CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. (2016)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.3929

SPECIAL ISSUE PAPER

Multilayer hybrid strategy for phishing email zero-day filtering

M. U. Chowdhury[1], J. H. Abawajy[1], A. V. Kelarev[1,*,†] and T. Hochin[2]

[1] *School of Information Technology, Deakin University, Locked Bag 20000, Geelong, Vic 3220, Australia*
[2] *Division of Information Science, Graduate School of Science and Technology, Kyoto Institute of Technology, Kyoto, Japan*

SUMMARY

The cyber security threats from phishing emails have been growing buoyed by the capacity of their distributors to fine-tune their trickery and defeat previously known filtering techniques. The detection of novel phishing emails that had not appeared previously, also known as zero-day phishing emails, remains a particular challenge. This paper proposes a multilayer hybrid strategy (MHS) for zero-day filtering of phishing emails that appear during a separate time span by using training data collected previously during another time span. This strategy creates a large ensemble of classifiers and then applies a novel method for pruning the ensemble. The majority of known pruning algorithms belong to the following three categories: ranking based, clustering based, and optimization-based pruning. This paper introduces and investigates a multilayer hybrid pruning. Its application in MHS combines all three approaches in one scheme: ranking, clustering, and optimization. Furthermore, we carry out thorough empirical study of the performance of the MHS for the filtering of phishing emails. Our empirical study compares the performance of MHS strategy with other machine learning classifiers. The results of our empirical study demonstrate that MHS achieved the best outcomes and multilayer hybrid pruning performed better than other pruning techniques. Copyright © 2016 John Wiley & Sons, Ltd.

Received 19 October 2015; Revised 26 April 2016; Accepted 06 July 2016

KEY WORDS: attribute selection; ensembles; filtering; multilayer strategy; phishing emails; pruning

1. INTRODUCTION

This paper proposes an advanced hybrid multilayer strategy for phishing email filtering, inspired by other multilayer approaches considered for different cyber security problems in [1–4].

The number of attacks mounted by phishing email creators has kept growing despite many excellent filtering methods developed by the researchers recently. The growth in cyber security threats from phishing emails is confirmed, for instance, by the most recent report [5], which tells us that in the fourth quarter of 2014 the incidence of phishing has increased again. This demonstrates the inventiveness of the phishing email distributors and their capacity to fine-tune or alter the employed trickery in order to defeat already known filtering. Therefore, it is vitally important for cyber security to continue active research work devoted to the design of new approaches for phishing email filtering. New advanced hybrid techniques combining arrays of different methods are required to make it rather difficult for the phishing email distributors to escape filtering.

Phishing has continued managing to cheat large proportion of users [6], whereas filtering of novel phishing emails, which apply new trickery to circumvent previous anti-phishing methods, remains a particular challenge [7]. Following analogy with the well-known term ‘zero-day malware’ [8], in

*Correspondence to: Andrei V. Kelarev, School of Information Technology, Deakin University, 221 Burwood Hwy, Melbourne, Vic 3125, Australia.
†E-mail: andreikelarev-deakinuniversity@yahoo.com

this paper, the words ‘zero-day phishing emails’ mean unknown phishing emails, which have not been detected in previous days. Accordingly, by zero-day filtering, we mean filtering of zero-day phishing emails.

To solve this problem, we propose a new advanced multilayer strategy and undertake an empirical study investigating it. Our paper introduces and combines a number of novel ideas as contributions to this research area. The novelty of our approach can be briefly summarized here as follows.

First, we introduce a new multilayer hybrid strategy (MHS) for phishing email filtering. It encompasses 10 layers and is a hybrid of several novel approaches. As a general inspiration for this method, we can indicate another layered approach studied in a different cyber security area of malware detection in [1]. Our MHS has more layers and tackles the new problem of filtering phishing emails during a new time span utilizing only the training data of a separate previous time span. This is inspired by the previous work in the different area of malware detection, where analogous task was investigated in [9, 10]. In the case of phishing email, this problem can be regarded as an advanced special case of zero-day phishing email filtering. This is why we have to use several separate datasets corresponding to separate time spans. To handle this problem, MHS also invokes a new hybrid process of attribute selection. It combines the selection of structural attributes and dynamic text-based attributes designed to fine-tune MHS for its application in the new time span.

Second, as an integral part of MHS, we introduce and investigate a new hybrid pruning approach. It is called multilayer hybrid pruning (MHP). Our empirical study presented in this paper investigates the effectiveness of MHP in the setting of the MHS strategy for zero-day phishing email filtering.

The first major contribution of this paper, MHS strategy, is the subject of thorough examination in a comprehensive empirical study utilizing separate datasets for different time spans. This empirical study pinpoints the values of the input parameters to MHS that can be used to achieve optimum output for zero-day phishing email filtering. The outcomes delivered by MHS are also compared with the results obtained by other machine learning classifiers.

The second main contribution of this paper is its new pruning method called MHP. Ensemble pruning is an important research direction. It aims at reducing the size of large ensembles at the same time maintaining or enhancing their performance. The reduction of size of classification systems is crucial, in particular, for the development of effective classifiers required for the cyber security of mobile applications, wireless sensor networks, smartphones, and for other cyber security applications that have been actively investigated recently. According to the taxonomy given in [11], the majority of pruning algorithms belong to the following three categories: ranking, clustering, and optimization-based methods. The MHS is a novel technique combining all three approaches – ranking, clustering, and optimization based in one hybrid scheme, as described in Section 3.

Section 2 gives an overview of work on filtering of phishing emails and pruning techniques. The structure of MHS and our new MHP pruning technique are elucidated in Section 3. The study of pruning techniques for ensemble classifiers is a large and important research area. Our paper makes a novel contribution to this area by introducing the new pruning approach MHP.

