NEURAL NETWORK-BASED DDOS DETECTION REGARDING HIDDEN LAYER VARIATION

[1]IMAM RIADI, [2]ARIF WIRAWAN MUHAMMAD, [3]SUNARDI

[1]Department of Information System, Ahmad Dahlan University, Yogyakarta, Indonesia

[2]Department of Informatics Engineering, Ahmad Dahlan University, Yogyakarta, Indonesia

[3]Department of Electrical Engineering, Ahmad Dahlan University, Yogyakarta, Indonesia

E-mail: [1]imam.riadi@is.uad.ac.id, [2]arif1508048009@webmail.uad.ac.id, [3]sunardi@mti.uad.ac.id

ABSTRACT

Distributed Denial of Service attack (DDoS) is a structured network attack coming from various sources and fused to form a large packet stream. DDoS attacks aiming to disrupt the services available in the target tissue by flooding the target bandwidth or processing capacity of the system by making the target network server becomes overloaded. Network packet classification is one method of network defense system in the organization of the Internet in order to avoid DDoS attacks. Network packet classification can be carried out either by utilizing the method of Artificial Neural Network (ANN). The proposed work of network traffic packet classification applying variation of hidden layer with Quasi-Newton method training function and statistical network traffic packet feature extraction have the result that ANN with two hidden layers outperformed than ANN with single or three hidden layers. ANN with two hidden layers gives overall consistent mse and convergence speed, also higher correct classification percentage at 99.04%. Quasi-Newton method (trainlm) is qualified and suit for classification task based on value of regression both in the training and validation phase.

Keywords: *DDoS, Classification, Neural-Network, Hidden Layer*

1. INTRODUCTION

Distributed denial of service (DDoS) is a structured network attack coming from various sources and fused to form a large packet stream. DDoS attacks, generally utilizing resources from the slave computer coordinated by the attacker to decrease the target network resources causing legitimate client can not access these resources. DDoS packet stream behaves as normal packet flow pattern so it is very difficult to distinguish between normal or DDoS packet stream [1].

DDoS attacks aiming to disrupt the available services in the target network by flooding the target bandwidth or processing capacity system to make the target network servers become overloaded [2]. DDoS attacks are mostly aimed to overwhelm the server's bandwidth such as ICMP / UDP Flood, Ping of Death, TCP SYN Flood, SMURF, UDP Storm, Syslogd, and Mailbomb. DDoS packet stream with a large volume [3] causes the target system can not handle and end up with a loss of resources such as system shutdown, loss of data, moreover, the system loses the overall of owned services.

DDoS attacks are reported as a form of attack that has the highest frequency in recent decades and seriously affected the Internet service provider and the world Internet community [4]. DDoS is a threat to the cyber world and became the main problem of cyber security. The DDoS attack is one of the hacker's main weapon in crippling targets and proved to be a perennial threat to the infrastructure, users, and organizations on the Internet as being a risk to the confidentiality, security, integrity, and availability of resources on the Internet [5][6].

DDoS attacks are not only done by computer but can also be carried out by a mobile device by applying slow DDoS technique in which small-sized DDoS packets transmitted continuously from a large number of mobile devices. From attacker's side, slow DDoS is very beneficial, because it requires only a small computational resource. A large number of mobile devices that are scattered throughout the world which exceeds the number of computers also influence the amount of amplification DDoS attack originating from mobile devices that have become bot [7].

Network packet classification is one method of network defense system in the Internet organization in order to avoid DDoS attacks. Network packet classification can be carried out either by utilizing the method of Artificial Neural Network (ANN). Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by biological neural cell system in the information processing and has the constituent components that work together to process information signals.

