Attack Classification Analysis of IoT Network via Deep Learning Approach

Bayu Adhi Tama and Kyung-Hyune Rhee*
Department of IT Convergence and Application Engineering
Pukyong National University
bayuat@pukyong.ac.kr; khrhee@pknu.ac.kr

Abstract

A variety of attacks in the transportation layer of IoT network seeks for a detection and prevention mechanism such as intrusion detection systems (IDSs). Anomaly detection is one of the most demanding task in IDSs. It requires a robust classifier model which is able to detect different kinds of attacks intelligently. This paper addresses deep neural network for classifying attacks in IoT network. The performance of the proposed method is evaluated on the three novel benchmarking datasets in wired and wireless network environment, i.e. UNSW-NB15, CIDDS-001, and GPRS. Furthermore, deep neural network combined with grid search strategy are utilized to obtain the best parameter settings for each dataset. The experimental results demonstrate the effectiveness of our approach using deep neural network in terms of accuracy, precision, recall and false alarm rate.

Keywords: Intrusion detection systems, deep neural network, benchmarking, internet of things

1 Introduction

A 'stay-connected' feature of wireless network allows users to have a connection in anywhere and anytime. Nowadays, the emerging technology of Internet of Things (IoT) makes people's life convenience but still security and privacy issues are major concern. Due to the heterogeneity nature of IoT devices where physical objects, i.e. embedded sensors, actuators, etc exchange a huge amount of data through wireless network, thus it gains the chances to get affected by malicious attackers. IoT lies in various platforms, protocols, and applications, i.e. smart home, smart city, smart grid, etc resulting a comfortable place for malvolent users to launch attacks without a hitch. Moreover, according to Gartner [6], the market of IoT devices is fully prepared to increase rapidly and will attain about 21 billion connected devices by 2020. Consequently, it makes sense that as the number of devices increase, security and privacy will still become a primary concern in the forthcoming years.

An IoT security framework is made of three layers, i.e. perception, transportation, and application layer [9]. Perception layer possesses perception node (sensors, controllers, etc) that is used for data acquisition. Secure communications between nodes, lightweight authentication are the main security issues in this layer. In addition, there exist three layers in the transportation layer, i.e. the access network, the core network, and local area network. Transportation layer offers a ubiquitous access information for perception layer using wireless network (WiFi, 3G, etc), ad-hoc network, etc. Thus, several attacks, i.e. information disclosure, network disability, DoS attack, etc become the prevalent security issues found in this layer. To overcome these problems, attack detection and prevention mechanism might be deployed before they make a huge loss in the entire layer.

Research Briefs on Information & Communication Technology Evolution (ReBICTE), Vol. 3, Article No. 15 (November 15, 2017)
*Corresponding author: (48513) 45 Yongso-ro, Nam-Gu, Busan, South Korea. Telp: +82 51 6296247, Fax: +82 51 6264887, Web: http://lisia21.net

Such mechanism could be materialized by employing an intrusion detection system (IDS). It is an intelligent system which is able to detect and repeal any malevolent behaviours within the network. Many researchers have proposed a plethora techniques to improve the detection accuracy of IDS, yet the span-new attacks are continuously mushrooming and on the top of that they become more and more sophisticated. Anomaly-based detection is one IDS technique that can be utilized to detect some novel attacks. It probes the deviation of the network patterns that differs significantly from the normal patterns. A classifier is trained to construct a classification model from the intrusion data and we will use the model to predict the forthcoming attacks. However, solving classification problem in anomaly-based IDS is not a straightforward task since the classifier always suffers from higher false alarm rate (FAR).