Three separate datasets used in our empirical study presented in this paper are introduced in Section 4. This paper handles the more difficult problem of designing a classifier to be trained on a dataset collected during one time span for the zero-day filtering of new phishing emails that are received during another separate time span. This is why we collected three datasets corresponding to three separate time spans. They are called Datasets A, B, and C. Section 4 also explains the structural email attributes and how Dataset A is to be divided into two subsets, Subset A1 and Subset A2, for attribute extraction.

Section 5 explains how dynamic text attributes are extracted. It also elucidates the generation, training, and pruning of an ensemble of random forests (RFs) in MHS.

In Section 6, we discuss the empirical study comparing the results obtained by MHS with several other machine learning classifiers. Besides, we undertake a comprehensive empirical study of the effectiveness of various options of the MHP as an ingredient of MHS, and comparing the performance of MHP pruning with several simpler basic pruning methods: directed hill climbing ensemble pruning, ensemble pruning via individual contribution ordering (EPIC), and K-means pruning (KMP). In Section 7, conclusions are recorded.

DOI: 10.1002/cpe

2. OVERVIEW OF PREVIOUS WORK

Let us refer to the survey papers [7, 12–15] and recent papers [2, 16, 17] for comprehensive bibliography of previous work on phishing email filtering. Here, we add a concise summary of other recent publications.

The applications of honeypots in cyber security are well known in the literature. For instance, an efficient anti-phishing framework based on honeypots was offered in [18]. A robust technique against phishing was proposed in [19]. An application of the latent Dirichlet allocation and conditional random field for email filtering was explored in [20]. In [17], an approach for profiling phishing activities employing a two-layer clustering procedure was developed. A large iterative construction of hierarchical classifiers for the detection of phishing websites was proposed in [21]. Random forest was applied to the classification of phishing emails in [22] achieving 98.45% accuracy in 10-fold cross validation for training set collected in the same time as validate set.

Ensemble pruning is a well known area important for the cyber security of smartphones, dynamic and distributed web services, wireless sensor networks, and other cyber security applications. This paper demonstrates the feasibility and effectiveness of MHP for phishing email filtering and compares it with several other pruning techniques. We compare the results of MHP with the optimization algorithm known as the directed hill climbing ensemble pruning (DHCEP), the ranking algorithm called the EPIC, and the K-means pruning (KMP). DHCEP is a classical method treated, for instance, in [11]. It starts with an empty ensemble and then iteratively adds to each subensemble the next classifier that makes the best improvement. EPIC was proposed in [23]. It introduces a heuristic that combines accuracy, diversity, and predictions in the minority group to evaluate contribution of classifiers. The classifiers are selected in decreasing order of the heuristic. KMP was introduced and investigated in [24] and [25]. It applies K-means clustering, and from each cluster, it selects the classifier most is as far away from all other clusters as possible.

Many ensemble classifiers have been developed to solve various cyber security challenges. Ensemble pruning algorithms can reduce the size of these ensembles while at the same time enhancing their effectiveness. We refer to the survey [11] for an overview of previous studies dealing with ensemble pruning methods and for taxonomy of to pruning approaches. Here, we discuss a few recent papers on pruning of ensembles. For example, an ensemble pruning algorithm utilizing frequent patterns and applying Boolean matrices was studied in [26]. A modified backtracking algorithm for ensemble was explored in [27]. The efficiency of a music inspired algorithm called Harmony search was studied in [28]. The paper [29] explored the ability of Harmony search to prune parallel ensembles for malware detection. In [30], a genetic algorithm was used as a part of ensemble pruning algorithm. A margin-based ordered aggregation was applied to prune ensembles in [31]. A competitive ensemble pruning utilizing cross-validation was introduced in [32]. Pruning techniques for ensembles handling both labeled and unlabeled data were studied in [33].

In [34], a new pruning algorithm was based on ballot and greedy randomized strategy. The paper [35] treated a ranking-based ensemble pruning using four data partitioning algorithms in the area of text categorization. Furthermore, [36] examined a new evolutionary-based pruning method. Reinforcement learning was applied in [37] to prune ensembles of classifiers. The paper [38] proposed a new metric for greedy ensemble pruning. A new measure to guide the directed hill climbing ensemble pruning was examined in [39]. A double algorithm for pruning was studied in [40]. The paper [41] examined an instance-based statistical pruning in a Bayesian framework. A pattern mining algorithm for pruning ensembles was applied in [42].

3. DESCRIPTION OF STRUCTURE OF THE MULTILAYER HYBRID STRATEGY STRATEGY AND MULTILAYER HYBRID PRUNING PRUNING

This section describes the main structure of our new MHS for zero-day filtering of phishing emails. MHS comprises 10 layers, which are illustrated in Figure 1, and are explained in the succeeding text. Two layers of the MHS strategy are each subdivided into three sublayers of our new MHP pruning procedure.

