Artificial Neural Network and Genetic Clustering based Robust Intrusion Detection System

Soumya Tiwari
M-Tech Research Scoholar
Department of Computer
Science & Engineering, NIIST

Umesh Lilhore
Head of Department
Department Of Computer
Science & Engineering,NIIST

Ankita Singh
Assistant Professor
Department Of Computer
Science & Engineering,NIIST

ABSTRACT
To improve network security different steps has been taken as size and importance of the network has increases day by day. In order to find intrusion in the network IDS systems were developed. In this paper main focus was done on finding the type of session i.e. normal or intrusion where if intrusion found than class of intrusion was detected. Here whole work was so designed that automatic clustering of various sessions are done by using genetic algorithm steps while clustered data is taken as the input in the neural network for training. So, the need of special identification was required in this work for session class. Error back propagation neural network was used by this work training and testing. Experiment was done on real dataset where various set of testing data was pass for comparison on different evaluation parameters.

Keywords
Anamoly, ANN, Clustering, Genetic Algorithm, Intrusion Detection.

1. INTRODUCTION
Giving network security to various web services on the web, distinctive network foundations, communications arrange numerous means that has been taken like encryption, firewall, and virtual private network and so forth organize Intrusion detection framework is a noteworthy advance among those. Intrusion detection field rises up out of most recent couple of years and built up a great deal which uses the gathered data from various sort of interruption attack, based on those distinctive business and open source programming items appear to solidify your network to enhance security of the diverse correspondence, service giving networks. As the quantity of network clients and machine are expanding step by step to give diverse sort of administrations and effortlessness for the smoothness of the world. Be that as it may, some unapproved clients or exercises from various sorts of aggressors which may inward attack or outer attack keeping in mind the end goal to hurt the running framework, which are known as programmers. The fundamental thought process of such sort of programmer and gatecrashers is to cut down cumbersome networks and web administrations. Because of increment in enthusiasm of network security of various kinds of attack, numerous scientists has included their enthusiasm for their field and wide assortment of conventions and in addition calculation has been produced by them, with a specific end goal to give secure administrations to the end clients. Among various kind of assault interruptions is a sort of assault that build up a business intrigue. Interruption discovery framework is presented for the insurance from interruption attack.

From the above exchange this work can close the primary point of the network Intrusion discovery framework is to identify all conceivable interruption which perform malevolent movement, PC assault, spread of infections, PC abuse, and so forth so a network interruption recognition framework investigations distinctive information bundles as well as screen that move over the web for such sort of vindictive action. So the smooth running of general network distinctive server needs to settle overall network which go about as network Intrusion detection framework that screen every one of the parcels developments and recognize their conduct with the noxious exercises.

2. RELATED WORK
Yogitha et. al. [1] Offered interruption discovery framework with Support Vector Machine (SVM). Affirmation is finished by coordinating explores on NSL-KDD Cup'99 data collection which is reformer type of KDD Cup'99 data index. By utilizing this NSLKDD Cup'99 data collection they have condensed wide time obligatory to shape SVM exemplary by achievement proper pre-training on data collection. In this association SVM made clustering of data. By obligation appropriate part accumulation assault location rate is opened up and false positive rate (FPT) is lessened. In this proposed work author has utilized Gaussian Circular Basis.

A.R. Jakhale, et. al [2] In this work the author portrays a anamoly discovery framework and its two stages particularly training and testing. The slipping window and bunching is accustomed to nursing the network movement by mining the repetitive examples utilizing calculations. The calculations are so genuine and utilized as a part of constant observing. The normal multi-design catching calculation has high location rate. At long last, increase the identification rate and reduced the false alert rate.
Research by Jiefei, Lobo and Russo [3] explores the event of Multi-way steered attack where an assault is divided and sent over different courses to endeavor to trick an IDS framework. This is influenced conceivable due to multi way TCP (MPTCP) which enables transmissions to course finished numerous ways between a source and target

Barolli et al [4] researches the utilization of IDS utilizing neural network for giving IDS arrangement in a Tor (The Onion Router) organize. Tests did utilized a Tor server and customer with back engendering NN to reproduce exchanges over the Tor organize while catching for examination. The framework proposed is a prepared ANN with information caught from Wireshark, at that point the server and customer information are analyzed, contrasts will recognize an interruption or misuse. The outcomes from testing were fruitful in giving viable exactness when assessed in the test condition.

