Botnet detection and feature analysis using backpropagation neural network with bio-inspired algorithms

Jen-Li Liao, Kuan-Cheng Lin* and Jyh-Yih Hsu*

Department of Management Information Systems,
National Chung Hsing University,
250 Kuo Kuang Rd., Taichung 402, Taiwan
Email: dlnx38@hotmail.com
Email: kclin@nchu.edu.tw
Email: hsu@nchu.edu.tw
*Corresponding authors

Abstract: Botnets has been the major type of cybercrime recently, the amount of infected computers gradually increasing each year. Many companies and schools are often troubled with problems, such as DDOS, phishing, spam, and stealing of personal data, because botnet is constantly changing its network structure, attack patterns and data transmission, making it more and more difficult to detect. In this paper, we proposed some new features to detect the botnet traffic, and we found the best solutions by using feature selection algorithm. These two methods are particle swarm optimisation and genetic algorithms, and by using backpropagation network as the classifier, we evaluate our subset feature on botnet detection that shows high detection rate, and we validate that own manufactured feature packet transmission time of regularity can be adopted, and the accuracy will change with the t-value.

Keywords: botnet; genetic algorithms; backpropagation network; BPN; particle swarm optimisation; PSO.

Reference to this paper should be made as follows: Liao, J-L., Lin, K-C. and Hsu, J-Y. (2018) 'Botnet detection and feature analysis using backpropagation neural network with bio-inspired algorithms', *Int. J. Cognitive Performance Support*, Vol. 1, No. 2, pp.132–142.

Biographical notes: Jen-Li Liao received his MS in the Department of Management Information Systems from the National Chung Hsing University, in 2013. His primary interests lie in the areas of botnet detection and machine learning.

Kuan-Cheng Lin received his BS in Chemistry from National Taiwan University in 1988 and a PhD in Applied Mathematics from the National Chung-Hsing University in 2000. Since 2015, he has been a Professor with the Department of Management Information Systems at National Chung-Hsing University, Taichung, Taiwan. His current research interests include affective computing, intelligent tutoring system and data mining.

Jyh-Yih Hsu received his BS in Agriculture Economics from National Chung-Hsing University in 1978 and a PhD in Natural Resource and Environmental Management from University of Hawaii in 1984. Since 2004, he has been a Professor with the Department of Management Information Systems at National Chung-Hsing University, Taichung, Taiwan. His main research interests include energy and environmental economics, multi-criteria decision-making, fair trade law, public utility policy, internet of things, and big data analytics.

1 Introduction

Botnet is composed by a group of infected computers which were remote controlled by herders, many computers are easily compromised by malware or bot, and as they quietly wait for the herders' command, the herders are able to direct the activities of these compromised computers through communication channels formed by standards-based network protocols such as IRC, P2P or HTTP. Botnet are mostly used for financial gain, a bot herder employs a number of attackers to extort money by threatening to attack websites or servers.

Botnet detection can be classified into two categories, behaviour-based and signature-based. Many researchers are using behaviour-based detection to find new viruses or unknown bots, but the false positive rate is still higher than signature-based detection, and the behaviour-based is hard to implementation for other researcher, however, the signature-based often used by researchers to detect, which is faster and more efficient on botnet detection.

In recent years, feature selection is often used in intrusion detection system, because it can enhance the accuracy of classifier. Therefore, we use feature selection to get the best subset features, making the detection faster and more efficient. Although researchers found the best solution which significantly shortens work time, they did not verify the accuracy and performance on a real detection system and did not clearly describe the steps on traffic collection. In order to understand the mode of operation of the botnet, we construct a virtual architecture of botnet to capture the network traffic.

Neural networks are best applied in pattern recognition, generalisation, and trend prediction. They are fast and tolerant of imperfect data, but rarely used in botnet detection, consequently, we proposed a botnet detection that is based on a neural network as the classifier, the classifier is constructed from the most simple and popular neural networks, backpropagation (BPN) algorithm. This is good for the nonlinear problem, especially when the problem is very complex.

One of the challenges is to add a feature which measures packet of transmission time, since we found that bot connected to the IRC server of time is very regular. We found that botnet is intensely associated with packet of transmission time, and it can be adopted for botnet detection.