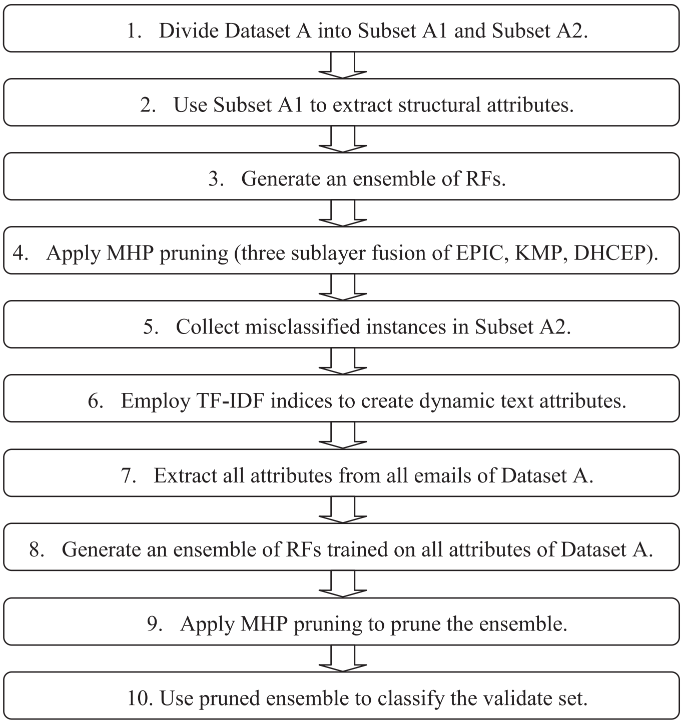
1. Divide Dataset A into Subset A1 and Subset A2.
2. Use Subset A1 to extract structural attributes.
3. Generate an ensemble of RFs.
4. Apply MHP pruning (three sublayer fusion of EPIC, KMP, DHCEP).
5. Collect misclassified instances in Subset A2.
6. Employ TF-IDF indices to create dynamic text attributes.
7. Extract all attributes from all emails of Dataset A.
8. Generate an ensemble of RFs trained on all attributes of Dataset A.
9. Apply MHP pruning to prune the ensemble.
10. Use pruned ensemble to classify the validate set.

Figure 1. Ten layers of the multilayer hybrid strategy strategy incorporating three sublayers of multilayer hybrid pruning (MHP) pruning.

Layer 1. Divide the training dataset (Dataset A) of phishing emails into two subsets collected during two separate time spans: Subset A1 and Subset A2.
Layer 2. Use Subset A1 collected during earlier time subspan to extract a fixed set of structural attributes of these emails.
Layer 3. Apply bagging to generate an ensemble of RFs employing the set of structural attributes extracted from Subset A1 of the training set Dataset A.
Layer 4. Apply MHP pruning to prune the bagging ensemble of RFs.
Layer 5. Employ the first pruned ensemble to collect all misclassified instances in Subset A2 of the training set.
Layer 6. Employ TF-IDF indices of all misclassified emails in Subset A2 to create a family of dynamic text attributes.
Layer 7. Combine fixed structural attributes and dynamic text attributes. Extract all these attributes from all emails of Dataset A.
Layer 8. Apply bagging to generate an improved ensemble of RFs trained on all structural and dynamic text attributes of Dataset A.
Layer 9. Apply MHP pruning to prune the bagging ensemble to improve its performance. (Our empirical study presented in this paper investigates several options applied in this layer in the design of our new pruning algorithm for this task and enables us to choose the best option.)
Layer 10. Use the final pruned ensemble to classify the validate set. (We perform two tests where Datasets B and C are used as the validate set, respectively.)

This paper concentrates on the effectiveness of our new MHS strategy for zero-day filtering of phishing emails using hybrid attribute selection and advanced ensemble classifier enhanced by a novel pruning technique and trained on the emails collected during one time span for application in a new separate time span. A general inspiration for our multilayer strategy came from the effectiveness of different previous multilayer approaches considered in [1, 2, 43–45]. More details pertaining to the 10 layers of MHS are given later.

Next, we describe the MHP pruning, which can be regarded as another main contribution of this paper. It combines together the ranking, clustering, and optimization pruning approaches in one unified strategy. MHP is used during Layers 4 and 9 of the MHS. The MHP pruning brings three sublayers to the whole MHS strategy: Sublayers 1, 2, and 3.

Sublayer 1 (first sublayer of MHP in MHS) uses EPIC ranking heuristic to rank all instances of the classifiers in the ensemble. The EPIC ranking heuristic combines accuracy, diversity, and predictions in the minority group to rank individual contribution of classifiers. Then, MHP selects a percentage of the higher ranked classifiers for further steps. To compute the EPIC heuristic, we selected a set of 50 emails in the training set. This set is called a *pruning set*. It is used to compute a vector representation of all the classifiers represented as binary vectors.

Sublayer 2 (second sublayer of MHP in MHS) applies the same clustering algorithm used in K-means clustering to obtain clusterings of the remaining classifiers in the same way as the KMP pruning does. This means that we apply the classical K-means to the binary vector representation of the classifiers given by the pruning set as mentioned earlier. In our implementation, Sublayer 2 uses SimpleKMeans available in WEKA [46]. It is an open source Java implementation of the well-known K-means algorithm for clustering data. It is explained in the monograph [47].

To work out the number of clusters, Sublayer 2 invokes the Silhouette index, which is a very well known measure of the quality of the groupings defined in [48]. Sublayer 2 begins with two clusters and then proceeds increasing the number of clusters as long as the Silhouette index continues improving. The next sublayer of MHP receives the clustering with the best value of the Silhouette index for further processing of the clusters.

Sublayer 3 (third sublayer of MHP in MHS) uses the greedy search strategy of DHCEP optimization within each of the formed clusters separately to choose classifiers for the final ensemble. For each cluster, DHCEP adds to each subensemble the next classifier that belongs to the same cluster and makes the most significant improvement to the whole ensemble.

As input parameters, MHP pruning accepts the required overall pruning percentage and the ratio of the number of classifiers pruned in Sublayer 1 to the number of classifiers pruned in Sublayer 3. In this way, the ideas of all three major pruning approaches – ranking, clustering, and optimization – are intricately combined to create an aggregated novel method.