Chuan Long [5] In this paper, author investigate how to display an interruption recognition framework in light of profound learning, and this work propose a profound learning approach for intrusion identification utilizing recurrent neural networks (RNN-IDS). Additionally, this work examine the execution of the model in paired classification and multiclass classification, and the quantity of neurons and distinctive learning rate impacts on the execution of the proposed display. This work contrast it and those of J48, artificial neural network, arbitrary woodland, bolster vector machine, and other machine learning strategies proposed by past analysts on the benchmark data index.

3. METHODOLOGY

Whole work is divide into different modules base on the steps of calculation from the user query to final output on the screen. In fig. it is seen that there are two different modules. first is separation of various class data into separate cluster with the help of genetic algorithm. Then in second module learning of clustered data was done where error back propogation neural network was used for this module.

3.1 Genetic Algorithm Based Clustering

As dataset consist of sessions which has normal behavior of network as well as abnormal behavior of network. Separation of session in these various categories is done by genetic algorithm. Here pre-processing is done for the sessions where unnecessary information is removed and data is arranged for training.

3.1.1 Pre-Processing

Here information like type of protocol, socket type, etc. are removed. As presence of these information increases the

confusion for the clustering process. This can be understood as let raw data session is

{0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal,20}

After applying pre-processing session will be:

{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal}

3.1.2 Generate Population

Here assume some cluster set that are the combination of different session. This is generate by the random function which select fix number of cluster for the centroid. This can be understand as let the number of centroid be Cn and number of session are N then one of the possible solution is {C1, C2,..Cn. In the similar fashion other possible solutions are prepared which can be utilized for creating initial population represent by ST matrix.

3.1.3 Fitness Function

Fitness of the solutions are calculate on the basis of feature obtained from the dataset after preprocessing. In case of feature distance between the sessions from the centroid are obtained by calculating the difference values between the sessions. Similarity is obtained by the distance function.