The remainder of this paper is organised as follows, Section 2 overviews related works that describe the botnet detection methods. Section 3 describes the data collection and experiment architecture. Section 4 presents the experiment design and experiment results. Finally, we draw a conclusion in Section 5.

2 Related work

2.1 Artificial neural network with botnet

This paper adopts neural network as a classifier, the neural network is divided into many types, such as ADALINE, SOM, BPN and CPN. After a long period of research, we selected BPN as the classifier in this experiment, because it is easy to create a number of examples of the correct behaviour, we will summarise recent papers which utilise neural network on botnet detection.

Langin et al. (2009) proposed a self-organising map self-trained to detect new bot or other network virus. The training data is collected from denied internet firewall log and can be divided into several groups. Although it can classify similar network activity and discover other malicious network traffic, it is not transferable to other locations, because this method requires re-training in another network. Nogueira et al. (2010) proposed a new way to identify the traffic patterns by a botnet detection system, which generates alarms if the malicious packet is detected. This method uses a neural network. Although the botnet detection system has high detection rates, we do not know which kind of neural network it use on this framework. Chen et al. (2012) used RBF neural network, a detection system that involves k-means clustering with back propagation algorithm. It can judge the DDOS attacker, the test results are accurate, and the system cost is low. Venkatesh and Nadarajan (2012) proposed the multi-layer feed forward neural network system that is constructed from bold driver BPN learning algorithm. The system identifies Http-based botnet behaviour and is able to achieve high accuracy and low false positive. Salvador et al. (2009) proposed a detection framework, a model is able to detect licit and illicit traffic on the new traffic, which uses different application profiles to build a detection framework. Its identification approach is also based on neural networks.

2.2 Feature selection with network traffic

Liu et al. (2009) proposed a mathematic model to improve accuracy, which used the artificial fish-swarm algorithm on intrusion detection, 41 features are optimised and found 13 features by feature selection, the dataset are using KDD-CUP 99. Alomari and Othman (2012) used the bees algorithm on anomaly detection, which used support vector machine (SVM) as classifier, it is very effectual to finding optimal solutions. Jain and Upendra (2012) using decision tree as classifier, and analysed four machine learning algorithms, there are J48, BayesNet, OneR and NB, after the experiment, the J48 show the highest accuracy and the lowest false positive rate, however, it can only use KDD dataset. Mukkamala and Sung (2003) using two learning machines for detection, there are artificial neural network and SVMs, the result show that SVM-based is better than ANN-based on performance, in addition, it using few features for detection can be enhanced efficiency and working time.

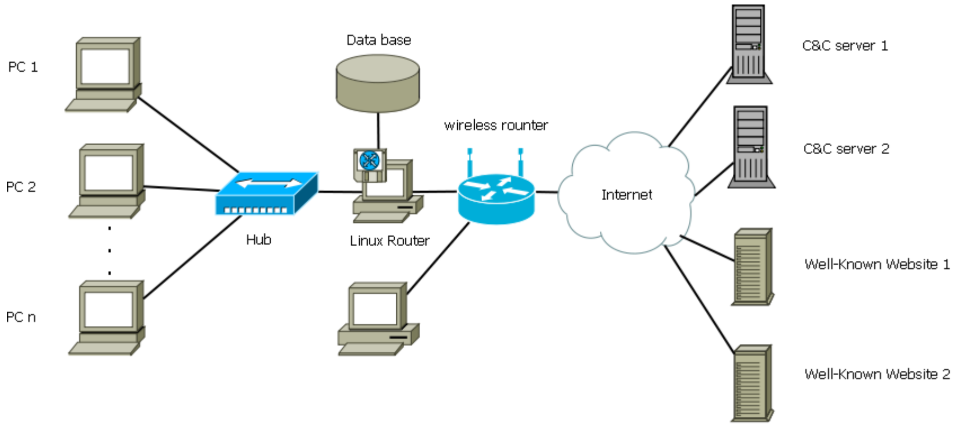
3 Feature extraction

3.1 Traffic collection

In this experiment, we use much network equipment and software to create real network architecture, as shown in Figure 1. We also make real bot to generate traffic in order to collect real malicious network traces. We generated four different trace that is traces 1 to 4, and they composed by four different bots.

