OPTIMAL FEATURE SELECTION FOR FIREWALL LOG CLASSIFICATION USING RANDOM FOREST AND HYBRID METAHEURISTIC ALGORITHMS

Seungwoo Han[1], **Gil Hong**[2], **Jewan Kim**[2*], **Jeuk Yu**[3], **Sangjun Lee**[2**], **Byeongok Cho**[2***], **Jusung Jeon**[4]
[1]Dept. of Intelligent Information System and Embedded Software Engineering, Kwangwoon University, Seoul, Korea,
[2]Dept. of Machatronics, Sahmyook University, Seoul, Korea,
[3]Dept. of Computer, Sahmyook University, Seoul, Korea, [4]Dept. of Car Machatronics, Sahmyook Univ.
[1]swoohan@kakao.com,
[2]wideroad95@nate.com, [2*]wpdhks6@gmail.com, [3]jipsina71@gmail.com,
[2**]tjdtlf1234@naver.com, [2***]winjbo1021@gmail.com, [4]gggg1371@naver.com,

May 6, 2021

ABSTRACT

Firewall log classification is important to monitor network traffic. In this paper, we applied a method of feature selection using bee swarm optimization with reinforcement learning to classify logs using optimal features. The average performance was obtained by accuracy, macro-averaged precision, macro-averaged recall, and macro-averaged F1 score in 10-stratified folds using a random forest classifier. As a result, 4 optimal features were selected and each performance was measured as 99.83%, 91.71%, 82.96%, 85.22%, respectively. The results demonstrate optimal feature combination outperforms all feature combination case.

1 Introduction

Firewalls of computers are used to ensure that the network is functioning properly and safely. In particular, as the security of personal information [1] becomes more important and robust, it need to be conscious to protect networks. Firewall logs are key evidence to identify intruder attacks, including insider and outsider threats [2]. In addition to the existing traditional classification methods, with the development of machine learning and deep learning, a study on log classification and intrusion prevention using this has been conducted [3, 4, 6, 7, 8]. Log analysis and intrusion detection defense should try to classify efficiently with fewer parameters for quick response. At this time, the number of features to use for classification or regression is based on the researcher's experience. For this reason, research is also being conducted to intensively select the optimal parameters and parameters required for machine learning. It has been mainly designed as information gain, and genetic algorithm [9]. In addition, studies have been conducted to select the optimal parameter using reinforcement learning [10], and bayesian optimization [11, 12]. In this paper, we performed optimal feature search using the bee swarm optimization algorithm (BSO) along with reinforcement learning [13] for firewall log classification.

2 Experiment

2.1 Data acquisition

In this paper, we used Internet Firewall Data Data Set [4] in UCI Machine Learning Repository [5]. The data contains 11 features and 4 labels. Total data points are 65532. Data profile is shown in Table 1. 11 feature is bytes, bytes received, bytes sent, destination port, elapsed time, NAT destination port, NAT source port, packets, packets received, packets sent, and source port, and 4 label is allow, deny, drop, reset-both, respectively.

Table 1: Data profile

Name	Data points per class
allow	37640
deny	14987
drop	12851
reset-both	54

2.2 Method

For feature selection, we used the fusion method of BSO and reinforcement learning [13]. First, the bee swarm optimization is an algorithm that is inspired by the social behavior of bees. Each bee is an object working together to solve the optimization problem, and they search for fitness function using a feature combination in iterations. Fitness function of this experiment is set to average accuracy. Second, reinforcement learning refers to an algorithm in which an agent defined in the environment recognizes the current state and finds an action that maximizes the reward among actions. The reinforcement learning algorithm applied in this paper is Q-learning [14].

Local search and experience of bee replace Q-learning algorithm. In this process, the reward is given differently depending on the accuracy of the current and next states. If the next state accuracy is higher than the current state, the reward of set to next state accuracy value, and if the current state accuracy is high, the reward set to (next state accuracy - current state accuracy) value. Additionally, if the number of features in the current state is greater than the number of features in the next state, the reward is set to (1/2 * next state accuracy). In the opposite case, the reward is set to (-1/2 * next state accuracy). As a result, the agent tries to get the best accuracy while getting fewer features. Moreover, to reduce the space in the search space, we applied the XOR operation on the best solution and the current state solution.

2.3 Hyper-parameter setting

Table 2 shows the hyper-parameters applied to this experiment. The parameters were chosen empirically.

Table 2: Parameters value of experiments

Name	Parameter	Summary
Filp	6	To calculate SearchRegion space (SearchRegion = length of all features / filp)
Max Chance	2	Number of chances to escape local minima
Number of Bee	5	Worker to solve
Max iteration	5	Number of total iteration
Local iteration	1	Number of iteration in local search
α	0.7	Learning rate of reinforcement learning
γ	0.3	Discount factor of reinforcement learning
ϵ	0.05	Probability of doing random actions
Max Depth	5	Maximum depth of tree
n-estimators	10	The number of trees in the forest

To get the average accuracy, this experiment applied 10-stratified folds. And we use the random forest classifier algorithm to calculate the accuracy.

3 Results

The optimal combination of features selected by the feature selection method is **destination port, packets, elapsed time, packets received**. Figure 1 shows the frequency of selected features during the total iteration. Performance was evaluated by average accuracy and average macro-averaged precision (Macro-precision), average macro-averaged recall (Macro-recall), average macro-averaged F1 (Macro-F1) score. We compared the optimal selected feature results with the case of applying all features. The results are shown in Table 3.

