SCIS 2016 2016 Symposium on
Cryptography and Information Security
Kumamoto, Japan, Jan. 19 - 22, 2016
The Institute of Electronics,
Information and Communication Engineers

Evaluation of ACA-based Intrusion Detection Systems for Unknown-attacks

Kyung-min Kim * Jina Hong † Kwangjo Kim ‡ Paul D. Yoo §

Abstract: Intrusion Detection System (IDS) monitors a network and detects users' malicious activities. Since new unknown-attacks are appearing continuously, IDS must have capability of detecting attacks without any specific prior knowledge. Also many devices are connected on network and produce enormous large volumes of network data. Labeling enormous network data manually is impractical task. Therefore, we should find a way to learn normal traffic and attack traffic by itself on the unlabeled dataset. In this paper, we propose two IDS for unknown-attacks based on Ant Clustering Algorithm (ACA). Our IDS can learn on the unlabeled dataset and detect unknown-attacks. Our proposed IDS are combination of ACA and other supervised learning algorithm. We combined Decision Tree and Artificial Neural Network with ACA separately and compared performance between them.

Keywords: Bio-inspired IDS, Swarm Intelligence, Ant-based IDS, Unknown-attack Detection

1 Introduction

Intrusion Detection System (IDS) monitors a network and detects users' malicious activities [1]. Two main detection types exist in IDS: Signature-based IDS and Anomaly-based IDS. Signature-based IDS has specific signatures of known-attacks and performs pattern matching between input traffic and signatures to detect intrusion. Anomaly-based IDS builds a profile of normal behavior and concentrates to detect violation of the profile as an intrusion. By the location of IDS sensor, IDS can be divided as a Network IDS (NIDS) and a Host IDS (HIDS). A NIDS is installed over a target network. A HIDS is installed on a certain host or device.

As Internet of Things services are coming, many devices are connected on the open network. They produce enormous volumes of network data. Labeling each data whether normal or attack manually is difficult and impractical task. Because of this difficulty, we don't have enough labeled data available which reflect state-of-the-art network environment. Also, we can get the labeled dataset only by simulating intrusions. This method has a limitation that we can know just cases of known-attacks. Unknown-attacks are appearing continuously and harmful to us. We should detects unknown-attacks but we cannot get a labeled dataset which contains labels of indicating unknown-attacks.

To solve these problems, we must find some methods to learn normal traffic and attack traffic by itself on the unlabeled dataset. Since the supervised learning model needs to know labels beforehands, the method should have the unsupervised model. The unsupervised model doesn't need to know labels and can learn by itself. In this paper, we propose new two IDS which based on the unsupervised learning algorithm. We combine ACA and other supervised learning algorithms: Decision Tree and Artificial Neural Network. As combining two machine learning algorithms, the proposed IDS doesn't need any prior knowledge about attacks and can detect unknown-attacks.

The rest of this paper is organized as follows: Section 2 describes background knowledge about detection types of IDS, unknown-attack detection, dataset description for evaluation, clustering algorithms, and some supervised learning algorithms. The approach of the proposed IDS is described in Section 3. Description about architecture of the proposed IDS is in Section 4. Evaluation is described in Section 5. Finally, the conclusion and future work are discussed in Section 6.

* School of Computing, KAIST, Daejeon, Republic of Korea, (saza12345@kaist.ac.kr)
† Graduate School of Information Security, KAIST, Daejeon, Republic of Korea, (jina3453@kaist.ac.kr)
‡ School of Computing, KAIST, Daejeon, Republic of Korea (kkj@kaist.ac.kr)
§ Department of Computing and Informatics, Bournmouth University, Poole, United Kingdom (paul.d.yoo@ieee.org)

2 Background

2.1 Detection Types of IDS

2.1.1 Signature-based IDS

Signature-based IDS has some signatures of known-attacks and makes rules for detection of known-attacks. When a traffic from the network comes into IDS, signature-based IDS extracts some signatures from the traffic. After extraction, signature-based IDS compares extracted signatures with the signatures which IDS already has. If some signatures are matched, signature-based IDS decides the traffic as an attack. Signature-based IDS only can detect attacks which has signatures. Therefore, signature-based IDS cannot detect any unknown-attack because signatures of unknown-attacks do not exist in signature-based IDS. Signature-based IDS decides an unknown-attack as a normal traffic.

