ScamSlam: An Architecture for Learning the Criminal Relations Behind Scam Spam

Edoardo Airoldi and Bradley Malin
Data Privacy Laboratory, Institute for Software Research International

May 2004
CMU-ISRI-04-121

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Unsolicited communications currently accounts for over sixty percent of all sent e-mail with projections reaching the mid-eighties. While much spam is innocuous, a portion is engineered by criminals to prey upon, or scam, unsuspecting people. The senders of scam spam attempt to mask their messages as non-spam and con through a range of tactics, including pyramid schemes, securities fraud, and identity theft via phisher mechanisms (*e.g.* faux PayPal or AOL websites). To lessen the suspicion of fraudulent activities, scam messages sent by the same individual, or collaborating group, augment the text of their messages and assume an endless number of pseudonyms with an equal number of different stories. In this paper, we introduce ScamSlam, a software system designed to learn the underlying number criminal cells perpetrating a particular type of scam, as well as to identify which scam spam messages were written by which cell. The system consists of two main components; 1) a filtering mechanism based on a Poisson classifier to separate scam from general spam and non-spam messages, and 2) a message normalization and clustering technique to relate scam messages to one another. We apply ScamSlam to a corpus of approximately 500 scam messages communicating the "Nigerian" advance fee fraud. The scam filtration method filters out greater than 99% of scam messages, which vastly outperforms well known spam filtering software which catches only 82% of the scam messages. Through the clustering component, we discover that at least half of all scam messages are accounted for by 20 individuals or collaborating groups.

Keywords: spam, scam, Internet fraud, e-mail filtering, text analysis, text classification, poisson classification models, single linkage clustering, information retrieval, semantic learning

Contents

1 Introduction **2**

2 Spam, Fraud, and E-mail **3**

3 ScamSlam Architecture **4**
3.1 Poisson Filter 5
3.2 Message Representation 6
3.3 Scam Clustering 6

4 Experiments **8**
4.1 Scam Filtering 8
4.2 Clustering Analyses 11

5 Discussion **11**
5.1 Spam, not ScamAssassin 12
5.2 It's All Scam To Me 12
5.3 Extending the ScamSlam System 13

6 Conclusions **13**

1 Introduction

In today's digital society, unsolicited electronic communications, or spam messages, are increasingly difficult to escape. As the simplest of computer users know all too well, spam consists of annoying, infuriating, and, quite possibly, insulting text or images that surreptitiously creep into your virtual life. Even for those of the digital sophisticate, who encounter it only when they glance at their junk mailbox for false filter classifications, spam may be less of a nuisance but remains ever present. Due to its continued growth, the burdens of spam on society are felt by many different groups, including the individual that cleans his mailbox, the ISP that monitors the network, as well as the governmental investigators that attempt to curb illicit actions. As a result, spam is widely recognized as a multifaceted problem that requires both technology and policy solutions. Furthermore, the spam issue has been catapulted into the public spotlight via the channels of media government. Daily, journalists compose and report on stories about any number of ways by which spam is destroying the Internet. Governments, from local to federal to international bodies, now deliberate and even pass laws to curb aspects of the spam problem. [1]

A major challenge of the spam problem is the difficulty in determining the identity and relationships of spammers. To understand this challenge, one must realize that spam itself takes on many different forms which, to some extent, are dependent on an individual's motivation for playing the role of a spammer. For example, the text of an e-mail generated by an individual who perceives spam as a legitimate mass direct-marketing tool will appear vastly different from an e-mail generated by an individual whose sole desire is only to clog inboxes and increase packet load on the Internet. One of the more malicious breeds of spammer is that which considers e-mail as a medium for conducting social engineering, grifting, or fraud. [2] These spammers attempt to mask their "scam spam" messages as non-spam and con people through a range of scams, including pyramid schemes, securities fraud, and identity theft via "phisher" mechanisms, such as the notorious PayPal and AOL redirection scams.[3] Thus, in this research we concentrate on the advance fee fraud, the most infamous of which is the "Nigerian", or 4-1-9, scam. Over the past several years, the number and type of spam messages imploring readers for monetary assistance today with the promise of future riches, has increased without signs of abating.

The problem with respect to Internet fraud consists of several social and technological problems which we address in this research. The initial question is how does one discern scam messages from spam and non-spam e-mail? Furthermore, can we, or law enforcement officials, learn and track the scams perpetrated by a specific criminal cell? A traditional law enforcement approach for spammer recognition is to detect when a large number of the same email message is sent to different recipients, often within a short time period. Yet, scam messages differ from other types of spam for several reasons. First, a set of scam messages sent by the same individual are not necessarily equivalent in text and story. Second, scam messages can be sent out over a longer time period than traditional bulk spam messages. Third, scam messages are not necessarily sent via the same physical routes as spam or via the same techniques, such the commandeering of an open relay.

To address certain aspects of these problems, we have developed the ScamSlam system, which approaches the problem of scam spam from a forensic perspective. Despite the differences between general spam and scam, there are particular notable aspects of scam messages useful for learning and analyzing patterns in the messages. Specifically, though scam spam messages are unique, they tend to be engineered by a single, or related group of individuals. As such, there exist features in the semantic and syntactic structures of scam messages, or the scam artist signatures, such as similarities in general story and writing style, which can be used to relate messages to one another. Thus, the ScamSlam system is designed to leverage certain aspects of writing style features to help determine how many different authors exist for a particular type of scam, as well as which scam spam messages were written by which author.

The goal of this work is to assist law enforcement agents track the criminal activities of a group of individuals for which some evidence has been gathered in the form of predatory email messages. From this perspective, it is not of great importance that one or more individuals may be writing and adapting scam messages. Rather, it is more important that we are able to identify which scam messages are similar in terms of specific features, such as general storylines, payment methods, or word choice, which may remain hidden when messages are simply read and not analyzed by statistical and computational means. By exploiting patterns in the scam messages, our methods empower law enforcement officials with the capability to investigate and traceback messages of high similarity to locate members within the same ring of criminals.

