Detecting Phishing E-mails by Heterogeneous Classification

M.Dolores del Castillo[1], Angel Iglesias[1], J. Ignacio Serrano[1]

[1] Instituto de Automática Industrial. CSIC, Ctra. Campo Real km. 0.200,
28500 Arganda del Rey. Madrid, Spain
{lola, iglesias, nachosm}@iai.csic.es

Abstract. This paper presents a system for classifying e-mails into two categories, legitimate and fraudulent. This classifier system is based on the serial application of three filters: a Bayesian filter that classifies the textual content of e-mails, a rule- based filter that classifies the non grammatical content of e-mails and, finally, a filter based on an emulator of fictitious accesses which classifies the responses from websites referenced by links contained in e-mails. This system is based on an approach that is hybrid, because it uses different classification methods, and also integrated, because it takes into account all kind of data and information contained in e-mails. This approach aims to provide an effective and efficient classification. The system first applies fast and reliable classification methods, and only when the resulting classification decision is imprecise does the system apply more complex analysis and classification methods.

Keywords: e-mail classification, web filtering, multistrategy learning

1 Introduction

Phishing is the term used to describe massive e-mails that trick recipients into revealing their personal or company confidential information, such as social security and financial account numbers, account passwords and other identity or security information. These e-mails request the user's personal information as a client of a legitimate entity with a link to a website that looks like the legitimate entity's website or with a form contained in the body of the e-mail. The aim of phishing is to steal a user's identity in order to make fraudulent transactions as if the phisher were the user.

According to Anti-Phishing Working Group [9] the number of phishing reports increased from 20,109 in May 2006 to 28,571 in June 2006 to make it the most ever recorded. Phishing attacks are increasing despite of the use of e-mail filters. Although only 0.001 percent of these e-mails sent are answered, this percentage is enough to provide a return on investment and keep the phishing industry alive [26].

The next section reviews different filter types currently available. Section 3 gives details of the anti-phishing system proposed. Section 4 discusses the empirical evaluation of this approach. Section 5 focuses on the learning ability of the system, and the final section presents the conclusions and point the way to future work on this subject.

2 Phishing Filtering

Information on techniques for avoiding electronic fraud is not readily available in the scientific and marketing literature. This lack of information prevents phishers from hitting upon for the circumvention of filters.

Nowadays, although different systems exist to deal with the problem of electronic fraud, these systems are far from optimal for classification purposes. The methods underlying these systems can be classified into two categories, depending on the part of the message chosen by the filter as the focus for deciding whether e-mail messages are legitimate or fraudulent:

1. Origin-based filtering. Origin-based filters focus on the source of the e-mail and verify whether this source is on a white verification list [8] or on a black verification list [1], [29].

2. Content-based filters focus on the subject and body of the e-mail. These can be also divided into two classes, depending on whether the analyzed content is textual or non-textual. Textual content filters classify e-mails using Bayesian classifiers [2], [7], [10], heuristic rules [23], [28] or a combination of them [10]. Non-textual content filters examine the links embedded in e-mails by different techniques. Some filters check whether the links belong to white and black verification lists [6], [10]. Other filters analyze the appearance of the alphanumeric string of links and look for well known phishing schemes [14], [15]. In [4], a combination of internal and external information relative to both textual content and links in e-mails is used to build an effective e-mail filter.

Besides the phishing e-mail filters, there exist other anti-phishing tools, the toolbars, which can be built into browsers for blocking access to or warning the user about web pages identified as possible or actual phish. Examples of these web filters are [5], [16], [25]. In [18] a comparative test of several toolbars is shown.

Most current commercial e-mail filters use verification lists to analyze e-mail senders and links contained in the body of e-mails. These filters are dynamic and update verification lists when new attacks are reported. However, the updating rate of filters is often overcome by the changing rate of the attacks because phishing e-mails are continuously modifying senders and link strings, many websites are only available one day, and continuous updating thus implies a high economic cost.

This paper describes a client-side system, which was designed and built to detect and filter phishing e-mail automatically, using different sources of information present in the content of e-mails which are handled by the processing methods most suitable for each information type.

