Volume 3, Issue 6, June 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Ensemble Clustering for Internet Security Applications

H.T Tanoj kumar , B.N.RamaChandra
M.Techt,1st year Dept of CSE,CEC
Mangalore, India

Verdine Saviola Noronha
Assistant Professor, Dept of ISE,CEC
Mangalore, India

Abstract-Malware and phishing website detection has been the Internet security topics that are now a day has great interests. Compared with malware attacks, phishing website fraud is a relatively new Internet crime. Over the past few years, many clustering techniques have been employed for automatic malware and phishing website detection. In these techniques, the detection process is generally divided into two steps: 1) feature extraction, where representative features are extracted to capture the characteristics of the file samples or the websites; and 2) categorization, where intelligent techniques are used to automatically group the file samples or websites into different classes based on computational analysis of the feature representations. This paper describes about automatic categorization system to automatically group phishing websites and malware samples using a cluster ensemble by aggregating the clustering solutions that are generated by different base clustering algorithms, and a principled cluster ensemble framework to combine individual clustering solutions that are based on the consensus partition, which can not only be applied for malware categorization, but also for phishing website clustering.

Index Terms—Cluster ensemble, malware categorization, phishing website detection.

I. INTRODUCTION

Malware such as virus, worms, Trojan Horses, spyware, backdoors, and root kits has presented a serious threat to the security of computer systems. Currently, the most significant line of defense against malware is Internet security software products, which mainly use a signature-based method to recognize threats in the Clients. Given a collection of malware samples, these venders first categorize the samples into families so that samples in the Same family shares some common traits, and generates the common string(s) to detect variants of a family of malware samples. Phishing Website Detection: Compared with malware attack, phishing website fraud is a relatively new Internet crime. Phishing is a form of online fraud, whereby perpetrators adopt social engineering schemes by sending e-mails, instant messages, or online advertising to allure users to phishing websites that impersonate trustworthy websites in order to trick individuals into revealing their sensitive information (e.g., financial accounts, passwords, and personal identification numbers) which can then be used for profit. To defend against phishing websites, security software products generally use blacklisting to filter against known websites. However, there is always a delay between website reporting and blacklist updating. Indeed, as lifetimes of phishing websites are reduced to hours from days, this method might be ineffective. The number of new phishing websites that are collected by the Antivirus Laboratory of Kingsoft is usually larger than 20000 per day, and the number of new malware samples with various families collected by the Antivirus Laboratory of Kingsoft is usually larger than 10000 per day. There is, thus, an urgent need of effective methods for automatic detection for these threats. Though the phishing websites and the malware samples evolve constantly, most of their essence or the inherent structure is relatively stable. For example, a family of malware samples typically exhibit similar behavior profiles. It has also been shown that phishing websites are not isolated from their targets but have strong relationships with them, which can be used as clues to cluster them into families and generate the signature for detection.

II. SYSTEM ARCHITECTURE

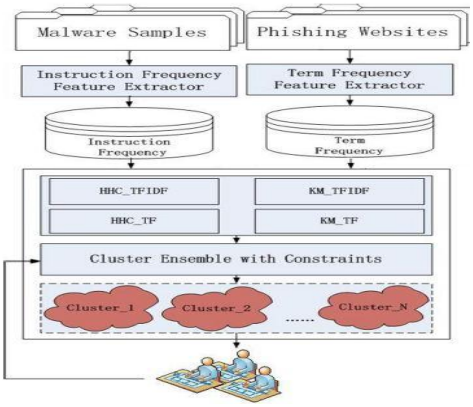


Fig1. Architecture of the ACS

Above shows the architecture of the ACS, and we briefly describe each component below.

1) *Term-frequency feature extractor***:**

For phishing website categorization, the ACS first uses the term-frequency feature extractor to extract the terms from the web pages of the collected phishing websites, and then transforms the data into term-frequency feature vectors. These vectors are stored in the database. The transaction data can also be easily converted to relational data if necessary.

2) *Instruction-frequency feature extractor*:

For malware categorization, the ACS first uses the instruction-frequency feature extractor to extract the function-based instructions from the collected Portable Executable (PE) malware samples, converts the instructions to a group of 32-bit global IDs as the features of the data collection, and stores these features in the signature database. These integer vectors are then transformed to instruction frequencies and stored in the database. The transaction data can also be easily converted to relational data if necessary.