Table 1: Related work of IDS methods based on neural-based classifiers

Study	Technique	Dataset	Performance metrics	Feature selection	Category
[18]	GA+SVM	DARPA 1999	DR, FP, FN	Yes	Anomaly detection
[15]	DT+SVM	KDD Cup 99	Accuracy	No	Anomaly and signature detection
[13]	SOM+ANN	DARPA 1998	DR,FA,FP	Yes	Anomaly and signature detection
[11]	SOM+SVM	DARPA 1998	Accuracy, FP, FN	No	Anomaly detection
[3]	GA+ANN	DARPA 1998	DR, FP	Yes	Anomaly detection
[23]	ANN+Fuzzy clustering	KDD Cup 99	Precision, recall, F-measure	No	Anomaly detection
[17]	DBN+SVM	NSL-KDD	Accuracy	No	Signature detection
[12]	DT+SVM	Private	DR, ROC curve	No	Anomaly and signature detection
[7]	SVM+Ant colony network	KDD Cup 99	DR, FP, FN	No	Anomaly detection
[10]	MLP	NSL-KDD	Accuracy, DR, FAR	Yes	Anomaly detection
This Study	DNN	UNSW-NB15, CIDDS-001, GPRS	Accuracy, precision, recall, FAR	No	Anomaly detection

Hitherto, anomaly-based IDS has been an active research in the purview of information security [19]. Some techniques include single and ensemble of classifier [20] have been widely implemented for IDS. Table 1 presents a review of the existing techniques of IDS using neural-based classifiers, i.e. neural network (NN) and support vector machine (SVM). In this paper, we discuss the state-of-the-art of neural classifiers and propose a deep learning network (DNN) architecture for anomaly detection. Instead of using obselet dataset, i.e. KDDCup 99 [4] and NSL-KDD [21], we report the performance of DNN on the three new benchmark datasets, i.e. UNSW-NB15 [14], CIDDS-001 [16], and GPRS [22] in terms of accuracy, precision, recall, and FAR metric.

The rest of the paper is structured as follows. Section 2 describes the overview of deep neural network, whilst Section 3 presents the experimental setup that comprises the description of datasets used in the experiment, validation method, and performance metrics. The results of our experiment is shown in Section 4 and finally some concluding remarks are drawn in Section 5.

2 Deep Neural Network

Deep neural network (DNN) was firstly promoted by [8] for a class of deep probabilistic models, so-called Deep Belief Networks (DBNs). DBNs are made up of several layers of Restricted Boltzmann

Machine (RBM), a type of neural network [24] [5]. The network possesses a two-layer architecture in which the visible binary stochastic v are connected to hidden binary stochastic h, where units within a layer are not connected.

There are several architectures for deep learning, but we focus on feedforward architecture as depicted in Figure 1. Feedforward network is made up of many layers of interconnected neurons which are the basic unit in the model. Let a training set of N instances is $\{(x^1,y^1),(x^2,y^2),...,(x^n,y^n)\}$. The input vector $\vec{x}$ is the feature vector comprising of the probability of the bit-symbol "1" and y is class label, assigned to each instance.

In the training process, the input vector $\vec{x}$ goes through the visible nodes in the network, in which initial weights $\vec{w}$ are given by DBN. Our goal is to minimize a cost function C:

$$C(\vec{w};\vec{x},y) = \frac{1}{2}\|h_w(\vec{x}) - y\|^2 \tag{1}$$

where $h_w(\vec{x})$ is hypothesis function yielding an estimated output. The overall cost is defined as:

$$C(\vec{w}) = \frac{1}{N}\sum_n C(\vec{w};\vec{x}^n,y^n) + \frac{\lambda}{2}\sum_k^K \sum_i^{L_m} \sum_j^{L_{m+1}} (w_{ij}^k)^2 \tag{2}$$

where K is the depth of the network, L_m is the number of nodes in the m-th layer, and $w_{ij}^k \in \vec{w}$ is the weight of the edges between a node i in the layer $k-1$ and a node j in the layer k. Thus, in order to minimize the overall cost function, we calculate the parameter set $\vec{w}^*$ as follows:

$$\vec{w}^* = \arg\min_w C(\vec{w}) \tag{3}$$

The $\vec{w}$ can be obtained by using back propagation algorithm which the weight vectors are updated from the top layer to the bottom layer by using the following equation:

$$w_{ij}^k = w_{ij}^{k-1} + \zeta \frac{\partial}{\partial w_{ij}^{k-1}} C(\vec{w}) \tag{4}$$

where ζ is an adaptation parameter.