Network packet classification for DDoS attacks detection in TOR network using ANN carried on research [8]. Research [8] utilizing optimization of a sinusoidal function as a feature extractor of the network packet. ANN used in [9] with Resilient-Backpropagation function combined with the ensemble of classifier outputs method and Neyman-Pearson cost minimization strategy for detection of DDoS attack based on DARPA and KDDCUP datasets. Research [10] adopted the ANN method to detect DDoS attacks based on darknet traffic. TCP/80 and UDP/53 packets used as input and optimized by Locally Sensitive Hashing methods. ANN used in [11] to recognize illegal packets in the network, by taking advantage of the Backpropagation functions. TCP, ICMP, and UDP packet used as inputs in the [11]. Research [12] proved the ANN method can be used to detect a new type of DDoS attacks, in Hadoop and Hbase environment.

Earlier research regarding DDoS detection using ANN method, does not address the parameters in ANN that underlie the accuracy level in detecting DDoS attacks. Therefore in this study we do research focuses on ANN parameters, one of which is the hidden layer numbers used in the training phase of the sets input patterns data. We aiming to find out wheter significant differences on ANN detection accuracy level by applying variation on hidden layer numbers and also to find out the best-hidden layer numbers for DDoS detection. We use DDoS packet traffic datasets published by the Center for Applied Internet Data Analysis (CAIDA) and normal packet traffic datasets published by the Computer Networks Laboratory of Ahmad Dahlan University Indonesia.

We limits this study focused to the effectiveness of hidden layer numbers that used in the training phase of the sets input patterns data by Quasi-Newton training function. We will compare the detection accuracy level that resulted by several variation of hidden layer numbers. By comparing these detection accuracy level, it will show the better hidden layer scheme.

2. BASIC THEORY

2.1 Artificial Neural Network

An Artificial Neural Network (ANN) is a biologically inspired model of computational composed by various processing elements (neurons). Neurons are connected with coefficients or weights which construct the neural network's structure. ANN have elements for information processing they are transfer function, weighted inputs, and outputs [13]. ANN is composed of single layer or multiple layer neurons as seen in Figure 1

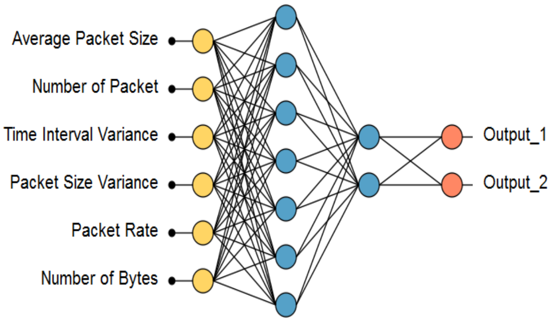


Figure 1: Artificial Neural Network

2.2 Mean Squared Error

Mean square error (mse) is the most ANN important parameter for performance evaluation of training functions parameters [14]. Mean square error reflect an absolute error of ANN training output pattern with desired output pattern as seen in equation 1.

$$\text{MSE} = \frac{\sum_{i=1}^{N}(y_i - o_i)^2}{N} \qquad (1)$$

Where y_i is the target and o_i is the observed output and N is the number of data set.

2.3 Packet Feature

To classify the network packet, the first step is to preprocess the datasets to reduce noise. In this study, we extracted network packet features with statistical method. The aim of feature extraction is to measure certain properties and attributes in original data that distinguish one input pattern from another pattern. Total six features are extracted from network traffic. These features are:

a. Average packet size.
Logically, DDoS attacks would overwhelm a target computer network to spend resources, so the longer DDoS attack occurs, then it is always followed by a rise in the value of average packet size [15].

b. Number of packets
DDoS attacks overwhelm a target computer network by sending many packets at a certain time lag. Therefore, DDoS always cause anomalies to the number of the packet. [15].

c. Time interval variance
DDoS attack delivers packages in large numbers occurred in a certain time span, the value of time interval variance will be smaller and nearly zero. As seen from equation 2 [16]

$$t_c^2 = \frac{\sum \left(t_n - \bar{t}\right)^2}{n} \qquad (2)$$

Where t_n is time of a packet is received and $\bar{t}$ is the rate of time a packet is received

d. Packet size variance
Logically, in the normal traffic, packet size variance values are high. In DDoS attacks, packet size variance value will result in a small value and is close to zero, due to the size of the package in a DDoS attack that was sent to overwhelm a target computer network worth monotony as seen from equation 3. [16]

$$p_c = \sqrt{\frac{\sum (p_n - \bar{p})^2}{n}} \qquad (3)$$

Where p_n is received packet size, and $\bar{p}$ is received packet size rate.