The remainder of this paper is organized as follows. In the following section we discuss background issues with respect to internet fraud and specific aspects of the Nigerian scam. In Section 3, we present the technical details of the ScamSlam system. As mentioned, the system consists of several components based on both supervised and unsupervised learning models. Each component of the system is addressed from the standpoint of statistical and mathematical formulation, as well as its relationship to the application and assumptions of the system. In Section 4, we use a real world dataset of over 500 Nigerian scam messages to study the filtering and relationship learning capabilities of SlamScam. Finally, in Section 5 we discuss the limitations of the system, as well as how the SlamScam system can be validated and applied to a law enforcement setting.

2 Spam, Fraud, and E-mail

The concept of spam is not a novelty limited to the electronic world of the Internet. For years, any individual or household with a mailbox in the physical world received their fair share of unsolicited "junk" mail. However the quantity of junk snail mail sent to individuals is limited by the fact that its marginal cost scales linearly with the amount of mail sent. In cyberspace, on the other hand, the current status quo of communication is such that marginal cost is negligible as the quantity of electronic mail (e-mail) is sent. In combination with other factors, including the increased implementation of e-mail as a direct marketing tool, the amount of spam sent over the Internet is continually growing. Statistics compiled by Brightmail, a well-respected antispam company, indicate that as of February 2003, approximately 42% of all messages sent over the Internet was spam. By April 2004 this number had increased to almost 65%. The growth curve of spam on the Internet over time is depicted in Figure 1.

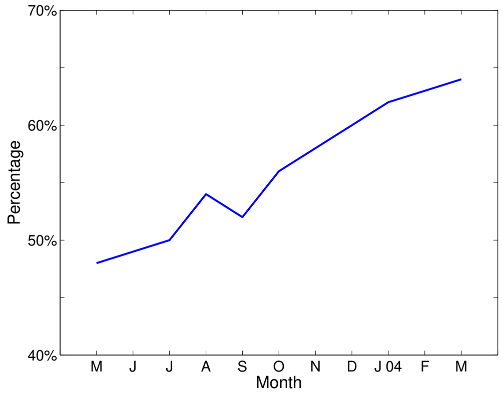

Figure 1: Monthly Percentages of total internet email identified as spam. Over 96 billion messages filtered in April 2004. Source: Brightmail, Inc. [4]

Similarly, the phenomenon of fraud is neither new nor trivial. For example, in 2003, the Federal Trade Commission (FTC) reported the American public lost over $400 million to fraudulent activities. [5] Scams communicated via e-mail and the Internet are on the rise as well. Brightmail reports that over three billion phishing scam emails are now sent monthly over the Internet, noting a 50% increase from January to April 2004 alone. [6] In March 2004, Zachary Hill was arrested by the FTC and the Department of Justice for identity theft and illegally attracting people via email to fake websites masquerading as AOL and PayPal. During the tenure of his scam, Hill obtained at least 471 credit card numbers, 152 bank account and routing numbers, and 541 user names and passwords. [3]

Though there exist many different kinds of fraud, the dataset studied in this research pertains to one specific type, namely the advance fee fraud. The advance fee fraud is a scheme in which a stranger with an unfortunate story requests an individual for some money, usually not a very large sum, to assist in the transfer of a large monetary sum. The hook is that once the stranger's money has been safely transferred, the investor will be paid a percentage of the sum for their assistance, which translates into a much larger amount than initially invested. However, this message being a ruse to bilk the investor out of their money, the return on investment is never realized, much to the investor's chagrin and frustration. The most well known version of this fraud is the "Nigerian", or 4-1-9, scam, named after the section of the Nigerian criminal code that explicitly prohibits such actions. The scam has been conducted since at least 1989 in the form of physical mail, fax, and most recently through e-mail. While the fraud is commonly referred to as "Nigerian", this is partially derivative of the common use of this country in much of the earlier versions of such communicated messages. In fact, it is quite common for the stranger to claim residence in any number of countries both within and outside the continent of Africa. The scam itself has proven to be quite lucrative, especially over the Internet. In 2003, MessageLabs reported that the Nigerian scam grossed an estimated $2 billion dollars, ranking it as one of the top grossing industries in Nigeria. [7]

3 ScamSlam Architecture

In this section, we introduce the ScamSlam system along with the underlying models and methods. During the course of this research, we refer to three types of e-mail messages, ham, spam, and scam, the general descriptions of which follow. In Figure 2 we depict the exclusive and inclusive relationships between e-mail types. As stated above, spam messages are unsolicited pieces of email. The scam messages are a subset of spam messages that are intelligent in design, such that they attempt to coax the individual to perform some action of illegal purpose beyond a simple "click me". In contrast, "Ham" (a term introduced by John Graham [8]), refers to legitimate e-mail messages.

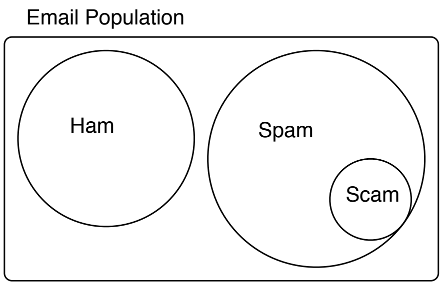

Figure 2: Different e-mail types and their exclusive and inclusive relationships. In general terms, ham corresponds to legitimite e-mail, while spam means non-legitimite. Scam messages are considered a subpopulation of spam.

Before delving into the technical details, we provide a brief sketch of the ScamSlam system. The ScamSlam system consists of three main components, as depicted in Figure 3: 1) a trained scam filter, 2) a message normalizer via a vector space projection method, and 3) an intelligent clustering engine.

