IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Issue-10 E-ISSN: 2347-2693

Effect of Genetic Algorithm on Artificial Neural Network for Intrusion Detection System

Amin Dastanpour[1*], Suhaimi Ibrahim[2]and RezaMashinchi[3]

[1,2]*Advanced Informatics School, University Technology Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia*
[3]*Faculty of Computing, University Technology Malaysia, Johor*

Available online at: www.ijcseonline.org

Received: 20/Sep/2016 Revised: 30/Sep/2016 Accepted: 16/Oct/2016 Published: 31/Oct/2016

***Abstract*—** By increasing the advantages of network based systems and dependency of daily life with them, the efficient operation of network based systems is an essential issue. Since the number of attacks has significantly increased, intrusion detection systems of anomaly network behavior have increasingly attracted attention among research community. Intrusion detection systems have some capabilities such as adaptation, fault tolerance, high computational speed, and error resilience in the face of noisy information. So, construction of efficient intrusion detection model is highly required for increasing the detection rate as well as decreasing the false detection. . This paper investigates applying the following methods to detect the attacks intrusion detection system and understand the effective of GA on the ANN result: artificial Neural Network (ANN) for recognition and used Genetic Algorithm (GA) for optimization of ANN result. We use KDD CPU 99 dataset to obtain the results; witch shows the ANN result before the efficiency of GA and compare the result of ANN with GA optimization.

Keywords*—Artificial Neural Network (ANN); Intrusion detection; Genetic algorithm (GA); Machine learning; Network Security;*

I. INTRODUCTION

Today, the internet is used by most people for communication. Therefore, they expect a secure network or a secure channel for communication[1]. In the past few years, a large number of research studies have been done in the area of network security to ensure the safety of the transmitted and stored data[2]. Intrusion detection system (IDS) is a tool that the administrators use for network protection against the malicious activities[3].

There is a limitation for this system; it is only capable of detecting the known attacks and there should be a frequent update for the attacks signatures [4]. On the other hand, they need to consider too many attributes. Therefore, the network traffic that needs to be dealt with is very large and the data distribution is highly imbalanced. Thus the recognition of abnormal behavior against the normal behavior becomes a challenge. To overcome this problem, various artificial intelligence methods are developed[5].

The objective of machine learning is to discover and learn and then adapt to the circumstances that might change over time and therefore improving the performance of the machine[6]. In the field of intrusion detection, the reference input is used for the algorithms of machine learning so that they "learn" In this paper used artificial neural network (ANN) for detection. ANN is the most popular techniques of machine learning and it has been used to solve the regression and classification problems. There are several advantages for the ANN, but one of them is considered as the most popular one on them and that the patterns of attacks. Then the algorithms are deployed on the input attacks that have been previously unseen in order to perform the actual process of detection. Aside from the ability of recognizing the new patterns of attacks, these algorithms have another capability. It is to sanitize the dataset with the redundant and irrelevant features, thus the dataset will be containing only a few numbers of key features and the process of detection will be optimized[7]. Its ability to learn from data set observation. In the mentioned way, ANN is applies as an approximation tool for random functions[8]. The task of these tools is to assist the estimation of the methods with the most ideality and the most cost effectiveness for reaching the solutions while they define the distributions of computing or functions of computing. Instead of the entire set of data, a data sample is taken by ANN for reaching the solutions. There are three interconnected levels in ANNs. The input neurons are in the first layer. The data is sent by these neurons to the next layer which is the 2nd one and in turn, the 2nd layer will send the outcome neurons to the 3rd layer[9].

ANN is applicable for data recognition and classification. However, for the purpose of classification and recognition, a large data set is required by the ANN. For optimizing this data type and overcome the accuracy problem of ANN, this paper proposes to use the Genetic Algorithm (GA) [10]. The

Corresponding Author: *Amin Dastanpour, amindastanpoure@gmail.com*
Advanced Informatics School, University Technology Malaysia / Jalan Semarak, 54100 Kuala Lumpur, Malaysia

aim of this paper is to propose the use of genetic algorithm for improving the mechanism of ANN [11, 12].

Genetic algorithm is one of the most popular and most used algorithms for the machine learning. Genetic algorithm is an exploratory and adaptive algorithm for work and search which has been base on the natural genetics evolutionary ideas[13]. In GA, a solution is represented by each one of these individuals for the problem[14]. Since GA is a parallel algorithm and is capable of finding a solution in a problem with multi subsets, it is considered to be suitable for IDS[15]. Besides, GA is capable of proposing a solution in a single solution with an optimal value. One other capability of the GA is that proper method for IDS, especially for the detection of attacks which are based on the behavior of the human[16].