4. THREE SEPARATE DATASETS: DATASET A, DATASET B, AND DATASET C

We use three datasets of emails collected during three separate time spans illustrated in Table I. These datasets were collected during three time spans: from March 1 to July 31, 2015, from August 1 to 31, 2015, and from September 1 to 30, 2015, respectively. Separate datasets were necessary to study the problem of zero-day filtering of phishing emails during separate time spans, which are the major question explored in this paper. The phishing emails were collected by members of our laboratory and industry partners using honeypots (Li and Schmitz, 2009) and email addresses displayed on private and work websites and posted on social networking sites. The legitimate emails, also called ham emails, were also collected by the members during the same time spans. The empirical study presented in this paper required three datasets to enable us to perform two large tests. Both of them use Dataset A as a training set. The first experiment used Dataset B as a validate set, and the second experiment used Dataset C as a validate set.

Table I. Datasets of phishing emails collected during three separate time spans.

	Dataset A	Dataset B	Dataset C	
Time spans	1 March to 31 July 2015	1 August to 31 August 2015	1 September to 30 September 2015	Total
Phishing emails	367	65	74	506
Ham emails	721	123	142	986
Total	1088	188	216	1492

DOI: 10.1002/cpe

Next, we discuss how Dataset A is used to extract structural attributes in Layer 2 of MHS. We begin by dividing the training dataset into two subsets. Subset A1 contained 653 emails collected during March, April, and May of 2015. Subset A2 contained 435 emails collected during June and July of 2015. Subset A1 was used to select the set of attributes based on the structure of the emails. Next, here, we present a list of these structural attributes.

- The number of images incorporated in the email.
- The number of characters in the text of the email.
- Whether html is used and the size of the html body of the email.
- The inclusion of a greeting and a signature in the text.
- The number of tables in the email.
- The number of hyperlinks included in the email.
- The number of forms included in the email.
- Inclusion of hidden text in the email.
- The email contains a link to IP address without any domain name service record.
- Number of spoofed html anchors where displayed text is different from the actual link.
- The email contains a link to a URL including the sign.
- The number of double slashes in URLs in the links included in the email.
- The number of dots within the URL addresses in links included in the email.
- An IP address is used in the email in the hostname part of the URL address instead of domain name.
- Email includes a link to a URL with domain name in the path of the URL.
- The top-level domain or country code of a link occurs more than once in the domain.
- The number of URLs included in the email with the use of hexadecimal representation of an IP address.
- The number of URLs included in the email with the use of Unicode representation.
- The email contains a link to URL with added prefixes or suffixes.
- The maximum number of sub-domains within an URL contained in the email.
- The email contains a link to URL where the port number part of a domain name does not match the protocol.
- The rank of domain name in the URL included in the email in the Google Toolbar PageRank.
- The age of domain name in the URL included in the email in a WHOIS search.
- The number of scripts included in the email.

We used Python programming to parse and analyze email messages to extract structural attributes. Notice that there are also other types of effective attributes well known in the zero-day filtering of emails, like those based on global black lists, local white lists, or grey lists. Lists of this type are broadly used and are quite effective in email filtering [49]. We do not include user generated white lists, grey lists, and global blacklists in our strategy, because their use in the empirical study presented in this paper may obscure the role of the contribution of other parts of the strategy, and besides, they rely on the user activity, which cannot be controlled by developing data mining methods.

5. ENSEMBLE OF RANDOM FORESTS AND DYNAMIC TEXT ATTRIBUTES

We refer to [47] for information on bagging and RF, which is available in WEKA [46]. In Layer 3, MHS uses bagging and structural attributes extracted from Subset A1 of Dataset A to generate and train a bagging ensemble of RFs. Layer 4 employs MHP pruning to improve its performance by pruning. Layer 5 applies the resulting pruned ensemble to Subset A2 of Dataset A and collects all misclassified instances in Subset A2.

Layer 6 uses these misclassified emails of Subset A2 to extract dynamic text attributes that reflect the email content for all the emails misclassified by the first version of the pruned ensemble. It applies the bag-of-words model and extracts an additional family of attributes. Term frequency-inverse document frequency (TF-IDF) word indices are utilized to pick out words as new dynamic attributes. These indices are well known in the literature (for example, [50, 51]). To compute TF-IDF indices, we used Gensim, a Python package for vector space modeling of texts.

If we denote by E the number of misclassified phishing emails, then the TF-IDF indices are defined as follows. For an email e and keyword k and, let $N(k, e)$ denotes the number of occurrences of k in e. Assuming that a family $F = \{k_1, k_2, \ldots, k_m\}$ of keywords $k_1, k_2, \ldots, k_m$ has been chosen, then the following symbols are then defined. The symbol $\mathrm{TF}(k, e)$ stands for the term frequency of keyword k from F in a email e:

$$\mathrm{TF}(k, e) = \frac{N(k, e)}{\sum_{i=1}^{m} N(k_i, e)} \tag{1}$$

The symbol $\mathrm{DF}(k)$ stands for the document frequency of the keyword k, which is equal to the number of emails containing k. To measure the significance of each keyword k, the inverse document frequency $\mathrm{IDF}(k)$ is used:

$$\mathrm{IDF}(k) = \log\left(\frac{|E|}{\mathrm{DF}(k)}\right) \tag{2}$$

Next, we define The TF-IDF index of keyword k in email e:

$$\text{TF-IDF}(k, e) = \mathrm{TF}(k, e) \times \mathrm{IDF}(k, e) \tag{3}$$

For each of the misclassified phishing emails, we determined a collection of 12 keywords with largest TF-IDF indices. Then, we compiled a ranked list of all of these keywords in all of the misclassified phishing emails collected during Layer 5. For each misclassified phishing email, we extracted six words with the highest TF-IDF indices. We compiled a ranked list of all of these keywords. To take into account the number of emails a word came from, and the value of its TF-IDF indices, if a word was present in several emails, we calculated the sum of its TF-IDF indices on all of these misclassified emails. Keywords with higher ranking are more significant for correcting future classifications. This creates a ranked collection of words with highest sums of TF-IDF indices in the whole dataset.