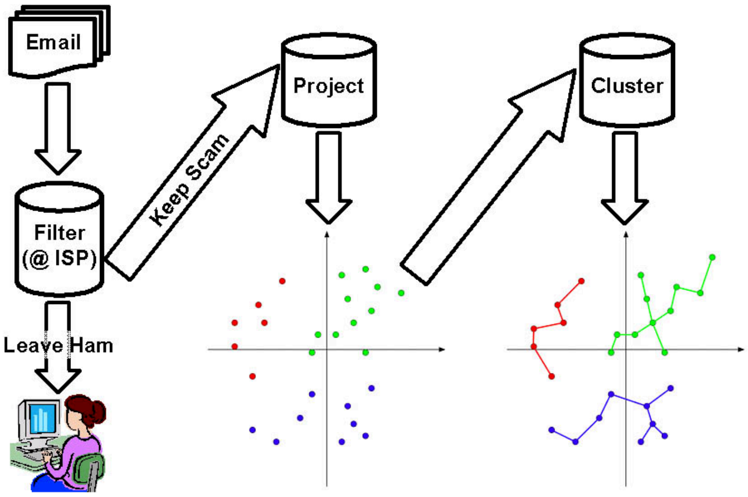

Figure 3: General overview of the the ScamSlam system. *Step 1)* Incoming messages from the general population of e-mails are filtered for scams. *Step 2)* Scam messages are projected into a Euclidean space for vector representation. *Step 3)* Messages are clustered based on similarity.

The first component of the system is a message filter that determines which messages contain the type of scam in question. The filter is trained to make a Boolean decision on a labelled dataset, where the labels are "scam" and "not scam". After the filter has been trained, it can be applied to messages incoming to a mail server in real time. Next, the scam messages are projected into a common space of representation. More specifically, the SlamScam system converts a scam message into a normalized vector of words. For each message, each word is assigned a weight that captures information about the frequency with which the word occurs in the message and in the set of scam messages under scrutiny. Once the documents have been normalized by the reweighting and representation process, the documents are clustered based on similarity. The current implementation of the system uses a hierarchical clustering method, specifically single linkage, which partitions the vector space into clusters of similar messages. The clustering method proceeds in a stepwise manner and terminates when no linkages can be constructed at a minimal level of message similarity. The minimal level, or threshold, is derived using a novel heuristic based on empirical observations of the studied scam messages.

In the following subsections, each component is described in further detail.

3.1 Poisson Filter

We begin our model with a short description of the filtration process. Briefly, a filter is a function that takes as input the word counts observed in a message and some parameters (to be defined below) and returns a decision about whether or not the message is scam. Specifically, our Poisson filter labels a message as scam if the probability of the message being scam given the counts of the words it contains is greater than the probability of the message not being scam given the counts.

Formally, we start with a corpus of p messages, $M = \{m_1, m_2, \ldots, m_p\}$, which are labelled as belonging to one of two categories, $C = \{Scam, Not\text{-}Scam\}$, so that $M = \cup_{c \in C} M_c$ is the union of disjoint sets of messages (M_c) in different categories. From M we extract a vocabulary of x unigrams, $V = \{v_1, v_2, \ldots, v_x\}$, defined as contiguous strings of letters. Let X_{mv} be a random variable denoting the counts for unigram v in message m. We assume that the counts for X_{mv} occur according to a Poisson distribution as in [9]:

$$p(X_{mv}|\omega_m,\, \mu_{v\,c}) \;=\; \begin{array}{l} \frac{e^{-\omega_m \mu_{v\,c}}\,(\omega_m \mu_{v\,c})^{x_{v\,m}}}{x_{mv}!}, \quad x_{mv} = 0, 1, 2, \dots . \\ \text{s.t.} \quad \omega_m > 0, \quad \mu_{v\,c} > 0, \end{array} \tag{1}$$

where ω_m is the length of message m in thousands of words, and μ_{vc} is the Poisson rate for unigram v in category c. The Poisson rate is the number of unigrams we expect to see in an arbitrary block of a thousand consecutive words of text from a messages of category c. During training, we assign a value to the parameter μ_{vc} of the Poisson model for both categories of messages by computing maximum likelihood estimates according to the following formula:

$$\hat{\mu}_{vc} = \frac{\sum_{m \in M_c} x_{mv}}{\sum_{m \in M_c} \omega_m}, \qquad \text{for each } c \in C. \tag{2}$$

Our filter is based on several simplifying independence assumptions. First, the random variables that represent unigram counts in a message, X_{vm}, are independent from one another. Second, the position of the random variables are independent within the text of the message. In our framework, we use the following ratio r_m to determine if it is probabilistically more likely that a message $m \in M$ is $Scam$ or not:

$$r_m = \frac{\prod_{v \in V} p(X_{mv} \mid \hat{\mu}_{v\,Spam})}{\prod_{v \in V} p(X_{mv} \mid \hat{\mu}_{v\,No\text{-}Spam})} \tag{3}$$

When r_m is greater than 1, we classify a message as $Scam$, otherwise it is classified as $Not\text{-}Scam$.

3.2 Message Representation

After filtering the scam spam messages, we project them into a normalized multi-dimensional space, the details of which are as follow. Recall that we represent the corpus of messages as a set $M = \{m_1, m_2, \dots, m_p\}$, from which we extract the vocabulary $V = \{v_1, v_2, \dots, v_x\}$, which is the set of distinct unigrams, or strings of contiguous letters, found in the messages. Each message $m_i \in M$ is converted into a vector model, such that each message is represented as a n-size vector, $\vec{m} = [x_{m1}, x_{m2}, \dots, x_{m|V|}]$, where each value x_{mv} corresponds to the observed number of times that term v appears in message m.[11]

Each vector is then re-weighted, or normalized, to account for the relative frequencies of terms in the set of messages M. The weights, components of a normalized vector, represent the term frequency - inverse document frequency scores. With respect to message m, term frequency (tf) corresponds to the number of times a term v is observed in a message, normalized by the maximum frequency term in m, such that term frequency for term t in message m is $tf_{mv} = \frac{x_{mv}}{max_t x_{mt}}$. While the term frequency weight accounts for the relative frequency of a term within a message, the inverse document frequency (idf) accounts for the relative frequency of a term among messages. Specifically, let obs_v represent the number of messages that term v is observed in, the inverse document frequency score idf_v equals $log(\frac{|M|}{obs_i})$. Combining term frequency and inverse document frequency, we re-weighted messages are represented as the $\vec{m'} = [w_{m1}, w_{m2}, \dots, w_{m|V|}]$, where $w_{mv} = tf_{mv} \times idf_v$.