First, in the infection phase, we create many virtual computers by VirtualBox (Oracle VM Virtubox, 2012), there have many different operation system, such as unpatched Windows XP SP3, unpatched Windows 7, and Linux. And execute these bot on ten virtual machines for 168 hours. Second, in the collection phase, these virtual computer connected to the C&C server through a Linux router, the Linux router create by Ubuntu, it will be capture and save to database by MySQL.

Figure 1 Botnet experiment architecture for data collection (see online version for colours)



3.2 Data pre-processing

3.2.1 Data numerical

Feature selection must transfer the raw data into training data, so we have to process packet data through SQL syntax and mathematical formulas, in this phase, we generate 12 features from database: Total_Count, Souce_Count, Port_Count, Avg_Length, Stddev_Length, Time_Regularity, Info_Char, and the destination IP address as the index in this table, these features of statement as the following:

1 Total_Count: It is the total number of packet for the destination IP address.

2 Souce_Count: It is the total number of unique external source IP address for the destination IP address.

3 Port_Count: It is the total number of unique destination ports for the destination IP address.

4 Low_Port: It is the lowest destination port for the destination IP address.

5 High_Port: It is the highest destination port for the destination IP address.

6 ICMP_Count: It is the total number of ICMP protocol for the destination IP address.

7 TCP_Count: It is the total number of TCP protocol for the destination IP address.

8 UDP_Count: It is the total number of UDP protocol for the destination IP address.

9 Avg_Length: It is the average of packet length for the destination IP address.

10 Stddev_Length: It is the standard deviation of packet length for destination IP address.

11 Time_Regularity: It is the packet transmission time of regularity for destination IP address, and the formulas as the following, we define γ as a fixed time interval array that contains $n - 1$ counters, i.e., $\gamma = \{\gamma_2, \gamma_3, \ldots, \gamma_n\}$, α as a frequently array, β as an infrequently array, and t as a constant value between 0 and 1, the default value is set to 0.5 [e.g., see equation (1)].

$$\gamma_i > \frac{2t\sum \gamma_i}{n}, \text{ then } \alpha_j = \gamma_i$$
$$\gamma_i \le \frac{2t\sum \gamma_i}{n}, \text{ then } \beta_k = \gamma_i \qquad (1)$$
$$\text{Time\_Regularity} = avg(\alpha) * \left(avg(\alpha) - avg(\beta)\right)$$

12 Info_Char: It is the packet of characters value for destination IP address, and the formulas as the following, we define C as ASCII value counter array that contains 95 counters, because ASCII only includes 95 printable characters, and others are non-printing control characters. i.e., $C = \{c_1, c_2, \ldots, c_{95}\}$, and α as a characters of ASCII value in packet of information [e.g., see equation (2)].

$$C_i = sum(\alpha_i)$$
$$Info\_Char = Max(C_i) \qquad (2)$$

Figure 2 Time interval of count (see online version for colours)

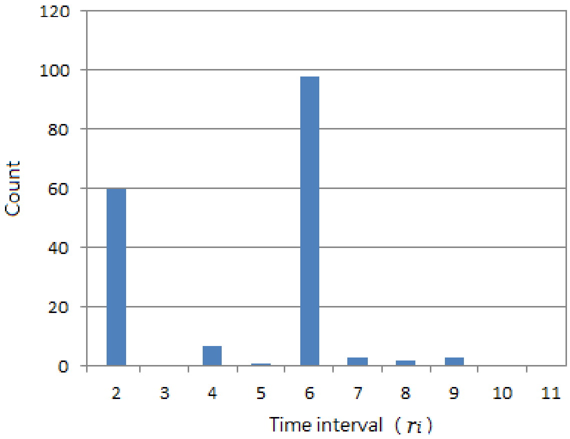
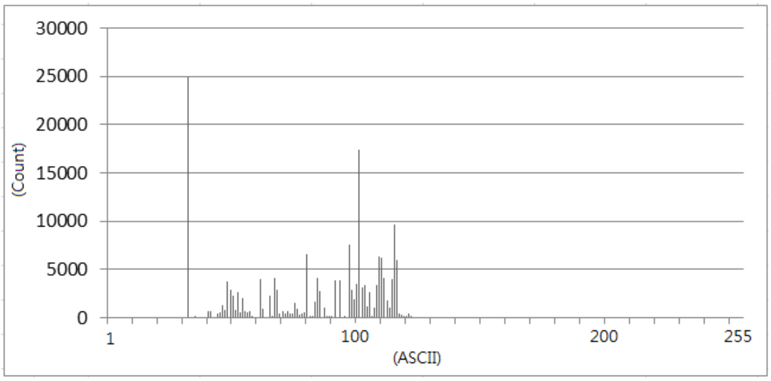