e. Packet rate.
Packet rate reflects the number of packets sent by the source address to a destination address within a specific time frame as seen from equation 4. [16]

$$n_p \times \frac{1}{(t_e - t_s)} \qquad (4)$$

Where n_p is the number of packets, t_e is end time a packet is received, t_s is the initial time a packet is received.

f. Number of bytes.
Logically DDoS attacks in the span of time will always be an increase in the number of bytes in constant. [16]

These six features are fed as input to multilayer Artificial neural network.

2.4 Hidden Layer

For complex problems, multilayer neural network is the best model as it overcomes the drawback of the single-layer neural network by the adding one or more layers between input and output layer, called hidden layer. In a feedforward multilayer neural network, the inputs signals are multiplied by the connection weights and summed together then directed to a transfer function to give an input for hidden layer neuron. The transfer function such purelin, hardlim,sigmoid and logistic executes on the weighted sum of the neuron's inputs [17].

2.5 Quasi-Newton Algorithm (trainlm)

There are numbers of batch training algorithms which can be used to train a network [18] one of which is Newton algorithm. The first step of Newton method is second derivatives called the Hessian matrix of the performance index at the current values of the weights and biases [17]. The weight update of Newton's method can be seen in from equation 4:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mathbf{A}_k^{-1}\mathbf{g}_k \qquad (4)$$

A_k is the Hessian matrix for an index of performance at the current values of the weights and biases. It is complex and has more time consumption to compute w_k+1 when A_k is large. Newton's method is fast to reach convergence than conjugate gradient methods, but Newton's method is complex and time-consuming to compute the Hessian matrix for feed forward neural networks [19]. Based on Newton's method there a new class of method is called a Quasi-Newton method which doesn't require calculation of second derivatives [20].

3. METHODOLOGY

The proposed work of network packet classification applying a variation of hidden layer involves two steps as follows:

(1) Step One
As seen in Figure 2, step one contain:
a. Get DDoS datasets from CAIDA and normal packet flow datasets from Computer Networks Laboratory of Ahmad Dahlan University Indonesia in .pcap format.
b. Open file DDoS datasets as seen in Figure 3, using packet sniffer software [21].

Open normal packet flow datasets as seen in Figure 4, using packet sniffer software [21].

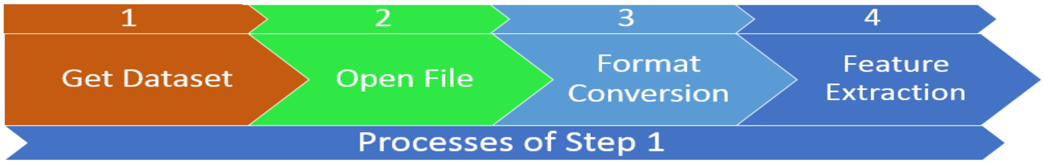
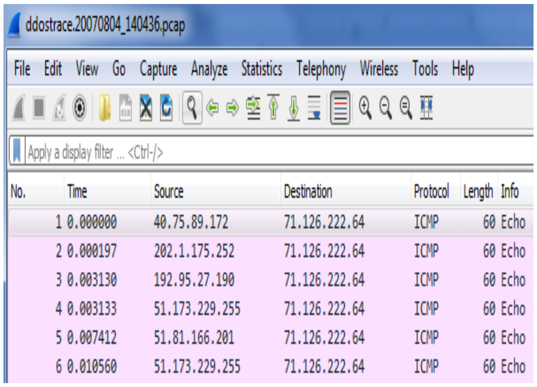