After that, in Layer 6, for each email of the whole Dataset A, we selected 12 dynamic text attributes. We determined 12 keywords with highest TF-IDF indices in this email, and for each of these keywords as a new dynamic attribute, we used its rank in the compiled ranked list explained in the preceding paragraph. To this end, we verified whether the keyword belongs to the compiled ranked list of words found in all the misclassified emails, and if so, then we included its rank in the ranked list as a new dynamic text attribute corresponding to this keyword. Thus, if all 12 words with the highest TF-IDF indices in an email do not belong to the compiled ranked list made up by keywords of the misclassified emails, then all 12 dynamic text attributes of the email are equal to zero.

Layer 7 combines all structural attributes and 12 new dynamic text attributes for all emails in the whole Dataset A. In Layer 8, bagging is used to generate a final bagging ensemble, trained on all structural and additional dynamic text attributes of the whole training set. It is generated in the same way as explained earlier, but now, we use the whole Dataset A and all structural attributes combined with the additional dynamic text attributes for each email.

Layer 9 uses MHP pruning as explained earlier to prune the bagging ensemble.

Layer 10 applies the second pruned bagging ensemble of RFs to the validate set collected during a separate time span to determine the effectiveness of MHS for zero-day filtering of phishing emails. Our empirical study used two datasets, Datasets B and C as validate sets.

6. EMPIRICAL STUDY

Precision and recall are very well known parameters evaluating the effectiveness of classifiers [47]. In this paper, we employed the F-measure (FM) [47], because it amalgamates both precision (P) and recall (R) into a combined convenient metric defined by

$$\mathrm{FM} = \frac{2 \times \mathrm{P} \times \mathrm{R}}{\mathrm{P} + \mathrm{R}}. \tag{4}$$

In WEKA, the standard output for all classifiers includes the weighted average F-measure evaluated using 10-fold cross validation.

We begin our empirical study presented in this paper by testing the MHS strategy for various values of input parameters employed in the Layers 4 and 9. These input parameters are the required overall pruning percentage and the ratio of the number of classifiers pruned in Sublayer 1 of MHP to the number of classifiers pruned in Sublayer 3 of MHP. The results of these tests for Dataset B as validate set are shown in Table II. It shows that the best performance was achieved when MHS prunes 40% of the total number of classifiers with the ratio equal to 6/4 = 3/2.

Our next diagram represents the results of experiment comparing what happens if we replace MHP with simpler basic pruning algorithms in the Layers 4 and 9 of MHS. It compares MHP with the use of several different pruning algorithms within MHS strategy. It includes the application of MHP pruning in Layers 4 and 9 of MHS, compared with the application of DHCEP in Layers 4 and 9, the use of utilizing EPIC, and finally, the application of KMP in Layers 4 and 9 of MHS.

In Figure 2, we included the outcomes of MHS achieved with the best option of input parameters found in Table II. We see that the best performance was achieved by MHS strategy employing MHP.

Next, we compare the performance of MHS in filtering phishing emails with several other classifiers available in WEKA [46]. Here, we look at BayesNet (BN), J48, kNN, RF, and SMO. The readers are referred to the monograph [47] for background information on these algorithms.

BayesNet is a WEKA implementation of Bayesian belief networks that can use an array of search algorithms. Our empirical study incorporates comparison of the performance of BN utilizing five

Table II. Performance of the multilayer hybrid strategy algorithm for various input parameters.

Percentage of pruning	Ratio of the number of classifiers pruned in Sublayer 1 to the number of classifiers pruned in Sublayer 3								
	1/9	2/8	3/7	4/6	5/5	6/4	7/3	8/2	9/1
90	0.90	0.90	0.91	0.92	0.92	0.92	0.92	0.91	0.90
80	0.90	0.92	0.93	0.93	0.94	0.94	0.94	0.93	0.92
70	0.91	0.92	0.94	0.95	0.95	0.96	0.95	0.94	0.92
60	0.92	0.93	0.95	0.96	0.96	0.97	0.96	0.95	0.94
50	0.92	0.94	0.95	0.96	0.97	0.97	0.97	0.96	0.94
40	0.92	0.94	0.96	0.97	0.97	**0.98**	0.97	0.96	0.94
30	0.93	0.94	0.96	0.97	0.97	0.97	0.97	0.96	0.95
20	0.92	0.94	0.95	0.96	0.97	0.97	0.97	0.95	0.95
10	0.92	0.93	0.94	0.95	0.96	0.96	0.96	0.95	0.95

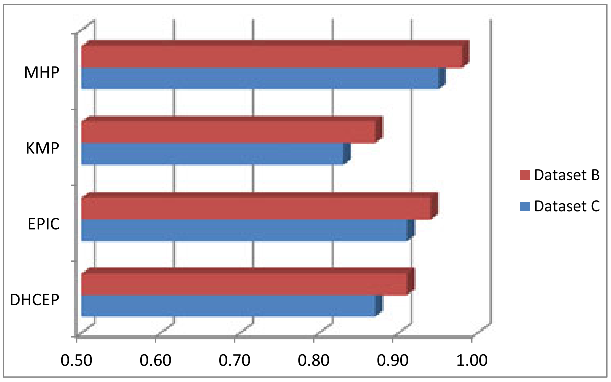

Figure 2. *F*-measure obtained by multilayer hybrid strategy employing multilayer hybrid pruning (MHP) and by several simplified versions of the filtering strategy invoking other pruning algorithms in Layers 4 and 9 instead of MHP. Dataset A is used as training set and Datasets B and C as validate sets.