3 Classification System

The range of procedures used by a phisher, which evolve quickly in order to evade filters available, makes it necessary to solve the problem of identifying phishing e-mails with a multistrategy and integrated approach. The system shown in this paper focuses on a global view of all the information provided by an e-mail and the different analysis methods for each kind of information.

One main feature of this system is effectiveness, measured by *precision* (percentage of predicted documents for a category that are correctly classified) and *recall* (percentage of documents for a category that are correctly classified), because it applies three different classifiers in order to obtain all the information needed at each decision point to classify an e-mail serially. Another relevant feature is efficiency, because the order in which the classifiers are applied aims to obtain fast and reliable classifications by minimizing the resources used in a first classification step and then, if it is necessary, more complex methods are used in following classification steps.

The system decides to assign an e-mail to the *Fraud* or *Legitimate* categories after applying first, a Naïve Bayes classifier that focuses on the textual content of e-mails and assigns them to the *Economic* or *Non-Economic* categories. It should be highlighted that it is very difficult to obtain financial legitimate e-mails because neither the financial sector nor users give their legitimate economic e-mails to generate legitimate economic corpuses that can be analyzed by the scientific community. Besides, the non-on-line financial entities are reluctant to use such means primarily due to the growing number of phishing attacks. This lack makes it difficult to train Bayesian classifiers that discriminate between the *Legitimate* and *Fraud* categories according to the textual content of e-mails. This is the reason for training both Bayesian classifiers on a corpus of e-mails labeled into the *Economic* and *Non-Economic* categories.

Next, a rule-based classifier, which focuses on non grammatical features of e-mails, classifies e-mails previously assigned to the *Economic* category by the first classifier into one of the *Legitimate*, *Fraud* or *Suspicious* categories. Last, a third classifier emulates a fictitious access to websites referenced by links contained in the body of the e-mails assigned to the *Suspicious* category by the rule-based classifier. This classifier analyzes the responses obtained from the fictitious access to these websites and classifies them into the *Fraud* or *Legitimate* categories.

The system includes a bias against generating false negatives, i.e., e-mails erroneously classified in the *Legitimate* category, and false positives, i.e., e-mails erroneously classified in the *Fraud* category. When a classifier lacks information at some decision point, the classifier assigns the e-mail to the category that allows the system to further analyze the e-mail at a deeper level. The bias tries to enforce the user's safety against false negatives and classification performance of the system against false positives. Although the outcome from wrongly classifying an e-mail as legitimate is more dangerous for the user than classifying an actual legitimate e-mail as fraudulent, the system takes into account both kinds of misclassifications.

The system can learn incrementally from past mistakes when the user prompts misclassified e-mails by an interface that shows the final classification results. The architecture of this classifier system is modular and flexible. This easily allows new analysis and detection methods to be added as they are developed.

3.1 Naïve Bayes Classifier for the Textual Content of E-mails

This classifier was developed to identify and filter e-mail based on the Naïve Bayes statistical classification model [12], [21]. Since there are two parts in an e-mail, subject and body, which can contain text, a Bayesian classifier is built for each part. In the training stage, the probabilities for each word conditioned to each category are estimated, and a vocabulary of words with their associated probabilities is created. The filter classifies a new text into a category by estimating the probability of the text for each possible category C_j, defined as $P\ (C_j \mid text) = P\ (C_j)\ .\ \Pi_i\ P\ (word_i \mid C_j)$, where $word_i$ represents each word contained in the text to be classified. Once these computations have been performed, the Bayesian classifier assigns the text to the category that has the highest probability value.

In the classification phase of a new e-mail, the Bayesian classifier composes a word vector associated with each part of the e-mail and computes the economic probability *Pec* and non-economic probability *Pnec* for each part. The parts are assigned to the category with the highest probability. The system integrates the classification decisions of both parts by assigning the e-mail to the *Economic* category whenever the classification result of any of the two parts is in the *Economic* category. This course of action claims to avoid generating false negatives.

