Phishing Detection using Text, Image and Tag Classification Approach

Pankaj H. Gawale
P. G. student, Department of Computer Engineering,
R. C. Patel Institute of Technology,
Shirpur, Dist.Dhule ,Maharastra, India
pankajgawale.be@gmail.com

D. R. Patil
Department of Computer Engineering,
R. C. Patel Institute of Technology,
Shirpur, Dist.Dhule, Maharastra, India,
dharmaraj.rcpit@gmail.com

Abstract—Phishing is an act of cracking by single person or group of persons to steal the personal confidential information such as credit card details, bank account details, passwords etc., from unknown sufferer for illegal activities. In this paper we have implemented the text classifier using Bayesian approach for phishing detection; image classifier is used the earth mover's distance to measure the visual similarity between web pages. Tag classifier is used the layout similarity of two web pages by comparing the HTML tags**.** Bayesian model is used to establish the threshold. A data fusion algorithm used to merge the results of the text classifier, the image classifier and tag classifier. The experimental results shows, 99.90% of phishing pages detection rate.

Keywords-*Text Classifier; Image Classifier; Tag Classifier, Bayesian approach; Detection Rate (DR); F-score; Matthews Correlation Coefficient (MCC); False Negative Ratio (FNR;, False Alarm Ratio (FAR).*

*****

I. INTRODUCTION

Phisher are generating phishing web page which mostly similar to real web page. The most commonly in phishing collects information such as bank account, credit card details, user name, password, social security numbers, mobile numbers, and birthdates by masquerading as honest entity in an electronic communication. Now days e-mail contains much type of links of different web sites. Phishing is carried out by quick messaging or e-mail spoofing and also fake web pages or phishing web sites directs users to enter details which is useful for phishers. A phishing web site is felt and look likes the real or legitimate web site. Phishing is responsible for large amount of personal data loss and money loss [1].

In general phishing attacks are done in following way:

- Firstly phisher set up the fake web site which is identical to legitimate web site.
- Phisher then send links to the phishing web site in the large amount of spoofed e-mails to the target user. Due to that the phisher are trying to convince the victims to visit their fake web sites.
- By clicking on the link the victims visits the fake websites and inputs its confidential information there.
- Phisher then steal the confidential information and use into their fraud such as transferring money from victim's account.

Phishers are technically introducing many new ideas and cam affords to invest money in technology. It is common misunderstanding that phishers are amateurs. Phishers can afford investment in technology adequate with illegal benefits gained by their crimes.

Phishing is done in many different ways. Some of them are: deceptive phishing, malware based phishing, key loggers and screen loggers, session hijacker's web Trojans hosts, file poisoning, system reconfiguration attack, data theft, DNS-based phishing ("pharming"), content-injection phishing, man-in-the-middle phishing, search engine phishing software and security provider, financial institution and academic researcher gives the much attention on finding phishing web page. Anti-phishing refers to methods that derived to detect and prevent phishing attacks. It provides security from phishing attacks. Many researchers have worked on anti-phishing for deriving the various anti-phishing techniques. From these techniques some of them have works on URLs of web sites, some on e-mails, some works on attributes of web sites and some works on content of web sites.

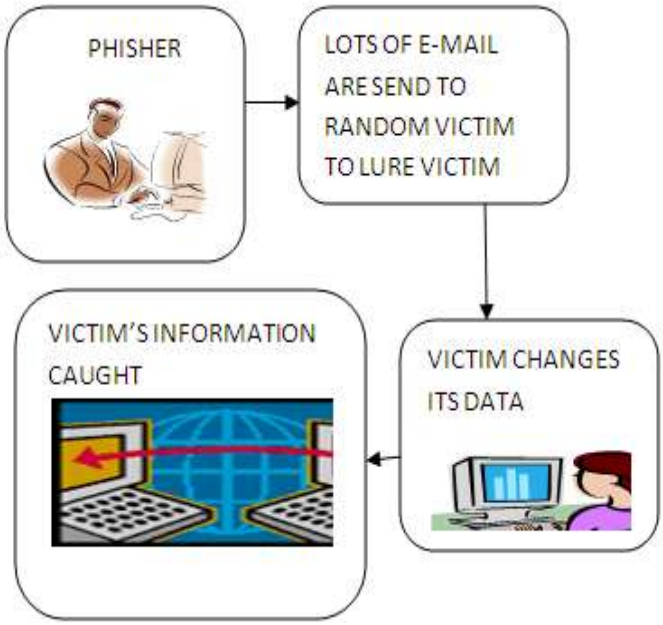

Fig. 1. A General way of phishing attack [1].