DOI: 10.1002/cpe

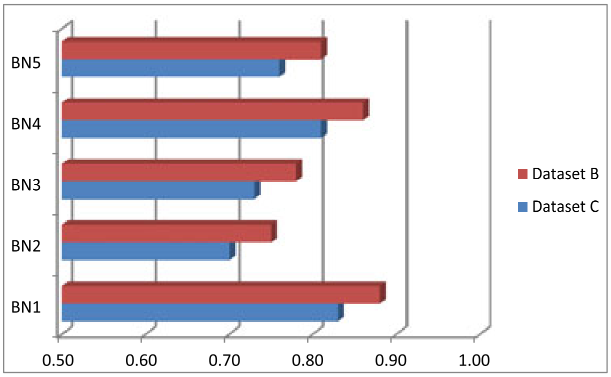

Figure 3. *F*-measure obtained by several search algorithms of BayesNet for Dataset A as training set and Datasets B and C as validate sets.

search algorithms. To present the results obtained by these algorithms and include them in the subsequent diagram, we use the following notation for the corresponding versions of BN:

BN1 stands for the BN algorithm using the genetic search,
BN2 is the BN algorithm utilizing the Hill climber search,
BN3 is the BN algorithm applying the K2 search,
BN4 is the BN algorithm using the simulated annealing,
BN5 is the BN algorithm applying the tabu search.

Empirical study presented in this paper compares the effectiveness of all these kernels. Figure 3 presents the results obtained in filtering phishing emails by these kernels of BN for Dataset A of Table I as training set and Datasets B and C of Table I as validate sets.

SMO is a WEKA implementation of the sequential minimal optimization algorithm. It is a fast version of support vector machine, which form an important class [52, 53]. SMO can be invoked in WEKA with four kernels. Our tests compare the performance of all four available kernels. To present these results, we use the following brief notation.

SMO1 stands for the SMO algorithm using the radial basis function kernel given by

$$\mathrm{RadialBasisFunctionKernel}(e_1, e_2) = e^{(-\gamma \times \langle e_1 - e_2, e_1 - e_2 \rangle^2)} \tag{5}$$

SMO2 stands for the SMO algorithm utilizing the polynomial kernel given by

$$\mathrm{PolyKernel}(e_1, e_2) = (\langle e_1, e_2 \rangle + 1)^p \text{ or } \langle e_1, e_2 \rangle^p \tag{6}$$

SMO3 stands for the SMO algorithm employing the normalized polynomial kernel given by

$$\mathrm{NormalizedPolyKernel}(e_1, e_2) = \frac{\mathrm{PolyKernel}(e_1, e_2)}{\sqrt{\mathrm{PolyKernel}(e_1, e_1) \cdot \mathrm{PolyKernel}(e_2, e_2)}} \tag{7}$$

SMO4 stands for the SMO algorithm utilizing the Pearson VII function kernel. Empirical study presented in this paper compares the performance of SMO with all available kernels. Figure 4 compares the results obtained in filtering phishing emails by these four kernels of SMO for Dataset A in Table I as training set and Datasets B and C of Table I as validate sets.

Finally, Figure 5 compares the results obtained in filtering phishing emails by MHP and several other classifiers for Dataset A in Table I as training set and Datasets B and C of Table I as validate sets. In Figure 5, we included the outcomes of MHS achieved with the best option of input parameters found in Table II, the best outcomes of BN from Figure 3 and the best outcomes of SMO from Figure 4. Figure 5 shows that the best performance was achieved by MHS strategy.

DOI: 10.1002/cpe

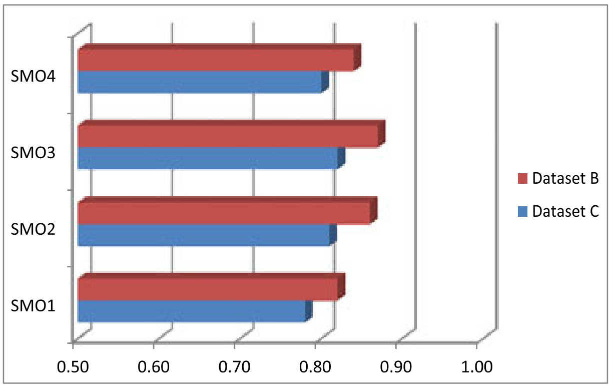

Figure 4. F-measure obtained by several kernels of SMO for Dataset A as training set and Datasets B and C as validate sets.

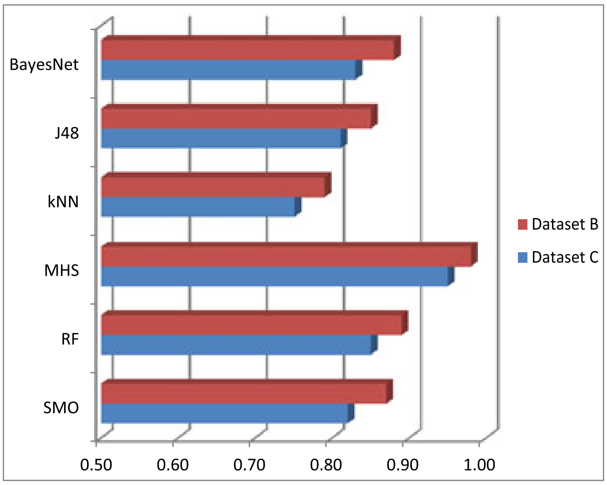

Figure 5. F-measure obtained by multilayer hybrid strategy and other classifiers for Dataset A as training set and Datasets B and C as validate sets.