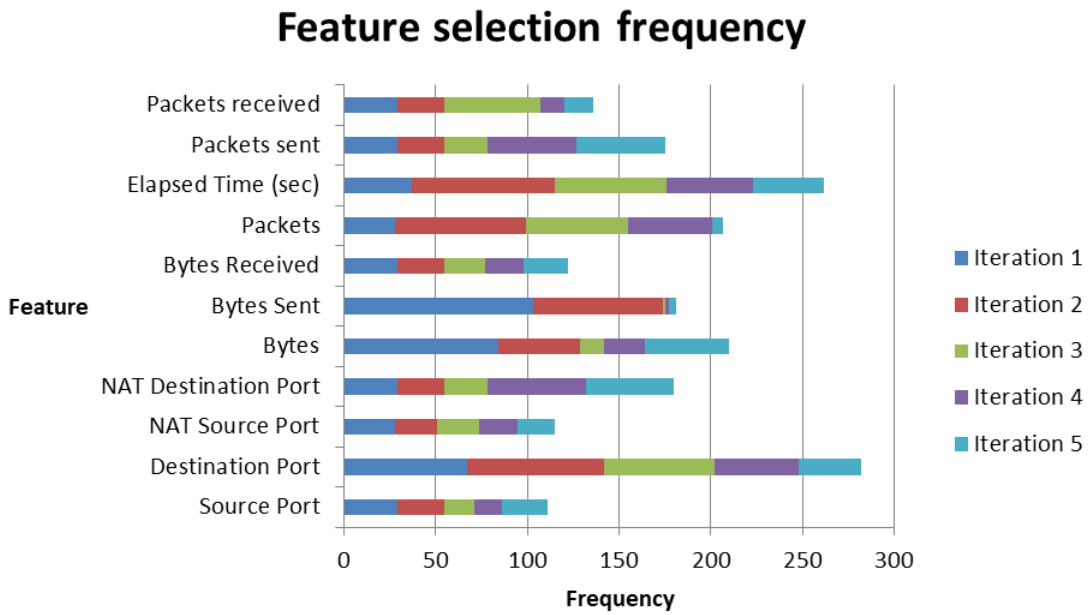

Figure 1: Feature selection frequency result using BSO with reinforcement learning

Table 3: Performance evaluation result

Feature selected	Average accuracy	Average Macro-precision	Average Macro-recall	Average Macro-F1 score
All feature	99.80%	84.79%	79.61%	81.26%
Optimal features	**99.83%**	**91.71%**	**82.96%**	**85.22%**

4 Discussion and Conclusion

In this paper, we classified firewall logs using optimal feature via BSO with reinforcement learning feature selection method. The results of using optimal features outperformed using all features and it could be applied to a firewall log analysis that can perform log classification using only a few features. In future plans, we will consider the hyper-parameter selection of algorithms for good results.

References

[1] J. A. Castañeda and F. J. Montoro, "The effect of Internet general privacy concern on customer behavior", In *Electronic Commerce Research*, vol. 7, no. 2, pp. 117–141, 2007.

[2] N. K. Singh, D. S. Tomar, and B. N. Roy "An approach to understand the end user behavior through log analysis", In *International Journal of Computer Applications*, vol. 5, no. 11, pp. 27–34, 2010.

[3] S. Allagi and R. Rachh, "Analysis of Network log data using Machine Learning", In *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*, pp. 1–3, 2019.

[4] F. Ertam and M. Kaya, "Classification of firewall log files with multiclass support vector machine", In *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, pp. 1–4, 2018.

[5] A. Frank, and A. Asuncion, "UCI Machine Learning Repository http://archive.ics.uci.edu/ml", Irvine, CA: University of California, School of Information and Computer Science. 2010.

[6] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection", In *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[7] D. C. Le, N. Zincir-Heywood, and M. I. Heywood, "Analyzing Data Granularity Levels for Insider Threat Detection Using Machine Learning", In *IEEE Transactions on Network and Service Management*, vol. 17, no. 1, pp. 30–44, Mar. 2020.

[8] A. Javaid, Q. Niyaz, W. Sun, and M. Alam "A deep learning approach for network intrusion detection system", In *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 21–26, 2016.

[9] S. Lei, "A Feature Selection Method Based on Information Gain and Genetic Algorithm", In *2012 International Conference on Computer Science and Electronics Engineering*, pp. 355–358, 2012.

[10] J. Janisch, T. Pevný, and V. Lisý, "Classification with Costly Features Using Deep Reinforcement Learning", In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3959–3966, 2019.

[11] S. Falkner, A. Klein, and F. Hutter, "BOHB: Robust and efficient hyperparameter optimization at scale", In *In ICML*, pp. 1437–1446, 2018.

[12] S. Han, H. Eom, J. Kim, C. Park, "Optimal DNN architecture search using Bayesian Optimization Hyperband for arrhythmia detection", In *2020 IEEE Wireless Power Transfer Conference (WPTC)*, pp. 357-360, 2020.

[13] S. Sadeg, L. Hamdad, A. R. Remache, M. N. Karech, K. Benatchba, and Z. Habbas, "QBSO-FS: A Reinforcement Learning Based Bee Swarm Optimization Metaheuristic for Feature Selection", In *International Work-Conference on Artificial Neural Networks*, vol. 33, pp. 785-796, 2019.

[14] C. J. Watkins and P. Dayan, "Q-learning", *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.