In the field of machine learning, the process in which a subset or a set is selected in a related feature in order to make a model of solution is called feature selection. When the feature is being used, it is assumed that the data includes some irrelevant and redundant information. Therefore, when it comes to machine learning and for overcoming this problem, the algorithm of feature selection is applied by the researchers to choose the useful and relevant information. In this study, the use of GA plays a significant role in feature selection and in order to understand this role, the algorithm of GA is compared with some other algorithms of related work.

Organization of this paper as a flow: section 2 in related work and try to prepare small literature review. Section 3 is methodology and tries to explain and expand method of this paper. Section 4 is experimental result and observation. Section 5 is conclusion try to conclude the whole objective in this paper.

II. PREVIOUS WORK

Previous researcher try to overcome this problem by various method such as, Combined a hierarchical clustering algorithm, a simple feature selection procedure and the fuzzy IF–THEN rules[17], Ant colony and support vector machine(SVM)[18], LCFS, FFSA and MMIFS[19],fuzzy rule based[20],and SVM Classification, GA optimization [21], cooperative game theory based framework[22] and four-angle-star[23]. Table 1 illustrate of these method in brief.

Author(s)	Year	technique(s)	Method(s)	Advantages	Disadvantages
Bin Luo et al. [23]	2014	four-angle-star visualized feature generation (FASVFG)	evaluate the distance between 5-class classification problem	high accuracy in validation experiment	Number of Features are not low enough
Dastanpour et al. [21]	2013	SVM and GA	Used the SVM for recognition of IDS pattern and GA for optimization	High rate of accuracy	Number of Features are not low enough
Li et al. [18]	2012	Removal feature selection and support vector machine(SVM)	Employed clustering method, Removal Feature Selection and (SVM) in their detection system	Reducing the data sets, preparing small training dataset	Rate of accuracy is not high enough
Xin Sun et al. [22]	2012	cooperative game theory based framework (CoFS)	Evaluate the power of each feature. The power can be served as a Metric of the each feature	handle the feature selection problem with the less feature	Rate of accuracy is not high enough
Amiri et al. [19]	2011	LCFS ,FFSA and MMIFS as a method for comparison	The forward feature selection and mutual Modified mutual information and linear correlation feature selection	To overcome the curse of high dimensionality with the High rate of accuracy	High number of features
Tsanget al. [17]	2007	fuzzy IF–THEN rules	proposed fuzzy rule-based system is evolved from an agent-based evolutionary framework and multi objective optimization	Feature selection wrapper to search for an optimal feature	Rate of accuracy is not high enough
Abraham et al. [20]	2007	fuzzy rule based classifiers	framework for Distributed Intrusion Detection Systems	lightweight and more accurate	High number of features

Table 1: Previous work

III. RESEARCH METHODOLOGY

A. Methodology Description

Figure 1 has illustrated the overall method and the main idea of this study. First of all, try to divide dataset in to the 2 part of data for testing and training. Then these methods try to make the standard format of dataset for the ANN recognition (explain in part 4.2). For the next step is when the ANN training is done, ANN try to classification of KDD CUP 99 testing dataset and out put the accuracy of the system detection and monitor or plot it as a system result. When the ANN recognition is finished, then the result of ANN is Data input of GA. In this step, GA tray to optimize the result of ANN recognition. At the final, GA after the

optimization of ANN result, plot it for compare the result of ANN and ANN with GA and understanding the effect of GA in the ANN recognition for the intrusion detection system with the KDD CUP 99 dataset. In this paper, parameters of ANN are used shows in table 3.Table 4 shows the division of ANN on this paper.