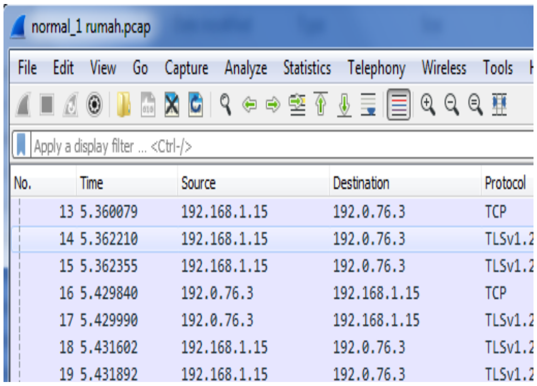
Figure 2: Convert Pcap File Format


ddostrace.20070804_140436.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	40.75.89.172	71.126.222.64	ICMP	60	Echo
2	0.000197	202.1.175.252	71.126.222.64	ICMP	60	Echo
3	0.003130	192.95.27.190	71.126.222.64	ICMP	60	Echo
4	0.003133	51.173.229.255	71.126.222.64	ICMP	60	Echo
5	0.007412	51.81.166.201	71.126.222.64	ICMP	60	Echo
6	0.010560	51.173.229.255	71.126.222.64	ICMP	60	Echo

Figure 3: Open DDoS Traffic Packet

c. Convert file format, from .pcap format to .csv format [22] as seen in Figure 5.


normal_1 rumah.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools
Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol
13	5.360079	192.168.1.15	192.0.76.3	TCP
14	5.362210	192.168.1.15	192.0.76.3	TLSv1.2
15	5.362355	192.168.1.15	192.0.76.3	TLSv1.2
16	5.429840	192.0.76.3	192.168.1.15	TCP
17	5.429990	192.0.76.3	192.168.1.15	TLSv1.2
18	5.431602	192.168.1.15	192.0.76.3	TLSv1.2
19	5.431892	192.168.1.15	192.0.76.3	TLSv1.2

Figure 4: Open Normal Traffic Packet

The .csv result can see in Figure 6

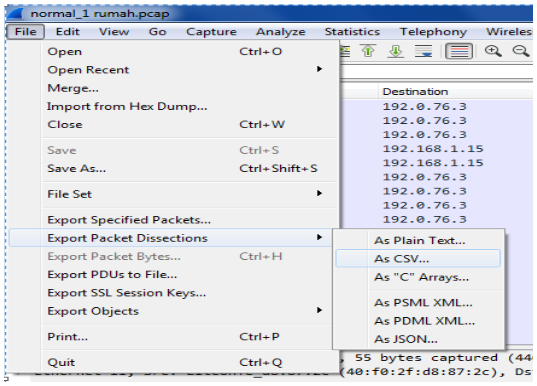
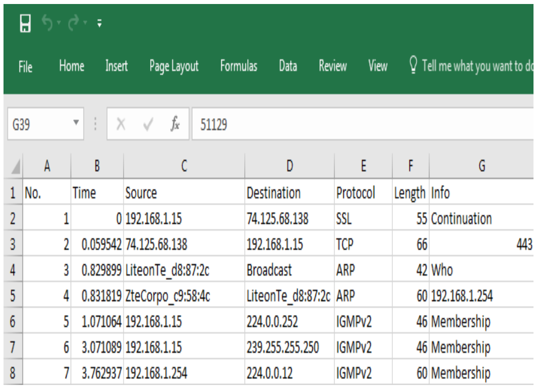


Figure 5: Convert Pcap File Format


File Home Insert Page Layout Formulas Data Review View Tell me what you want to do

G39 51129

	A	B	C	D	E	F	G
1	No.	Time	Source	Destination	Protocol	Length	Info
2	1	0	192.168.1.15	74.125.68.138	SSL	55	Continuation
3	2	0.059542	74.125.68.138	192.168.1.15	TCP	66	443
4	3	0.829899	LiteonTe_d8:87:2c	Broadcast	ARP	42	Who
5	4	0.831819	ZteCorpo_c9:58:4c	LiteonTe_d8:87:2c	ARP	60	192.168.1.254
6	5	1.071064	192.168.1.15	224.0.0.252	IGMPv2	46	Membership
7	6	3.071089	192.168.1.15	239.255.255.250	IGMPv2	46	Membership
8	7	3.762937	192.168.1.254	224.0.0.12	IGMPv2	60	Membership

Figure 6: Convert Csv Result

d. Extract network packet feature, by the statistical method and resulting:
   i. Average packet size
   ii. Number of packets
   iii. Time interval variance
   iv. Packet size variance
   v. Packet rate
   vi. Number of bytes

The features above can be used as ANN input to classify network packets as DDoS or normal.