3.2 Rule-Based Classifier for the Non Grammatical Content of E-mails

Since knowing whether an e-mail is economic is not sufficient to classify it into the *Legitimate* and *Fraud* categories, the goal of the system is only attained by analyzing other kinds of data contained in e-mails previously classified into the *Economic* category by the Bayesian classifier. This analysis aims to find out the values of the discriminatory features present in economic e-mails: links, images, and forms. An algorithm based on AQ learning [11] trained on a corpus of fraudulent and legitimate economic e-mails, which are represented by these three features, builds the descriptions and thus the rules for each category.

The first rule states that if the body of an economic e-mail does not contain forms, images or links, then the e-mail is assigned to the *Legitimate* category. The second rule determines that an e-mail that requests information directly from a form contained in the body is not safe at all. Accordingly, the classifier categorizes all e-mails dealing with economic topics and not offering any security when requesting personal information into the *Fraud* category. Since both fraudulent and legitimate e-mails can contain images or links, these features do not determine with certainty the category of e-mails. In order to avoid generating false positives, the rule-based classifier uses a third rule to decide whether an e-mail is potentially dangerous to users. This rule states that if an e-mail deals with economic topics and it contains links in the text or links in an image, then the danger could come from the websites referenced by these links. So, the classifier classify the e-mail into the *Suspicious* category and a third classifier further processes the e-mail. An e-mail categorized as *Suspicious* is considered by the system lacking of knowledge to make a decision. Each rule has a dynamic confidence level. This confidence level can be modified as the system learns from misclassifications.

3.3 A Classifier Based on an Emulator for the Content of Websites Addressed by the Links Contained in E-mails

This classifier processes the links contained in the body of the e-mail and assigns a suspicious e-mail to the *Legitimate* or *Fraud* categories in a procedure consisting of three steps: 1) a meta-searcher extracts the structure of the websites addressed by links, 2) an emulator fills in the website structure with fictitious data, and 3) a finite state automaton tries to recognize the answer given by the website.

The meta-searcher is a procedure for obtaining the links contained in e-mails and extracting the patterns of the forms contained in the websites referenced by these links. First of all, the meta-searcher verifies whether the websites actually exist using an Internet search engine. Since fraudulent websites are available for a short period of time, this classifier determines that if there is no information about a website, then there is a high probability that the website is fraudulent. So, the e-mail that references this website is classified into the *Fraud* category.

If the website exists, then the meta-searcher verifies that the accessed website uses a secure *http* connection (*https*). If a website is not safe, then the e-mail is classified into the *Fraud* category. If the website exists and it is safe, then the meta-searcher looks for the forms contained in the website and extracts all the fields that need to be filled in. The meta-searcher analyzes the kind of data of every field and detects whether a website requests sensitive information, like a password. If a website does not request personal information, then the meta-searcher determines that this website is legitimate and thus, the e-mail is classified into the *Legitimate* category.

If a website requests personal information, then the emulator fills in the forms using fictitious data and then it submits the information in order to obtain the website response. At this point, the website response needs to be analyzed to know its nature and classify the e-mail.

The classification model used for recognizing the response given by the website is based on the construction of a finite state automaton. The automaton represents words contained in a certain language and collects the grammar presented in this language. The language used in this case consists of the sentences contained in the responses given from legitimate financial websites to fictitious accesses. All of these responses warn the user of an error that occurred while processing the information requested by the website with a different vocabulary of words and a different grammar. The automaton is a generalized description of the instances of the class “legitimate responses to fictitious accesses”. Fictitious access to fraudulent websites return no error message since their sole goal is to collect users’ confidential information. This classification model allows a high degree of generalization in the classification and it is very efficient because it is not time-consuming and uses few resources.

A grammar inference algorithm called ECGI [19] takes the sentences and the probabilities of all bigrams (groups of two words) and generates the automaton states and the edges between the states. Every edge connecting two states is weighted with the probability of the corresponding bigram. When the automaton is going to recognize a new sentence, the words of the sentence are processed according to their written order [22]. If there is a state representing the processed word and this state is reachable from the actual state, then the automaton follows that connection. If there is no state reachable representing the word, then the automaton follows the connection that has the highest probability. After following this connection, the automaton processes the next word. When the automaton reaches the final state, the automaton response is the sequence of words with the highest probability and similarity related to the sequence of words included in the sentence analyzed.