2.1.2 Anomaly-based IDS

Anomaly-based IDS doesn't maintain list of signatures of known-attacks. Anomaly-based IDS concentrates on violations of normal behavior. Anomaly-based IDS builds a profile of normal behavior based on the dataset and detects violations of the profile as attacks. Irrespective of known or unknown to IDS, any activity which causes abnormal state of a network can be detected by anomaly-based IDS. Therefore, anomaly-based IDS can detect unknown-attacks.

2.2 Unknown-attack Detection

Unknown-attack detection is a research about detecting attacks without any prior knowledge of attacks. Signature-base IDS cannot be used for unknown-attack detection since it can detect attacks which has prior knowledge only. Unlike signature-based IDS, anomaly-based IDS can detect unknown-attacks. Therefore anomaly-based IDS is mainly used for unknown-attack detection.

Building robust profile of normal behavior is the most significant factor for anomaly-based IDS. It directly affects to performance of anomaly-based IDS. Many researches on anomaly-based IDS utilize machine learning and data mining algorithms to robust modeling of normal behavior profile. For example, Support Vector Machine, Decision Tree, and Artificial Neural Network in supervised learning algorithms and k-Means clustering, Density-Based Spatial Clustering of Application with Noise (DBSCAN), and Ant Clustering Algorithm (ACA) in unsupervised learning algorithms are used for anomaly-based IDS.

2.3 KDD Cup 1999 Dataset

KDD Cup 1999 Dataset is widely used dataset for evaluation of IDS performance. The dataset is a version of dataset of DARPA 1998 Intrusion Detection Evaluation Program. The dataset is used in the 1999 KDD intrusion detection contest [2]. The dataset contains five types of traffic: Normal, DoS, U2R, R2L, Probe. The type of dataset are described as below:

- Normal : not attack
- DoS : denial-of-service (*e.g.*, syn flood attack)
- U2R : unauthorized access to local superuser privileges (*e.g.*, buffer overflow attack)
- R2L : unauthorized access from a remote machine (*e.g.*, guessing password)
- Probe : surveillance and other probing (*e.g.*, port scanning)

The dataset has 4,898,431 traffic data instances. Each data instances has 41 features and label for traffic type. Table 1 shows the traffic distribution of KDD Cup 1999 Dataset.

Table 1: Traffic distribution of KDD Cup 1999 Dataset

Type	# of traffics	Proportion (%)
Normal	972,781	19.86
DoS	3,883,370	79.28
U2R	52	0.00
R2L	1,126	0.02
Probe	41,102	0.84
Total	4,898,431	100

2.4 Clustering Algorithms

Clustering is one area of the unsupervised machine learning algorithms. Since the unsupervised machine learning algorithm doesn't need to know labels of data instances, we can use clustering algorithms to learn on the unlabeled dataset. Clustering algorithm make clusters which has similar features on the unlabeled dataset. Among many clustering algorithms, some are used in anomaly-based IDS.

(1) k-Means clustering algorithm partitions the dataset into k clusters. Each instance of the dataset is assigned to exactly one cluster. To assign a cluster of a data instance, the Euclidean distance is mainly used [3]. k-Means clustering is very sensitive to initial state of the cluster centroids.

(2) DBSCAN finds clusters based on the estimated density distribution of the dataset [3]. Unlike k-Means clustering, DBSCAN is insensitive to initial state. But DBSCAN is sensitive to data density and data dimension [4].

(3) ACA is a heuristic algorithm and one of the swarm intelligence algorithms. The swarm intelligence algorithms belong to one area of the bio-inspired algorithms. ACA is inspired by the brood sorting activities of ants. ACA was modeled by Deneubourg *et al.* [5]. Their model is referred to as the basic model [6].