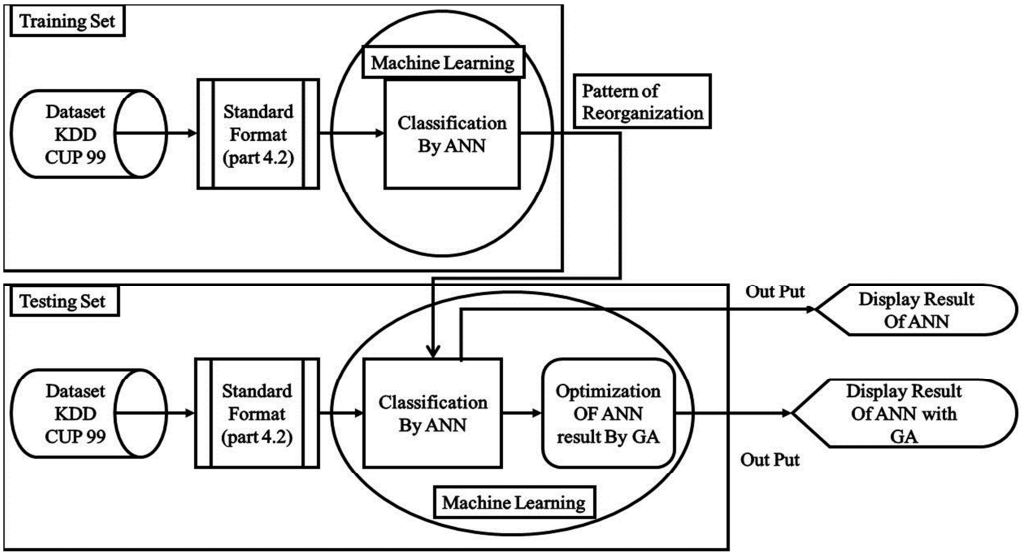


Figure1: Overall method of this paper

Name of parameters	Parameters used in this paper
Epoch	20 iteration for each feature. Total 20*41= 820
Hidden layer	15
Out put	1
Data division	Random
Training	Levenberg-marquadt
Performance	Mean squared error
Type of train	Trainlm

Table 2: Parameters used on this paper for ANN

Total number of sample	41
training	70%of all record (494020) include 29 sample
validation	15% of all record (494020) include 6sample
testing	15% of all record (494020) include 6sample

Table 3: Division of ANN

In this paper, the error of ANN counted by the equation (1)[24]

$$\boldsymbol{E} = \sum_{\boldsymbol{i=1}}^{\mathbf{n}} \left(\frac{|\boldsymbol{RN-ID}|}{\boldsymbol{SID}}\right) * \mathbf{100} \qquad (1)$$

In this formula: E = error, RN = result of ANN detection, ID = identify of data (it can be only 0 or 1), SID=total number of record are used, In this case, total number of record the part of testing is 15% of all data, that is mean 15% of 494020 and equal to 74103.

The syntax of below shows the main part of coding in this paper. *From 1 to 41 do*

Select feature from training data
Select feature from testing data
Train ANN with 20 repetition of each train and Kind of training is trainlm
Get back the result of ANN
Test ANN
Calculate error by equation number 2
End

B. Standard Format of dataset

This Dataset contains 494'020 record and 41 features; in fact, this data set has 494'020 rows and 41 columns. In this

paper, we add a column to identify each record as an attack or normal in the step of standard format; while in additional columns, we assume 1 for attack data and 0 for normal data. Figure 2shows a sample dataset used in this paper. The reason is to use for counting and understanding of recognition error in ANN.

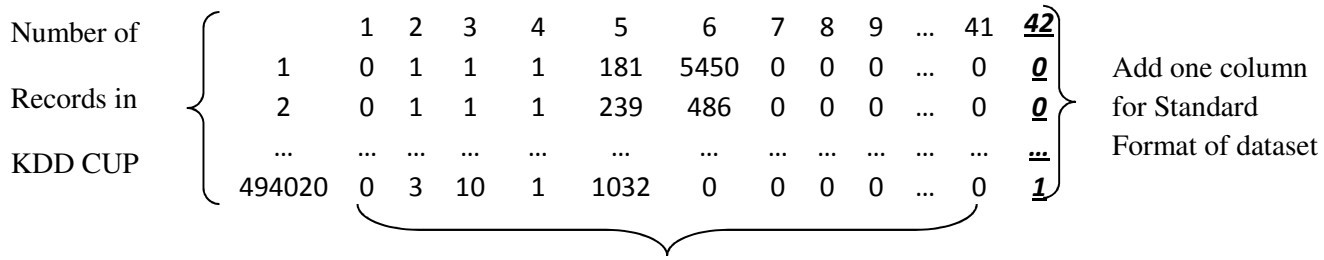


Figure 2: Example of dataset used in this paper

C. Data Description

The data set that has been used in this paper is the KDD cup 1999 .The reason is this type of dataset is complete and it is known as the most popular IDS Researcher. There are 24 attack types in this dataset and they can be classified into four categories[25].

DOS: This attack type is used for user behavior understanding[26].

R2L: Some packets are sent by this type of attack into the network to gain the network accessibility as a known local user[27].

U2R: This type of attacks is known as the attacks in which the attacker will have access to the system and will be able to exploit the vulnerabilities for achieving the key permissions[28].

Probing: The network is scanned by this type of attack for data collection about the targeted host[29].

D. Intrusion Detection Using Genetic Algorithm

In this research we used GA to optimize the ANN result. The parameter and setting of GA is shows in table 2.