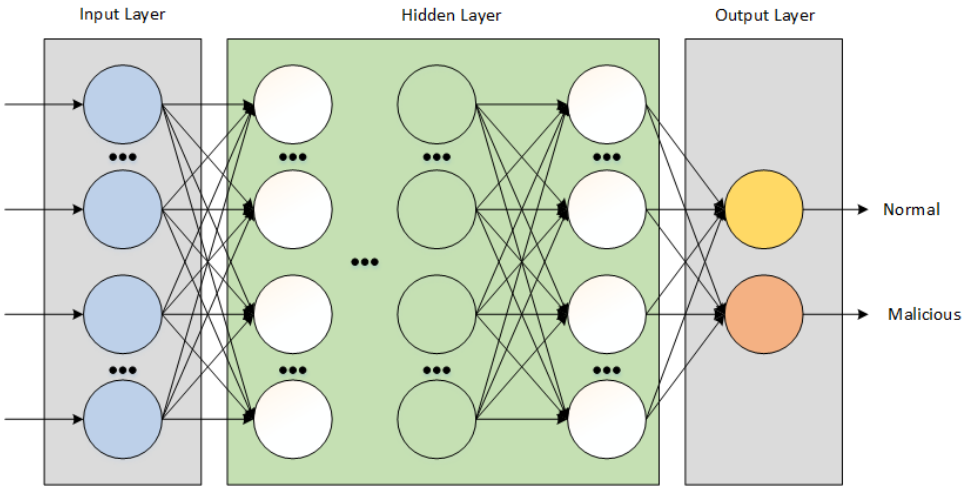


Figure 1: Deep neural network architecture for anomaly-based IDS

3 Experimental Setup

In this section, the details of experiment are discussed. It comprises the materials, method for validating the results, and performance measures used in the experiment.

3.1 Datasets

The aforementioned datasets used in our experiment are UNSW-NB15 [14], CIDDS-001 [16], and GPRS [22]. UNSW-NB15 was proposed since evaluating IDS using the existing datasets, i.e. KDD Cup 99 and NSL-KDD does not portray the satisfactory results. This because of three main issues exist: (1) they do not cover modern attack patterns, (2) they lack of modern normal traffic patterns, and (3) distribution between training and testing sets are different. UNSW-NB15 contains 42 attributes, and two classes, i.e. normal (31.94%) and malicious (68.06%).

Consecutively, CIDDS-001 was firstly introduced by Ring, *et al* [16]. It is a labelled flow-based dataset for evaluation of anomaly-based IDS. Specifically, it provides an up-to-date dataset as it is not improper to test current IDS using obsolete dataset. To generate malicious traffic, some attacks such as DoS, Brute Force, and Port Scans were performed within the network. The network traffic data has been obtained from an OpenStack environment and an external server. Furthermore, original version of CIDDS-001 consists of 10 attributes and 5 classes, i.e. normal, suspicious, unknown, attacker, and victim. Since our objective is to evaluate anomaly based IDS, we only included normal and attacker class as the final dataset. It contains 146,500 instances with the proportion of normal class is 91.6%.

Lastly, GPRS is a dataset specific to wireless environment (IEEE 802.11 standard) [22]. It was generated from two different topologies, i.e. WEP/WPA and WPA2. 9600 instances and 15 features were generated in WEP/WPA topology. The proportion of normal and malicious class is 62.5% and 37.5%, respectively. Furthermore, 7500 samples and 16 features were successfully obtained from WPA2 topology. It is made up of normal class (60%) and malicious class (40%). Table 2 summarizes the descriptions of all datasets used in our experiment.

Table 2: Description of datasets

Dataset	No of features	No of instances	Ratio between normal and malicious class
UNSW-NB15	42	175,341	1.00 : 2.13
CIDDS-001	10	146,500	10.90 : 1.00
GPRS-WEP	15	9,600	1.67 : 1.00
GPRS-WPA2	16	7,500	1.50 : 1.00

3.2 Validation Method and Performance Metrics

Concerning validation method, we employ three different resampling strategies in order to lessen the variability of the datasets as follows.

- *Cross-validation*. It is a resampling strategy in which *k*fold cross-validation, a dataset *D* is splitted into *k* subsets of equal size. In the *n*-th of the *k* loopings, the *n*-subsets is drawn for testing, whilst the blend of the remaining parts indicate the training set. In this case, 10fold cross-validation $(10FCV)$ is considered.