We measure the similarity between a pair of messages $\vec{m_i}$, $\vec{m_j}$ using the cosine of the angle between the two vectors as explained in the following section.

3.3 Scam Clustering

ScamSlam clusters messages using single linkage over the corresponding weighted vector representations. Single linkage is a hierarchical clustering technique that targets messages which display high similarity between pairs. [12] As clustering proceeds, each message belongs to one and only one cluster at any particular time during the clustering process. The way clustering proceeds is as follows. Let $thresh$ be a threshold of similarity which defines the boundary at which two messages can be considered to belong to the same cluster or not. Initially, each message is a singleton cluster consisting of only itself, so there exist $|M|$ clusters. As clustering proceeds, two arbitrary clusters l_i and l_j are merged into a single cluster if there exists one message m_a in l_i and one message m_b in l_j such that the distance between them does not exceed $thresh$. ScamSlam uses a distance measure, $dist(\vec{m_i}, \vec{m_j})$, induced by the cosine similarity:

$$dist(\vec{m_i}, \vec{m_j}) = 1 - \frac{\sum_{k=1}^{n} w_{ik} \times w_{jk}}{\sqrt{\sum_{k=1}^{n} w_{ik}^2} \times \sqrt{\sum_{k=1}^{n} w_{jk}^2}}. \quad (4)$$

The choice of single linkage addresses one of the observed means by which scam spam authors operate. Specifically, a very useful component of single linkage clustering is its ability to permit messages within a cluster to be very different from each another. Over time, the writers of scam spam can change any number of features, such as the motive for money transfer of the name and title subject of who is in need of help. Moreover, sections of the story or plead may change as well, such as when a paragraph of the message is removed or added. It is not uncommon to find that over time, there is a continual tweaking of the scam, where a part of the scam is changed while keeping most parts in common.

For example, The left panel of figure 4 shows 10 messages as two-dimensional vectors of tf-tdf scores, and the corresponding clusters formed by using a unit threshold. Notice that messages 5 and 3 are closer than messages 1 and 7, but they do not cluster together because $d(5,3) > D_*$, whereas in the rightmost cluster all messages connect to a neighbor closer than D_*.

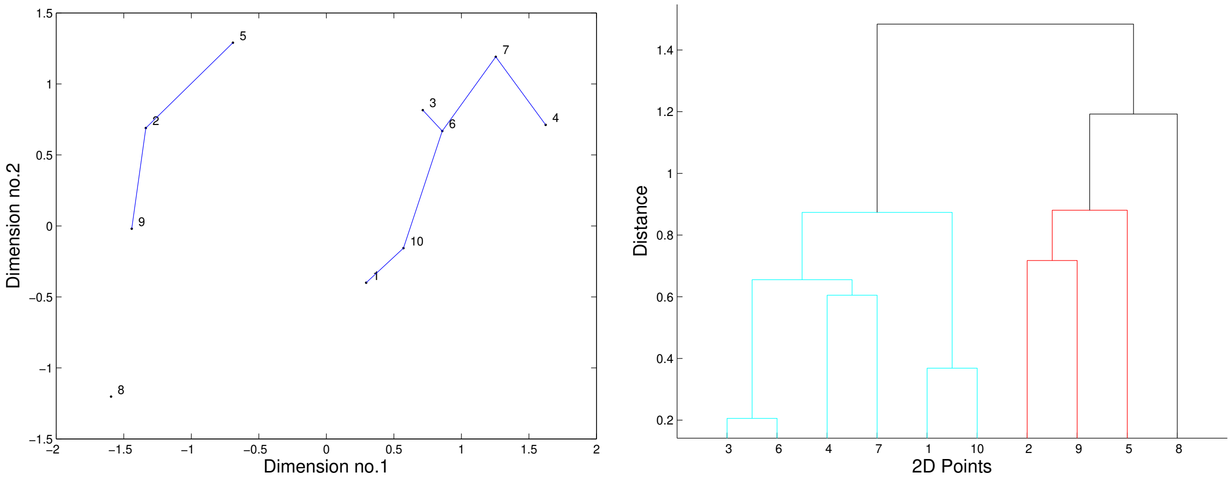

Figure 4: Left: Example clusters for randomly generated messages from a vocabulary of 2 words (2D vectors); we used a threshold $D_* = 1$ to form the clusters. Notice that single linkage allows for a situation where $D_* < d(5,3) < d(1,7)$, and where the messages $\{3,5\}$ do not belong to the same cluster but the messages $\{1,7\}$ do. Right: The dendrogram corresponding to the 2D vectors in the left panel, obtained using single linkage.

The clustering methods we use are unsupervised, which means there is no feedback provided to the linkage process. In other words, if given the opportunity, clustering would proceed until there is only one cluster! Clearly this is undesirable and is counterproductive to the goal of partitioning messages into sects of similarity. As a result, the process must embed some type of stopping criterion. In the description of the single linkage clustering method above we termed this arbitrary criterion $thresh$. More formally, we use distance as a threshold parameter for our model and term it the maximum distance of membership D_*. This distance serves as a threshold that facilitates the decision of whether a message belongs to a certain cluster

(C). For example, we assign message m to cluster C if the distance between m and any of the messages already in cluster C is less than D_*.