Figure 3 Packet of ASCII value count



3.2.2 Data normalisation

In order to shorten the gap between data value, we have to transfer the data into function, the value will transfer between 0 and 1 [e.g., see equation (3)].

$$value_{jk} = \frac{f_{jk}}{\max\left(f_j\right) - \min\left(f_j\right)} \tag{3}$$

3.2.3 Label the training data

In this phase, we have to increase a column for label a destination IP, if the destination IP address are malicious, then label '1', or '0'. In order to allow the feature selection algorithm to training, so it must be labelled.

3.3 Feature selection

In this paper, we proposed two feature selection approaches for botnet detection. These two methods are particle swarm optimisation (PSO) and genetic algorithms (GA), and it uses the wrapper approach as a random search method and the BPN network (BPN) as the classifier technique.

3.3.1 Fitness function

A feature selection algorithm needs a value to find the best solution, the value called fitness, we used BPN to build fitness function, a feature is selected depends on the PSO and GA algorithm, these features will be put into BPN to get a fitness.

3.3.2 The proposed PSO-BPN approach

The PSO-BPN approach consists of steps as following:

Step 1 Initialising the parameters, there are number of particles, dimension, and randomly assigned particle of position and particle of velocity.

Step 2 Calculate fitness of particle.

Step 3 Update the best local of fitness if the current fitness is better.

Step 4 Update the best global of fitness if the current local of fitness is better.

Step 5 Calculate particle of velocity and update each particle of position.

Step 6 Repeat step 2 to step 5 until the end condition is satisfied.

Step 7 Return the best solution.

Every particle has a position and velocity in PSO algorithm, whether a feature is selected depend on particle of position value, if it is greater than 0.5, then it will be selected, or not.

Table 1 Particle of position value

Feature	*Feature 1*	*Feature 2*	*Feature 3*	*Feature 4*	*Feature 5*
Position	0.23	0.65	0.87	0.16	0.42
Selected or not		Selected	Selected		

3.3.3 The proposed GA-BPN approach

The GA-BPN approach consists of steps as following:

Step 1 Initialising the parameters, and generate random population of n chromosomes, each solution consists of a string of randomly mixed 1 and 0.

Step 2 Calculate fitness of each chromosome in population.

Step 3 Select two best fitness of chromosomes from a population.

Step 4 With a crossover probability cross over the parents to generate a new child.

Step 5 There is a chance that mutation will occur, and some of the child's bits will be changed 1 or 0.

Step 6 Replaced worst two fitness of chromosomes by child of chromosomes in population.

Step 7 Repeat step 3 to step 6 until the end condition is satisfied.

Step 8 Return the best solution.

Each chromosome consists of a string of mixed 1 and 0. Whether a feature is selected depend on chromosomes of bit value, if it equal to 1, then it will be selected, or not.

Table 2 chromosomes of feature value

Feature	*Feature 1*	*Feature 2*	*Feature 3*	*Feature 4*	*Feature 5*
Chromosomes	1	1	0	1	0
Selected or not	Selected	Selected		Selected	

3.3.4 Solution of result

We use the processed dataset in this experiment, it provides botnet of traffic data in the period of one week in a virtual network, and the dataset have 12 features, the file size is about 876 MB. The dataset have four bots in this experiment.

The BPN classifier parameters as shown in Table 3 and the feature selection parameters as shown in Table 4.