- *Repeated cross-validation (RepCV)*. We select 5×2 cross-validation ($5 \times 2CV$). It is performed by 5 repetitions of a $2FCV$, which give us 5 training and 5 testing subsets at 50%. In this case, the overlapping samples exist but it is less strikingly than $10FCV$.
- *Subsampling*. A dataset D is randomly divided into two subsets in accordance with a given percentage, i.e. 70% training and 30% testing set. We repeated this experiment 10 times in order to obtain the same number of elements as in $10FCV$ and $RepCV$.

Table 3: Contingency table

	Predicted	
Actual	Normal	Anomaly
Normal	True positive	False negative
Anomaly	False positive	True negative

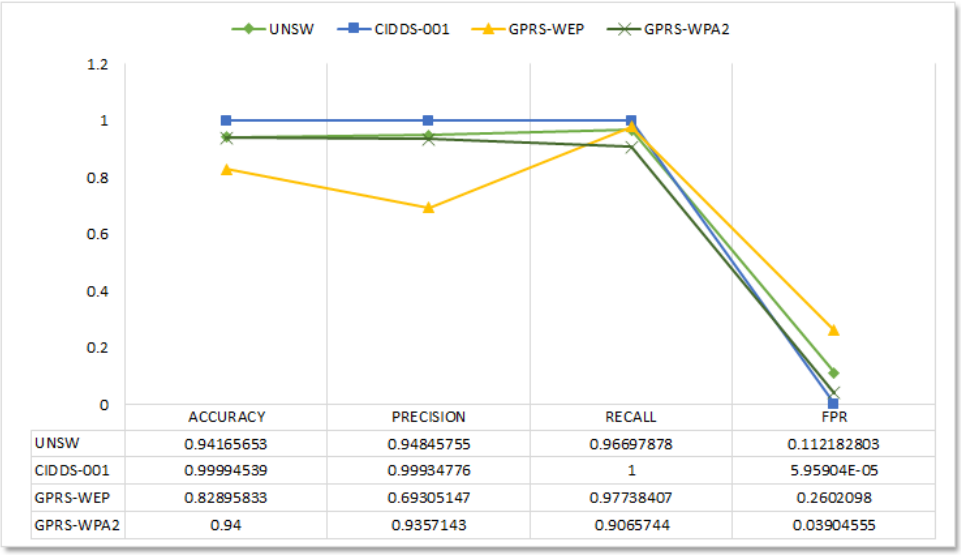
Furthermore, by taking into account Table 3 we can derive several performance measures, i.e. accuracy, precision, recall and false alarm rate as follows.