In ACA, each data instances of the dataset is randomly scattered in a 2D space. Each ant moves randomly in the 2D space and picks up or drops down data instances based on probability. Equations for calculating probability of picking up and dropping down are presented in below:

$$P_{pick} = (\frac{k_1}{k_1 + f})^2 \quad (1)$$

where f is the perceived fraction of items in the neighborhood of the ant and k_1 is a threshold item.

$$P_{drop} = (\frac{f}{k_2 + f})^2 \quad (2)$$

where k_2 is another threshold constant.

ACA has self-organizing characteristic. The characteristic named as self-organizing means that accomplishing overall process by coordination out of the local interactions between smaller components. Therefore, ACA can makes clusters on the initially disordered dataset by itself. ACA doesn't require to predefine the number of clusters because of self-organizing characteristic. Also, ACA is insensitive to initial state, data density, and data dimension.

2.5 Supervised Learning Algorithms

2.5.1 Decision Tree

Decision Tree (DT) is a predictive model which using tree-like graph. DT can be trained as a rule-based structure. DT makes a tree based on the training set. DT extracts some rules to classify the training set correctly. Each branch of DT represents a decision rule. Each leaf node represents a set of data instances has same class. DT makes decision rules until every data instances are classified correctly. The goal of DT is to create a model that predicts the value of a target output based on several inputs. Two main types are exist in DT: classification tree and regression tree. When the predicted output is a finite set of classes, classification tree is used. When the predicted output is continuous value, regression tree is used.

DT has some advantages than other supervised machine learning algorithms. First, DT can allow the addition of new possible scenarios. It is appropriate advantage for unknown-attack detection. Second, DT is a white-box model algorithm. Therefore, unlike black-box model, we can analyze why the algorithm predicts a certain output. When using DT as detection algorithm, a security expert can analyze a certain attack and extract some rules after detection of the attack to detect the attack next time.

2.5.2 Artificial Neural Network

Artificial Neural Network (ANN) is a computational model inspired by the natural neurons. Natural neurons receive signals through synapses located on the dendrites or membrane of the neuron. When the signals received are strong enough, the neuron is activated and emits a signal though the axon. This signal might be sent to another synapse, and might activate other neurons. The complexity of real neurons is highly abstracted when modelling artificial neurons. These basically consist of inputs, which are multiplied by weights, and then computed by a mathematical function which determines the activation of the neuron. Another function computes the output of the artificial neuron. ANN combines artificial neurons in order to process information [7].

ANN is composed of several layers: an input layer, hidden layers, and an output layer. Each layer has artificial neurons and adjacent layers are connected to each other. The network sends their signals to next layer and the errors from the output are propagated backwards. Base on the given dataset, ANN modify weights between neurons until output errors of the network is converged. After the convergence of the network, ANN can predict a class of a certain input.

ANN has some advantages than other supervised machine learning algorithms. First, ANN can learn non-linear structured function. More layers in the network improve capability to learn more complex function. Second, if the cost function and learning algorithm are selected appropriately, the network has extreme robustness. The robustness of the network can improve the performance of anomaly-based IDS.

3 Approach

3.1 Assumption

KDD Cup 1999 Dataset has a drawback in distribution of traffic. Over 80% of data instances is attack traffic in the dataset. This proportion doesn't reflect general network traffic environment. In general network, normal traffic overwhelms attack traffic. Therefore, we