This method of scoring and clustering provides law enforcement officials with the capabilities to pursue two strategies for searching and persecuting criminals. First, in the presence of evidence from a criminal group, progressive clustering via an increasing value for D_* provides an ordered list of suspects by ranking the messages closest to the cloud of messages that constitute the evidence. Second, in the absence of evidence, law enforcement officials can increase the minimum distance D_* and grow clusters, each of which can be regarded as a possible pocket of criminal activities worthy investigating further, again ranked by similarity. An aspect of interest is a good heuristic to decide whether there is enough evidence in the data to justify the fusion of small pockets of illegal activity. In order to answer this question we use the following metric F_D:

$$F_D = \frac{\sum_{i=1}^{|M|}\sum_{j=i+1}^{|M|} \phi(dist(m_i, m_j))}{\frac{|M|(|M|-1)}{2}}, \qquad \text{where } \phi(x) = \begin{cases} 1, & \text{if } x \leq D \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

which measures the fraction of all message pair distances within threshold D. This measure leverages the geometry of the vector space of messages. More specifically, F_D measures how clusters grow, and we set D_* at the point where the growth rate is slow or stagnant for a period of time. The intuition behind this heuristic is that if there are defined clusters, we will discover them when D_* equal to approximately the radius of the majority of the clusters, but less than the distance needed for these well defined clusters to merge. Thus, even if after the period of stagnancy there is an increase in the rate of growth, we suspect that this growth is due to the merging of clusters which should remain independent will begin merging. The lack of growth in cluster sizes is found by minimizing a smoothed version of the first derivative of the F_D.

4 Experiments

For our experiments, we used five different datasets, one for the scam messages, and two for each of the remaining types of messages, spam and ham. The scam corpus consists of 534 messages posted to the Nigerian Fraud Email Gallery.[1] [10] Each message was previously been classified as the Nigerian 4-1-9 scam by the proprietor of the website. The messages dates span the time period from April 2000 to April 2004 and are distinct, such that no two messages are duplicates. The spam-A and ham-A corpora were collected and supplied by Greg Hartman (a graduate student at Carnegie Mellon University), who collected the messages over a four month period. There are 2944 spam and 7651 ham messages. The spam-B corpus was collected by Dr. Latanya Sweeney (Carnegie Mellon University); it contains 2532 spam messages. Finally, we assembled the ham-B corpus by selecting 75 posts from each of seven newsgroups, for a total of 525 ham messages. There are approximately 200,000 distinct unigrams in the combined spam-B and ham-B corpus.

4.1 Scam Filtering

Before studying the relationships within a set of scam spam messages, we must address how one goes about filtering scam messages from the deluge of messages flowing through the Internet. We performed a preliminary study to assess how well widely used spam filters would be at recognizing scam messages as spam. To do so, we subjected the combined scam, spam-A, ham-A corpus to analysis and classification by SpamAssassin™, the popular open source spam filter. [13] SpamAssassin uses a set of rules and a

[1] The corpus is publicly available and can be found at http://potifos.com/fraud/

Bayesian classifier to determine if a message is spam or not. It ultimately assigns a message with a total score which denotes the degree to which SpamAssassin considers a message as spam. The more negative a SpamAssassin score is, the lower the probability that the message is spam.

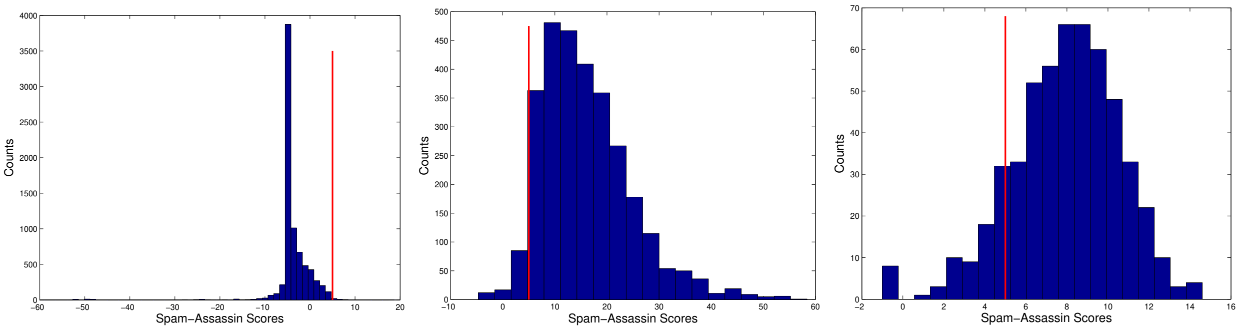

Figure 5: Distribution of SpamAssassin scores for test corpora. Scores for *left*) ham, *center*) spam, and *right*) Nigerian scam corpus. The thin vertical line at $x = 5$ represents the default threshold value for which messages are considered spam, (*i.e.* a message with a score greater than 5 is considered spam). We notice an increase in the "falsely classified as ham" rate from $\approx 4\%$ for spam to $\approx 12\%$ for scam.

The messages were scored using SpamAssassin. While users of SpamAssassin are afforded with the ability to set their threshold for spam classification, the default value for SpamAssassin is 5.0. Thus, if the score for a spam or scam message was less than 5.0 we consider the message to be misclassified. Similarly, for ham messages that score greater than or equal to 5.0. Side-by-side histograms of the resulting scores are depicted in Figure 5 with the threshold score depicted by a thin vertical line. The classification and misclassification rates are provided in Table 1. Based on the observed scores, SpamAssassin does very well at classifying ham as ham, However it has a more difficult time classifying the other message types and disproportionately so for spam versus scam. As seen in Figure 5, SpamAssassin misclassifies about $\approx 4.1\%$ and $\approx 11.8\%$ of the spam and scam messages as ham, respectively.

		SpamAssassin Prediction	
		Ham	**Spam**
Reality	**Ham**	7624 (99.65%)	27 (0.35%)
	Spam	122 (4.14%)	2822 (96.86%)
	Scam	63 (11.8%)	471 (88.2%)

Table 1: Average confusion matrix for SpamAssassin (*7651 ham, 2944 spam, and 534 scam messages*).

It appears that whereas SpamAssassin performs extremely well on the task it was engineered for, separating spam from ham, it is not able to accurately extract scam messages, which is reasonable as this type of email is not very frequent. For ScamSlam, however, the identification of scam messages is a crucial step to learn hidden criminal patterns, and we need to be more accurate on scam than on spam. Therefore we further explored the problem of scam classification. In order to do so, we trained and tested a Poisson classifier [9] using a balanced 5-fold cross-validation scheme[2], and performed an additional set of experiments. In the first Poisson classification test, the ham-B corpus was considered as one class and we combined both the

[2]This means that all messages are split into two classes A and B, each of which is partitioned into 5 equal-sized exclusive sets of messages (i.e. $A_1, A_2, \ldots, A_5$, such that $A_1 \cup A_2 \cup \cdots \cup A_5 = \emptyset$). The classifier was trained on eight of the partitions, four from each class, and we tested the trained classifier on the remaining two classes. This scheme was used to test the classifier in five separate runs, such that each of the partitions for each class is tested one time.