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

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

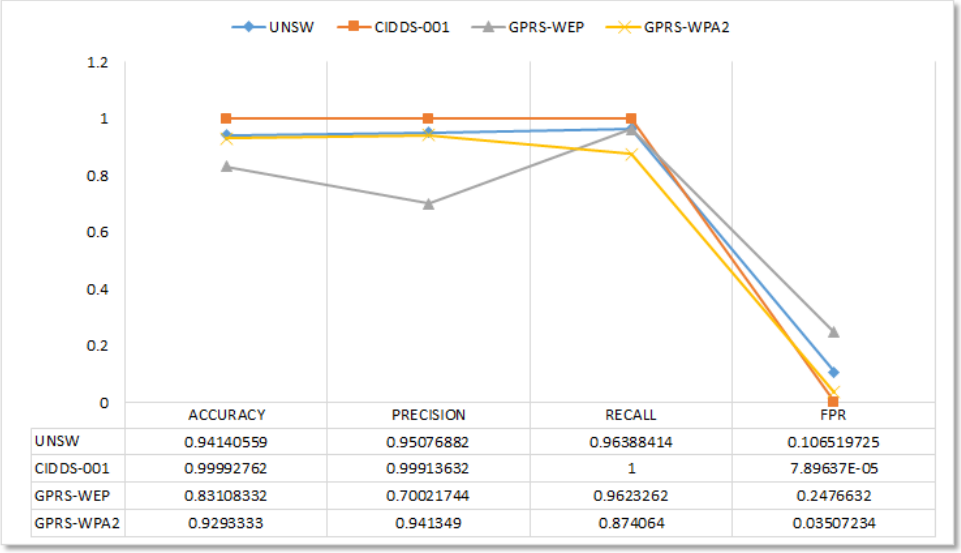
$$Recall = \frac{TP}{TP+FN} \tag{7}$$

$$FPR = \frac{FP}{FP+TN} \tag{8}$$



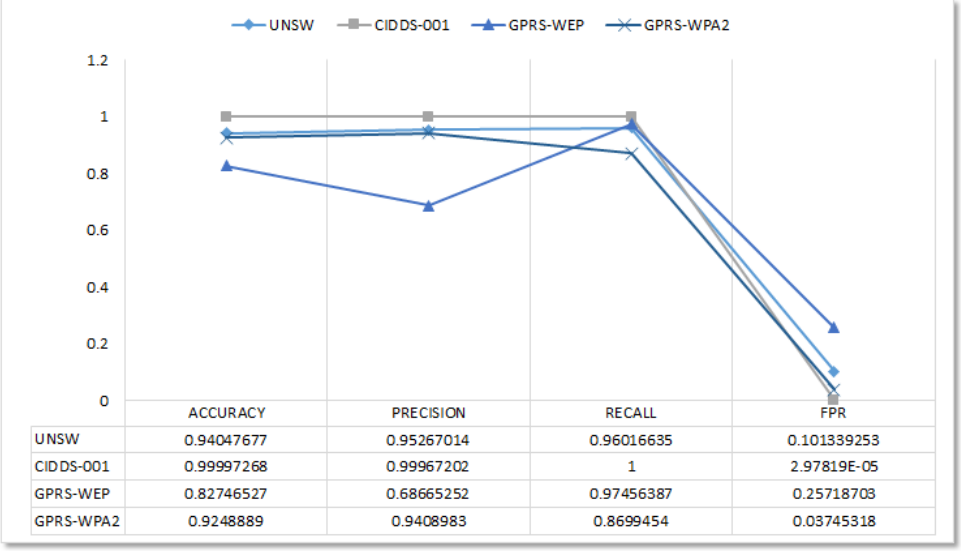
	ACCURACY	PRECISION	RECALL	FPR
UNSW	0.94165653	0.94845755	0.96697878	0.112182803
CIDDS-001	0.99994539	0.99934776	1	5.95904E-05
GPRS-WEP	0.82895833	0.69305147	0.97738407	0.2602098
GPRS-WPA2	0.94	0.9357143	0.9065744	0.03904555

Figure 2: Average performance of DNN using $10FCV$



	ACCURACY	PRECISION	RECALL	FPR
UNSW	0.94140559	0.95076882	0.96388414	0.106519725
CIDDS-001	0.99992762	0.99913632	1	7.89637E-05
GPRS-WEP	0.83108332	0.70021744	0.9623262	0.2476632
GPRS-WPA2	0.9293333	0.941349	0.874064	0.03507234

Figure 3: Average performance of DNN using $5 \times 2CV$



	ACCURACY	PRECISION	RECALL	FPR
UNSW	0.94047677	0.95267014	0.96016635	0.101339253
CIDDS-001	0.99997268	0.99967202	1	2.97819E-05
GPRS-WEP	0.82746527	0.68665252	0.97456387	0.25718703
GPRS-WPA2	0.9248889	0.9408983	0.8699454	0.03745318

Figure 4: Average performance of DNN using *subsampling*

4 Experimental Result

In this section we provide analysis of the performance results of DNN. First of all, we set learning parameters of DNN by performing *grid search* which exhaustively generates a number of candidates from a grid of parameter values specified with the given parameter. These parameters include $activation\_fuction$, $adaptive\_learning\_rate$, $learning\_rate\_annealing$, ℓ_1 regularization, ℓ_2 regularization, maximum *weight*, and *distribution* function. However, the number and size of each hidden layer in the model are specified as 3 and 150, respectively. We adopt an efficient implementation of DNN using H_2O in *R* [2] for conducting classification task. Also, *mlr* package [1] is used to run several experiments of the aforementioned resampling strategies. The results presented in this paper are average value of 10 elements obtained from each resampling method.