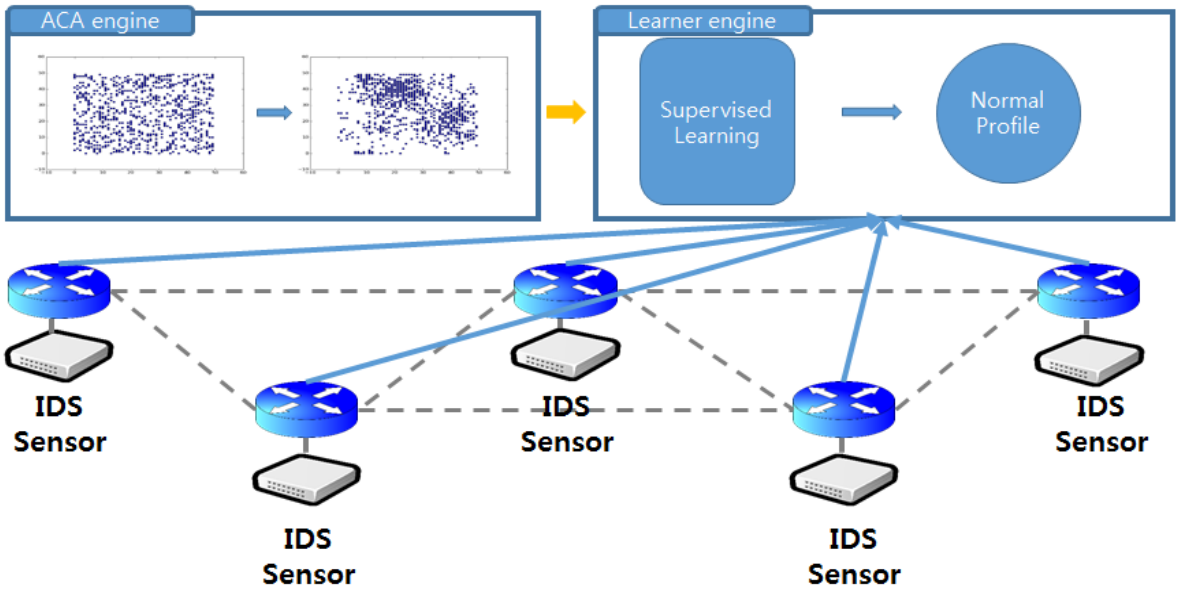

Figure 1: Architecture of our proposed IDS

need to reflect general network traffic environment into our experiments.

The proposed IDS is based on a major assumption to reflect general network environment. The assumption is that there is overwhelmingly much normal traffic than attack traffic in target network. Based on the assumption, we filtered and construct the dataset as 98% of normal traffic and 2% of attack traffic [8].

3.2 Clustering

To make clusters from the unlabeled dataset, we used ACA. Due to self-organizing characteristic, ACA can makes clusters by itself. Initially, we assigns a cluster to each data instances. In main loop of ACA, when an ant drops down a data instance, the ant looks around and assigns cluster of the data instance as majority cluster around his neighbor. After certain iterations, some clusters are remained with more members than initial state and other clusters are disappeared. Based on the clustering result, we can make labels to each data instances.

3.3 Cluster Labeling

Since we are considering to learn on the unlabeled data, we don't have access to labels during training. Therefore, we need to find some other way to decide which clusters contain normal instances and which contain attack instances. Under our assumption about normal traffic constituting an overwhelmingly large portion, over 98%, it is highly probable that clusters containing normal data will have a much larger number of instances associated with them then would clusters containing attacks. Therefore, we label some percentage of the clusters containing the largest number of instances associated with them as 'normal'. The rest of the clusters are labeled as 'attack'[8].

3.4 Detection

After all labels of data instances made by ACA, we can use the dataset as a labeled dataset with the labels by ACA. Therefore, we can train intrusion detector using supervised learning method based on the dataset and labels made by ACA to detect unknown-attacks. Among many supervised machine learning algorithms, we choose decision tree and artificial neural network algorithms to train intrusion detector. The trained detector monitors a network and can detect unknown-attacks.

4 Our proposed IDS

Our proposed IDS combine a clustering algorithm and a supervised machine learning algorithm to detect unknown-attacks. We use ACA as a clustering algorithm. We makes clusters and builds a profile of normal behavior by using a supervised learning algorithm. As a supervised learning algorithm, we propose two algorithms: DT and ANN. Because we choose clustering model, the proposed IDS can build a profile of normal behavior on the unlabeled dataset. Figure 1 illustrates the architecture of the proposed IDS. The proposed IDS is composed of two main engines: the ACA engine and the Learner engine.