3) *Base clustering algorithms***:**

Base clustering solutions are generated by applying different clustering algorithms that are based on the feature representations. The HC algorithm and KM partitioned approach are applied on the Term-frequency vectors or instruction-frequency vectors with the TF-IDF and TF weighting schemes, which are widely used for document representation in IR (information retrieval.

4) Cluster ensemble with constraints:

Cluster ensemble is used to combine different base clustering. The cluster ensemble is also able to utilize the domain knowledge in the form of website-level/sample-level constraints.

5) *Domain knowledge:*

Our system provides a user-friendly mechanism to incorporate the expert knowledge and expertise of human experts. Internet security experts can look at the partitions and manually generate website level/ sample-level constraints. These constraints can be used to improve the categorization performance

III. Malware Categorization And Phishing Website Detection

1) Malware Feature Extraction

Features are the characterization of the behavior of a program under analysis. They are used as the input to data mining algorithms and can be derived from different levels of abstractions, including instruction level, API level, and cross-module level. There are generally three categories of feature extraction methods: dynamic, static, and hybrid. Dynamic analysis techniques observe the execution of the malware to derive features. The execution can be on a real or virtual processor. Well-known techniques include debugging and profiling. Example tools include Valgrind, QEMU, and strace. One advantage of dynamic feature extraction is that the environment- or configuration-dependent information has been resolved during the extraction, e.g., a variable whose value depends on the hardware, system configuration, or program input.

One disadvantage of dynamic analysis is its limited coverage. Static analysis techniques analyze the malware without running it. The target of analysis can be binary or source code. Static analysis has the advantage that it can explore all possible execution paths in the malware; therefore, it can be exhaustive in detecting malicious logic. One disadvantage of static analysis is its inability to address certain situations due to undesirability, e.g., indirect control transfer through function pointers. Hybrid analysis is an approach that combines static and dynamic analysis to gain the benefits of both. In our study, built on our previous work we use the instruction-frequency feature extractor to extract the function-based instructions from the collected PE samples.

2) Malware Categorization

Various classification approaches including association classifiers, support vector machines, and Naive Bayes have been applied in malware detection. HOLMES detects malware families by combing frequent sub graph mining and concept analysis to synthesize discriminative specifications. Research efforts have been reported on combining different classification methods using different learning methods with possible different feature representations from malware detection. These classification methods require a large number of training samples to build the classification models. In recent years, there have been several initiatives in automatic malware categorization using clustering techniques. Bayer et al. used locality sensitive hashing and hierarchal clustering to efficiently group large datasets of malware samples into clusters. Lee and Mody adopted KM clustering approach to categorize the malware samples. Several efforts have also been reported on computing the similarities between different malware samples using edit distance (ED) measure or statistical tests.

3) Phishing Website Detection

Phishing website, a semantic attack which targets the user rather than the computer, is a relatively new significant security threat to the Internet in comparison with malware. Recently, many classification methods such as support vector machines and Naive Bayes have been used for anti phishing. However, to date, to the best of our knowledge, there are only limited efforts that focus on phishing website clustering for phishing prevention. Given an unknown webpage, Liu et al. proposed the following method for phishing detection: The method first finds the associated web pages with the given page, then mines the features (such as links relationship, ranking relationship, webpage text similarity, and webpage layout similarity relationship) between the given webpage and its associated web pages, and, finally, applies DBSCAN clustering algorithm to decide if there is a cluster around the given webpage. If such cluster is found, the given webpage is then regarded as a phishing webpage; otherwise, it is identified as a legitimate webpage.

Layton proposed the following framework for phishing website clustering: It first extracts the bag-of-word representation from the source of the websites and then principal component analysis (PCA) for feature selection, and, finally, uses certain clustering algorithms (such as k-means, DBSCAN) for detection. For example, the experiments of were performed based on 8745 phishing web pages and 1000 legitimate web pages, while Layton evaluated their proposed methods based on a dataset containing 24 403 websites. We believe that the further progress can be made in clustering particular sets of malware samples or sets of phishing websites. In particular, existing clustering methods usually apply a specific clustering method on a feature representation. Different clustering methods have their own advantages and limitations in malware detection. In our study, we propose a principled cluster ensemble framework to integrate different clustering solutions.