		Poisson Prediction	
		Ham	**Spam+Scam**
Reality	**Ham**	516 (98.29%)	9 (1.71%)
	Spam+Scam	74 (2.41%)	2992 (97.59%)

Table 2: Average confusion matrix of Poisson classifier obtained via 5 fold cross validation, over a corpus of 3591 messages (*525 ham, 534 scam, and 2532 spam*).

		Poisson Prediction	
		Ham+Spam	**Scam**
Reality	**Ham+Spam**	2803 (99.57%)	13 (0.43%)
	Scam	0 (0%)	534 (100%)

Table 3: Average confusion matrix of Poisson classifier obtained via 5 fold cross validation, over a corpus of 3591 messages (*525 ham, 534 scam, and 2532 spam*).

scam and spam-B corpora for the second class. With classes defined as such, this classification experiment is equivalent to the traditional spam filter (or spam classification problem). In the second Poisson experiment, we consider the problem of directly filtering scam from the general population of email messages. Therefore, the first class consists of both ham-B and spam-B messages, while the second class consists solely of scam messages. Using 5000 unigrams, we observe the results as shown in tables 2 and 3. We chose to use 5000 unigrams in both our experiments, since this number minimizes the cross-validated misclassification error (ham erroneously tagged as spam or scam) as shown in the right panel of Figure 6.

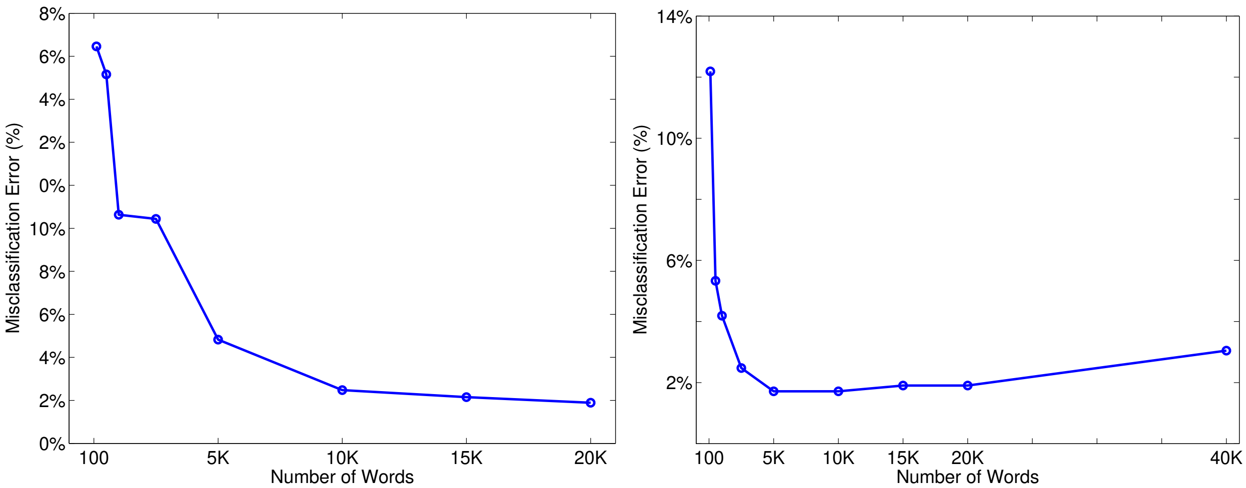

Figure 6: Poisson misclassification ham as one class and spam and scam (spam+scam) combined as a second class. *left*) Spam+Scam misclassified as ham. *right*) Ham misclassified as spam+scam.

It is worth noting that a decision about the number of words, or equivalently about the threshold for SpamAssassin, is essentially a policy decision about which type of mistake is more important. The cross-validated misclassification error plots in figure 6 decreases sharply as more strongly discriminating words are used, and eventually starts increasing after too many weakly discriminating words are used. The Poisson classifier makes a decision by weighting and composing into a linear combination the probabilities of each message being of one category rather then the other; ideally we would want few strongly discriminating words pushing the sum in one direction or another, whereas too many small terms introduce confusion and, in the end, misclassification errors. In our experiments we assessed how good of a discriminator each unigram was on the training set, for each fold, according to their information gain, and that is the ordering that we used for the X axes in figure 6.

4.2 Clustering Analyses

For the following unsupervised clustering experiments, we continue with the Nigerian scam corpus. All header information was removed so that clustering was performed with only the text of the messages. One of the assumptions that we incorporate into this analysis is that messages which form clusters are scattered at nonuniform levels of density in the vector space of tf-idf weights. Since, the measure F_D captures the density of message clustering in the vector space, we empirically tune D_* according to the observed growth rate. We observe in Figure **??** the growth rate of F_D is minimized at a distance of 0.6. Though the global minimum is realized at the boundary point, this is an artifact of the fact that all messages are clustered at distance equal to 0.9. While the growth rate in messages clustered continues to grow beyond 0.6, this is mainly due to the uniform distribution of single message clusters. At this point we begin to observe that large clusters which are well defined at a relatively low threshold (below 0.6) begin merging.

At D_* equal to 0.6, we uncover approximately 20 clusters of size 5 or larger, where the largest cluster consisted of 40 messages. These clusters account for approximately half of the total corpus. A section of the distribution of messages to criminal clusters is shown in the dendrogram to the right of Figure 8.

