Journal of Korea Multimedia Society Vol. 21, No. 5, May 2018(pp. 617-625)
https://doi.org/10.9717/kmms.2018.21.5.617

A Comparative Study of Phishing Websites Classification Based on Classifier Ensembles

Bayu Adhi Tama[†], Kyung-Hyune Rhee[††]

ABSTRACT

Phishing website has become a crucial concern in cyber security applications. It is performed by fraudulently deceiving users with the aim of obtaining their sensitive information such as bank account information, credit card, username, and password. The threat has led to huge losses to online retailers, e-business platform, financial institutions, and to name but a few. One way to build anti-phishing detection mechanism is to construct classification algorithm based on machine learning techniques. The objective of this paper is to compare different classifier ensemble approaches, i.e. random forest, rotation forest, gradient boosted machine, and extreme gradient boosting against single classifiers, i.e. decision tree, classification and regression tree, and credal decision tree in the case of website phishing. Area under ROC curve (AUC) is employed as a performance metric, whilst statistical tests are used as baseline indicator of significance evaluation among classifiers. The paper contributes the existing literature on making a benchmark of classifier ensembles for web phishing detection.

Key words: Phishing Website, Classifier Ensembles, Performance Comparison, Significance Test

1. INTRODUCTION

Hitherto, phishing has gained a lot of attention from cyber security researchers and practitioners due to its widespread escalation. Phishing deals with the trial to acquire personal information such as usernames, passwords, and credit card details, apparently for malevolent intention, by camouflaging as a trusted users over the Internet. Users can be targeted either through email scam, websites, or short message service. Using such a fake company branding, for instance, a well-known company with large customer base is a priceless target for brand-jacking; an attempt to abuse the trademarks of a company to fool targets. According to the report, nearly 90% of users have faced a security incident originating with a deceptive email, making an increase of phishing attacks at 65% in 2017 in comparison with the previous year [1].

Web phishing detection using machine learning techniques have been proposed in order to establish a deterrent action taken to countermeasure threat [2]. It detects threat intelligently using pre-defined model which is built by classification algorithm. The task of constructing classification model could be considered as binary classification problem which the classifier is trained to classify web phishing data set either as normal or malicious. Solving binary classification problem is non-trivial process as it oftentimes suffers high false positive rate (FPR). Most prior works have been focused on single classifier which might not appropriate to

※ Corresponding Author : Kyung-Hyune Rhee, Address: A12-1305, Daeyeon Campus, Pukyong National University, Yongso-ro 45, Nam-gu, Busan, (48513), Republic of Korea , TEL : +82-51-629-6247, FAX : +82-51-626-4887, E-mail : khrhee@pknu.ac.kr

Receipt date : Feb. 1, 2018, Revision date : Mar. 29, 2018
Approval date : Apr. 13, 2018

[†] Dept. of IT Convergence and Application Engineering, Pukyong National University
(E-mail : bayuat@pukyong.ac.kr)
[††] Dept. of IT Convergence and Application Engineering, Pukyong National University

※ This research is supported by a Research Grant of Pukyong National University (2017 year)

enhance detection accuracy and to reduce FPR at once [3].

Classifier ensembles train multiple learners to predict the final output. They aggregate several weak classifiers whose individual class prediction are incorporated in some techniques, i.e. voting, averaging, and so forth to establish final class prediction. Instinctively, classifier ensembles solve different problems that might be difficult to be tackled by only a single classifier [4] [5]. Because of this benefit, classifier ensembles have been adopted in many real-world applications. Furthermore, 'no free lunch' theorem shows that there is no single classifier which is applicable for any problems [6]. Thus, in practical point of view, it is not straightforward to seek a good single classifier.

This paper investigates the performance of classifier ensembles for automatic web phishing detection. Several ensemble learning approaches are included in the study such as random forest (RF) [7], rotation forest (RotFor) [8], gradient boosted machine (GBM) [9], and extreme gradient boosting (XGBoost) [10]. Since these ensembles are constructed using a number of weak classification models, several tree-based classifiers, i.e. decision tree (DT) [11], classification and regression tree (CART) [12], and credal decision tree (CDT) [13] are also incorporated in our experiment.