IV. Feature Representation

1) Instruction Frequencies Of Malware Samples

There are mainly two ways for feature extraction in malware analysis: static extraction and dynamic extraction. Dynamic feature extraction can well present the behaviors of malware files and especially perform well in analyzing packed malware. However, it has limited coverage.

Only Executable files can be executed or simulated. Actually, from the daily data collection of the Kingsoft Internet Security Laboratory, more than 60% of malware samples are dynamic link library files, which cannot be dynamically analyzed. In addition, dynamic feature extraction is time consuming. We use the dissembler K32Dasm which was developed by the Kingsoft Internet Security Laboratory to dissemble the PE code and output the file of decrypt or unpacked format as the input for feature extraction.

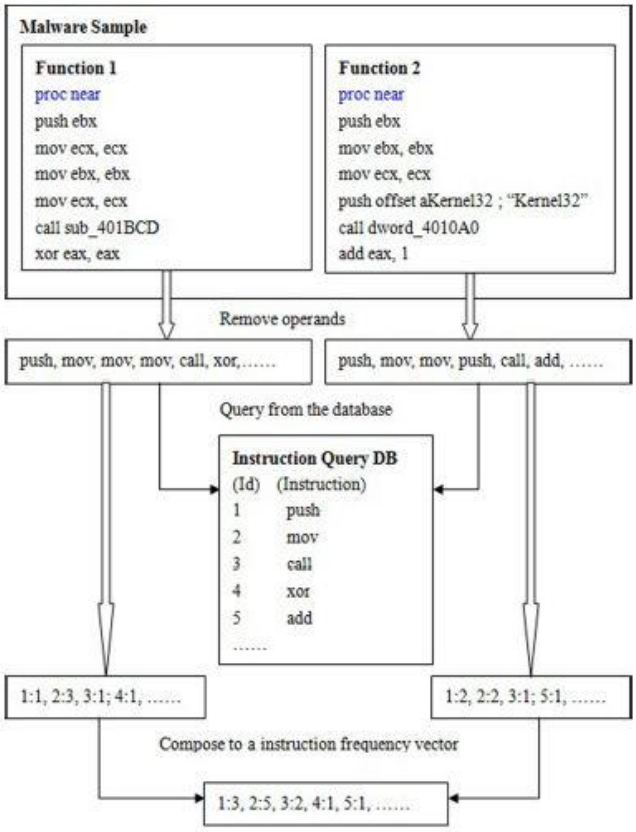
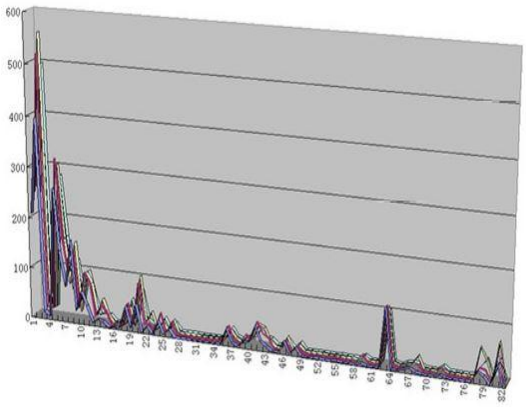


Fig2.Malware feature extraction and Transformation processes of the ACS

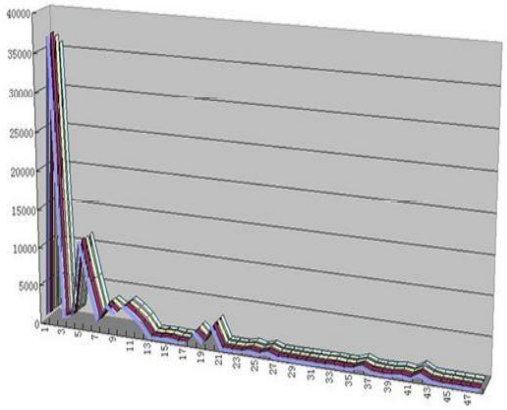
This paper uses the instruction frequencies for malware representation. The extraction and transformation processes are shown in Figure 2. Comparing with other static features, such as construction phylogeny tree, control flow graph, Windows API calls, or arbitrary binaries, the instruction frequencies and function-based instruction sequences for malware representation have great ability to represent variants of a malware family, high coverage rate of malware samples, good semantic implications, and high efficiency for feature extraction .