4.1 ACA engine

The ACA engine makes clusters based on the unlabeled dataset. Due to self-organizing characteristic, the ACA engine can builds clusters by itself. We don't need to consider about the number of clusters. After making clusters, the ACA engine sorts clusters by the number of cluster members. And the ACA engine makes labels to certain percentage of large clusters as 'normal'. Other clusters and outliers are labeled as 'attack' by the ACA engine. The ACA engine passes the dataset and labels made by the engine to DT engine. Figure 2 illustrates the process of the ACA engine.

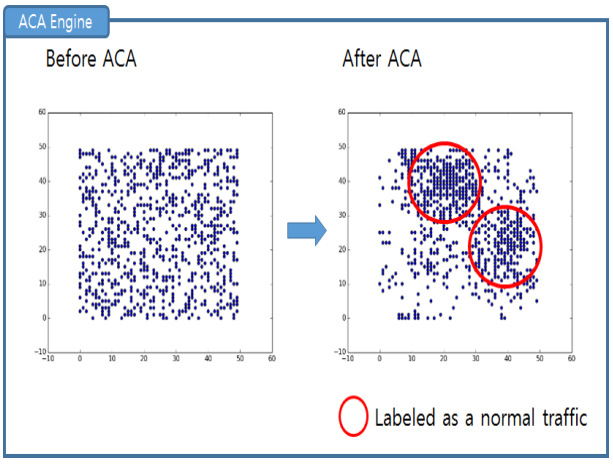

Figure 2: Process of the ACA engine

4.2 Learner engine

The Learner engine receives the dataset with labels by the ACA engine. In the Learner engine's view, the engine has a labeled dataset. Therefore, the Learner engine can train an intrusion detector by using supervised learning method such as DT or ANN. If we use DT as a supervised learning algorithm of the Learner engine, we can analyze certain unknown-attack by security experts to extract some signatures of certain unknown-attack because DT is a white-box model. If we use ANN as a supervised learning algorithm of the Learner engine, we can learn more complex structure of certain unknown-attacks due to a characteristic of ANN. The trained intrusion detector monitors a network and can detect unknown-attacks.

5 Evaluation

5.1 Dataset Description

Under our major assumption about the proportion of normal traffic overwhelms attack traffic, we construct the training set and the test set as 98% of normal traffic and 2% of attack traffic. To make 2% of attack traffic, we filtered attack traffic of KDD Cup 1999 Dataset. When we filtered attack traffic, we tried to include all of attack types to prevent biased training result. Also we used 10% version of KDD Cup 1999 Dataset in the experiment. We extract the training set and the test set in the different area of 10% version of KDD Cup 1999 Dataset. Therefore, we can say that the proposed IDS doesn't know any information about the test set. Tables 2 and 3 show the distribution of the training set and the distribution of the test set in the experiment, respectively. After filtering, we don't use labels of the dataset because we choose the clustering model. We just use features of each data instances to build a profile of normal behavior.

Table 2: Traffic distribution of the training set

Type	# of traffic	Proportion (%)
Normal	78,010	98.00
DoS	761	0.96
U2R	35	0.04
R2L	398	0.50
Probe	398	0.50
Total	79,602	100

Table 3: Traffic distribution of the test set

Type	# of traffic	Proportion (%)
Normal	19,268	98.00
DoS	277	1.41
U2R	17	0.09
R2L	1	0.00
Probe	98	0.50
Total	19,661	100