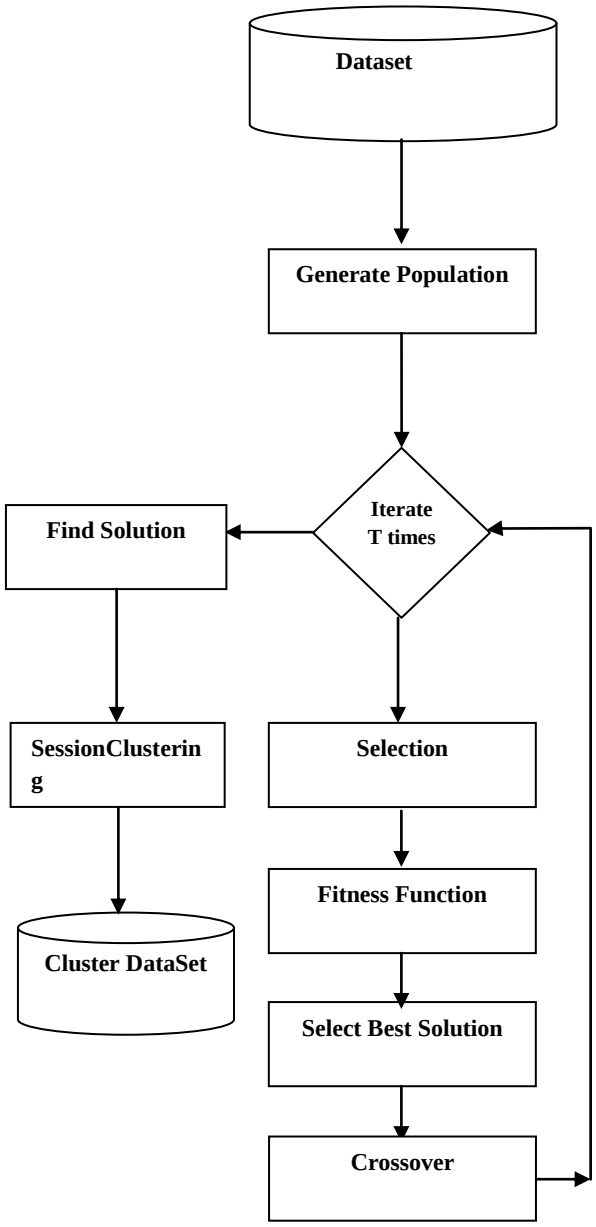

Fig 1: Proposed work first module

Eludician Distance: This can be understand as Let X be a session matrix and Y be the other session matrix. Then distance between them is calculated by:

$$D = \sqrt{sum((X - Y)^2)}$$

Here both X, Y are vectors of elements from dataset feature. Base on the minimum distance value between query and dataset session rank is assigned to the minimum distance cluster. This is considered as final rank of the work.

3.1.4 Select Best Solution

Main motive of this step is to find best solution from the generated population. Here each possible solution is evaluated for finding the distance from each centroid so that the images closer to the centroid are clustered together. Then calculate the fitness value which give overall rank of the possible solution.

3.1.5 Cross-Over

Top possible solution after sorting will act as the best for other possible solutions. Now selected solution will modify other possible solution by replacing fix number of centroid as present in best solution. By this all possible solution will learn from best solution.

This difference modifies the existing solution according to the following expression

$X_{new,i}$ = Crossover ($X_{best,i}$, $X_{,i}$)

Accept Xnew if it gives a better function value. Once this cycle is over then check for the maximum iteration for the genetic if iteration not reach to the maximum value then GOTO step of finding fitness function and rest of steps in regular fashion of genetic algorithm stop learning and the best solution from the available population is consider as the final centroid of the work. Now sessions are cluster as per centroid.

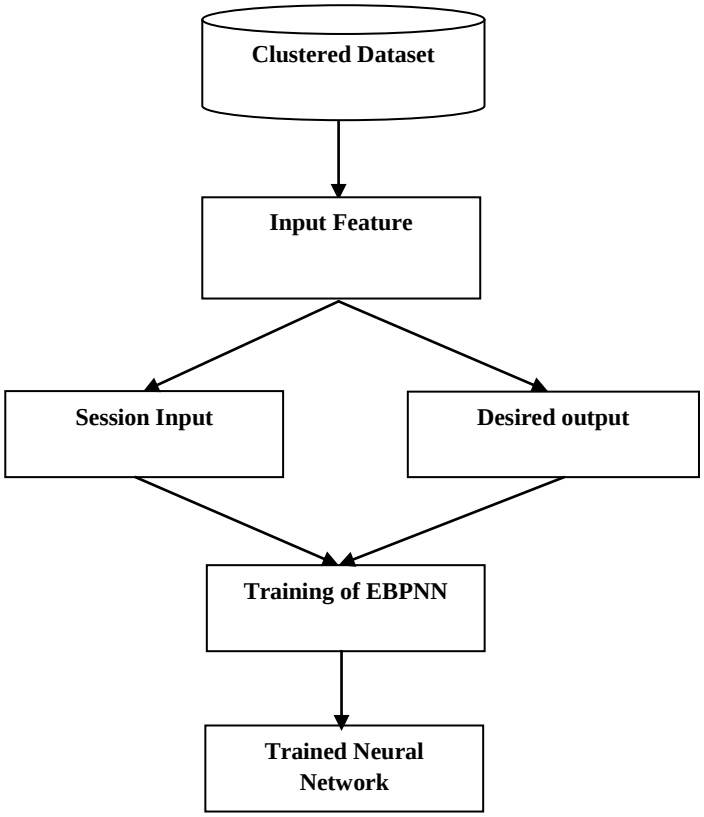

Fig 2: Proposed work testing module.

3.2 Error Back Propagation Neural Network

In this module cluster dataset of sessions are utilize to train the EBPNN. Then trained dataset is utilized for the testing of unknown attack session.