2) Term Frequencies Of Phishing Websites

There are several feature extraction methods for phishing website representation: URL of the website, user interface associated web pages of the website, webpage block, layout, and overall style, terms of given webpage with the TF-IDF scores, etc. Considering the expression ability of the website and the complexity for the categorization inputs, in this paper, we extract the term frequencies from the web pages of their corresponding websites. We first extract the terms from the "Title," "Keywords," "Description," "Copyright," and "Alt" of the web pages.



*TrojanQQ,dm_C2e5bff27fef9dfb5f4facac27339b56 ,
*Trojan QQ, dm_e1231626fbac92382c08cac010135287 *TrojanQQ,dm_f30e50767a370dd33eb9863faec05f36
*Trojan QQ, dm_f99114b8b693495f959b9f7a9f83e76f



*Adware.Mnless.aul_0fbfdale0ba534111aa224407f2a9c1a *Adware.Mnless.aul_23cdc79010917cb712267f1ea376602
*Adware.Mnless.aul_5877e8f9ebd5ff99063978f60829ff4b
*Adware.Mnless.aul_8109506d2e0a9340b93fc69d200c77c

Fig3. Shapes of instruction-frequency patterns are shared by the same malware family and differ between different families

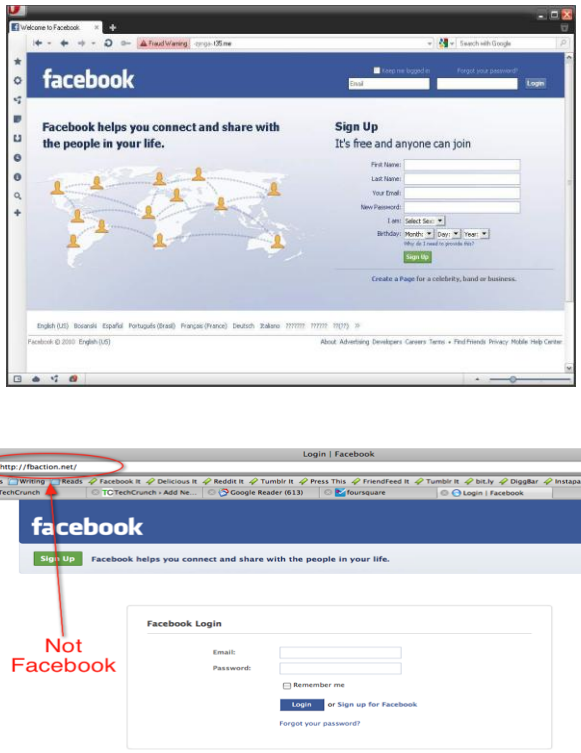


Fig4. Two phishing websites of the same family share similar term features.

The description of the extraction is illustrated as follows.

1) Title: extracting the content from the title tag of the webpage, i.e., the content between "<TITLE> . . . </TITLE>."

2) Keywords: extracting the keyword information of the website from the meta tag of the webpage, i.e., the content between "<META name=description content=. . .>."

3) Description: extracting the description information of the website from the meta tag of the webpage, i.e., the content between "<META name=keywords content=. . .>."

4) Copyright: extracting the copyright information of the website from the meta tag of the webpage, i.e., the content between "<META name=copyright content=. . .>."

5) Alt: extracting the text from the Alt tag of the webpage, i.e., the content between "<IMG alt=. . .>."

C. Characteristics of the Feature Representation Note that phishing websites represented by the term frequencies of the webpage content share similar characteristics with malware samples represented by the instruction frequencies. First, the feature representation is representative and can well group the instances of the same cluster. It has been observed in practice that malware samples in the same family or derived from the same source code share similar shapes of instruction-frequency patterns. Figure 3 illustrates that the shapes of instruction-frequency patterns are similar for the same malware family, and they are different for different malware families. For websites, the extracted terms can well summarize the content of the full web pages, while eliminating a large amount of "redundant" information. As shown in Fig. 4, the two websites "http://www.nanhang10.tk/"and"http:/ /www.zgnhair. com" belong to the same family (sharing similar term features), which both masquerade as the real China Southern Airline to trick people into ordering the flight tickets and remitting money to the perpetrators.