(2) Step Two

As seen in Figure 7, step two contain:

a. Variating ANN hidden layer by 10 type as seen in Table 1, for reason to get the best classification percentage in identifying network traffic (normal or DDoS). Kolmogorov formula (2n+1) where n is the number of input layer neuron generally used to establish the number of hidden layer neurons. Until now there is no certainty as to the number of hidden layers is best used in resolving a problem with a network [15]

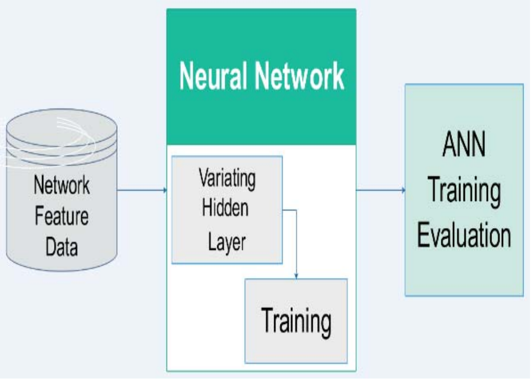


Figure 7: Processes of Step Two

b. Training ANN applying hidden layer variation. In this paper, we use Quasi-Newton (trainlm) training algorithms to classify network packet, for reason Newton's training algorithms gives fast optimization than other algorithms [13] [19].
c. Evaluation of ANN training using comparison parameters such as, mse, the number of the epoch at the end of training, correct classification, regression on training, and regression on validation.

4. RESULT

In this paper, experiments were carried out on Windows 7 (64-bit) operating system with an i7 processor and 4GB of RAM. ANN training processes are coded in Matlab 2010R environment using ANN Toolbox. The overall experimental dataset consists of 100000 traffic data by six features. The dataset consists of 50000 (50%) DDoS packet flow data, and 50000 (50%) normal packet flow data. In purpose of learning, the dataset was divided as Matlab 2010R default into sets for training (70%), validation (15%), and testing (15%).

Distribution of the data set for training, validation, and testing carried out by the random function (dividerand) to avoid the tendency to bias in the sample pattern. The training process in the hidden layer using sigmoid transfer function, whereas the training process in the output layer uses a linear transfer function. The basic parameters used in the training process is epoch = 20000, performance function = mse, goal = 0.01, maximum fail = 6, minimum gradient = 1.00e-10, mu = 1.00e+10. All ANN variation is trained until performance function mean squared error (mse) is less than 0.01. Comparison of ANN training results is presented in Table 1.

The Quasi-Newton method (trainlm) training process is fast and allows higher learning rates while maintaining its stability. CPU elapsed time at the end of training is less than ten seconds for each ANN type. Within less than 60 epoch (iteration), Quasi-Newton method (trainlm) can achieve performance goal (mse). Small mse value indicates a good performance of ANN. In another hand, achieving performance process on Quasi-Newton method (trainlm) have a drawback in memory requirement that relatively big depends on the size of ANN layer scheme, the more number of neuron in hidden layer, more big its memory requirement. Overall, single hidden layer ANN (ANN type 1, 4, 7, 9) produce mse value lower than the ANN with two and three hidden layers. In another side, ANN with two hidden layers (ANN type 2, 5, 8, 10) provides a consistent mse value. Whereas ANN with three hidden layers tends to have a gap on the mse value. Adding more hidden layer affect the overall value of mse. The more hidden layer, the greater the mse value, as seen in Figure 8. Detailed information about training performances of these function provides in Table 1.