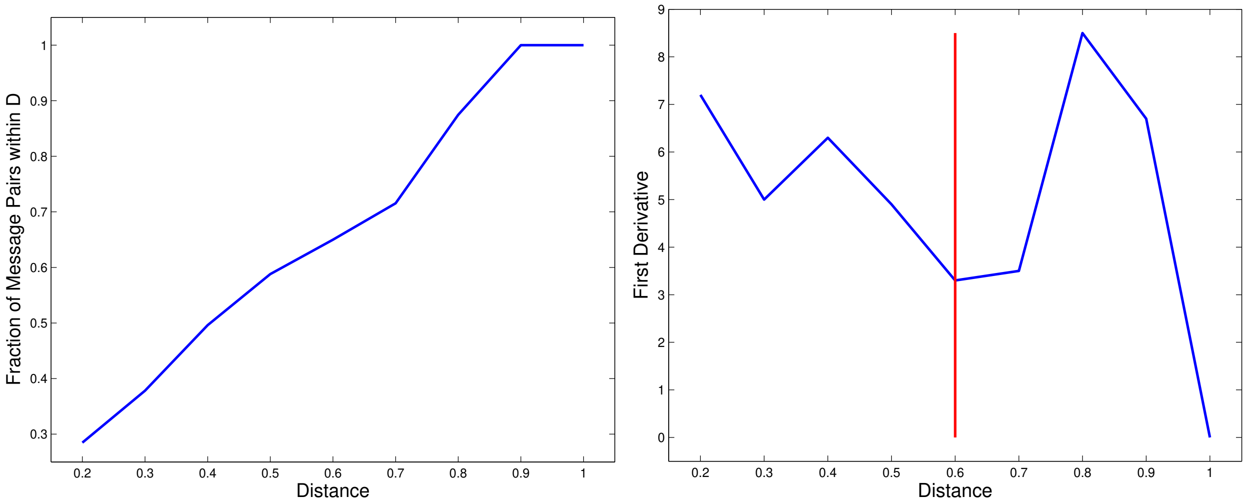

Figure 7: *Left*) F_D, the fraction of pairs of messages far apart less than D, versus the distance D in the Nigerian dataset. *Right*) the first derivative of F_D versus D in the Nigerian dataset suggests a distance $D_* = 0.6$ as a good value for the threshold that controls the number of clusters.

5 Discussion

Under the current ScamSlam implementation, the scam filter is trained and validated on labelled data. The hidden relationship learner (consisting of the latter two system components of message projection and clustering), however, is trained and tested on data that is independent of the reality regarding the actual relationships between authors. This is a limitation of the system which derives from a lack of available data

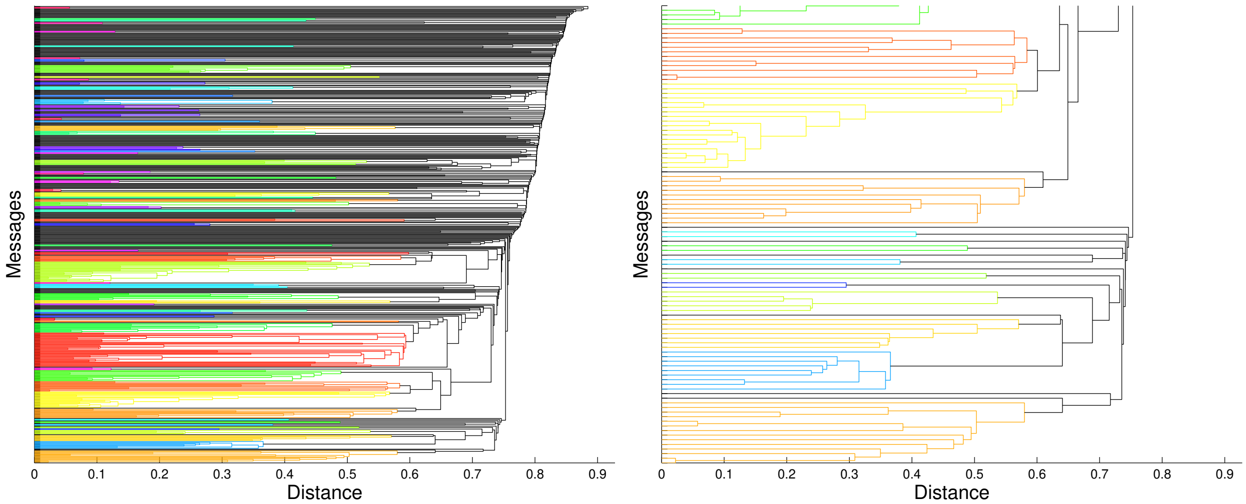

Figure 8: Dendrograms derived from the Nigerian dataset using the cosine distance metric and single linkage clustering. Different colors represent independent clusters at $F_D = 0.6$. *Left*) All messages and clusters. *Right*) Detailed portion of the dendrogram.

for validation. With the incorporation of validation data, we will be to determine an optimal value for the clustering threshold. In addition, we will be able to compare different combinations of distance metrics and heuristic methods to correct the current threshold value. Thus, one the next steps in this research is to obtain a labelled dataset which supplies real authorship and/or criminal relationships behind the messages.

Despite this limitation, there are several findings from our studies that are of notable interest, which we now elaborate on.

5.1 Spam, not ScamAssassin

First, in order to study the relationships between scam messages there must exist some method by which scam messages are captured. This was our initial reason for testing SpamAssassin's ability to filter scam messages. As the results demonstrate, SpamAssassin is not as capable of filtering scam from ham than spam from ham. This difference is significant, given that we observe a threefold difference in SpamAssassin's false classification of such messages. Based on these findings it is clear that a different type of system is necessary for filtering scam messages from the general population of e-mail. This is not overly surprising since one would expect the typical scam message used in our studies to be much more similar to the average ham than spam message. Moreover, the overall goal of SpamAssassin is the classification of spam in general, of which scam is only a fraction. This is supported by the disproportional misclassification rate and as the, distribution of the range of scores, observed in the SpamAssassin filtering experiments as shown in Figure 5.