The rest of the paper is structured as follows. Section 2 presents state-of-the-art review of phishing web detection found in the literature, while Section 3 describes overview of classification algorithms used in this study. The data set, validation technique, and significance test based on statistics are provided in Section 4. Next, Section 5 further discusses the experimental result, and finally the paper is concluded in Section 6.

2. PHISHING WEB DETECTION: A REVIEW

Prior researches have considered various machine learning algorithms for phishing web detection. However, most related works have been focused on a single classifier. Even though there exist a plethora of detection methods have been previously proposed such as LibSVM [14], fuzzy classifier [15], and to name a few, we restrict only several researches which data set in [16] is used for the experimentation. Phishing web detection using self-structuring neural network is proposed by [17] [18]. The proposed algorithm show competitive results in terms of various evaluation metrics. A study in [19] suggests hybrid approach for identifying phishing websites. The proposed approach eliminates unused features from previous works.

Rule based phishing detection is proposed by [20]. The experiment reveals that the error-rate has decreased for all the algorithms, CBA classifier algorithm has the lowest error-rate with 4.75%. A performance comparison of machine learning algorithms for web phishing detection has been conducted by [21]. Several classifiers have been included in the study, i.e. RF, DT, REP Tree, decision stump, and so forth. The authors claimed that RF with REP Tree is the best performer. The combination of computational based feature selection and a number of classification algorithms, i.e. RIPPER, PART, and C4.5 have been suggested to improve the performance of web phishing detection [22]. According to the their experiment, there are slightly performance drop when comparing full feature set against reduced set for RIPPER classifier. However, for PART and C4.5 show undeviating performance.

A novel web phishing website based on probabilistic neural network (PNN) has been presented in [23]. A *k*-medoids clustering approach is also incorporated in order to evaluate the complexity of the proposed method. Based on their experiment, an effective detection model with a reduced complexity can be built without sacrificing detection accuracy. Finally, a hybrid feature selection technique by combining scores from two effective feature selection methods, i.e. information gain and chi-square is described in [24]. The results obtained from applying the proposed method against full feature set, it has been revealed that the proposed method is able to pick relevant features that impact on the phishing detection rate.

3. CLASSIFICATION ALGORITHMS

3.1 Single Classifier

In this section, different single classification algorithms used in our study are briefly illustrated as followings.

- Decision Tree (DT)

Tree are produced in a top-down approach from root to nodes. The selection of the feature for a node is based on the impurity of the distribution of the class label. The impurity might be quantified in different way, e.g. entropy-based and Gini index. In order to avoid *over-fitting* in the training set, it is recommended to apply pruning strategy in order to generalize the tree generated by generating sub tree during the growing stage. The two main alternatives for constructing trees are the ID3 algorithm and the C4.5 algorithm, however, in this experiment, we use C50 algorithm which is the most renowned tree-based implementation [25]. There are several parameter in C50, i.e. *confident factor* (CF), *sampling factor* that specifies the random proportion of the data should be used to train the model, and *global pruning step* to simplify the tree.

- Classification and Regression Tree (CART)

The classifier is a tree-constructing technique which identifies splitting variables based on an exhaustive search. It has a number of advantages over other classification methods i.e. it can handle numerical data that are highly skewed and it has sophisticated method for dealing with missing variables. For CART, there are two parameters, i.e. the number of folds in the internal cross-validation (f) and the minimal number of observations at the terminal nodes (t). We considered f and t are 5 and 2, respectively. Furthermore, heuristic process for binary split of nominal attributes and the pruning strategy are used. We used CART which space parameter of tree is optimized by evolutionary algorithm.

- Credal Decision Tree (C-DT)

Unlike DT which uses imprecise information gain ratio as split criterion to select the split attribute at each branching node, C-DT comes with an imprecise probabilities and the application of uncertainty measures for the original split criterion [26]. It uses reduced-error pruning (with *back-fitting*) and sorted values for numeric attributes. Missing values and numeric attributes are treated like C4.5. Learning parameter setting of C-DT includes *fold numbers* which specifies the amount of data used for pruning, *root attribute* which is used to fix the root node of the tree, and S value which is a parameter in the Imprecise Dirichlet Model to obtain imprecise probabilities as intervals.