A fitness function is a particular type of objective function that is used to summaries, as a single figure of merit, how close a given design solution is to achieving the set aims. Each design solution, therefore, needs to be awarded a figure of merit to indicate how close it came to meeting the overall specification, and this is generated by applying the fitness function to the test, or simulation, results obtained from that solution [30]. In this paper F is fitness function and equation (2) is fitness function in this study [31-34].

$$f = \frac{\alpha}{A} - \frac{\beta}{B} \qquad (2)$$

Whereα the number of correctly identified attacks,A is the total number of attacks in the training dataset, β is the number of normal connections incorrectly characterized as attacks, andB is the total number of normal connections in the training dataset.

Genetic algorithm parameter	Genetic algorithm Amount
Number of iterations	49 in this case
Population size	20
Mutation rate	0.15
Fraction of population of kept	0.5
Total number of bits in a chromosome	41
Crossover	single point
Type of GA	Binary GA

Table4: Parameters of Genetic Algorithm

In this paper, stopping criteria is: If fi - fi-5 ≤ 10-6 then stop iterations.

E. Mutation

Mutation is a genetic operator used to maintain genetic diversity from one generation of a population of genetic algorithm chromosomes to the next. It is analogous to biological mutation. Mutation alters one or more gene values in a chromosome from its initial state. In mutation, the solution may change entirely from the previous solution. Hence GA can come to better solution by using mutation. Mutation occurs during evolution according to a user-definable mutation probability. This probability should be set low. If it is set too high, the search will turn into a primitive random search. The classic example of a mutation operator involves a probability that an arbitrary bit in a genetic sequence will be changed from its original state.

Bit string mutation

The mutation of bit strings ensue through bit flips at random positions.

Example:

1 0 1 0 0 1 0

↓

1 0 1 0 1 1 0

The probability of a mutation of a bit is $\frac{1}{l}$, where L is the length of the binary vector. Thus, a mutation rate of 1 per mutation and individual selected for mutation is reached.

F. Selection

Selection is the stage of a genetic algorithm in which individual genomes are chosen from a population for later breeding (using the crossover operator).

A generic selection procedure may be implemented as follows:

- The fitness function is evaluated for each individual, providing fitness values, which are then normalized. Normalization means dividing the fitness value of each individual by the sum of all fitness values, so that the sum of all resulting fitness values equals 1.
- The population is sorted by descending fitness values.
- Accumulated normalized fitness values are computed (the accumulated fitness value of an individual is the sum of its own fitness value plus the fitness values of all the previous individuals). The accumulated fitness of the last individual should be 1 (otherwise something went wrong in the normalization step).
- A random number R between 0 and 1 is chosen.
- The selected individual is the first one whose accumulated normalized value is greater than R.

G. Crossover

In this paper used the Crossover technique. In genetic algorithms, crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. In this paper used fitness proportionate selection as a selection method of chromosomes for crossover part. Fitness proportionate selection is the individual is selected on the basis of fitness. The probability of an individual to be selected increases with the fitness of the individual greater or less than its competitor's fitness. Fitness proportionate selection, also known as roulette wheel selection, is a genetic operator used in genetic algorithms for selecting potentially useful solutions for recombination. In fitness proportionate selection, as in all selection methods, the fitness function assigns fitness to possible solutions or chromosomes. This fitness level is used to associate a probability of selection with each individual chromosome.

In a part of crossover, many crossover techniques exist for organisms which use different data structures to store themselves such as Single -point crossover, Two-point crossover, Cut and splice, Uniform Crossover and Half Uniform Crossover, Three parent crossover and Crossover for Ordered Chromosomes. In this paper used the Single -point crossover. A single crossover point on both parents' organism strings is selected. All data beyond that point in either organism string is swapped between the two parent organisms. The resulting organisms are the children shown in figure 3.

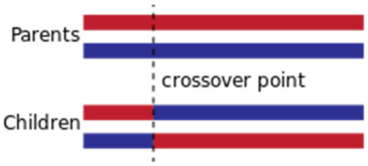


Figure 3: Organisms the children in GA

IV. OBSERVATION

In this paper, try to apply the ANN as the detection system. In IDS, ANN has a rule of pattern recognition and it makes the pattern of Attack and normal for the detection of data. Figure 4 is result of ANN accuracy or detection without any optimization by each feature.