3.2.1 Input Feature

As cluster dataset provide the sessions in group as per the genetic algorithm so it has to divide into two group first act as neural network input while other act as desired output. Considering first input feature vector which consist of numeric values where clustered numeric values are arrange in the input matrix. While second desired output vector consist of the class of session which was obtained from the genetic algorithm. This can be understand by below example where

Training of Error Back Propagation Neural Network (EBPNN): Here feature vector obtained are used as the input in the neural network while desired output make proper weight adjustment in the network. So with fix number of iteration or epochs work will get trained neural network

3.2.2 Proposed Algorithm: Neural Network Algorithm

Input: D // Dataset

Output: EBPNN // Trained Neural Network

1. PD←Pre_Process(D) // Preprocessed Dataset
2. P←Generate_population(c, s) //c: number of classes, s: population size, P: population
3. Loop 1:iter // iter: number of iterations
4. F←Fitness_function(P, PD) // F: fitness value of each probable solution
5. Best←Min(F)
6. Loop 1:s
7. P←Crossover(Best, P[s])
8. EndLoop
9. EndLoop
10. CD←Cluster(PD, Best) // CD: Clustered Dataset
11. Loop 1:n // n : number of session in the dataset
12. S←CD[n]
13. [X D]←Input_Feature(S) // X input session feature and D desired output
14. EndLoop
15. EBPNN←Initialize(In, Hn, On,) // In: neurons in inputs layer, Hn Number of Hidden layer, On: number of output layer
16. Loop 1:n
17. Loop 1: iter // iter: number of iterations
18. EBPNN←Train(EBPNN, X[n], D[n])
19. EndLoop
20. EndLoop

3.2.3 Testing of EBPNN

In this step input query is preprocess as done in the training module, similarly feature vector is create for input in the neural network. Finally feature vector is input in the EBPNN which give output. Now analysis of that output is done that whether specified class is desired one or not.

4. EXPERIMENT AND RESULTS

Data Set: For the evaluation of the whole work the dataset used is NSL KDD [12] about which previous chapter has already explained and the collection of the all evaluating vectors look like. Where numeric terms are used for feature learning and at the end of each vector it has the corresponding class. The pre-processing step and its requirement have already been explained.

Evaluation Parameter

To test our result this work use following measures the accuracy of the, that is to say Precision, Recall and F-score. These parameters are depend on the TP, TN, FP and FN.

$$\mathrm{Precision} = \frac{True\_Positive}{True\_Positive + False\_Positive}$$

$$\mathrm{Recall} = \frac{True\_Positive}{True\_Positive + False\_Negative}$$

$$F\_Score = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

In order to make the better evaluation for this work one more parameter has introduced that is accuracy of the class of the intrusion. Accuracy of the work is calculate by:

Accuracy = (true positives +false negatives)/(Total_Normal + Total_Intrusion)

4.1 Results

Table 1. Precision value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	Precision Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	0.879694	0.981552
6000	0.877468	0.981516
9000	0.876332	0.983288

From above Table1, it is obtained that with the increase in dataset size precision value rate increases. As the number of patterns are more in the dataset so the results are more accurate. Here it was shown that use of genetic algorithm increase the precision value.

Table 2. Recall value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	Recall Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	0.978754	0.987204
6000	0.977995	0.987705
9000	0.976717	0.987621

From above table 2 it is obtained that with the increase in dataset size recall value rate increase. As number of patterns are more in the dataset so results are more accurate. Here it was shown that use of genetic algorithm increase the recall value.

Table 3. F-Measure value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	F-Measure Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	0.926584	0.98437
6000	0.925008	0.984601
9000	0.923805	0.98545

From above table 3 it is obtained that use of EBPNN in proposed work has high F-measure value as compared to previous work. Here it was shown that use of new approach of neural network training reduce the execution time as compared to RNN used in previous method.

Table 4. Execution time value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	Training Execution time (second) Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	19.1139	6.5586
6000	43.0747	10.3615
9000	76.064	16.8743

From above table 4 it is obtained that with the increase in dataset size execution time value increase. Here it was shown that use of new approach of neural network training reduce the execution time as compared to RNN used in previous method.