The emulator-based classifier uses a function that computes the similarity between an input sentence and the sentence obtained by the automaton by calculating the relation between the number of overlapping words in both sentences, and the total number of words in each sentence. If the minimum of the two percentages calculated is greater than or equal to 50%, then it is considered that the two sentences are similar. If the minimum is less than 50%, then the function computes the sum of the document frequency (the number of sentences contained in the base of responses which include that word) of the overlapping words in order to give a higher score to the words that appear in more sentences of the base of responses. If the sum of the document frequency of the overlapping words is greater than or equal to a threshold, whose value has been empirically determined, then the classifier considers that the two sentences are similar. If the sum is less than the threshold, then the sentences are not considered similar.

Therefore, if the emulator-based classifier considers that a textual response obtained after a fictitious access is similar enough to the automaton response, then the textual response is considered legitimate. So the e-mail that references the website is classified into the *Legitimate* category. If the responses are not similar enough then the e-mail is classified into the *Fraud* category.

4 Empirical Evaluation

The classifier system was evaluated on a set of messages. This dataset is composed of 1,038 economic messages, divided into 10 legitimate messages and 1,028 fraudulent messages, and 1,006 non-economic messages. A small fraction (4 e-mails) of the legitimate messages, i.e., economic messages coming from legitimate financial entities, and all non-economic messages were extracted from [24]. The remaining legitimate messages (6) were received by the authors of this paper during a given period of time. From 1,028 fraudulent economic messages, 833 were extracted from [13] and the remaining messages came from the inboxes of the authors of this paper. In the evaluation shown in this paper, messages from a sub corpus of [24] were carefully analyzed by hand for extracting the non-economic messages. Currently, the vocabulary of terms needed by the Bayesian filter for classifying messages into the economic or non-economic categories has been learned from the textual contents of two corpus of webpages dealing with economic topics and a third corpus dealing with general topics.

The Bayesian classifiers were trained and tested on this dataset using 5-fold cross validation, that is, four folds were the training data and the fifth was the test data. The content of both textual parts of messages was preprocessed by applying a stop list, and stemming the words [17]. Next, words recurring below an experimentally determined threshold value of the function tf.idf [20] were removed. All words were sorted by the Chi-square statistical measurement [20] and only the 50% highest ranked words of the vocabulary of each part were retained. After that, both textual Bayesian classifiers were built.

Table 1 shows the *precision* (*Pr*), *recall* (*Rc*) and *F*-measure (*F*) (F = (2 * precision * recall) / (precision + recall)) values obtained by the integration of both Bayesian classifiers for the *Economic* (E) and *Non-Economic* (N-E) categories. The last column presents the macro averaged values and the other five columns (Si) present the result of each execution of the cross validation.

Table 1. Performance measurements resulting from integrating the classifications obtained by the two textual Bayesian classifiers (Categories: *Non-Economic* N-E*, Economic,* E).

	S1	S2	S3	S4	S5	Average
Pr N-E	0.956	0.976	0.995	0.995	0.926	0.969
Rc N-E	0.975	0.995	0.985	0.975	0.683	0.922
F N-E	0.965	0.985	0.989	0.984	0.786	0.942
Pr E	0.975	0.995	0.986	0.976	0.753	0.937
Rc E	0.957	0.976	0.995	0.995	0.937	0.972
F E	0.965	0.985	0.990	0.985	0.834	0.952

A supervised learning algorithm trained on the fraudulent and legitimate e-mails of the dataset generated the decision rules for the *Fraud*, *Legitimate* and *Suspicious* categories. These rules were used to classify all the e-mails in the five folds which were previously categorized as *Economic* by the Bayesian classifier. In a first experimental setting, suspicious e-mails were classified in the *Fraud* class by default. The effectiveness values of this classifier for the *Fraud* (F) and *Legitimate* (L) categories is reported in Table 2.