II. RELATED WORK

Fu et al. have developed earth movers distance method to estimate the similarity of the images. In this method they have firstly converted web page in to finding visual similarity. This

approach only detects phishing web page at pixel level not at text level [2].

Zang et al. have proposed effectiveness of anti-phishing toolbars from their study & analysis. They have worked on browsers security indicators and two user studies of three security toolbar. They conducted two studies for detecting which attacks are more effecting than others & generate the anti-phishing tool bar [4].

Liu et al. have worked on use of semantic link network to identify phishing web page. They have proposed the novel approach to finding the phishing web page by calculating the reasoning on the semantic link network .They firstly find the associated web pages of given web page & the constructing a semantic link network for all web pages [5] .

Zang et al. have proposed a novel content based approach to finding phishing web sites based on TF_IDF information retrieval algorithm CANTINA i.e.Carnegie mellon anti-phishing & Network analysis tool. This method first evaluating TF_IDF of each term then an retrieval algorithm is used in information retrieval to generate a lexical signature provides to a search engine & then tallies the domain name of web page is phishing or not [6].

Likarish et al. have developed B-PAT anti-phishing tool bar that helps users to identify phishing websites using Bayesian approach. B-PAT is developed to found the phishing websites by using open source Bayesian filter on the basis of tokens which are extracted from document object module analyzer [7].

Liu et al. have developed concept of visual approach to phishing detection which is oriented by document object module based visual similarity. Its firstly decomposes the HTML Web pages in to visually differentiable block regions. There are three metrics namely over all style, block level similarity and layout used to evaluate the visual similarity between two web pages [8].

Chandrasekaran et al. have proposed a novel approach to developed classification method based on structural characteristics of phishing e-mails. In this method they used support vector machine for phishing classification. This technique is compared with other widely used machine learning techniques. Identification of phishing e-mail was based on a number of structural features such as domain name, presence of form tag, presence of JavaScript [9].

Zang et al. have developed new content based anti-phishing system using Bayesian approach. New features like text classifier, image classifier and fusion algorithm was developed. For generation of text classifier naive bayes rule was used and for image classifier earth mover's distance method was used. According to the experimental results they shows that the detection rate 99.87% [10].

III. METHODOLOGY

A. *Overview of System*

As shown in Fig. 2 Anti-phishing approach contains the following parts: A text classifier using the Bayes rules to handle the text content extract from a current web page. An image classifier developed by using the Earth Movers Distance (EMD) for similarity assessment of a given web page [3]. A tag classifier similar approach to text classifier to work on HTML tags pairs extract from source code of web pages. A Bayesian approach used to estimate the threshold practice classifiers during offline training. A data fusion algorithm is used to fuse the results from the text classifier, the image classifier, and tag classifier. The above framework contains a training section, which is to calculate the statistics of historical data i.e., web page training set. A testing section is observing the incoming testing web pages. The detection results are finally sent to the user or the web browsers.