3.2 Classifier Ensembles

Now, we discuss about different orchestration techniques to combine single classifier as follows.

- Random Forest

This generates a number of trees. Random trees are grown without pre- or post-pruning, which contributes to their diversity. At each node, the feature to split upon is chosen from a randomized split of the original feature. Classification accuracy is gained some increase since the diversity of the trees. There are only two parameters in RF, i.e. number of trees and the number of variables to try at each split. Because selecting large number of trees leads to reduce the performance of ensemble, we consider the number of trees is 500 and set the number of variables to the square root of the total number of predictors.

- Rotation Forest

Rotation forest depends upon unstable classifiers, i.e. decision tree regarding rotation of the space. It emphasize on the idea that diversity can be implemented without jeopardizing either data objects or features. The potential accuracy loss of the base classifiers is counterbalanced by increasing diversity. The feature set F is randomly partitioned into L subsets, PCA is run separately on each subset, and a new set of the extracted attributes is constructed by pooling all principal components. Then the data are transformed into the new feature space. An iteration number is one parameter which represents the number of iterations to be performed. Besides, base classifier and projection technique can also be specified.

- Gradient Boosted Machine

Gradient boosted machine is constructed to improve the performance of CART. Final class prediction is made through the same type of base classifiers forming the ensemble. One of the main problem in the tree learning is to find the best split. To solve this issue, exact greedy algorithm is commonly used. We use original GBM algorithm found in [9] and a fast implementation of GBM, so called XGBoost [10]. Like RF, different parameters can be assigned such as *num_trees* is 500, *nrounds* = 10, α, λ, and *max_depth*.

4. EXPERIMENT CONFIGURATION

4.1 Data set

We employed publicly available data set which is specifically for web phishing detection [16]. Even though there exist plethora of researches about detecting phishing website have been done, no dependable training data set has been proposed for evaluating machine learning algorithms. The data set is made up of 30 input features with 1 class label feature. The number of samples is 11,055 instances with the proportion of samples belonging to negative (-1) and positive (+1) class is 44.31% and 55.69%, respectively. The data set has been pre-processed by the authors of [20] [18] into a unified format and no missing values are found. In the data set, some new features are introduced and only the important features that have been proved to make an effective phishing website detection are included.

4.2 Resampling Approaches

Performance evaluation of a machine learning algorithm relies on the model selection procedures. Resampling procedures offer a performance approximation in terms of repeatedly dividing data set D to form a training set and a test set. Suppose Tr_i depicts the training set and Te_i is the test set, in the i-th iteration of the resampling procedure, such that:

$$Tr_i \cap Te_i = \emptyset \text{ and } Tr_i \cup Te_i = D \quad (1)$$

In this experiment, different resampling procedures are used such as k-fold cross validation, subsampling, and bootstrap. In the k-cross validation, make k disjoint partitions of approximately equal size. Each k iteration, a different partition is used for testing and the others for training. Subsampling is a hold-out with repetition, where only a subset of the data set is used in each iteration. Furthermore, bootstrap obtain Tr_i by sampling n items with replacement from D and $Te_i = D/Tr_i$.

In order to acquire the same element at each resampling procedure, we are interested to investigate the following methods: 2 times repeated 10-fold (2×10f), 4 times repeated 5-fold (4×5f), 20 times repeated 2/3 hold-out (20×ho), and 20 times repeated boostrap (20×boot). Area under ROC curve (AUC) is used as a performance metric and Demsar procedure [27] is followed to determine whether a statistically significance exist in the performance of multiple classifiers. Demsar proposes Friedman test as a wide-ranging nonparametric paired test. The test calculates the *p*-value based on the null hypothesis that all classifiers have performed 'equivalent' with respect to their rankings. If the Friedman test indicates 'significant', a post-hoc test using Nemenyi is recommended.