ANN classification percentage, are not much affected according to the number of neurons in their hidden layer. According to Table 1, ANN classification percentage gap between the minimum (97,03%) and maximum (99,04%) are less than 2,5%. Table 1 provides information detail on classification percentage of Quasi-Newton method.

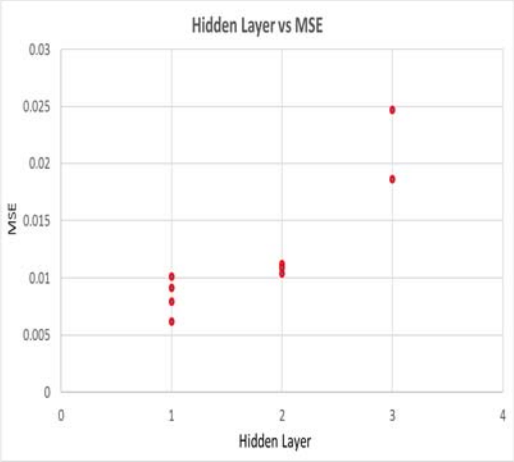


Figure 8: Hidden Layer-MSE

In this study, ANN with two hidden layer and three neurons (ANN type 9, 10) produce the highest classification level with a value of 99.04%, as seen in Figure 9.

Table 1: ANN Layer Variation Scheme

ANN Type	Input Neuron (n)	Hidden Layer Variation	Total Hidden Layer
1.	6	13	13
2.		8-5	
3.		9-2-2	
4.	6	12	12
5.		8-4	
6.		9-2-1	
7.	6	6	6
8.		3-3	
9.	6	3	3

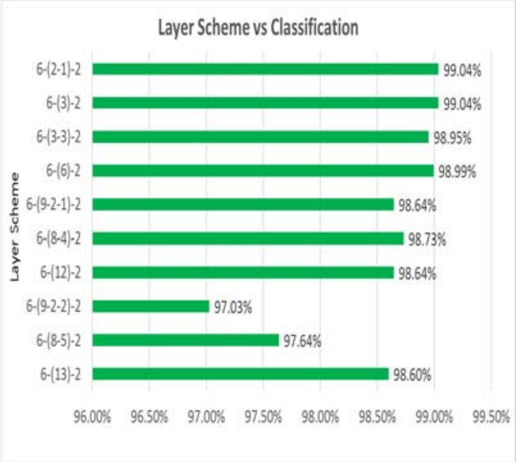


Figure 9: Layer Scheme-Classification

In this proposed work, training epoch of Quasi-Newton method (Matlab-trainlm) is affected by increasing the number of hidden layers. Figure 10 shows the number of iterations (epochs) at the end of training.

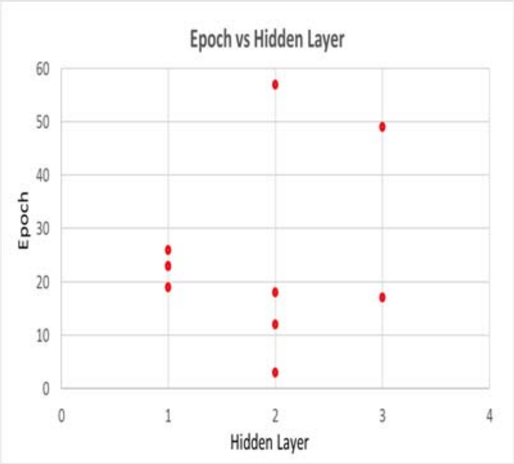


Figure 10: Epoch-Hidden Layer Scheme

From Figure 9, we can conclude that most efficient ANN type regarding number of hidden layer, in this case, is ANN with two hidden layers because it gives the overall minimum number of epoch although there is one anomaly in which ANN with two hidden layers produces more than 50 epoch.

The regression analysis compares the actual outputs of each ANN type with corresponding desired outputs (targets). Regression analysis returns the correlation coefficient (R) number between actual and corresponding desired output, the slope and also the intercept of the best-linear-fit equation. R number range between 0.0 to 1.0. The more values of R near to 1.0 show the more correct response of the each ANN type. Figure 11 show the regression value on training based on ANN scheme.