5.2 It's All Scam To Me

The results reported in this research are based on a particular scam, the advance fee fraud. Scam messages of this type are susceptible to analysis by ScamSlam partially as a result of being several paragraphs in length and somewhat verbose. The combination of these characteristics permits the use of a significant number of discriminative features for both filtering and learning scam authorship. Based on our finding, we expect similar results with other types of e-mail scams, such as securities and bank fraud. An extension to our analyses is to determine the usefulness of the ScamSlam system with types of e-mail fraud that communicate much less information in the message body. Given the rise of phisher fraud e-mail over time, the AOL, PayPal, and Ebay scams are of particular interest. However, even though phisher frauds may communicate less information in an email, the websites which they redirect individuals to are amenable to study via ScamSlam as well. This is because ScamSlam, at its core, is basically a text analysis tool, which permits analysis on e-mail messages, webpages, or any other type of information communicated via text.

5.3 Extending the ScamSlam System

In general, the only criteria that single linkage clustering requires is that there must exist a logical path of data points between any two data points in the cluster. As a result of this criteria, clusters learned via single linkage tend to have a bias to be more "elongated" in the vector space than clusters learned through other clustering criteria. In certain settings this is considered a limitation, however, this method is a preferred representation for a hypothesis regarding how scam messages are used by groups of authors. Recall that our hypothesis of scam authorship is that scam messages are reused, such that each time the message is recycled a certain component of the message is changed, but not the whole of the message. With each change, the new scam message deviates a little further from the previous version of the initial scam message.

While the scam dataset is devoid of the reality regarding relationships, the temporal aspect of our hypothesis may permit its validation via an alternative route. If scam messages are both reused and changing over time, then it is possible that scam clusters can be modelled as an evolutionary process. That is, the spam message within a cluster can be partially ordered on the dates messages were sent. If the cluster is indeed an evolutionary process, then we expect that several features will be observable. First, one would expect that the linkages within clusters will reveal the partial ordering on time. The temporal ordering may be the result of a continual changing of messages, such that each scam message is message is changed only one time to yield the next scam message. Second, as in many evolutionary processes there may exist bifurcations in the family tree of scam. Such bifurcations will manifest when a single scam message is used as the basis for two or more lines of message augmentation, each of which can sustain an independent line of evolution. It is interesting to note that the single linkage criteria provides an ideal setting for analyzing such patterns since the returned clusters represent spanning trees over a set of messages. The search for such patterns within scam messages is a fruitful direction for research, especially in the absence of validated data. Though we have yet to attempt such analysis, this is a logical progression of our research.

In addition, the temporal aspect of e-mail may assist in the design of useful heuristics for clustering. For example, one simple heuristic based on time is to incorporate the message date as a feature for measuring the distance between messages. Caution and intuition must be used with such a heuristic since it may predispose messages to cluster in a manner such that authorship relations are eroded. This would more likely be the case if date was considered as part of the cosine measure of distance. Used in this way, clusters would bias toward messages of similar time points, which would not necessarily help to discern between criminal groups perpetrating during the same time period. Rather, it seems more feasible that such a heuristic would be more useful to guide the addition of messages already assigned to a particular cluster, possibly as a tie-breaker criteria. For instance, if a message is equidistant from two or more messages in the same cluster, then it appears more intuitive to assign a linkage between the documents which are closer in date.

6 Conclusions

The methods used in this research integrate hierarchical clustering and geometric insights in the message similarity space for a simple heuristic to establishing common source behind disparate scam messages. In combination, the methods developed in this research enable the learning of relationships between criminal sects sending scam spam messages. As a result, this work provides the basis for a novel forensic tool to assist law enforcement agencies in tracking criminals for which some evidence has been gathered in the form of electronic content. In particular, leveraging scientifically validated linkages, our methods strengthen the case against individuals and criminal rings by using fragments of evidence to construct a stronger case for legal intervention. We have confidence that our methods are useful for law enforcement and surveillance purposes, however, one barrier to the adoption of such methods is the current validation through unsupervised learning techniques. This work would be greatly benefited if we could obtain a labelled dataset, which denotes the reality regarding the individuals and groups engineering e-mail scams. With such information we will be able to not only validate our techniques, but formally tune our heuristic parameters.

Acknowledgements

The authors wish to extend thanks to the members of the Data Privacy Laboratory, especially to Latanya Sweeney, for their insightful suggestions, discussion, and support. This research was supported by the Data Privacy Laboratory in the Institute for Software Research International in the School of Computer Science at Carnegie Mellon University.

References

[1] Federal Trade Commission. Controlling the assault of non-solicited pornography and marketing act of 2003 (CAN-SPAM Act). RIN 3084-AA96. 16 CFR Part 316. Nov 22, 2003.

[2] Federal Trade Commission Press Release. Law enforcement posse tackles internet scammers, deceptive spammers. Federal Trade Commission. May 15, 2003.

[3] FTC vs. Zachary Keith Hill. United States District Southern Court of Texas. File No. 032-3102. December 8, 2003.

[4] Brightmail, Inc. http://www.brightmail.com/

[5] Federal Trade Commission. National and State Trends in Fraud and Identity Theft: January - December 2003. Federal Trade Commission. January 22, 2004.

[6] Warner B. Billions of "phishing" scam emails sent monthly. Reuters News Service. May 6, 2004.

[7] Leonard D. E-mail threats increase sharply. IDG News Service. December 12, 2002.

[8] Graham J. Fooling and poisoning adaptive filter systems. Sophos White Paper. Sophos Inc, USA. November 2003.

[9] Airoldi E.M. and Fienberg S.E.. Who wrote Ronald Reagans radio addresses? Technical Report no. CMU-STAT-03-789, Department of Statistics, Carnegie Mellon University, 2003.

[10] Sullivan B. Nigerian scam continues to thrive. MSNBC News. March 5 2003. Available online at http://msnbc.msn.com/id/3078489/

[11] Baeza-Yates R. and Ribeiro-Neto B. Modern Information Retrieval. Addison Wesley. New York. 1999.

[12] Duda R., Hart P., and Stork D.. Pattern Classification, Second Edition. John Wiley & Sons, Inc. New York. 2001.

[13] SpamAssassin^{TM}. Available online at: http://www.spamassassin.org/