5.2 Evaluation Criteria

Generally, datasets and real network traffic data are used to evaluate performance of IDS. Researchers can train their IDS based on the training dataset and test by using the test dataset. They compare prediction results of detection algorithm and analyze the results of actual value of test dataset. Four categories are exist to evaluate detection result: True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN). In IDS case, TP is a case that a malicious traffic referred to as an attack by IDS. TN is a case that a normal traffic referred to as a normal by IDS. FP is a case that a normal traffic referred to as an attack and FN is a case that a malicious traffic referred to as a normal traffic. Based on these four categories, three important criteria for evaluation performance of IDS can be calculated. First one is Detection Rate (DR) which is defined as the number of intrusion in-

stances detected by IDS, same as TP, divided by the total number of intrusion instances in the real traffic. DR is a criteria which indicates how well IDS detects attacks. High DR means that IDS can detect attacks more than IDS which has low DR. Second one is False Positive Rate (FPR) which is defined as the number of normal instances classified as attack, same as FP, divided by the total number of normal instances in the real traffic. High FPR means that IDS can misclassify a normal traffic as an attack more frequently than IDS which has low FPR. Final one is Accuracy (ACC) which is defined as the number of corrected classified instances by IDS, same as sum of TP and FN, divided by the total number of instances in the real traffic. ACC is related to how well IDS classify normal and attacks correctly. Equations for calculation DR, FPR, and ACC are presented in below:

$$DR = \frac{TP}{TP + FN} \tag{3}$$

$$FPR = \frac{FP}{FP + TN} \tag{4}$$

$$ACC = \frac{TP + FN}{TP + FP + TN + FN} \tag{5}$$

Also, Hosseinpour *et al.*[3] proposed similar approach with our approach to detect unknown-attacks. They proposed two combinations of unsupervised machine learning algorithm and supervised machine learning algorithm. One combination is k-Means clustering and Artificial Immune System (AIS). The other is a combination of DBSCAN and AIS. Since their approach is similar with us, we compared performance of our proposed IDS and their IDS.

5.3 Experimental Result

5.3.1 Clustering Result

Since ACA needs more parameters than k-Means clustering and DBSCAN, we did many experiments in diverse parameter setting. Among them, the best parameter setting case is 1000 ants, 600 × 600 size of 2D grid, 500,000 iterations, 3 × 3 of local area of an ant, and 15 of constant for calculating probability. In this parameters, the ACA engine made 795 clusters based on our training set. Figures 3 and 4 illustrate the initial state and the final state of the 2D grid, respectively.

As shown in Figure 4, many small clusters are exist in the clustering result. Many small clusters mean that ACA did clustering task based on strict criteria for similarity between data instances. It can help us to build more precise profile of normal behavior than when several big clusters exist in the result.

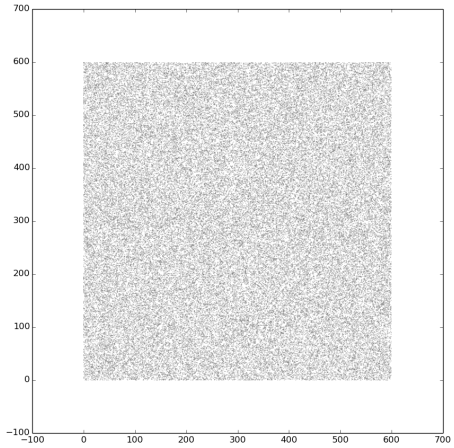

Figure 3: Initial state of the 2D grid

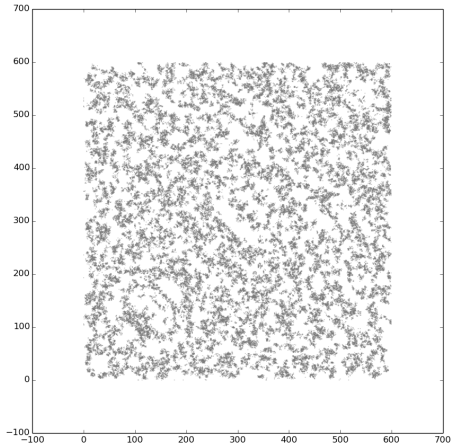

Figure 4: Final state of the 2D grid

5.3.2 Performance of the proposed IDS

As Hosseinpour *et al.*[3] proposed similar approach with our approach, we compared the performance of our IDS with Hosseinpour *et al.*[3]. The compared criteria are DR, FPR, and ACC. Table 4 shows comparison of performance between Hosseinpour *et al.*[3] and our IDS.