Table 3 BPN parameter setting

BPN of parameters	*Value*
Learning rate	0.1
Number of input layer	Selected features of length
Number of hidden layer	(Selected features of length) / 2
Number of output layer	1
Training rate	0.95

Table 4 PSO and GA parameter setting

PSO of parameters	*Value*	*GA of parameters*	*Value*
W	1.3	Crossover rate	0.5
C1, C2	1.6	Mutation rate	0.2
Dimension	7	Genetic length	7
Number of particle	250	Number of population	200
Iterator	1,000	Iterator	1,000

We define F_1, F_2, …, F_{12}, respectively Total_Count, Souce_Count, Port_Count, Low_Port, High_Port, ICMP_Count, TCP_Count, UDP_Count, Avg_Length, Stddev_Length, Time_Regularity, Info_Char. Table 5 shows the feature selection accuracy, the subset D is the best solution, and we used eight features of Langin et al. (2009) as subset G to compare, the main purpose of this work is to find a better features subset to detection the botnet, so we take the best subset to detection botnet experiment, it is depend on accuracy in Table 5.

Table 5 The experimental result of feature selection

Subset	*Solution*	*Num.*	*PSO-BPN*	*GA-BPN*
Subset D	F_2, F_9, F_{10}, F_{11}	4	94.25	93.77
Subset G (Langin et al., 2009)	$F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8,$	8	82.14	86.45

4 Botnet detection

4.1 Experiment design

In order to collect real network packet, we reconstruct the network architecture in laboratory, the purpose is to let the experimenter's computer can connect to the internet through a Linux router, each computer normally use a variety of software, such as playing computer games, surfing the internet, checking e-mail, using IRC software to chat, and using P2P software, and infected several computers in the virtual machine for generate malicious network traces, each trace collection for a week, then record the malicious destination IP address, so that it can be used to calculate the detection system of accuracy. The collection packet traffic for more than a month and it is divided into four datasets, named trace 1 to trace 4. We use accuracy (ACC) to evaluate four subsets of performance [e.g., see equation (4)].

$$\frac{True\ positive\ (TP) + True\ negative\ (TN)}{Positive\ (P) + Negative\ (N)} \tag{4}$$

4.2 Experiment result

We take four subset of feature to detect botnet in this experiment, we set t-value as 0.4 and 0.5, the average accuracy respectively 95.09 and 94.06, it show as Table 6, the subset D has four features, there are F_2, F_9, F_{10}, $F11$, it mean they have a close relationship with botnet behaviour.

In this paper, we propose two reasons to explain the subset. First, we found the packet transmission time of regularity is very important, because the infected computer will constantly send DNS query to find their C&C servers, and these packets are very regular and intensive. Second, the standard deviation of packet length, this feature is used to illustrate a destination IP transmission packet length of similarity, malicious bot needs to connect C&C servers for confirmation and it will be sent the same message to connection, or if it is running the hackers to task, such as denial of service attacks, they can be set fixed-length packets to attack a victim.

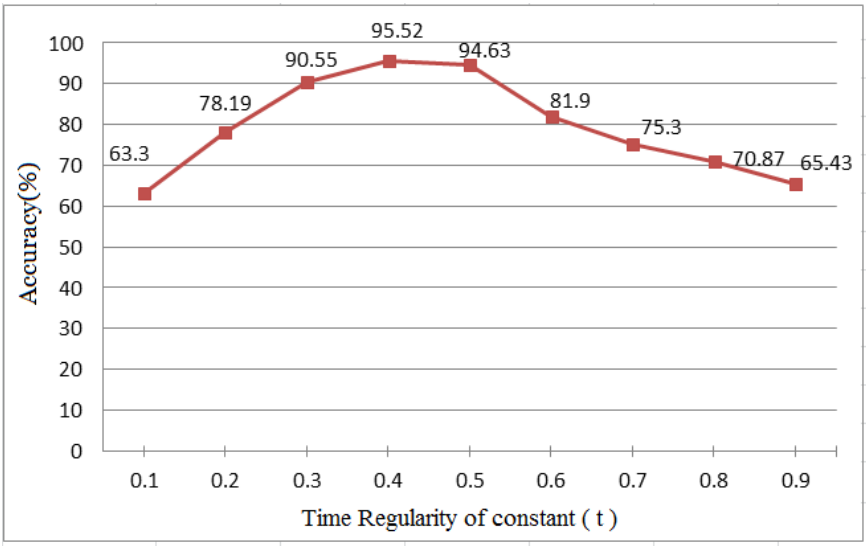
In this paper, we use mathematical formulas to transfer the feature called F_{12}, it is the packet of characters value, although it has not been selected in feature selection, but it let we know that it is unrelated with malicious packet of characters. Perhaps this is not a valid mathematical formula on Info Char, but I believe it can be used in other methods.