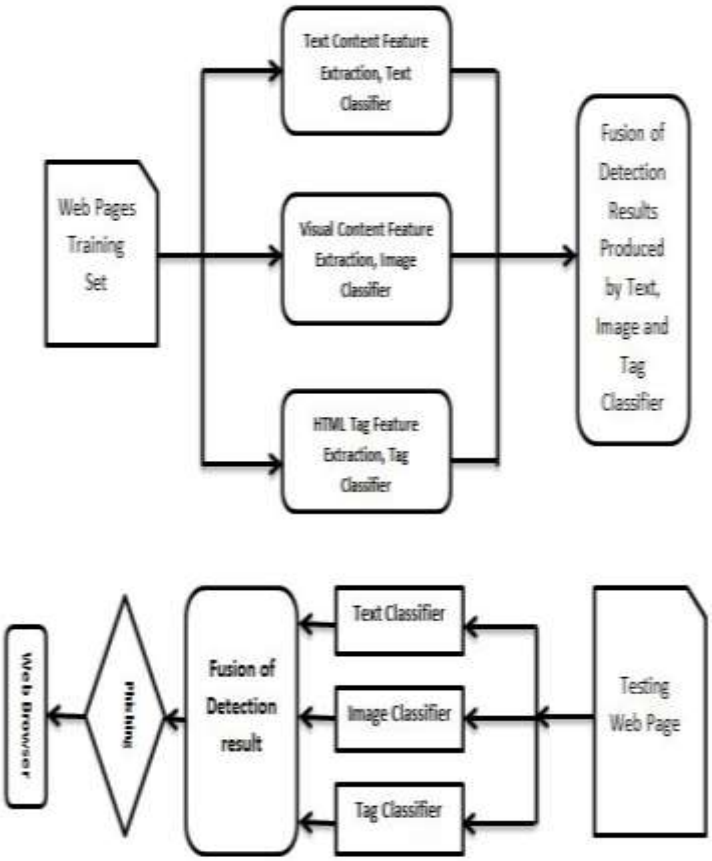

Fig. 2 Overview of system

B. *Text Based Classification*

In preprocessing firstly extracted all the text i.e. words from web page by removing HTML tags. Then we built the vocabulary to form the histogram vectors for each word from given web page. After extracting all words we apply stemming process to each word. Then stem words are used as basic feature for generation of text classifier instead that extracted words. Given a web page, we then form a histogram vector (h1, h2, . . . , hn), where each component represents the term frequency (a term appears in the web page) and *n* denotes the total number of components in the vector. We explain three points here.

- We do not extract words from all the web pages in a dataset to construct the vocabulary, because phishers usually only use the words from a targeted web page to scam unwary users.
- For the simplicity, we do not use any feature extraction algorithms in the process of vocabulary construction.
- We do not take the semantic associations of web pages into account, because the sizes of most phishing web pages are small.

In this paper, we have used the Bayes classifier to classify the text content of web pages. In the classification process, the Bayes classifier outputs probabilities that a web page belongs

to the corresponding categories. These probabilities also can be regarded as the similarities or dissimilarities that given web pages have with the protected web page. Let G = {g1, g2, . . . , g j , . . . , gd } denote the set of web page categories, where d is the total number of categories. In fact, for anti-phishing problem only two categories are included: the phishing web page category *g*1 and the normal web page category g2. Given a variable vector (v1, v2, . . . , vn) of a web page, the classifier is employed to determine the probability P(g j |v1, v2, . . . , vn) that the web page belongs to category g j . Applying the Bayes rule, the posterior probability P(g j |v1, v2, . . . , vn) is calculated by[10],

$$P(gj \mid v1, v2, ..., vn) = \frac{P(v1, v2, ...vn \mid gj)P(gj)}{P(v1, v2, ...vn)} \quad \ldots (1)$$

Let Cj = {c j,1, c j,2, . . . , c j,K j } be the set of training web pages belonging to category g j, where K j is the number of web pages in set Cj, and let Hl = (hl,1, hl,2, . . . , hl,n) (l =1, 2, . . . , K j) denote the histogram vector of the l-th web page in Cj corresponding to the word vocabulary (u1, u2, . . . , un). Conditioning on category g j . Thus, given a testing web page T , the probability P(g j |T) that the web page T belongs to category g j is calculated by

$$P(gj \mid T) = \frac{P(gj)\prod_{i=1}^{n} P(ui \mid gj)^{\frac{hiT}{R}}}{\sum_{s=1}^{d} P(gs)\prod_{i-1}^{n} P(ui \mid gs)^{\frac{hiT}{R}}} \quad (2)$$

Where, hi,T represents the frequency of the i th word appearing in the web page T, and R is the total number of words extracted from the protected web page. Here, the term R is used to enlarge the value of the term P(ui |g j)hi,T /R such that the denominator of above equation will not be close to zero, because for most phishing cases the phishing web pages include much more term frequencies than the normal web pages. We then compare the probability P(g1|T) of the web page T belonging to the phishing category g1 to a threshold θ_T which is estimated later by using the Bayesian theory. If the probability P(g1|T) exceeds the threshold θ_T , the web page is classified as phishing, otherwise, the web page is classified as normal [10]

C. Image Based Classification

In image classifier we have used approach as Earth Mover's Distance (EMD) method to measure the visual similarities between the incoming web page and real certified web page [3]. We have fetched both doubtful web pages and certified web pages from web and then we generate signatures, which are used for estimation of EMD between them. All images used are in JPEG format. Original images of web pages are converted into normalized size (e. g. 100×100) using Lanczos algorithm [11]. We used these normalized images to generate the signature of each web page. A signature of web page is nothing but a feature vector, is used to represent image of web page. Feature vector includes two components: a degraded color and the centroid of its position distribution in the image. Let Fσ = {σ,Cσ } be the feature, where σ represents the degraded color (i.e., a 4-tuple < A, R, G, B >, in which the components represent alpha, red, green, and blue, respectively), and Cσ represents the centroid of the degraded color. The calculation of the centroid is given by Cσ = _Nσ i=1 (cσ,i /Nσ), where cσ,i is the coordinate of the i th pixel that has the degraded color σ, and Nσ is the total number of pixels that have the degraded color σ (i.e., the frequency). The weight corresponding to the feature Fσ is the color's frequency Nσ . Thus, a complete signature S is described as