From Table 4, both of our IDS have much higher DR than Hosseinpour *et al.*[3]. Also, both of our IDS have much higher ACC and low FPR at the same time. This means that the proposed IDS builds more robust and accurate profile of normal behavior than Hosseinpour *et*

Table 4: Comparison of performance between Hosseinpour *et al.*[3] and the proposed IDS

	[3]	[3]	Proposed IDS 1	Proposed IDS 2
Algorithm	k-Means + AIS	DBSCAN + AIS	ACA + DT	ACA + ANN
DR(%)	43.1	58.9	97.4	93.4
FPR(%)	15.6	0.8	9.7	0.2
ACC(%)	60.7	77.1	90.4	99.6

al.[3]. Also it means that ACA performed better clustering result than k-Means clustering and DBSCAN.

There are small differences in performance of IDS between two proposed IDS. The combination of ACA and DT case has higher DR than the combination of ACA and ANN case. At the same time, the ACA and DT case has lower ACC and higher FPR than the ACA and DT case. We think these performance differences came from characteristics of each supervised learning algorithms. DT tries to find rules which can correctly classify given data instances until all of the data instances are classified correctly. Due to this characteristic, DT can classify attack traffic well. But DT caused more false positive cases than ANN because DT build simple structure of normal behavior profile by rules. If a normal traffic can classify as a normal only condition on complex profile, DT may misclassify the traffic. ANN can learn complex structured function and nonlinear function. Due to this characteristic, ANN can build more complex profile of normal behavior than DT. Therefore, ANN can classify correctly. But ANN also has robustness of the network. Because of robustness, ANN is hard to allow new scenario and it may cause false negative cases. We think these differences caused performance difference between two proposed IDS.

5.4 Discussion

In this section, we discuss some considerable points. Although the proposed has much higher DR and ACC, FPR still higher than Hosseinpour *et al.*[3] in the combination of ACA and DT case. This means that the ACA engine assigned same cluster to normal traffic and attack traffic sometimes. The reason for this issue comes from the fact that each ant in the ACA engine look around within their local area only, not global area. Because ACA is a heuristic algorithm and has self-organizing characteristic, overall task is done by sum of smaller tasks. Therefore ACA has a local optima problem. Making larger local area of each ant will be a possible solution to mitigate this problem.

Our experimental result is based on the 10% version of KDD Cup 1999 Dataset. It can be thought that the experimental result is biased to the 10% version of KDD Cup 1999 Dataset. More experimental cases on various datasets are needed to generalize the proposed IDS. But the performance of our proposed IDS will decrease as doing experiments in various cases because if the proportion of attack traffic increase, more attack traffic will be blended into normal traffic clusters. We expect that the performance of our proposed IDS will be degraded as the proportion of attack traffic increase.

Also, KDD Cup 1999 Dataset was announced at 1998. As many things were changed on network environment since 1998, KDD Cup 1999 Dataset no longer contains attack traffic enough. Therefore, we need to perform experiments on the latest dataset to reflect contemporary network environment. Kyoto 2006+ Dataset[9] can be a considerable candidate for solution of this problem. Kyoto 2006+ Dataset built on the 3 years of real traffic data from November 2006 to August 2009. It consist of total 24 features.

6 Conclusion and Future work

This paper proposes two new IDS architectures that can detect unknown-attacks by combining ACA and two supervised learning algorithms: DT and ANN. The proposed IDS can learn on the unlabeled dataset in unsupervised manner. The capability of learning on the unlabeled dataset is appropriate to enormous amount of network traffic environment such as internet of things. The capability can let us be free from labeling attack or not manually.

We assumed that normal traffic overwhelms attacks traffic to reflect general network environment. Therefore, we make the training dataset and the test dataset be composed of 98% of normal traffic and 2% of attack traffic, respectively.