7. CONCLUSIONS

This paper introduces a new MHS and presents the results of empirical study investigating it for a novel application of phishing email filtering during new time span based on training available only in a separate previous time span. MHS is based on a new pruning method MHP introduced in this paper. The empirical study presented in this paper shows that MHS is effective and produced the best outcomes with F-measure of 0.98%. MHP performed better than other pruning methods in two layers of MHS. The results also show that the accuracy of filtering decreases for the more distant time span, because phishing email creators can vary their tools. This means that MHS has to be retrained as new data become available. Because the F-measure of the performance for Dataset B as validate set is high enough for practical filtering, the outcomes show that it suffices to retrain the ensemble classifier after 1 month, which is easy to implement in practice. An interesting open question for future research is to develop and study a mathematical model of the MHS strategy.

ACKNOWLEDGEMENTS

The authors are grateful to three reviewers for thorough reports that have helped to improve this paper.

DOI: 10.1002/cpe

REFERENCES

1. Liu T, Guan X, Qu Y, Sun Y. A layered classification for malicious function identification and malware detection. *Concurrency and Computation: Practice and Experience* 2012; **24**:1169–1179.
2. Islam R, Abawajy J. A multi-tier phishing detection and filtering approach. *Journal of Network and Computer Applications* 2013; **36**:324–335.
3. Ezzati-Jivan N, Dagenais MR. Cube data model for multilevel statistics computation of live execution traces. *Concurrency and Computation: Practice and Experience* 2015; **27**:1069–1091.
4. Miao X, Jin X, Ding J. A new hybrid solver with two-level parallel computing for large-scale structural analysis. *Concurrency and Computation: Practice and Experience* 2015; **27**:3661–3675.
5. APWG. Phishing activity trends report, 2015. (Available from: http://www.antiphishing.org/resources/apwg-reports/), [Accessed on 21 October 2015].
6. Alsharnouby M, Alaca F, Chiasson S. Why phishing still works: user strategies for combating phishing attacks. *International Journal of Human-Computer Studies* 2015; **82**:69–82.
7. Zeydan HZ, Selamat A, Sallehm M. Survey of anti-phishing tools with detection capabilities. *Proceedings of the 2014 International Symposium on Biometrics and Security Technologies, ISBAST*, Kuala Lumpur, Malaysia, 2014a; 2014–2019.
8. Alazab M, Venkatraman S, Watters P, Alazab M. Zero-day malware detection based on supervised learning algorithms of API call signatures. *Data Mining and Analytics 2011, Proceedings of the Ninth Australasian Data Mining Conference, AusDM2011, CRPIT*, Vol. 121, Ballarat, Australia, 2011; 171–182.
9. Islam R, Tian R, Moonsamy V, Batten L. A comparison of the classification of disparate malware collected in different time periods. *Journal of Networks* 2012; **7**:956–955.
10. Islam R, Altas I, Islam MS. Exploring timeline-based malware classification. *Proceedings of the 28th IFIP TC International Conf. Security and Privacy Protection in Information Processing Systems, SEC 2013, IFIP Advances in Information and Communication Technology*, Vol. 405, Auckland, New Zealand, 2013; 1–13.
11. Tsoumakas G, Partalas I, Vlahavas I. An ensemble pruning primer. *Applications of Supervised and Unsupervised Ensemble Methods, Studies in Computational Intelligence*, Vol. 245, Springer, Verlag, 2009; 1–13.
12. Almomani A, Wan TC, Manasrah A, Altaher A, Almomani E, Al-Saedi K, Alnajjar A, Ramadass S. A survey of learning based techniques of phishing email filtering. *International Journal of Digital Content Technology and its Applications* 2012; **6**:119–129.
13. Almomani A, Gupta BB, Atawneh S, Meulenberg A, Almomani E. A survey of phishing email filtering techniques. *IEEE Communications Surveys & Tutorials* 2013; **15**:2070–2090.
14. Khonji M, Iraqi Y, Jones A. Phishing detection: a literature survey. *IEEE Communications Surveys & Tutorials* 2013; **15**:2091–2121.
15. Zeydan HZ, Selamat A, Sallehm M. Current state of anti-phishing approaches and revealing competencies. *Journal of Theoretical and Applied Information Technology* 2014b; **70**:507–515.
16. Hamid IRA, Abawajy J. Hybrid feature selection for phishing email detection. *International Conference on Algorithms and Architectures for Parallel Processing, ICA3PP 2011, LNCS*, Vol. 7017, Melbourne, Australia, 2011; 266–275.
17. Hamid IRA, Abawajy JH. An approach for profiling phishing activities. *Computers & Security* 2014; **45**:27–41.
18. Li S, Schmitz R. A novel anti-phishing framework based on honeypots. *Proceedings of the eCrime Researchers SummiteCRIME'09*, Tacoma, WA, USA, 2009; 1–13.
19. Barraclough PA, Hossain MA, Tahir MA, Sexton G, Aslam N. Intelligent phishing detection and protection scheme for online transactions. *Expert Systems with Applications* 2013; **40**:4697–4706.
20. Ramanathan V, Wechsler H. Phishing detection and impersonated entity discovery using conditional random field and latent Dirichlet allocation. *Computers & Security* 2013; **34**:123–139.
21. Abawajy J, Beliakov G, Kelarev A, Chowdhury M. Iterative construction of hierarchical classifiers for phishing website detection. *Journal of Networks* 2014; **9**:2089–2098.
22. Akinyelu AA, Adewumi AO. Classification of phishing email using random forest machine learning technique. *Journal of Applied Mathematics* 2014:1–6. Article ID 425731.
23. Lu Z, Wu X, Zhu X, Bongard J. Ensemble pruning via individual contribution ordering. *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2010*, Washington, DC, USA, 2010; 871–880.
24. Giacinto G, Roli F, Fumera G. Design of effective multiple classifier systems by clustering of classifiers. *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, Spain, 2000; 160–163.
25. Lazarevic A, Obradovic Z. Effective pruning of neural network classifier ensembles. *Proceedings if the 2001 IEEE/INNS International Joint Conference on Neural Networks*, Washington, DC, USA, 2001; 796–801.
26. Zhou H, Zhao X, Wang X. An effective ensemble pruning algorithm based on frequent patterns. *Knowledge-Based Systems* 2014; **56**:79–85.
27. Dai Q, Liu Z. ModEnPBT: a modified backtracking ensemble pruning algorithm. *Applied Soft Computing* 2013; **13**:4292–4302.
28. Sheen S, Aishwarya SV, Anitha R, Raghavan SV, Bhaskar SM. Ensemble pruning using harmony search. *LNAI* 2012; **7209**:13–24.
29. Sheen S, Anitha R, Sirisha P. Malware detection by pruning of parallel ensembles using harmony search. *Pattern Recognition Letters* 2013; **34**:1679–1686.