Second, the term frequencies of the webpage content and the instruction frequencies of file samples have similar distribution patterns. Figure 5 shows the distribution of term frequency on a set of 2004 phishing websites with 3038 dimensions as well as instruction frequency on a sample dataset with 1434 malware samples with 1222 dimensions. These two features with TFIDF scheme have been extracted, and PCA is performed to extract the first two important dimensions for visualization As shown in Figure5, the distributions of phishing websites and malware samples are typically skewed, irregular, and of varied densities.

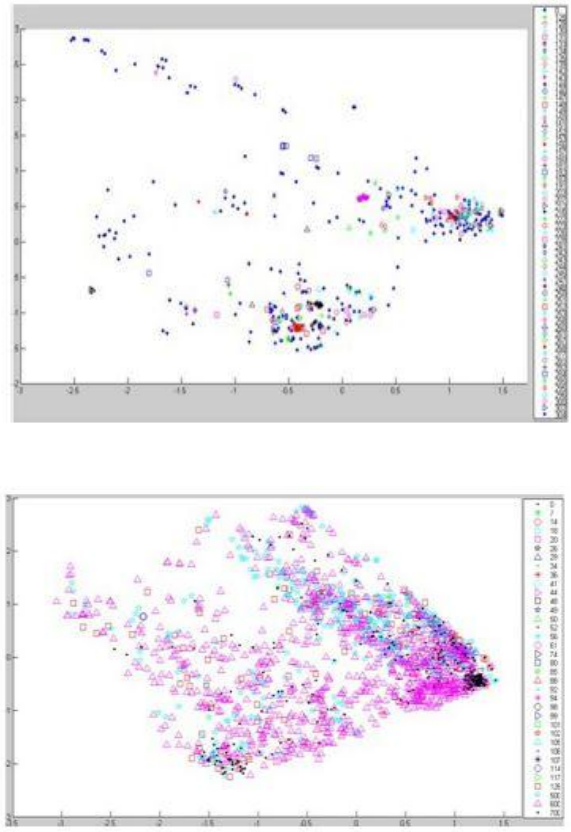


Fig5. Feature distributions after PCA transformation.

V. Base Clustering's

In application, a cluster is a collection of phishing websites or malicious files that share some common traits between them and are "dissimilar" to the phishing websites or malware samples belonging to other clusters. Hierarchical and partitioning clustering are two common types of clustering methods, and each of them has its own traits. The HC method can deal with irregular dataset more robustly, while partitioning clustering like KM is efficient and can produce tighter clusters especially if the clusters are of globular shape. The choice of clustering algorithms is largely dependent on the

underlying feature distributions. Since the feature distributions of malware samples and phishing websites are compel (as shown in Figure 5), in our study, both HC and KM algorithms will be applied to generate base clustering's.

1) Hierarchical Clustering Algorithm

Hierarchical algorithms can be categorized into two subcategories agglomerative algorithms and divisive algorithms. Because of its lower computation cost, in our application, we utilize the agglomerative HC algorithm as the frame starting with N singleton clusters, and successively merges the two nearest clusters until only one cluster remains.

The outline of the adopted hierarchical clustering (HC in short) algorithm suitable for both phishing website and malware categorization is described in Algorithm 1. Here, we adopt cosine similarity to measure the similarity between two data points, because of its independent data length. The definition of cosine similarity measure is described.

$$\mathbf{Dij = cos\alpha = x = \frac{xi\ T\ xj}{Xi|xi_{xj}|}xTi}$$

Where xi or xj represent the vectors of the two data points. There are a variety of ways to compute the similarity from C to all other clusters: complete linkage, single linkage and average linkage . Complete linkage is strongly biased toward producing clusters with roughly equal diameters, and it can be severely distorted by moderate outliers. Single linkage sacrifices performance in the recovery of compact clusters in return for the ability to detect elongated and irregular clusters. Considering the characteristics of both term-frequency and instruction-frequency feature representations, average linkage is used in our application. For validity index, we use the Fukuyama–Sugeno index (FS)to measure the quality of the clustering results. FS evaluates the partition by exploiting the compactness within each cluster and the distances between the cluster representatives.