Two proposed IDS's combine ACA and two supervised learning algorithms to detect unknown-attacks. The proposed IDS is composed of two main engine:

the ACA engine and the Learner engine. The overall flow of the proposed IDS is as follows: firstly, the ACA engine builds clusters on the unlabeled training dataset by self-organizing characteristic. After clustering phase, the ACA engine secondly attaches labels certain percentage of large clusters as 'normal' and the others as 'attack'. Thirdly, the ACA engine passes the training dataset with labels made by the ACA engine to the Learner engine. Fourthly, the Learner engine trains intrusion detector based on the received dataset with labels by supervised manner. Finally, the trained intrusion detector monitors a network and can detect unknown-attacks without any specific knowledge.

Under our assumption, two proposed IDS has much better performance than Hosseinpour *et al.* [3] which has a similar approach with ours. When combining ACA and DT, DR, FPR, and ACC are verified to have 97.4%, 9.7%, and 90.4%, respectively. On the other hand, when combining ACA and ANN, DR, FPR, and ACC are verified to have 93.4%, 0.2%, and 99.6%, respectively. It means that our proposed IDS can build more robust and accurate profile of normal behavior than Hosseinpour *et al.* [3].

However, future research still remains. Diverse experiments on various datasets are needed. We just performed our experiments on one dataset. To verify our IDS in detail, we need to perform more diverse experiments on various datasets. Similarly, the latest dataset is needed since KDD Cup 1999 Dataset is too outdated. KDD Cup 1999 Dataset doesn't include contemporary traffic types of a network. Therefore, we need to perform experiments on the other dataset which contains contemporary traffic types.

Acknowledgement

This work was partly supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (B0101-15-1270, Research on Communication Technology using Bio-Inspired Algorithm) and the KUSTAR-KAIST institute, KAIST, Korea.

References

[1] Karen Scarfone and Peter Mell, "Guide to intrusion detection and prevention system", NIST Special Publication 800, 2007.

[2] Salvatore Stolfo, Wenke Lee, and Andreas Prodromidis, "Cost-based modeling for fraud and intrusion detection: Result from JAM project", DARPA Information Survivability Conference and Exposition (DISCEX 2000), 2000.

[3] Farhoud Hosseinpour, Payam Vahdani Amoli, Fahimeh Farahnakian, Juha Plosila, and Timo Hamalainen, "Artificial Immune System Based Intrusion Detection: Innate Immunity using an Unsupervised Learning Approach", International Journal of Digintal Content Technology & its Applications 8.5, 2014.

[4] Kyung-min Kim, HakJu Kim, Kwangjo Kim, "Design of an Intrusion Detection System for Unknown-attacks based on Bio-inspired Algorithms", Computer Security Symposium 2015 (CSS 2015), Nagasaki, Japan, Oct. 21-23, 2015.

[5] Jean Louis Deneubourg, Simon Goss, N Franks, Ana Sendova-Franks, Claire Detrain, and L Chrestien, "The Dynamics of Collective Sorting Robot-like Ants and Ants-like Robots", Proceedings of the First International Conference on Simulation of Adaptive Behavior: From Animals to Animats, pp. 356-363, 1991.

[6] O.A. Mohamed Jafar and R. Sivakumar, "Ant-based Clustering Algorithms: A Brief Survey", International Journal of Computer Theory and Engineering, Vol.2, no.5, pp.787-796, 2010.

[7] Carlos Gershenson, *Artificial Neural Networks for Beginners*, In: Cognitive and computing sciences, University of Sussex.

[8] Leonid Portnoy, Eleazar Eskin, and Sal Stolfo, "Intrusion Detection with unlabeled data using clustering", Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001), 2001.

[9] Jungsuk Song, Hiroki Takakura, Yasuo Okabe, Masashi Eto, Daisuke Inoue, and Koji Nakao, "Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation", Proceeding of the First Workshop on Building Analysis Datasets and Gathering Experience Return for Security, pp. 29-36, 2011.