Figure 2 shows the average performance of DNN on each dataset when it is validated using 10-fold cross validation. In terms of all performance measures, DNN has performed perfectly applied on CIDDS-001. It achieves about 100% of accuracy, precision, and recall. Furthermore, it lowers FPR significantly by 5.96E-05. Since precision is a measure of a classifier exactness, a low precision depicts a large number of false positive (anomaly class that is predicted as normal class). From Figure 2 we can observe that DNN have performs worst on detecting malicious attack on GPRS-WEP. It can also be seen from a higher FPR. On the contrary, a low recall indicates many false negative, thus it is obvious that DNN reduces the misclassification error of anomaly detection on GPRS-WEP in comparison with UNSW and GPRS-WPA2.

The average performance of DNN on each dataset validated using $5 \times 2CV$ and *subsampling* are presented in Figure 3 and Figure 4, respectively. The results are almost similar to the previous. DNN is able to minimize the misdetection error of anomaly in WPA2 topology as it has a higher precision and lower FPR than in WEP topology. On the other hand, DNN is unable to lessen the number of false negative in WPA2 topology since its performance value in terms of recall is lower than in WEP and UNSW. Based on the above-mentioned experimental results, we draw some several remarks as follows.

- DNN always performs best while it is applied on CIDDS-001 dataset regardless of validation method used. This bias results may occur as the CIDDS-001 dataset has imbalanced problems, which the distribution of one class is significantly lower than the other class (see Table 2). For the current experiment, we will rebuilt the dataset by aggregating the traffic samples collected from external server and OpenStack. If the imbalanced data still occurs, we might consider ensemble technique by combining DNN and other classifiers.

- Different validation methods have not significantly affected the performance value of DNN. This might be happened since the number of element is too small (10 in our case). For the future work, we will investigate larger value of the experiment repetition.

- We are unable to observe whether there are performance differences between DNN and other algorithms. As for thorough comparison, we will consider a statistical test to prove the significant difference between DNN and other similar deep learning architecture, i.e. stacked auto-encoders (SAEs) and convolutional neural networks (CNNs).

5 Conclusion

In this paper, we analyzed and discussed attack classification in IoT network using deep neural network (DNN). The performance of DNN is assessed using various validation methods, i.e. cross-validation, repeated cross-validation, and subsampling on different novel benchmark datasets, i.e. UNSW-NB15, CIDDS-001, and GPRS. We conducted a *grid search* to discover the best learning parameters of DNN for each dataset. DNN gave a satisfactory performance, in particular the attack detection performance in wireless environment. Furthermore, the experimental results suggest that studies about anomaly detection should include several validation methods and datasets, as Table 1 is evidently not the case. We should pay attention on some classification problems, i.e. imbalance dataset, bias results, etc since there is no a panacea to improve the performance of the classifier.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2014R1A2A1A11052981), and partially supported by the MSIT

(Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-2015-0-00403) supervised by the IITP (Institute for Information & communications Technology Promotion).

References

[1] B. Bischl, M. Lang, L. Kotthoff, J. Schiffner, J. Richter, E. Studerus, G. Casalicchio, and Z. M. Jones. mlr: Machine learning in r. *Journal of Machine Learning Research*, 17(170):1–5, 2016.

[2] A. Candel, V. Parmar, E. LeDell, and A. Arora. Deep learning with h2o, 2015.

[3] Y. Chen, A. Abraham, and B. Yang. Hybrid flexible neural-tree-based intrusion detection systems. *International journal of intelligent systems*, 22(4):337–352, February 2007.

[4] K. Cup. Dataset, 1999. `http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html` [Online; Accessed on October 3, 2017].

[5] L. Deng, D. Yu, et al. Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387, June 2014.

[6] N. Eddy. Gartner: 21 billion IoT devices to invade by 2020. *InformationWeek, Nov*, 10, 2015.

[7] W. Feng, Q. Zhang, G. Hu, and J. X. Huang. Mining network data for intrusion detection through combining SVMs with ant colony networks. *Future Generation Computer Systems*, 37:127–140, July 2014.

[8] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, July 2006.