It is defined as

$$\mathbf{FS=\sum_{i}^{n}\sum_{j=1}^{nc} uij(||xi - uj||\ 2 - ||vj - v||2)}$$

Where vj is the center [19] of cluster Cj, v is the center of the whole data collection, and A is an n × n positive definite, symmetric matrix where n is the feature dimension. It is clear that for compact and well-separated clusters, we expect small values for FS.

2) K-Medoids Clustering Approach

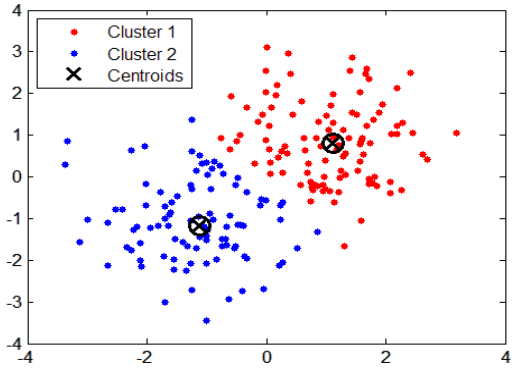


Fig6. k-Medoid Clustering Approach

Another well-known clustering algorithm for categorization is squared error-based partitioning clustering, such as K-means and KM, which assigns a set of data points into clusters using an iterative relocation technique. A cluster is represented by one of its real data point (called medoids) or by the mean of its data points (called centroid) in KM and K-means methods, respectively. They are very simple, but effective and widely used in many scientific and industrial applications. Considering that the distributions of phishing websites and malware samples are typically skewed, irregular, and of densities, in order to well deal with the outlier problem, we use KM instead of K-means for categorization. The algorithm procedure for KM is described in Algorithm 2. For the KM clustering algorithm, we use the same data point distance measure and validity index calculation methods as the aforementioned HC algorithm.

VI. Cluster Ensemble

Clustering algorithms are valuable tools for malware categorization. However, clustering is an inherently difficult problem due to the lack of supervision information. Different clustering algorithms and even multiple trials of the same algorithm may produce different results due to random initializations and stochastic learning methods. In our study, we use a cluster ensemble to aggregate the clustering solutions that are generated by different both hierarchical and partition clustering algorithms. We also show that the domain knowledge in the form of website level/sample-level constraints can be naturally incorporated into the cluster ensemble. To the best of our knowledge, this is the first work of applying such cluster ensemble methods for Internet security including phishing website and malware categorizations.

1) Formulation

Formally, let X = {x1, x2, . . . , xn} be a set of n data points (phishing websites or malware samples). Suppose that we are given a set of T clusterings (or partitioning)
P = {P1, P2, . . . , P T } of the data points in X. Each partition Pt (t = 1, . . . , T) consists of a set of clusters Ct ={Ct1, Ct2 , . . . , CtK t}, where Kt is the number of clusters for partition Pt and X = _K_=1 Ct_ . Note that the number of clusters K could be different for different clustering's. We define the connectivity matrix M(Pt) for the partition Pt as

$$Mij\ (Pt) = \begin{cases} 1 \text{ if xi and xj belong to the same cluster in Ct} \\ 0 \text{ Otherwise.} \end{cases}$$

Using the connectivity matrix, the distance between two partitions Pa, Pb can be defined as follows

$$d(Pa, Pb) = \sum_{i,j=1}^{n} dij\ (Pa, Pb)$$
$$= \sum_{i,j-1}^{n} |Mij\ (Pa) - Mij\ (Pb)|$$
$$= \sum_{i,j-1}^{n} [Mij\ (Pa) - Mij\ (Pb)]2 .$$

Note that |Mij (Pa) −Mij (Pb)| = 0 or 1.

A general way for cluster ensemble is to find a consensus partition P* which is the closest to all the given partitions:

$$\mathbf{Min} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{d(pt, p*)}$$

$$P* = \frac{1}{T} \sum_{t=1} \mathbf{T} \sum_{j=1}^{n} [\mathbf{Mi, j(Pt) - Mij(p*)}]\mathbf{2}$$

Since J is convex in M(P*), by setting ∇ M(P *)J = 0, we can easily show that the partition P* that minimizes is the consensus (average) association: the ijth entry of its connectivity matrix is.

