of training epochs. Then the training feature vectors and the corresponding desired outputs can be fed to the neural network to begin training.

Our work is divided into two parts. First part is for classification of attacks using multilayer perceptron. In this, we have used connection records belonging to Normal, Smurf and Neptune attacks. If any connection record is named as normal then it is of normal connection and does not belong to any attack category. The Smurf attack is a denial-of-service attack that uses spoofed broadcast ping messages to flood a target system. Neptune is a denial of service attack to which every TCP/IP implementation is vulnerable (to some degree). For distinguishing Neptune attack, network traffic is monitored for a number of simultaneous SYN packets destined for a particular machine. The host sending these packets is usually unreachable.

In this part, we have trained a neural network for classifying these attacks. Three sets of data are manually prepared from the KDD Cup corrected data. First set consists of training data, second set consists of validation data and third set consists of testing data which are collected manually from KDD Cup data. Validation set is used to monitor over fitting condition of the network.

In the second part we have built neural networks which belong to Normal, Smurf and Neptune attacks. These neural networks are built by using feature relevant analysis from (Mehdi, Mohammad, 2004). following features are most relevant for attack detection.

Normal: 1, 6, 12, 15, 16, 17, 18, 19, 31, 32, 37.

Smurf: 2, 3, 5, 23, 24, 27, 28, 36, 40, 41.

Neptune: 4, 25, 26, 29, 30, 33, 34, 35, 38, 39.

By using the above features, we have built each of individual neural network modules for these types of attacks. Training and test data are prepared manually from the data set and tested for each of these individual neural networks. Finally, accuracy details are calculated from the results.

Here, accuracy details refer to calculating detection rate and false alarm rate. The detection rate is equal to the number of intrusions detected divided by the total number of intrusions in a data set. False alarm rate refers to the number of normal instances detected as intrusions divided by the number of normal instances in a data set.

Results and discussions.

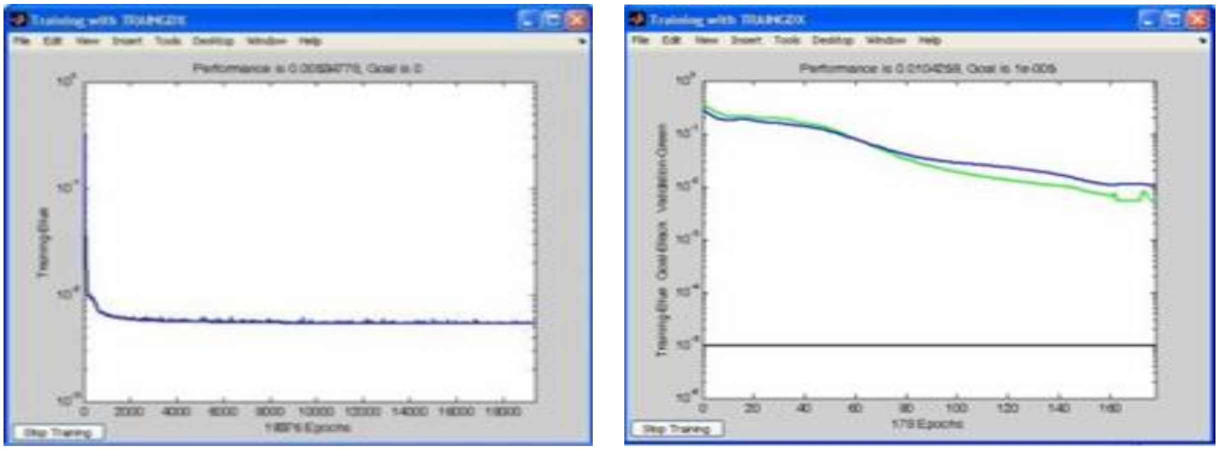
In the first part, classification module consists of one input layer, two hidden layers and one output layer. Input vector fed into the neural network consists of 41 elements feature vector. The number of nodes in the both hidden layers is 40 and output layer consists of 3 neurons. The training function used is back propagation with momentum and adaptive learning. Momentum is used for to get over small bumps in the error function. It often converges in fewer steps. Adaptive learning method assigns each weight a learning rate. That learning rate is determined by the sign of the gradient of the error function from the last iteration.

If the signs are equal, then it is more likely to be a shallow slope so the learning rate is increased. The signs are more likely to differ on a steep slope so the learning rate is decreased. This will speed up the advancement when there are gradual slopes.

Validation neural network module and Test neural network module also consist of input layer of 41 neurons, two hidden layers each of 40 neurons and output layer of 3 neurons and training function is same as classification. As we have already mentioned, data is divided into three sets i.e. training, validation and testing. Details of these are shown in the table II.

Table II. Details of Number of Connection Records Taken for Classification.