Table 6 Botnet detection of accuracy

Dataset	*t = 0.4*	*t = 0.5*	*Destination IP*
Trace 1	93.94	93.68	147
Trace 2	96.52	94.5	98
Trace 3	94.21	93.91	124
Trace 4	95.72	94.15	118
Average	95.09	94.06	121.75

After this experiment, we try to change the data conversion formula at F_{12}, but it still has not been selected in feature selection. In the 12 features, the packet transmission time of regularity is a very important feature called F_{11}, so we try to change the feature of conversion formula of constant t value, set this value in the range of 0.1 to 0.9, and with the other four features F_2, F_9, F_{10}, F_{11} to detect, the result as shown in Figure 4, the accuracy rate will change with constant of t-value, when t-value closed to 0.4, the accuracy will be the highest in Figure 4, and if the t-value approaching 0.1 and 0.9, the accuracy rate will decline along, this phenomenon indicates that the value of t in the formula can affect the quality of a feature.

Figure 4 Time regularity of accuracy with t value (see online version for colours)



5 Conclusions

The GA and PSO are easy to implement and it is very efficient in optimal solutions, so we adapt two methods to find the best solution, it combines BPN classification, and we evaluate our subset of feature on botnet detection that show high detection rate, the highest average accuracy rate is 95.09, and we also used a feature selection algorithm to validate our own manufactured feature F_{11} can be adopted, and the accuracy will change with the t-value.

We use a number of real bots to generate botnet traffic to evaluate the best solution, the main sample patterns is IRC botnet traffic, although we do not have a way to capture new botnet traces, but we find some important features in this experiment, and let us know about the operation of the botnet.

To sum up, we considered that it will become easier to implement and modify botnet, making the type of botnet change constantly and spread much fast.

Furthermore, many communication channels can change into a new botnet, as long as we understand its changes form and new operation mode. We believed that it will bring huge benefits for the botnet detection in the future.

References

Alomari, O. and Othman, Z.A. (2012) 'Bee algorithm for feature selection in network anomaly detection', *Journal of Applied Sciences Research*, March, Vol. 8, No. 3, p.1748.

Chen, J-H., Zhong, M., Chen, F-J. and Zhang, A-D. (2012) 'DDoS defense system with turing test and neural network', *IEEE International Conference on Granular Computing*, August, pp.38–43, Hangzhou, China.

Jain, Y.K. and Upendra (2012) ‘An efficient intrusion detection based on decision tree classifier using feature reduction’, *International Journal of Scientific and Research Publications*, January, Vol. 2, No. 1, pp.1–6.

Langin, C., Zhou, H., Rahimi, S., Gupta, B. and Zargham, M. (2009) ‘a self-organizing map and its modeling for discovering malignant network traffic’, *IEEE Symposium on Computational Intelligence in Cyber Security*, April, pp.122–129, Nashville, TN.

Liu, T., Qi, A.I., Hou, Y.B. and Chang, X.T. (2009) ‘Feature optimization based on artificial fish-swarm algorithm in intrusion detections’, *Networks Security, Wireless Communications and Trusted Computing*, April, pp.542–545, Wuhan, Hubei.

Mukkamala, S. and Sung, A.H. (2003) ‘Feature selection for intrusion detection using neural networks and support vector machines’, *Transportation Research Record: Journal of the Transportation Research Board*, January, Vol. 1822, pp.33–39.

Nogueira, A., Salvador, P. and Blessa, F. (2010) ‘a botnet detection system based on neural networks’, *International Conference on Digital Telecommunications*, June, pp.57–62, Athens, TBD, Greece.

Oracle VM Virtubox (2012) [online] http://www.virtualbox.org/ (accessed 31 August 2016).

Salvador, P., Nogueira, A., Franca, U. and Valadas, R. (2009) ‘Framework for zombie detection using neural networks’, *International Conference on Internet Monitoring and Protection*, May, pp.14–20, Venice.

Venkatesh, G.K. and Nadarajan, R.A. (2012) ‘HTTP botnet detection using adaptive learning rate multilayer feed-forward neural network’, *International Conference on Information Security Theory and Practice*, pp.38–48, Berlin.