5. RESULT AND DISCUSSION

In this section, the results of our experiment are discussed. Fig. 1 depicts the mean average of classifier's performance with respect to AUC value. Random forest (RF) has shown the best performance, followed by extreme gradient boosted machine (xgboost), and C50. Surprisingly, the worst algorithm goes to gradient boosted machine (GBM). Among single classifiers, C50 denotes the most effective algorithms for web phishing detection, followed by credal decision tree (C-DT) and CART. In addition, the results indicate that in the group of ensemble learners, RF outperforms xgboost and there is a notable performance gap between xgboost and GBM as well.

Table 1. The result of significant test using Friedman test

Resampling approaches	Chi-squared	*p*-value
2×10f	111.73	< 2.2E-16
4×5f	112.74	< 2.2E-16
20×ho	116.96	< 2.2E-16
20×boot	117.43	< 2.2E-16

Some classifiers have no performance differences regardless of resampling strategies used, i.e.xgboost, GBM, and RF, whilst other classifiers, i.e. rotation forest (RoF) and C-DT show their performance variability with reference to resampling approaches. In addition, performance result of each classifier ensemble with respect to standard cross-validation technique, i.e. 10fold cross-validation (10f) is presented in Fig. 2.

It is obvious that the top performer among ensemble algorithms is RF, whilst GBM have per-

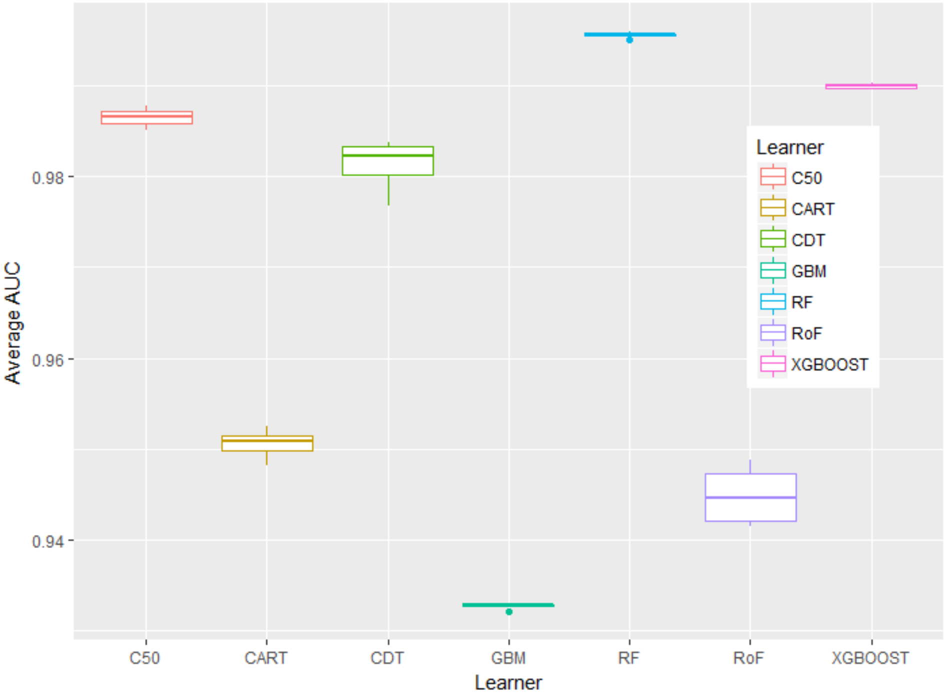

Fig. 1. Performance average in terms of AUC value over different resampling strategies.