$$\mathbf{Mij} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{d(pt)}$$

In our application, we construct four base categorizers using the algorithms that are described in Section V. 1) Two clustering's are obtained by applying HC on the term-frequency vectors or instruction-frequency vectors with TF-IDF and TF weighting schemes (denoted by HC_TFIDF and HC_TF); and 2) two clustering's by applying KM on the term-frequency vectors or instruction-frequency vectors with TF-IDF and TF weighting schemes with two different number of clusters: one is generated by HC_TFIDF, while the other is generated by HC_TF.
Based on Proposition 6.1, we could derive the final clustering from the consensus association _Mij. The ijth entry of _Mij represents the number of times that data point i and j have co-occurred in a cluster. We could then use the following simple strategy to generate the final clustering. 1) For each data point pair, (i, j), such that _Mij is greater than a given threshold (in our application, the threshold is 0.5 × 4 = 2), assign the data points to the same cluster. If the data points were previously assigned to two different clusters, then merge these clusters into one. 2) For each remaining data point not included in any cluster, form a single element cluster. Note that we do not need to specify the number of clusters.

2) Incorporating Sample-Level Constraints

We also show that the domain knowledge in the form of website-level/sample-level constraints can be naturally incorporated into the cluster ensemble. In this scenario, in addition to t partitions, we are also given two sets of pair wise constraints:
1) must-link constraints
A = {(xi1, xj1), . . . , (xia, xja)}, a = |A|
Where each pair of points are considered similar and should be clustered into the same cluster,
2) cannot-link constraints
B = {(xp1, xq1), . . . , (xpb, xpb)}, b = |B|
Where each pair of points are considered dissimilar, and they cannot be clustered into the same clusters. Such constraints have been widely used in semi supervised clustering however, few research efforts have been reported on incorporating constraints for cluster ensemble.
To incorporate the constraints in M and C into cluster ensemble, we need to solve the following problem:

$$\text{Min } j = \frac{1}{T} \sum_{t=1} T \sum_{j=1}^{n} [Mi, j(Pt) - Mij(p*)]2$$

s.t. Mij (P*) = 1, if (xi, xj) ∈ A
Mij (P*) = 0, if (xi, xj) ∈ B. (6)

Equation (6) is a convex optimization problem with linear constraints. Let C = A_B be the set of all constraints; then c = |C| = |A| + |B|. We can represent C as C =
{(xi1, xj1, b1), . . . , (xic, xjc, bc)}, where bs = 1if (xis, xjs) ∈ A, and bs = 0if (xis, xjs) ∈ B, s = 1. . . c.We can then rewrite (6) as

$$\text{Min } j=\frac{1}{T}\sum_{t=1} T \sum_{j=1}^{n}[Mi, j(Pt) - Mij(p*)]2$$

s.t. (**e**is)M(P∗)**e**js = bs, s = 1, 2, . . . , c (7)

Where **e** is∈ Rn×1 is an indicator vector with only the is the element being 1 and all other elements being 0. Now, we introduce a set of Lagrangian multipliers {αi}ci
=1 and construct the
Lagrangian for problem (7) as

$$L=J+\sum_{s} \alpha s\big((ei)T\ M(p*)ej, -b\big). \qquad (8)$$

Note that (**e**i)TM(P*)**e**js = Mi js (P*). Hence, we can show that the solution to problem (7) is

$$\textbf{Mij(Pt)} = \begin{cases} \frac{1}{T}\sum_{t=1}^{T} \mathbf{d(pt)} \\ \textbf{bs Otherwise.} \end{cases} \textbf{if(is,js) is not in C} \quad (9)$$

In other words, the solutions for regular elements in _Mij do not change and for constrained elements, according to (9), we need to set the corresponding entries of the consensus association _Mij to be the exact values based on their constraints.438 malware samples.

VII. Conclusion

In this paper, we seen an ACS which can not only be applied for phishing website categorization, but also for categorizing malware samples into families that share some common traits by an ensemble of different clustering solutions that are generated by different clustering methods. Empirical studies on large and real daily datasets that are collected by the Kingsoft Internet Security Laboratory illustrate that our ACS system performs well for real phishing website categorization as well as malware categorization applications. There are many avenues for future works. First, will explore various base clustering algorithms (e.g., recent probabilistic clustering methods and subspace clustering) with different feature representations. Second will extend clustering ensemble framework for anomaly detection. Third will investigate new ways to represent domain knowledge and novel methods to incorporate domain knowledge into the detection process.