Table 5 Execution time value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	Testing Execution time (second) Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	51.5539	18.2219
6000	68.0965	37.6935
9000	92.9525	69.0244

From above table 5.5 it is obtained that with the increase in dataset size execution time value increase. Here it was shown that use of new approach of neural network training reduce the execution time as compared to RNN used in previous method.

Table 6. Execution time value comparison of RNN and EBPNN at different Dataset Size

Data-Set Size	Accuracy Value Comparison	
	RNN (Existing)	EBPNN (Proposed)
3000	0.927	0.983672
6000	0.924333	0.983669
9000	0.923111	0.984557

From above table 6 it is obtained that with the increase in dataset size execution time value increase. Here it was shown that use of new approach of neural network training reduce the execution time as compared to RNN used in previous method.

5. CONCLUSION

Network security is one of the most important nonfunctional requirements in a system. Over the years, many software solutions have been developed to enhance network security and this paper provides an efficient system which has been a promising one for detecting intrusion of different kind where, one can get the detail of the class of attack as well. Results show that all type of attacks are identified accurately by the system as the accuracy value is above 96%. In future it needs to be improved by putting data on the unsupervised network, so it automatically updates the new behavior of the intruder. One more issue that remains in this work is to use dynamic adaptable technique for learning new type of attack.

6. REFERENCES

[1]Yogita B. Bhavasar, Kalyani C. Waghmare "Intrusion Detection System Using Data Mining Technique: Support Vector Machine" 2013 International Journal of Emerging Technology and Advance Engineering volume 3, Issue 3, March 2013.

[2]A.R. Jakhale, G.A. Patil, "Anomaly Detection System by Mining Frequent Pattern using Data Mining Algorithm from Network Flow", International Journal of Engineering Research and Technology, Vol. 3, No.1, January 2014, ISSN. 2278-0181.

[3]Aljurayban, N.S.; Emam, A. (21-23 March 2015). Framework for Cloud Intrusion Detection System Service. Web Applications and Networking (WSWAN), 2015 2nd World Symposium on, p1-5

[4] Barolli, Leonard; Elmazi, Donald; Ishitaki, Oda, Tetsuya; Taro; Yi Liu, Uchida, Kazunori. (24-27 March 2015). Application of Neural Networks for Intrusion Detection in Tor Networks. Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on, p67-72.

[5]Koushal Kumar, Jaspreet Singh Batth "Network Intrusion Detection with Feature Selection Techniques using Machine-Learning Algorithms" International Journal of Computer Applications (0975 – 8887) Volume 150 – No.12, September 2016.

[6]R.Karthik, Dr.S.Veni, Dr.B.L.Shivakumar "Improved Extreme Learning Machine (IELM) Classifier For Intrusion Detection System" International Journal of Engineering Trends and Technology (IJETT) – Volume-41 Number-2 - November 2016.

[7]Premansu sekhara rath, 2manisha mohanty, 3silva acharya, 4monica aich "optimization of ids algorithms using data mining technique" International Journal of Industrial Electronics and Electrical Engineering, ISSN: 2347-6982 Volume-4, Issue-3, Mar.-2016

[8]Mohammadreza Ektefa, Sara Memar, Fatimah Sidi, Lilly Suriani Affendey "Intrusion Detection Using Data Mining Techniques", 978-1-4244-5651-2/10/$26.00 ©2010 IEEE

[9]YU-XIN MENG," The Practice on Using Machine Learning For Network Anomaly Intrusion Detection" Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, 978-1-4577-0308-9/11/$26.00 ©2011 IEEE

[10]Liu Hui, CAO Yonghui "Research Intrusion Detection Techniques from the Perspective ofMachine Learning"2010 Second International Conference on MultiMedia and Information Technology 978-0-7695-4008-5/10 $26.00 © 2010 IEEE

[11]Chuanlong Yin , Yuefei Zhu, Jinlong Fei, And Xinzheng He. "A Deep Learning Approach For Intrusion Detection Using Recurrent Neural Networks" current version November 7,2017 .Digital Object Identifier

[12]NSL-KDD dataset source:https://github.com/defcom17/NSL_KDD/blob/master/Original%20NSL%20KDD%20Zip.zip