S = {(Fσ1, Nσ1), (Fσ2, Nσ2), . . . , (FσN , NσN)}…….(3)

Where N is the total number of selected degraded colors, in this signature representation, the feature weighted units in S are ranked in the descending order of their weights, i.e., Nσi ≥ Nσi+1 for 1 ≤ i ≤ N − 1 [3]. The EMD is adopted to measure the distance (or dissimilarity) of two web page images, because it supports many-to-many matching for feature distributions [3, 10]. Suppose we have two web page images a and b with signature Sa and Sb, respectively, where Sa has m feature units and Sb has n feature units. We first calculate the distance matrix D = [di j] (1 ≤ i ≤ m, 1 ≤ j ≤ n), where di j = Dnorm(Fσi , Fσj). Dnorm(Fσi , Fσj) is a normalized feature distance between feature Fσi and feature Fσj , which is defined by Dnorm(Fσi , Fσj) = μ·‖σi−σj‖+η ·‖Cσi−Cσj‖……(4)

Where μ + η = 1. Then the flow matrix Fab = [fi j] is calculated through linear programming and the EMD between Sa and Sb is calculated by

$$EMD(Sa, Sb, D) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} fij.dij}{\sum_{i=1}^{m}\sum_{j=1}^{n} fij} \ldots\ldots\ldots\ldots (5)$$

We define the EMD-based visual similarity of two images as

$$S_{visual}(Sa, Sb) = 1 - (EMD(Sa, Sb, D))^{\alpha} \ldots\ldots\ldots (6)$$

Where α ∈ (0, +∞) is the amplifier of visual similarity. If S_{visual}(Sa, Sb) = 1, the two images are completely identical, and if S_{visual}(Sa, Sb) = 0, the two images are completely different, because EMD(Sa , Sb, D) ∈ [0, 1]. If the visual similarity S_{visual} between a doubtful web page and the real web page exceeds the threshold θ_V, the web page is classified as phishing; otherwise, the web page is classified as normal. The complete implementation of image classifier is given as follows [3]:

- Get the images of web pages from its URL and carry out normalization.
- Generate visual signature of image.
- Estimate the EMD and visual similarity between input web page image and the real web page image using (5) and (6).
- Classify the web page into corresponding category on the basis of comparison of visual similarity and threshold θv.

D. Tag Classifier

Our approach was nearly related to the work presented in [12]. We have used Bayesian approach for tag classification. Tag classification done in two parts, firstly we analyze the layout similarity of two web pages by comparing the HTML tags of the pages and comparison with threshold. The system calculates the similarity between the input page and the real page. If this similarity is more than a predefined threshold value, then the web site is considered fake. We have analyzed web pages based on the simple tags comparison of the doubtful and real one web page

We have firstly extracted all source code of given web page. Then we have found HTML tags in Pair e.g.we will obtain the following pairs: (HTML, HTML), (div, div),

(BODY, BODY). These pairs used as histogram vectors as similar to text classifier. After this we have calculated the similarity between the current page and the original web page that is stored in the database. Then we have allocated the input web page into corresponding class according to the comparison of the Tag similarity and the threshold. [12].

E. Threshold Generation

The threshold is calculated using following [10].

$$\theta = \arg\max_{\hat{d}} \left(\frac{K(s > \hat{di}, O}{K(s > \hat{di}, O) + K(s > \hat{di}, N)} \right)$$

$$\left(\hat{di} \in \left\{ \arg\max_{di} K(s > di, O) \right\} \right) \quad (7)$$

Where K(s > di, O) and K(s > di, N) denote the numbers of phishing and normal web pages, the similarities of which exceed di , respectively, K(O) and K(N) denote the number of phishing and normal web pages in the training set, respectively, and KT = K(O) + K(N) denotes the total number of web pages in the training set. It is noted that the posterior probability P(O|s > θ) and P(O|s > θ) <= 1. If P(O|s > θ) =1, then P(s > θ |N) = 0, i.e., K(s > di , N) = 0. It indicates that we can select a threshold as large as possible to let K(s > di , N) = 0. But it is noted that K(s > di , O) also decreases when the threshold saturates to 1.