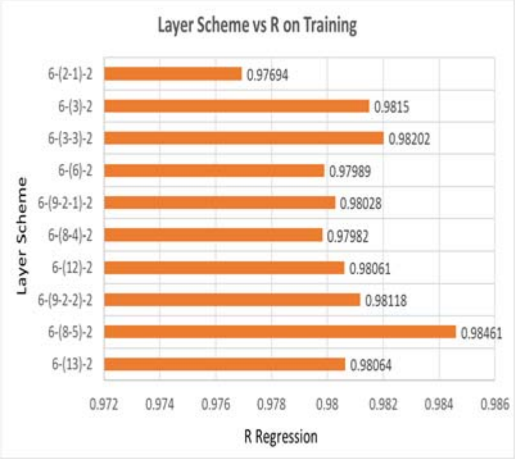


Figure 11: Layer Scheme-R Training

Figure 12 show the regression value on validation based on ANN scheme. Figure 11 and Figure 12 clearly indicates that Quasi-Newton method (trainlm) is qualified and suit for the classification task. It gives regression value over 0.94 both in the training and validation.

On this proposed work we found significant differences applying hidden layer variation in the neural network.

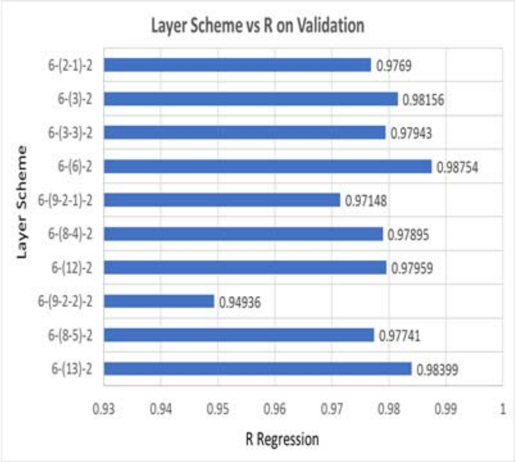


Figure 12: Layer Scheme-R Validation

Increasing the number of layers in ANN does not give much effect on correct classification percentage for all ANN type they are in acceptable range, only the convergence speed decreases. The convergence speed have strong relation to epochs. The more epoch means the convergence speed is decreased. Single hidden layer ANN has higher convergence speed than two or three hidden layers ANN. We further analyzed on other parameters like increasing the sample size of input patterns that presented to the ANN, reducing error goal and use more training method

5. CONCLUSION

ANN can be used as an effective tool for network packet classification with the appropriate combination of learning, transfer, hidden layer and training functions. ANN with two hidden layers gives overall consistent mse and convergence speed, also higher correct classification percentage at 99.04% and also Quasi-Newton training function method (Matlab-trainlm) is qualified and suit for classification task, based on value of regression both in the training and validation phase.

ACKNOWLEDGEMENT

The author would like to thank all the individuals who were involved in this research and experiment process. Special thank to Paul Hick from CAIDA who provide the DDoS dataset and also Sukma Aji from Computer Networks Laboratory of Ahmad Dahlan University Indonesia who provide the Normal dataset. This research is carried out without any conflict of interest from any party.

REFERENCES

[1] Mahadev, V. Kumar, and K. Kumar, “Classification of DDoS Attack Tools and its Handling Techniques and Strategy at Application Layer,” *2016 2nd Int. Conf. Adv. Comput. Commun. Autom.*, pp. 1–6, 2016.

[2] Arbor Networks, “Worldwide Infrastructure Security Report,” vol. XI, no. September, 2016.

[3] A. Iswardani and I. Riadi, “Denial of Service Log Analysis Using Density K-Means Method,” *Journal of Theoritical and Applied Information Technology (JATIT)*, vol. 83, no. 2, pp. 299–302, 2016.

[4] S. T. Zargar, J. Joshi, and D. Tipper, “DiCoTraM : A Distributed and Coordinated DDoS Flooding Attack Tailored Traffic Monitoring,” *IEEE IRI 2014*, no. August 13-15, pp. 120–129, 2014.