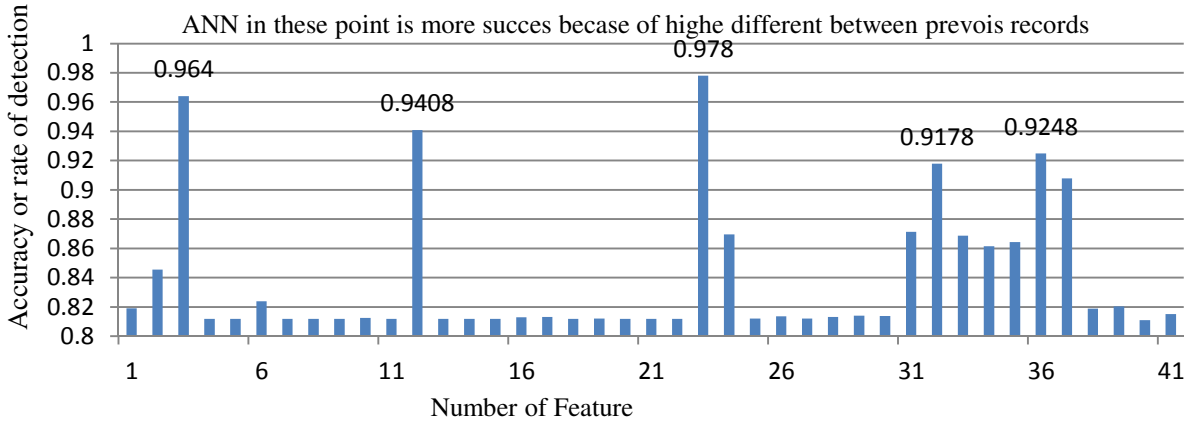


Figure 4: Result of ANN Accuracy in testing step without any optimization

This paper, try to show the performance of the GA on the result of ANN to improve the accuracy of ANN. In IDS with the ANN classifier algorithm, the result of ANN not completely satisfy because it cannot achieve the high detection rate of accuracy and need some changes.

In the situation the GA has role of optimization of ANN classification. GA optimizes and improves the performance of ANN classification. Figure 5 shows the result of ANN accuracy with the GA optimization.

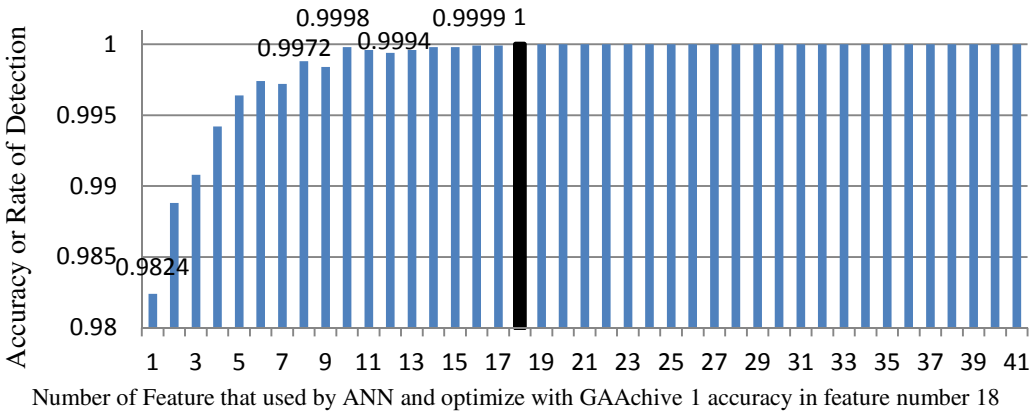
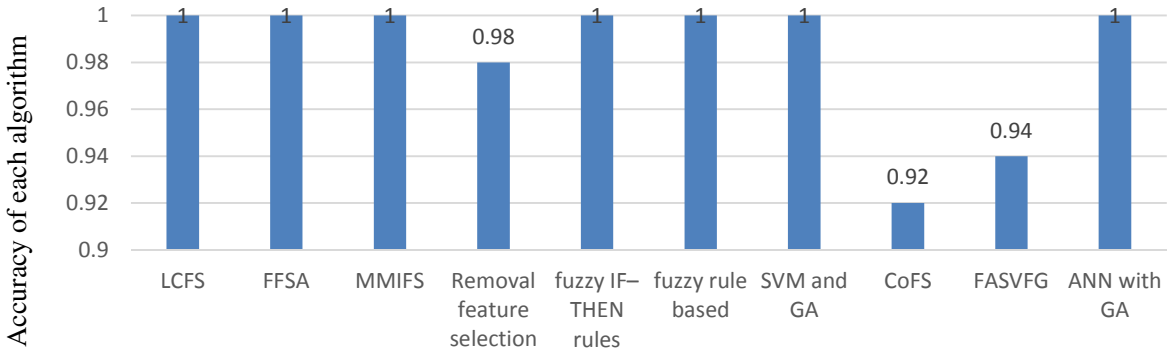


Figure 5: Result of ANN with GA optimization after testing step

In this study after the installing the GA for support ANN, the system can achieve almost 1.0 accuracy in the feature number 18. That is showing the system can be achieve 1.0 accuracy with the only 18 number of feature so GA after optimize the ANN result also can minimize the number of feature. In the testing part of machine learning, ANN with GA can achieve 1.0 accuracy only with the 18 number of featured. To understand better performance of propose system in this paper, table 5 has showing the comparison with other algorithm witch selected on the related work. In the figure 6 is comparison of these algorithms by the accuracy. For the better comparison of these algorithms in the section of feature selection, figure 7 is show.