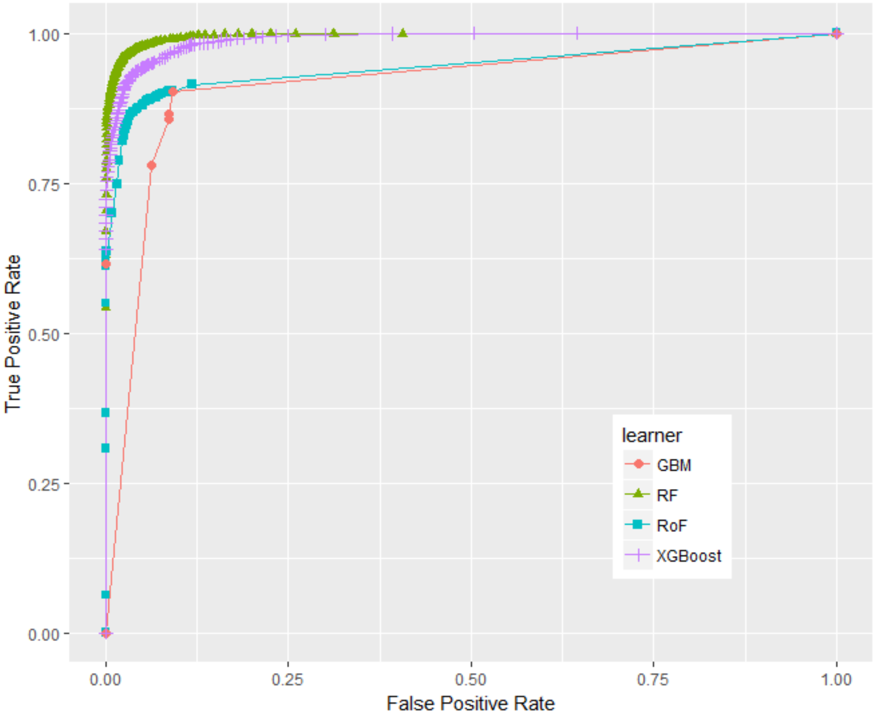

Fig. 2. Performance average of classifier ensembles for 10f.

formed worse in phishing web detection. In order to provide an ample comparative study, the performance differences of all classifiers are subsequently benchmarked using statistical test. First of all, the result of Friedman test is shown in Table 1. The Friedman test indicates that there exist a highly significant (*p < 2.2E-16*) difference among classifiers regardless of resampling approaches used. Since Friedman test points out the significance of these results, it is worthwhile to conduct Nemenyi post-hoc test. The results of the post-hoc test at each resampling approach are visually represented with critical difference (CD) diagram as shown in Fig. 3-6. We are interested to use significant level α=0.01.

The comparison of the groups of classifiers against each other are described in Fig. 3-6. The groups of classifiers that are not significantly different with other groups are connected with the straight line, whilst the top line indicates the interval of the Friedman CD's value. The graphs indicate that the performance of RF, xgboost, and C50 are not significantly different regardless the validation methods used. In addition, the Friedman test reveals that GBM performs significantly worse than RF and xgboost, which seems to possess equivalent AUC value in all resampling approaches. Furthermore, CART and RoF share equivalent performance in terms of 4x5f and 20×ho approaches.

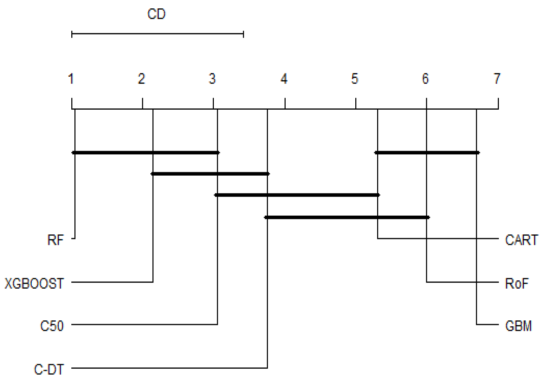

Fig. 3. Critical difference diagram of 2×10f.

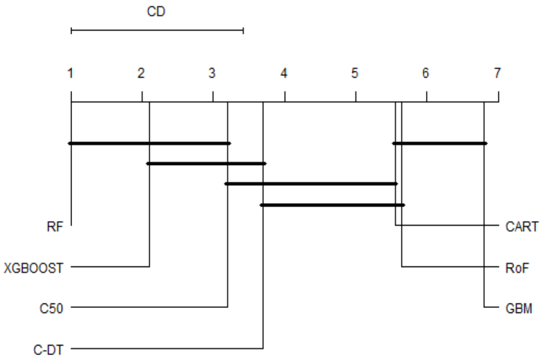

Fig. 4. Critical difference diagram of 4×5f.