F. Fusion Algorithum

For all three classifier we have calculated posterior probability from following equation given in [10].

$$P_T(C \mid l_t) = \frac{K_T(l_t, C)}{K_T(l_t, C) + K_T(l_t, I)} \quad \ldots\ldots\ldots\ldots (8)$$

Where K_T (lt ,C) and K_T(lt , I) denote the numbers of correctly classified and incorrectly classified web pages associating their similarity measurements belonging to the subinterval lt , respectively, KT (C) and KT (I) denote the number of correctly classified and incorrectly classified web pages, respectively, based on the trained text classifier, and $K_F = K_T$ (C)+ K_T (I) denotes the total number of web pages in the training set. Likewise, we have determined the posterior probability P_V(C|lv) conditioning on a sub-interval lv = [Lv−1, Lv] for the image classifier by:

$$P_V(C \mid l_v) = \frac{K_V(l_v, C)}{K_V(l_v, C) + K_V(l_v, I)} \quad \ldots\ldots\ldots\ldots (9)$$

Where K_V(lv, C) and K_V(lv, I) denote the numbers of correctly classified and incorrectly classified web pages associating their similarity measurements belonging to the subintervals, respectively. Likewise, we determined the posterior probability P_{Tag}(C| l_{tag}) conditioning on a sub-interval $l_{tag} = [L_{tag}-1, L_{tag}]$ for the tag classifier by

$$P_{Tag}(C \mid l_{tag}) = \frac{K_{Tag}(l_{tag}, C)}{K_{Tag}(l_{tag}, C) + K_{Tag}(l_{tag}, I)} \quad \ldots(10)$$

Also the decision factor δ is calculated by following equation

$$\delta = \frac{P_T(C \mid l_t)}{P_V(C \mid l_v)} \quad \ldots\ldots\ldots\ldots (11)$$

The overall fusion algorithm works as follows:

Step 1: Input the training set, train a text classifier, an image classifier and a tag classifier, and then collect similarity measurements from different classifiers.
Step 2: Partition the interval of similarity measurements into sub-intervals for all classifier.
Step 3: Estimate the posterior probabilities conditioning on all the sub-intervals for the text classifier using (8).
Step 4: Estimate the posterior probabilities conditioning on all the sub-intervals for the image classifier using (9).
Step 5: Estimate the posterior probabilities conditioning on all the sub-intervals for the tag classifier using (10).
Step 6: For a new testing web page, classify it into corresponding category by using the text classifier and the image classifier. If it is classified into different categories, locate the sub-interval that the similarity measurement of the web page belongs to and execute step 7), if else, execute step 8).
Step 7: Calculate the decision factor for the testing web page for text and image classifier or fusion result of text and image classifier and tag classifier.
Step 8: Likewise step 6 we find fusion result and combines with tag classifier and classify in corresponding category. If it is classified into different categories, locate the sub-interval that the similarity measurement of the web page belongs to and execute step 7), if else, execute step 9)
Step 9: Return the final classification results to a user or a web browser.

IV. Dataset and Experimental Result

A. Dataset

We have used dataset of 10 272 homepages URLs. The entire dataset consists of eight sub-datasets corresponding to the real web pages. The web page distribution of the phishing and normal categories for different sub-datasets used in this paper are eBay, PayPal, Rapidshare, HSBC, Yahoo, Alliance-Leicester, Optus, and Steam. We have used 50% dataset for training our system and other 50% for testing.

B. Experimental Results

We have conducted a large-scale experiment to evaluate the performances of text, image and tag classifier. We randomly used 50% data set for training our system and 50% data set used for testing purpose. All the experiment were performed on a PC with Intel(R) Core(TM) i3-2370M CPU @ 2.40Ghz 4Gb RAM. We calculate the result on basis of different classifiers based on five criteria: Detection Rate (DR), the calculation of which is given by the ratio of number of correctly classified web pages and total number of web pages, F-score, a weighted average of the precision and recall where the score reaches its best value at 1 and worst value at 0. Matthews Correlation Coefficient (MCC), a balanced measure that describes the confusion matrix of true/false positives and negatives-such measure can be used even if the classes are of very different sizes, False Negative Ratio (FNR), the calculation of which is given by the ratio of number of false negatives and number of phishing web pages, False Alarm Ratio (FAR), the calculation

of which is given by the ratio of number of false alarms and number of normal web pages [10]