Name of algorithm	Accuracy (between 0 and 1)	Feature #
LCFS[19]	1.0	21
FFSA[19]	1.0	31
MMIFS[19]	1.0	24
Removal feature selection[18]	0.98	21
fuzzy IF–THEN rules[17]	1.0	20
fuzzy rule based[20]	1.0	41
SVM and GA[21]	1.0	21
CoFS[22]	0.92	8
FASVFG[23]	0.94	20
ANN with GA	1.0	18

Table 5: Comparing of accuracy



Name of Algorithm that studed in previows work section

Figure 6: Comparative of accuracy

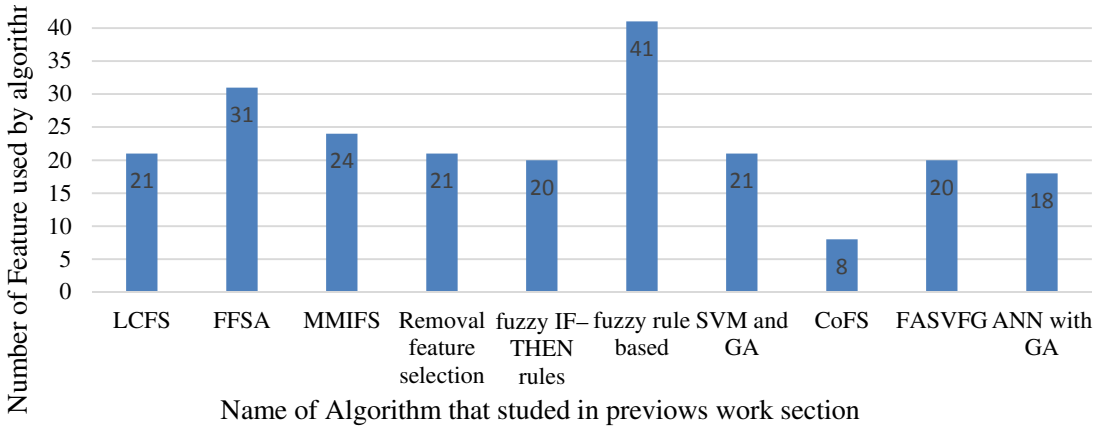


Figure 7: Number of feature selection for comparison

In this comparison (Figure 5), although other algorithms achieve 1.0 accuracy like model of this paper (ANN with GA), the number of feature selection is not less than ANN with GA. Although in the figure 6 other algorithm selected feature less than ANN with GA, the accuracy in not as high as ANN with GA. To sum it up, ANN with GA achieves high accuracy with the less number of features. The result of total error detection of ANN counted by the main syntax of this paper is show in parentage on figure 8.

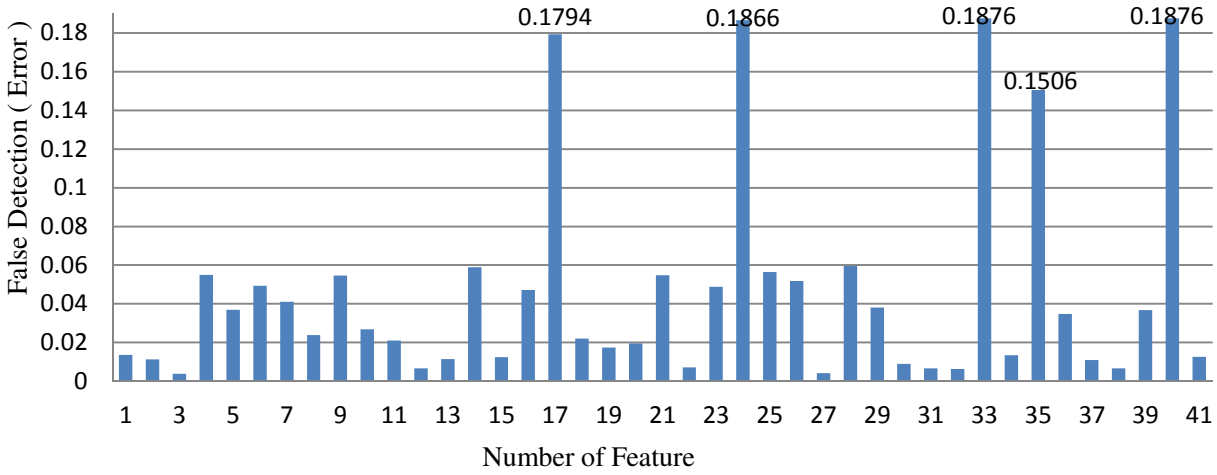


Figure8:

Rate of ANN Error and Highlights the heist Error

V. SUMMARY AND FEATURE WORK

In brief, GA has been proposed in this study for optimization of ANN Result to achieve high accuracy with the less features. The results indicate that the highest rate for detection and lowest number of features are achieved by GA with ANN in comparison to other algorithm, which study about that in related work. In This study using the KDD Cup 99 dataset for detection of four network attacks categories. In the future work, try to apply other optimization on the ANN for intrusion detection system.

VI. ACKNOWLEDGEMENT

This research is funded by the Research University grant of University Technology Malaysia (UTM) under the Vote no. 08H28. The authors would like to thank the Research Management Centre of UTM and the Malaysian ministry of education for their support and cooperation including students and other individuals who are either directly or indirectly involved in this project.

VII. REFERENCES

[1] O. A. Soluade and E. U. Opara, "Security Breaches, Network Exploits and Vulnerabilities: A Conundrum and an Analysis."

[2] H.-H. Gao, H.-H. Yang, and X.-Y. Wang, "Ant colony optimization based network intrusion feature selection and detection," in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, 2005, pp. 3871-3875.

[3] Z. Y. M. Liu, "Intrusion Detection Systems," in *Applied Mechanics and Materials*, 2014, pp. 852-855.

[4] A. Simmonds, P. Sandilands, and L. van Ekert, "An ontology for network security attacks," in *Applied Computing*, ed: Springer, 2004, pp. 317-323.

[5] K. M. Shazzad and J. S. Park, "Optimization of intrusion detection through fast hybrid feature selection," in *Parallel and Distributed Computing, Applications and Technologies, 2005. PDCAT 2005. Sixth International Conference on*, 2005, pp. 264-267.

[6] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin, "Intrusion detection by machine learning: A review," *Expert Systems with Applications,* vol. 36, pp. 11994-12000, 2009.

[7] A. Tamilarasan, S. Mukkamala, A. H. Sung, and K. Yendrapalli, "Feature ranking and selection for intrusion detection using artificial neural networks and statistical methods," in *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, 2006, pp. 4754-4761.

[8] V. T. Goh, J. Zimmermann, and M. Looi, "Towards intrusion detection for encrypted networks," in *Availability, Reliability and Security, 2009. ARES'09. International Conference on*, 2009, pp. 540-545.

[9] R. Ma, "Neural Networks for Intrusion Detection," 2009.

[10] A. Dastanpour, S. Ibrahim, R. Mashinchi, and A. Selamat, "Comparison of genetic algorithm optimization on artificial neural network and support vector machine in intrusion detection system," in *Open Systems (ICOS), 2014 IEEE Conference on*, 2014, pp. 72-77.

[11] O. Linda, T. Vollmer, and M. Manic, "Neural network based intrusion detection system for critical infrastructures," in *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, 2009, pp. 1827-1834.

[12] G. P. a. N. Mishra, "Optimal Feature Selection in Stream Data Classification Using Improved Ensemble Classifier for High Dimension Data," *International Journal of Computer Sciences and Engineering,* vol. 04, pp. 12-18, Sep -2016.

[13] M. S. Hoque, M. Mukit, M. Bikas, and A. Naser, "An implementation of intrusion detection system using genetic algorithm," *arXiv preprint arXiv:1204.1336,* 2012.

[14] M. H. Mashinchi, M. R. Mashinchi, and S. M. H. Shamsuddin, "A Genetic Algorithm Approach for Solving Fuzzy Linear and Quadratic Equations," *World Academy of Science, Engineering and Technology,* vol. 28, 2007.

[15] P. Gupta and S. K. Shinde, "Genetic Algorithm Technique Used to Detect Intrusion Detection," in *Advances in Computing and Information Technology*, ed: Springer, 2011, pp. 122-131.

[16] V. K. Kshirsagar, S. M. Tidke, and S. Vishnu, "Intrusion Detection System using Genetic Algorithm and Data Mining: An Overview," *International Journal of Computer Science and Informatics ISSN (PRINT),* pp. 2231-5292, 2012.

[17] C.-H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *Pattern Recognition,* vol. 40, pp. 2373-2391, 2007.