Attack type	Training Data	Validation data	Test Data	Detection accuracy (DA) rate (ontest data)	False alarm (FA) rate (ontest data).
Normal	909	201	204	91.6	8.3
Smurf	1712	240	90	96.6	3.3
Neptune	745	235	148	95.3	4.2



(a)

(b)

Fig. 1(a), 1(b). Graphs of classification module before and after applying validation method.

Fig. 1(a) shows the training graph of single classifier before applying validation method. Fig. 1(b) shows the same after applying validation method. From both, we can analyze that there is no over fitting condition.

In the second part of proposed approach, neural networks for each of the Normal, Smurf and Neptune attacks are built based on feature relevant analysis. All of these neural networks consist of one input layer, two hidden layers and one out layer. Input vectors are of size 11, 10 and 10 respectively. Each hidden layer consists of 40 nodes and output layer consists of 1 node. Training function taken here is back propagation with momentum. Here data set is divided into training and testing data. Details of these are shown in table III.

Table III. Details of Connection Records Taken for Detection based on Feature relevant analysis.

Attack type	Training	Testing	DA rate	FA rate
Normal	1010	204	83	17
Smurf	2010	90	100	0
Neptune	571	148	100	0

In table III, results are showing after rounding the values that have output layer values of more than 0.95. For normal connections, this proposed method with the test data shows detection accuracy of 83% and false alarm rate of 17%. But for Smurf and Neptune attacks, results are showing that 100% detection accuracy; however, we do not do that claim full 100% accuracy for Smurf and Neptune attacks, because the test data that we have used contains only few variant labeled connection records. This is the reason for these Smurf and Neptune attack accuracy results. But this method has shown less time compared to the classification method.

CONCLUSIONS.

A. Conclusions.

In this paper, an approach for Intrusion Detection System using neural networks based on feature relevant analysis is proposed. For each type of attack in KDD Cup data, separate neural networks are developed and accuracy results are calculated.

In this work, proposed approach is tested with by focusing on Normal, Smurf and Neptune attacks with the anomalous data to the neural network. We also developed classification module which is used to classify different attacks in KDD Cup database using Neural Networks and accuracy details are calculated.

Selecting 41 features as mentioned in KDD Cup, for training the neural network in the classification module makes the training time more for neural network learning, which is the basic drawback of any kind of Neural Network. This feature relevant based neural network approach makes the training time less and also intrusions can be detected in time which is the basic requirement of current Intrusion Detection Systems.

B. Future Scope of the work.

In our practical implementation, we have developed Neural Network classifier for distinguishing between three types of attacks. Separate Neural Network modules those we considered belong to same type of attacks. Those are Normal, Smurf and Neptune attacks from KDD Data Cup. Further classification of more attacks is required.

Another thing is the features those we used are based on from previous research those are also based on thorough observation of KDD Cup data. There is need for variant labeled data set for evaluating machine learning approaches. Our approach is based on using offline data. Further research of finding most relevant features for Intrusion Detection System in real time data is required.

C. Acknowledgment.

The authors express their sincere gratefulness for Majmmah University for his encouragement and support throughout the paper of his wish and also for facilitating technical and literature facilities, required in the development of this work

BIBLIOGRAPHIC REFERENCES.

1. Khorramabad, Iran. (2015). Implementation of Support Vector Machines and Clustering of Intrusion Detection System for Computer Networks.
2. Idris, D., Shanmugam, N.B (2005). Artificial Intelligence Techniques applied to Intrusion Detection, IEEE Indicon 2005 Conference, Chennai, India, pp. 11-13, (2005).
3. KDD CUP Data, (1999). The Fifth International Conference on Knowledge Discovery and Data mining, 1999.
4. Susan C. L., David V. H, (2001). Training a Neural-Network Based Intrusion Detector to Recognize Novel Attacks", IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans, Vol. 31, No. 4, pp. 294-299, July 2001.
5. Mehdi, M., Mohammad, Z. (2004). A Neural Network Based System for Intrusion Detection and Classification of Attacks, Proc. of 2004 IEEE International Conference on Advances in Intelligent Systems - Theory and Applications, Luxembourg-Kirchberg, Luxembourg, November 15-18, 2004.
6. Cannady, J (1998). Applying Neural Networks for Misuse Detection", Proc. 21st National Information Systems Security Conference, pp. 368-381, 1998.

DATA OF THE AUTHORS.

1. M. A. Rahim Khan. College of Computer and Information Sciences. Al- Majma'ah University. E-mail: m.khan@mu.edu.sa

2. Dr. Mohmmed Al-shehri, College of Computer and Information Sciences Majma'ah University Al- Majma'ah E-mail: ma.alshehri@mu.edu.sa

RECIBIDO: 2 de abril del 2019. **APROBADO:** 19 de abril del 2019.