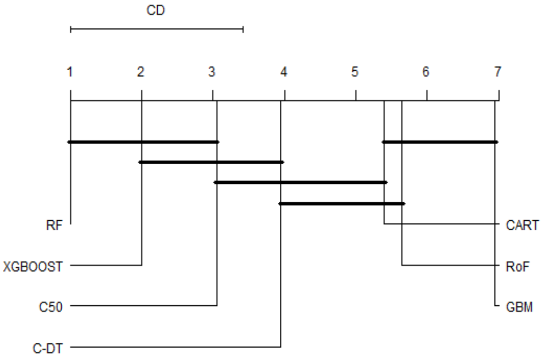

Fig. 5. Critical difference diagram of 20×ho.

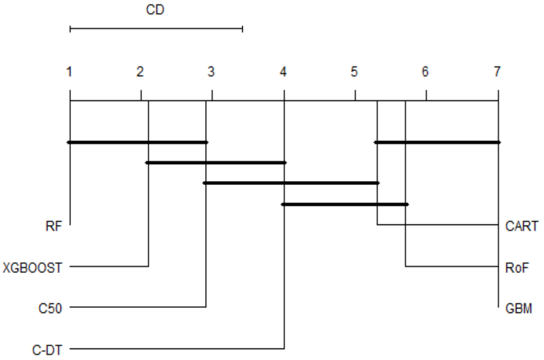

Fig. 6. Critical difference diagram of 2×10f.

6. CONCLUSION

This paper provided a comparative study of classifier ensembles for phishing web detection. A number of ensembles algorithms and single classification algorithms were included in the experiment. Their detection performance were evaluated using AUC value with respect to different resampling approaches. The experimental results revealed that random forest was superior to other ensembles, i.e. xgboost, rotation forest and GBM and to single classifiers, i.e. C50, C-DT, and CART. Further study should include other web phishing data set in order to provide a more comprehensive benchmark.

REFERENCES

[1] A.P.W. Group, *White Paper: Phishing Response Trends*, Technical Report, 2017.

[2] S.C. Jeeva and E.B. Rajsingh, “Intelligent Phishing URL Detection Using Association Rule Mining,” *Human-Centric Computing and Information Sciences,* Vol. 6, No. 1, pp. 1-19, 2016.

[3] B.A. Tama and K.H. Rhee, “Performance Analysis of Multiple Classifier System in DoS Attack Detection,” *Proceeding of International Workshop on Information Security Applications,* pp. 339-347, 2015.

[4] K.S. Komariah, C. Machbub, A.S. Prihatmanto, and B.-K. Shin, “A Study on Efficient Market Hyphothesis to Predict Exchange Rate Trends Using Sentiment Analysis of Twitter Data,“ *Journal of Korea Multimedia Society,* Vol. 19, No. 7, pp. 1107-1115, 2016.

[5] N.C. Oza and K. Tumer, “Classier Ensembles: Select Real-World Applications,” *Information Fusion,* Vol. 9, No. 1, pp. 4-20, 2008.

[6] D.H. Wolpert, “The Lack of a Priori Distinctions Between Learning Algorithms,” *Neural Computation,* Vol. 8, No. 7, pp. 1341-1390, 1996.

[7] L. Breiman, “Random Forests,” *Machine Learning*, Vol. 45, No. 1, pp. 5-32, 2001.

[8] J.J. Rodriguez, L.I. Kuncheva, and C.J. Alonso, “Rotation Forest: A New Classifier Ensemble Method,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 28, No. 10, pp. 1619-1630, 2006.

[9] J.H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *The Annals of Statistics,* Vol. 29, No. 5, pp. 1189-1232, 2001.

[10] T. Chen and C. Guestrin, "XGboost: A Scalable Tree Boosting System," *Proceeding of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* pp. 785-794, 2016.

[11] J.R. Quinlan, *C4.5: Programs for Machine Learning,* Calif : Morgan Kaufmann Publishers, San Mateo, 2014.