[18] Y. Li, J. Xia, S. Zhang, J. Yan, X. Ai, and K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method," *Expert Systems with Applications,* vol. 39, pp. 424-430, 2012.

[19] F. Amiri, M. Rezaei Yousefi, C. Lucas, A. Shakery, and N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," *Journal of Network and Computer Applications,* vol. 34, pp. 1184-1199, 2011.

[20] A. Abraham, R. Jain, J. Thomas, and S. Y. Han, "D-SCIDS: Distributed soft computing intrusion detection system," *Journal of Network and Computer Applications,* vol. 30, pp. 81-98, 2007.

[21] A. Dastanpour and R. A. R. Mahmood, "Feature selection based on genetic algorithm and SupportVector machine for intrusion detection system," in *The Second International Conference on Informatics Engineering & Information Science (ICIEIS2013)*, 2013, pp. 169-181.

[22] X. Sun, Y. Liu, J. Li, J. Zhu, H. Chen, and X. Liu, "Feature evaluation and selection with cooperative game theory," *Pattern recognition,* vol. 45, pp. 2992-3002, 2012.

[23] B. Luo and J. Xia, "A novel intrusion detection system based on feature generation with visualization strategy," *Expert Systems with Applications,* vol. 41, pp. 4139-4147, 2014.

[24] Y. Chen, B. Yang, and A. Abraham, "Flexible neural trees ensemble for stock index modeling," *Neurocomputing,* vol. 70, pp. 697-703, 2007.

[25] M. K. Siddiqui and S. Naahid, "Analysis of KDD CUP 99 Dataset using Clustering based Data Mining," *International Journal of Database Theory & Application,* vol. 6, 2013.

[26] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, "Denial-of-service attack-detection techniques," *Internet Computing, IEEE,* vol. 10, pp. 82-89, 2006.

[27] M. Sabhnani and G. Serpen, "KDD Feature Set Complaint Heuristic Rules for R2L Attack Detection," in *Security and Management*, 2003, pp. 310-316.

[28] M. Sabhnani and G. Serpen, "Formulation of a Heuristic Rule for Misuse and Anomaly Detection for U2R Attacks in Solaris Operating System Environment," in *Security and Management*, 2003, pp. 390-396.

[29] G. Zargar and P. Kabiri, "Identification of effective network features for probing attack detection," in *Networked Digital Technologies, 2009. NDT'09. First International Conference on*, 2009, pp. 392-397.

[30] A. L. Nelson, G. J. Barlow, and L. Doitsidis, "Fitness functions in evolutionary robotics: A survey and analysis," *Robotics and Autonomous Systems,* vol. 57, pp. 345-370, 2009.

[31] M. Pillai, J. H. Eloff, and H. Venter, "An approach to implement a network intrusion detection system using genetic algorithms," in *Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, 2004, pp. 221-221.

[32] T. P. Fries, "A fuzzy-genetic approach to network intrusion detection," in *Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation*, 2008, pp. 2141-2146.

[33] P. Jongsuebsuk, N. Wattanapongsakorn, and C. Charnsripinyo, "Network intrusion detection with Fuzzy Genetic Algorithm for unknown attacks," in *Information Networking (ICOIN), 2013 International Conference on*, 2013, pp. 1-5.
[34] S. Dhopte and M. Chaudhari, "Genetic Algorithm for Intrusion Detection System."

Author Profile

Amin Dastanpour received BS computer engineering from Kerman University, Iran, in 2009 and received MS degree in University Putra Malaysia, Malaysia, in 2013. He is currently a Ph.D. student of Advanced Information School at University Technology Malaysia, Malaysia International Campus, Kuala Lumpur, Malaysia. His research interests include Intrusion Detection System.

Suhaimi Ibrahim received the Bachelor in Computer Science (1986), Master in Computer Science (1990), and PhD in Computer Science (2006). He is an Associate Professor attached to Dept. of Software Engineering, Advanced Informatics School (AIS), Universiti Teknologi Malaysia International Campus, Kuala Lumpur. He currently holds the post of Deputy Dean of AIS. He is an ISTQB certified tester and being appointed a board member of the Malaysian Software Testing Board (MSTB). His research interests include software testing, requirements engineering, Web services, software process improvement, mobile and trusted computing.

Reza Mashinchi has received his MS in computer engineering from SEUA, Armenia, and PhD in computer science and information systems from UTM, Malaysia. Reza is involved in academia such as computer society of Iran (CSI), IEEE Malaysia, and PMI. In addition, he is serving as a reviewer, an editorial board member, and a member of program committee in number of international conferences and journals. His research interests include soft computing techniques, granular computing approaches, fuzzy systems, meta-heuristics, predictive modeling, and regression analysis.

