OPTIMAL FEATURE SELECTION RESEARCH FOR FIREWALL LOG ANALYSIS USING BEE SWARM OPTIMIZATION WITH REINFORCEMENT LEARNING

Seungwoo Han1, Gil Hong2, Jewan Kim2∗, Jeuk Yu1, Sangjun Lee2**, Byeongok Cho2***, Jusung Jeon4
1Dept. of Intelligent Information System and Embedded Software Engineering, Kwangwoon University, Seoul, Korea,
2Dept. of Machatronics, Sahmyook University, Seoul, Korea,
3Dept. of Computer, Sahmyook University, Seoul, Korea, 4Dept. of Car Machatronics, Sahmyook Univ.
swoohan@outlook.kr, wideroad95@nate.com, wpdhks6@gmail.com, jipsina71@gmail.com,
tjdtlf1234@naver.com, winjbo1021@gmail.com, ggg1371@naver.com,

February 22, 2021

ABSTRACT

Firewall log analysis is important to monitor network traffic and determine whether to block or allow specific traffic. In this paper, we applied a method of feature selection using bee swarm optimization with reinforcement learning to classify logs using optimal features for firewall log analysis. 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, and it could be applied to a firewall log analysis using few features.

1 Introduction

A firewall is a major component of a network. In particular, as the security of personal information [1] becomes more important, firewalls must operate their tasks well on computers. Meanwhile, the firewall log is an important clue for identified intruders in case of intrusion to determine how to break in. With the development of machine learning, research was progressed to classify logs using machine learning and prevent intrusion [2, 3, 5, 6]. The log analysis and intrusion detection defenses should attempt classification with relatively few parameter for efficiency. Number of features to be used for classification or regression are based on the experience of researchers and developers. For that reason, research also has been conducted to select the optimal features and parameter required for machine learning intensively. It has been mainly designed as information gain, and genetic algorithm [7]. In addition, studies have been conducted to select the optimal feature using reinforcement learning [8], and bayesian optimization [9, 10]. In this paper, we propose the optimal feature selection using bee swarm optimization algorithm (BSO) with reinforcement learning [11] for firewall log classification.

2 Experiment

2.1 Data acquisition

In this paper, we used Internet Firewall Data Data Set [3] in UCI Machine Learning Repository [4]. 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, respectively.
2.2 Method

For feature selection, we used the fusion method of BSO and reinforcement learning [11]. 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 [12].

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>
<thead>
<tr>
<th>Name</th>
<th>Parameter</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flip</td>
<td>6</td>
<td>To calculate SearchRegion space (SearchRegion = length of all features / filp)</td>
</tr>
<tr>
<td>Max Chance</td>
<td>2</td>
<td>Number of chances to escape local minima</td>
</tr>
<tr>
<td>Number of Bee</td>
<td>5</td>
<td>Worker to solve</td>
</tr>
<tr>
<td>Max iteration</td>
<td>5</td>
<td>Number of total iteration</td>
</tr>
<tr>
<td>Local iteration</td>
<td>1</td>
<td>Number of iteration in local search</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>0.7</td>
<td>Learning rate of reinforcement learning</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>0.3</td>
<td>Discount factor of reinforcement learning</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>0.05</td>
<td>Probability of doing random actions</td>
</tr>
<tr>
<td>Max Depth</td>
<td>5</td>
<td>Maximum depth of tree</td>
</tr>
<tr>
<td>n-estimators</td>
<td>10</td>
<td>The number of trees in the forest</td>
</tr>
</tbody>
</table>

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.
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