References

[01] Ricardo Baeza-Yates and Berthier Ribeir Neto. Modern Information Retrieval. Addison Wesley, 1999.

[02] Alexander P. Topchy, Anil K. Jain, and William F. Punch. Clustering ensembles: Models of consensus and weak partitions. IEEE Trans. Pattern Anal. Mach. Intell.,27(12):1866–1881, 2005.

[03] Aristides Gionis, Heikki Mannila, andPanayiotis Tsaparas. Clustering aggregation. In ICDE, pages 341–352, 2005.

[04] Tao Li and Chris Ding. Weighted Consensus Clustering. In SIAM Data Mining, 2008.

[05] R.xu and D Wunsch, "Survey of clustering algorithms," IEEE Trans.Neural Netw.,vol.16,no.3,pp.645-678,May 2005.

[06] X. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graphpartitioning," in *Proc. 21st Int. Conf. MacLearn.*, 2004, p. 36.

[07] I. Gurrutxaga, O. Arbelaitz, J. M. Perez, J. Muguerza, J. I. Martin, and I. Perona, "Evaluation of Malware clustering based on its dynamic behaviour.," in *Proc. 7th Australas. Data Mining Conf.*, 2008

[08] Y. Ye, T. Li, Y. Chen, and Q. Jiang, "Automatic malware categorization using cluster ensemble," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 95–104.

[09] T. Li and C. Ding, "Weighted Consensus Clustering," in *Proc. SIAM Int. Conf. Data Mining*, 2008, pp. 798–809.

[10] R. Layton and P. Watters, "Determining provenance in phishing websites using automated conceptual analysis," in *Proc. eCrime Res. Summit*,2009, pp. 1–7.

[11] U. Bayer, P. M. Comparetti, C. Hlauschek, C. Kruegel, and E. Kirda, "Scalable, behavior- based malware clustering," in *Proc. 16th Annu. Netw.Distributed Secur. Symp.*, 2009.

[12] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databasewith noise," in *Proc. ACM nt. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.

[13] Y. Fukuyama and M. Sugeno, "A new method of choosing the number of clusters for the fuzzy C- means method," in *Proc. 5th Fuzzy Syst. Sym.*, 1989, pp. 247–250.

[14] M. Aburrous,M. A. Hossain, K. Dahal, and F. Tha btah, "Predicting phishingwebsites using Classification mining techniques with experimental casestudies," in *Proc. 7th Int. Conf. Inf. Technol.*, 2010, pp. 176–181.

[15] L. Kaufman and P. J. Rousseeuw, "Partitioning around medoids (program PAM)," *Finding Groups in Data: An Introduction to Cluster Analysis*,1990.

[16] J. Hartigan andM.Wong, "Algorithm AS136: A k- means clustering algorithm," *J. Roy. Stat. Soc., Appl. Stat.*, vol. 28, pp. 100–108, 1979.

[17] M. Gheorghescu, "An automated virus classification system," in *Proc. VIRUS BULLETIN CON.*, Oct. 2005.

[18] M. Fredrikson, S. Jha, M. Christodorescu, R . Sailer, and X. Yan, "Synthesizing near-optimal malware specifications from suspicious behaviors," in*Proc. IEEE Symp. Secur. Priv.*, Washington, DC IEEE Computer Society, May 2010, pp. 45–60.

AUTHORS PROFILES

Mr. H. T. Tanojkumar received his B.E degree in Computer Science and Engineering from AITM Bhatkal in 2012. Currently he is perusing M.Tech degree in Computer Science and Engineering at Canara Engineering College, Mangalore. His areas of interest are Web security, Cyber security, Information Security.

Mrs. Verdine Saviola Noronha received her B.E degree in Computer Science & Engineering from Canara Engineering College ,Manglore in the year 2007. She Completed her M. Tech in Computer Science & Engineering from NMAMIT, Nitte, Dakshina Kannada in 2010.Presently working as Asst. Professor at Canara Engineering College, Mangalore. Her area of interest are Computer Networks, Microprocessor and Network security.

Mr. B. N. RamaChandra
received his B.E degree in Computer Science and Engineering from KVG College of Engineering Sullia in 2012. Currently he is perusing M.Tech degree in Computer Science and Engineering at Canara Engineering College, Mangalore. His areas of interest are Digital Image Processing, Data mining, Information Security, Algorithms.