DOI: 10.1002/cpe

30. Abdi L, Hashemi S. GAB-EPA: a GA based ensemble pruning approach to tackle multiclass imbalanced problems. *LNAI* 2013; **7802**:246–254.
31. Guo L, Boukir S. Margin-based ordered aggregation for ensemble pruning. *Pattern Recognition Letters* 2013; **34**:603–609.
32. Dai Q. An efficient ensemble pruning algorithm using one-path and two-trips searching approach. *Knowledge-Based Systems* 2013; **51**:85–92.
33. Zhang G, Zhang S, Wang C, Cheng L. Ensemble pruning for data dependent learners. *Applied Mechanics and Materials* 2012; **135-136**:522–527.
34. Dai Q. A novel ensemble pruning algorithm based on randomized greedy selective strategy and ballot. *Neurocomputing* 2013; **122**:258–265.
35. Toraman C, Can F. Squeezing the ensemble pruning: faster and more accurate categorization for news portals. *LNCS* 2012; **7224**:508–511.
36. Bhowan U, Johnston M, Zhang M. Ensemble learning and pruning in multi-objective genetic programming for classification with unbalanced data. *AI 2011: Advances in Artificial Intelligence, 24th Australasian Joint Conference on Artificial Intelligence, LNAI* Wang D, Reynolds M (eds), Vol. 7106, Perth, Australia, 2011; 192–202.
37. Partalas I, Tsoumakas G, Vlahavas I. Pruning an ensemble of classifiers via reinforcement learning. *Neurocomputing* 2009; **72**:1900–1909.
38. Guo H, Zhi W, Han X, Fan M. A new metric for greedy ensemble pruning. *LNAI* 2011; **7003**:631–639.
39. Partalas I, Tsoumakas G, Vlahavas I. An ensemble uncertainty aware measure for directed hill climbing ensemble pruning. *Machine Learning* 2010; **81**:257–282.
40. Soto V, Martinez-Munoz G, Hernandez-Lobato D, Suarez A. A double pruning algorithm for classification ensembles. *LNCS* 2010; **5997**:104–113.
41. Hernandez-Lobato D, Martinez-Munoz G. Statistical instance-based pruning in ensembles of independent classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2009; **31**:364–369.
42. Zhao QL, Jiang YH, Xu M. A fast ensemble pruning algorithm based on pattern mining process. *Data Mining and Knowledge Discovery* 2009; **19**:277–292.
43. Islam R, Abawajy J, Warren M. Multi-tier phishing email classification with an impact of classifier rescheduling. *Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks,* 2009; 789–793.
44. Islam R, Zhou W, Chowdhury MU. Email categorization using (2+1)-tier classification algorithms. *Proceedings – 7th IEEE/ACIS International Conference on Computer and Information Science, IEEE/ACIS ICIS 2008, In conjunction with 2nd IEEE/ACIS Int. Workshop on e-Activity, IEEE/ACIS IWEA 2008*, Portland, OR, USA, 2008; 276–281.
45. Islam R, Zhou W, Gao M, Xiang Y. An innovative analyser for multi-classifier email classification based on grey list analysis. *Journal of Network and Computer Applications* 2009; **32**:357–366.
46. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software: an update. SIGKDD. *Explorations* 2009; **11**:10–18.
47. Witten IH, Frank E, Hall MA. *Data Mining: Practical Machine Learning Tools and Techniques* (3rd edn.) Elsevier/Morgan Kaufman: Amsterdam, 2011.
48. Rousseeuw P. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational & Applied Mathematics* 1987; **20**:53–65.
49. Islam R, Zhou W, Chowdhury M. Minimizing the drawbacks of grey list analyser of synthesis based spam filtering. *Journal of Electronics and Computer Science* 2009; **11**:89–96.
50. Yearwood J, Webb D, Ma L, Vamplew P, Ofoghi B, Kelarev A, Data Mining and Analytics 2009 Proc. 8th Australasian Data Mining Conference. Applying clustering and ensemble clustering approaches to phishing profiling Kennedy PJ, Ong K, Christen P (eds), Vol. 101, ACS, Melbourne, Australia, 2009; 25–34.
51. Peng T, Liu L, Zuo W. PU text classification enhanced by term frequency-inverse document frequency-improved weighting. *Concurrency and Computation: Practice and Experience* 2014; **26**:728–741.
52. Huda S, Abawajy J, Alazab M, Abdollalihian M, Islam R, Yearwood J. Hybrids of support vector machine wrapper and filter based framework for malware detection. *Future Generation Computer Systems* 2016; **55**:376–390.
53. Villar-Rodriguez E, Del Ser J, Torre-Bastida AI, Bilbao MN, Salcedo-Sanz S. A novel machine learning approach to the detection of identity theft in social networks based on emulated attack instances and support vector machines. *Concurrency Computat.: Pract. Exper* 2015; **27**. DOI: 10.1002/cpe.3633.