Table 2. Performance measurements of the serial application of the Bayesian and rule-based classifiers (Categories: *Fraud* F, *Legitimate* L).

Pr F	*Rc* F	*F* F	*Pr* L	*Rc* L	*F* L
0.941	0.949	0.944	0.943	0.921	0.930

In order for the system to make a confident decision about the class of a message, messages assigned to the *Suspicious* class were finally processed by the emulator-based classifier.

A base of responses was built to create the finite state automaton used by the emulator classifier. This base collects responses from different legitimate financial websites to fictitious accesses. These responses were preprocessed by using a stop list and a stemming algorithm [17]. The stemming algorithm does not reduce verbal endings that are used to determine verb tenses ("*Introduce* the information" is different from "The information *introduced*").

In the classification phase of the test responses, the classifier did not take the whole sentences obtained from accessing. Instead, it considered groups of three consecutive words. The choice of using groups of three words is based on the fact that most of the sentences included in the base of responses consisted of three words.

Table 3. Performance measurements of the serial application of the Bayesian, rule-based and emulator-based classifiers (Categories: *Fraud* F, *Legitimate* L).

Pr F	*Rc* F	*F* F	*Pr* L	*Rc* L	*F* L
0.962	0.948	0.955	0.949	0.962	0.955

The third classifier classifies the messages previously assigned to the *Suspicious* class by the rule-based classifier. Table 3 shows the overall performance measurements of the system for the *Fraud* (F) and *Legitimate* (L) categories. All effectiveness values are better than those obtained by only applying the Bayesian and the rule-based classifiers. The measurements indicate that this third classification step produces improvements of the precision value for the *Fraud* class and the recall value for the *Legitimate* class because it allows the system to classify more certainly the e-mails containing links and thus to generate less false positives.

5 Evolution of the System

Fraudulent e-mails will continue to evolve in an attempt to evade the filters available. In order to prevent increased misclassification and, consequently, reduced classification performance, the system has to be able to learn from past mistakes. Classification errors can be of two types: 1) false positives or *False Fraud*, which are e-mails incorrectly classified as fraudulent when they are legitimate, and 2) false negatives or *False Legitimate*, which are fraudulent e-mails incorrectly classified as legitimate.

As highlighted in previous sections, the system applies a bias that tries to protect users from the most dangerous error, the *False Legitimate*. Users are putting themselves at risk if they decide to access to an email classified into *Legitimate* class when it is actually *Fraud*. This is the reason why the system learning ability focuses on *False Legitimate* e-mails. When users identify a *False Legitimate*, they can prompt the e-mail misclassified by an interface designed for this purpose. The decision about this wrong classification can be caused by any of the three classifiers in the overall system. In order to know which classifiers are responsible for the misclassification, the system must save a classification track of the decisions taken by all the classifiers. Thus, the classifier that made the error can be identified. Depending on the classifier that wrongly decided, the learning ability differs:

1) If the Bayesian classifiers assign an economic e-mail into the *Non-Economic* class, both vocabularies are updated in terms of the number of words and word probabilities associated with the economic and non-economic categories.

2) The rule-based classifier assigns a fraudulent e-mail into the *Legitimate* class when there are no signs of danger present in this e-mail, i.e., the e-mail does not

contain any form or link. In this instance, the classifier decreases the confidence level of the rules implied in the classification by an empirically determined amount.

3) The emulator-based classifier assigns a suspicious e-mail to the *Legitimate* class when it is actually a fraudulent e-mail, because the automaton has not recognized correctly the responses to fictitious accesses to the websites referenced by the links contained in the suspicious e-mail. Here, learning implies that the sentences wrongly recognized by the automaton are added to a sentence stoplist. When the automaton receives a sentence, it verifies whether such sentence is included in the stoplist. If it is, the automaton does not process it and further analyzes the remaining sentences in the response.

6 Conclusions and Future Work

The system presented in this paper is based on a hybrid approach that takes advantage of applying different processing methods to multiple data sources. It is an effective system for avoiding the creation of false positives and negatives. The major novelty comes from the emulator method used to classify the webpages addressed by the links contained in e-mails. The analysis of all kind of data present in e-mails allows the classification to be independent of external information sources, like verification lists or web reputation servers, leading to a more efficient decision making.