[9] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu. Security of the internet of things: Perspectives and challenges. *Wireless Networks*, 20(8):2481–2501, November 2014.

[10] S.-H. Kang and K. J. Kim. A feature selection approach to find optimal feature subsets for the network intrusion detection system. *Cluster Computing*, 19(1):325–333, January 2016.

[11] L. Khan, M. Awad, and B. Thuraisingham. A new intrusion detection system using support vector machines and hierarchical clustering. *The International Journal on Very Large Data Bases*, 16(4):507–521, October 2007.

[12] G. Kim, S. Lee, and S. Kim. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. *Expert Systems with Applications*, 41(4):1690–1700, March 2014.

[13] G. Liu, Z. Yi, and S. Yang. A hierarchical intrusion detection model based on the PCA neural networks. *Neurocomputing*, 70(7):1561–1568, March 2007.

[14] N. Moustafa and J. Slay. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *Military Communications and Information Systems Conference (MilCIS'15), Canberra, Australian Capital Territory, Australia*, pages 1–6. IEEE, November 2015.

[15] S. Peddabachigari, A. Abraham, C. Grosan, and J. Thomas. Modeling intrusion detection system using hybrid intelligent systems. *Journal of network and computer applications*, 30(1):114–132, January 2007.

[16] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho. Flow-based benchmark data sets for intrusion detection. In *Proc. of the 16th European Conference on Cyber Warfare and Security (ECCWS'17),*, pages 361–369. ACPI, June 2017.

[17] M. Salama, H. Eid, R. Ramadan, A. Darwish, and A. Hassanien. Hybrid intelligent intrusion detection scheme. *Soft computing in industrial applications*, 96:293–303, 2011.

[18] T. Shon and J. Moon. A hybrid machine learning approach to network anomaly detection. *Information Sciences*, 177(18):3799–3821, September 2007.

[19] B. A. Tama and K. H. Rhee. A combination of PSO-based feature selection and tree-based classifiers ensemble for intrusion detection systems. In *Proc. of the Advances in Computer Science and Ubiquitous Computing*, volume 373 of *Lecture Notes in Electronic Engineering*, pages 489–495. Springer, Singapore, 2015.

[20] B. A. Tama and K.-H. Rhee. Data mining techniques in DoS/DDoS attack detection: A literature review. *Information*, 18(8):3739–3748, August 2015.

[21] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani. A detailed analysis of the KDD Cup 99 data set. In *Proc. of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*

(CISDA'09), Ottawa, Ontario, Canada, pages 1–6. IEEE, July 2009.

[22] D. W. Vilela, T. F. Ed'Wilson, A. A. Shinoda, N. V. de Souza Araujo, R. de Oliveira, and V. E. Nascimento. A dataset for evaluating intrusion detection systems in IEEE 802.11 wireless networks. In *Proc. of the 2014 IEEE Colombian Conference on Communications and Computing (COLCOM'14), Bogota, Colombia*, pages 1–5. IEEE, June 2014.

[23] G. Wang, J. Hao, J. Ma, and L. Huang. A new approach to intrusion detection using artificial neural networks and fuzzy clustering. *Expert systems with applications*, 37(9):6225–6232, September 2010.

[24] D. Yu and L. Deng. Deep learning and its applications to signal and information processing [exploratory dsp]. *IEEE Signal Processing Magazine*, 28(1):145–154, January 2011.

Author Biography

Bayu Adhi Tama completed a master degree of information technology (M.IT) from The University of Indonesia in 2008. He is currently a PhD candidate in the Laboratory of Information Security and Internet Applications (LISIA), Pukyong National University, Republic of Korea. During his doctoral study, he receives a scholarship and an award from Korean Government for his excellent academic achievement. Besides serving as a reviewer in several prestigious journals and conferences such as Complexity, ECML-PKDD, DEXA, IEEE TENCON, etc, he also has successfully published his works in some high-impact journals such as Artificial Intelligence Review, Neural Computing and Applications, IEICE Transaction, etc. His research interests include machine learning and data analytics for cyber-security applications.

Kyung-Hyune Rhee received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide, the University of Tokyo, and the University of California, Irvine. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on key management and its applications, mobile communication security and security evaluation of cryptographic algorithms.