[12] W.Y. Loh, "Classification and Regression Trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* Vol. 1, No. 1, pp. 14-23, 2011.

[13] C.J. Mantas and J. Abellan, "Credal-C4.5: Decision Tree Based on Imprecise Probabilities to Classify Noisy Data," *Expert Systems with Applications,* Vol. 41, No. 10, pp. 4625-4637, 2014.

[14] R.B. Basnet, S. Mukkamala, and A.H. Sung, "Detection of Phishing Attacks: A Machine Learning Approach," *Soft Computing Applications in Industry,* Vol. 226, pp. 373-383, 2008.

[15] M. Aburrous, M.A. Hossain, K. Dahal, and F. Thabtah, "Intelligent Phishing Detection System for E-Banking Using Fuzzy Data Mining," *Expert Systems with Applications,* Vol. 37, No. 12, pp. 7913-7921, 2010.

[16] M. Lichman, UCI Machine Learning Repository, 2013. (accessed Jan., 8, 2018)

[17] F. Thabtah, R.M. Mohammad, and L. Mc Cluskey, "A Dynamic Self-Structuring Neural Network Model to Combat Phishing," *Proceeding of Neural Networks 2016 International Joint Conference on IEEE*, pp. 4221-4226, 2016.

[18] R.M. Mohammad, F. Thabtah, and L. Mc Cluskey, "Predicting Phishing Websites Based on Self-Structuring Neural Network," *Neural Computing and Applications,* Vol. 25, No. 2, pp. 443-458, 2014.

[19] M. Dadkhah, M. Dadkhah, S. Shamshirband, S. Shamshirband, and A.W.A. Wahab "A Hybrid Approach for Phishing Web Site Detection," *The Electronic Library,* Vol. 34, No. 6, pp. 927-944, 2016.

[20] R.M. Mohammad, F. Thabtah, and L. Mc Cluskey, "Intelligent Rule-Based Phishing Websites Classification," *IET Information Security*, Vol. 8, No. 3, pp. 153-160, 2014.

[21] A. Hodzic, J. Kevric, and A. Karadag, "ComParison of Machine Learning Techniques in Phishing Website Classification," *Proceeding of International Conference on Economic and Social Sciences,* pp. 249-256, 2016.

[21] F. Thabtah and N. Abdelhamid, "Deriving Correlated Sets of Website Features for Phishing Detection: A Computational Intelligence Approach," *Journal of Information and Knowledge Management,* Vol. 15, No. 04, pp. 1-17, 2016.

[23] E.S.M. El-Alfy, "Detection of Phishing Websites Based on Probabilistic Neural Networks and K-Medoids Clustering," *The Computer Journal*, Vol. 60, No. 12, pp. 1-5, 2017.

[24] K.D. Rajab, "New Hybrid Features Selection Method: A Case Study on Websites Phishing," *Security and Communication Networks,* Vol. 2017, pp. 1-10, 2017.

[25] R. Quinlan, Data Mining Tools See5 and C5.0, 2004. http://www.rulequest.com/see5-info.html (accessed Jan., 8, 2018)

[26] J. Abellan and S. Moral, "Building Classification Trees Using the Total Uncertainty Criterion," *International Journal of Intelligent Systems,* Vol. 18, No. 12, pp. 1215-1225, 2003.

[27] J. Demsar, "Statistical Comparisons of Classifiers Over Multiple Data Sets," *Journal of Machine Learning Research,* Vol. 7, No. Jan, pp. 1-30, 2006.

Bayu Adhi Tama

He received his PhD degree from Pukyong National University, Rep. of Korea in 2018. He is currently a postdoctoral research fellow at the School of Management Engineering, Ulsan National Institute of Science and Technology (UNIST), Rep. of Korea. His current research interests include data analytics for business process management, information security, and healthcare informatics.

Kyung-Hyune Rhee

He received his MS and PhD degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively He is currently a Professor in the Department of IT Convergence and Application Engineering at Pukyong National University, Republic of Korea. His research interests are data mining and its applications, multimedia engineering, and information security.