Besides the learning capabilities of the system, thanks to its modular and flexible design, it does allow for straightforward upgrading as new processing methods become available or new features included in future phishing attacks can be identified.

Currently, this approach is being easily extended to build webpage classifiers. Here, the webpage textual content is categorized as economic or non-economic by integrating the predictions made by several classifiers that categorize each one of the four possible textual parts (url, meta-text, plain-text, links) of a webpage [3]. If the webpage is economic and it contains forms requiring confidential information, the emulator fills in the forms with fictitious data and it submits them. Next, the answer obtained is analyzed and classified and accordingly the webpage is also categorized.

References

1. Aladdin eSafe. http://www.aladdin.com (2005)
2. Androutsopoulos, I., Paliouras, G., Karkaletsis, G., Sakkis, G., Spyropoulos, C., Stamatopoulos, P.: Learning to Filter Spam E-mail: A Comparison of a Naive Bayesian and a Memory-Based Approach. Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases (2000)
3. Castillo, M.D., Serrano, J.I.: A Multistrategy Approach for Digital Text Categorization from Imbalanced Documents. ACM SIGKDD Explorations, 6 (2004) 70-79
4. Fette, I., Sadeh, N., Tomasic, A.: Learning to Detect Phishing Emails. WWW 2007, Banff, Canada (2007)
5. GeoTrust TrustWatch. http://www.trustwatch.com/ (2004)

6. GoDaddy. http://www.godaddy.com/ (2006)
7. Graham, P.: Better Bayesian Filtering. Proc. of Spam Conference 2003, MIT Media Lab., Cambridge (2003)
8. Iconix eMail ID. http://www.iconix.com (2005)
9. June Phishing Activity Trends Report. http://www.antiphishing.org (2006)
10. McAffee SpamKiller. http://www.spamkiller.com (2003)
11. Michalsky, R.S.: A Theory and Methodology of Inductive Learning. Machine Learning: An Artificial Intelligence Approach. Springer-Verlag (1983) 83-134
12. Mitchell, T.M.: Machine Learning. McGraw-Hill (1997)
13. Monkey.Org Inc. http://www.monkey.org/~jose/wiki/doku.php (2006)
14. Microsoft Outlook 2003. SP 2. http://office.microsoft.com (2005)
15. Mozilla Thunderbird 2. http://www.mozilla.com/thunderbird (2005)
16. Netcraft. http://news.netcraft.com/ (2007)
17. Porter, M.F.: An Algorithm for Suffix Stripping. Program, Vol. 14(3) (1980) 130-137
18. Robichaux, P., Ganger, D.L.: Gone Phishing: Evaluation Anti-Phishing Tools for Windows, 3Sharp LLC (2006)
19. Rulot, H.: ECGI. Un Algoritmo de Inferencia Gramatical mediante Corrección de Errores. Phd Thesis, Facultad de Ciencias Físicas, Universidad de Valencia (1992)
20. Salton, G., Buckley, C.: Term-weighting Approaches in Automatic Text Retrieval. Inform. Proccesing & Management, Vol. 24(5) (1988) 513-523
21. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys, Vol. 34(1) (2002) 1-47
22. Serrano, J.I., Araujo, L.: Statistical Recognition of Noun Phrases in Unrestricted Texts. Lecture Notes on Computer Science, Vol. 3646 (2005) 397-408
23. Sophos Email Security and Control. http://www.sophos.com (2005)
24. SpamAssassin. http://spamassassin.apache.org/publiccorpus/ (2006)
25. Spoofstick. http://www.spoofstick.com (2005)
26. Suckers for spam. http://www.Internetnews.com (2005)
27. Tagged Message Delivery Agent Homepage. http://tmda.net (2006)
28. Tumbleweed MailGate Email Firewall. http://www.tumbleweed.com/ (2006)
29. Verisign Messaging security http://www.verisign.com (2006)