[5] R. Wankhede, “Intrusion Detection System using Classification Technique,” vol. 139, no. 11, pp. 25–28, 2016.

[6] I. Riadi, J. E. Istiyanto, A. Ashari, and Subanar, “Internet Forensics Framework Based-on Clustering,” *International Journal of Advanced Computer Science and Application*, vol. 4, no. 12, pp. 115–123, 2013.

[7] P. Farina, E. Cambiaso, G. Papaleo, and M. Aiello, “Understanding DDoS Attacks From Mobile Devices,” *3rd Int. Conf. Futur. Internet Things Cloud Underst.*, 2015.

[8] T. Ishitaki, D. Elmazi, Y. Liu, T. Oda, L. Barolli, and K. Uchida, “Application of Neural Networks for Intrusion Detection in Tor Networks,” *Proc. - IEEE 29th Int. Conf. Adv. Inf. Netw. Appl. Work. WAINA 2015*, pp. 67–72, 2015.

[9] M. Kale and D. . Choudhari, “DDOS Attack Detection Based on an Ensemble of Neural Classifier,” *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 14, no. 7, pp. 122–129, 2014.

[10] S. H. A. Ali, S. Ozawa, T. Ban, J. Nakazato, and J. Shimamura, “A Neural Network Model for Detecting DDoS Attacks using Darknet Traffic Features,” *2016 Int. Jt. Conf. Neural Networks*, no. November 2014, pp. 2979–2985, 2016.

[11] A. Saied, R. E. Overill, and T. Radzik, “Detection of known and unknown DDoS attacks using Artificial Neural Networks,” *Neurocomputing*, vol. 172, pp. 385–393,

2015.

[12] T. Zhao, D. C. T. Lo, and K. Qian, "A Neural Network Based DDoS Detection System Using Hadoop and HBase," *Proc. - 2015 IEEE 17th Int. Conf. High Perform. Comput. Commun. 2015 IEEE 7th Int. Symp. Cybersp. Saf. Secur. 2015 IEEE 12th Int. Conf. Embed. Softw. Syst. H*, pp. 1326–1331, 2015.

[13] S. Haykin, *Neural Networks and Learning Machines*, Third Ed. New York: Pearson Prentice Hall, 2008.

[14] H. Demuth, *Neural Network Toolbox Users Guide*, Sixth Ed., vol. 24, no. 1. Natick, Massachuset: The MathWorks, Inc, 2002.

[15] C. J. Hsieh and T. Y. Chan, "Detection DDoS attacks based on neural-network using Apache Spark," *2016 Int. Conf. Appl. Syst. Innov. IEEE ICASI 2016*, pp. 1–4, 2016.

[16] T. P. Thwe Thwe Oo, "A statistical approach to classify and identify DDoS attacks using UCLA dataset," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 2, no. 5, p. 1766, 2013.

[17] L. V. Fausset, *Fundamental of Neural Networks Architectures, Algorithms, and Application*. Englewood Cliffs, New York: Prentice-Hall, 1994.

[18] N. Pise and P. Kulkarni, "Algorithm Selection for Classification Problems," *SAI Comput. Conf. 2016*, pp. 203–211, 2016.

[19] Y. H. Hu and J.-N. Hwang, *Handbook of Neural Network Signal Processing*, First Edit. New York: CRC Press, 2002.

[20] M. Anthony and P. L. Bartlett, *Neural Network Learning : Theoretical Foundations*, First Edit. New York: Cambridge University Press, 2009.

[21] T. V.Lillard, C. P. Garrison, and C. A. Schiller, *Digital Forensic for Network, Internet, and Cloud Computing*, First. Burlington, Massachusets: Syngress Elsevier, 2010.

[22] M. Zulfadhilah, I. Riadi, and Y. Prayudi, "Cyber Profiling using Log Analysis and K-Means Clustering A Case Study Higher Education in Indonesia," *International Journal of Advanced Computer Science and Application*, vol. 7, no. 7, pp. 430–435, 2016.