Table 1. Fusion classification result

Protected Web page	Thr	DR	F-score	MCC	FNR	FAR
eBay	0.20	0.996	0.983	0.985	16/818	0/4145
PayPal	0.25	1.00	0.988	0.982	8/1275	0/4146
RapidShare	0.10	1.00	0.986	0.984	0/226	0/4146
HSBC	0.10	0.998	0.987	0.985	0/226	0/4145
Yahoo	0.05	0.995	0.982	0.984	0/102	0/4145
Alliance-Leicester	0.05	0.999	0.983	0.981	0/91	0/4146
Optus	0.05	0.993	0.988	0.988	1/50	1/4145
Steam	0.20	1.000	0.983	0.987	0/48	1/4145

The classification result for different sites is shown in table 1. It is clearly observed that the all three classifier using Bayesian approach to determine threshold has better performance on DR. Which indicate that our system is more correctly classify the web page either phishing or original. It is observed that our fusion algorithm is capable of fusing the results from different classifiers in an efficient manner. Compared to use single classifier, i.e., either the text classifier or the image classifier or tag classifier, the DR has been improved by using our fusion algorithm, and significant improvement in terms of other evaluation measures has also been achieved as shown in Table 1. The experimental results show that our system has given better performance in terms of DR, F-score, and MCC.

V. CONCLUSION

In this paper, we have implemented text classifier using Bayesian approach, image classifier using EMD method, and tag classifier using Bayesian approach for phishing web page detection. We have implemented three classifier based anti-phishing system. We have evaluated our system on large dataset of 10272 homepages URLs. Our experimental results show that detection rate (DR) increases up to 99.90%.

REFERENCES

[1] G A. Emigh. (2005, Oct.). *Online Identity Theft: Phishing Technology, Chokepoints and Countermeasures. Radix Laboratories Inc.*, Eau Claire, WI [Online]. Available: http://www.antiphishing.org/phisging- dsh-report.pdf

[2] A. Y. Fu, W. Liu, and X. Deng, "Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD)", *IEEE Trans. Depend. Secure Comput.*, vol. 3, no. 4, pp. 301–311, Oct.-Dec. 2006.

[3] N. Chou, R. Ledesma, Y. Teraguchi, and D. Boneh, "Client-side defense against web-based identity theft", in *Proc. 11th Annu. Netw. Distribut. Syst. Secur. Symp.*, San Diego, CA, Feb. 2005, pp. 119–128.

[4] Y. Zhang, S. Egelman, L. Cranor, and J. Hong, "Phinding phish: Evaluating anti-phishing tools", in *Proc. 14th Annu. Netw. Distribut. Syst. Secur. Symp.*, San Diego, CA, Feb. 2007, pp. 1–16.

[5] W. Liu, N. Fang, X. Quan, B. Qiu, and G. Liu, "Discovering phishing target based on semantic link network", *Future Generat. Comput. Syst.*, vol. 26, no. 3, pp. 381–388, Mar. 2010.

[6] Y. Zhang, J. Hong, and L. Cranor, "CANTINA: A content-based approach to detecting phishing web sites", in *Proc. 16th Int. Conf. World Wide Web*, Banff, AB, Canada, May 2007, pp. 639–648.

[7] P. Likarish, E. Jung, D. Dunbar, T. E. Hansen, and J. P. Hourcade, "B-APT: Bayesian anti-phishing toolbar", in *Proc. IEEE Int. Conf. Commun.*, Beijing, China, May 2008, pp. 1745–1749.

[8] W. Liu, X. Deng, G. Huang, and A. Y. Fu, "An antiphishing strategy based on visual similarity assessment", *IEEE Internet Comput.*, vol. 10, no. 2, pp. 58–65, Mar.–Apr. 2006.

[9] M. Chandrasekaran, K. Narayanan, and S. Upadhyaya, "Phishing email detection based on structural properties", in *Proc. 9th Annu. NYS Cyber Secur. Conf.*, New York, Jun. 2006, pp. 2–8.

[10] H. Zang , G. Liu, Tommy W., S. Chow ,"Textual and visual content based anti-phishing : a Bayesian approach", *IEEE Transaction of neural network*, 1532- 1446, 2011.

[11] C. R. John, *The Image Processing Handbook*. Boca Raton, FL: CRC Press, 1995.

[12] A.P.E. Rosiello, E. Kirda, C. Kruegel, and F. Ferrandi, "A layout-similarity-based approach for detecting phishing pages", ;in Proc. SECURECOMM, 2007, pp.454-463.