View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Calhoun, Institutional Archive of the Naval Postgraduate School

Calhoun
Institutional Archive of the Naval Postgraduate School

Calhoun: The NPS Institutional Archive

Faculty and Researcher Publications Faculty and Researcher Publications

2007-06

# Do Word Clues Suffice in Detecting Spam and Phishing?

Rowe, Neil C.

Monterey, California. Naval Postgraduate School

Proceedings of the 8th IEEE Workshop on Information Assurance, West Point, NY, June 2007.

DUDLEY
KNOX
LIBRARY

Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

# Do Word Clues Suffice in Detecting Spam and Phishing?

Neil C. Rowe, David S. Barnes, Michael McVicker, Melissa Egan,
Richard Betancourt, Rommel Toledo, Douglas P. Horner,
Duane T. Davis, Louis Guiterrez, and Craig H. Martell

**Abstract –** Some commercial antispam and anti-phishing products prohibit email from "blacklisted" sites that they claim send spam and phishing email, while allowing email claiming to be from "whitelisted" sites they claim are known not to send it. This approach tends to unfairly discriminate against smaller and less-known sites, and would seem to be anti-competitive. An open question is whether other clues to spam and phishing would suffice to identify it. We report on experiments we have conducted to compare different clues for automated detection tools. Results show that word clues were by far the best clues for spam and phishing, although a little bit better performance could be obtained by supplementing word clues with a few others like the time of day the email was sent and inconsistency in headers. We also compared different approaches to combining clues to spam such as Bayesian reasoning, case-based reasoning, and neural networks; Bayesian reasoning performed the best. Our conclusion is that Bayesian reasoning on word clues is sufficient for antispam software and that blacklists and whitelists are unnecessary.

**Index terms** – spam, phishing, clues, words, testing

*This paper appeared in the Proceedings of the 8th IEEE Workshop on Information Assurance, West Point, NY, June 2007.*

## I. INTRODUCTION

In January of 2004 Bill Gates of Microsoft announced at the World Economic Forum that spam would disappear by 2006. Apparently Microsoft felt that its technology for correlating broadcast mailing combined with its blacklisting and whitelisting was working so effectively on Microsoft's Hotmail email service that it could similarly work on other big Internet email services that own Microsoft software, and it would solve the spam problem for everyone. That however did not occur, though the volume of spam email did decrease by 2006 [1]. What happened was that spammers turned to other techniques such as using less obvious language, buying their own domains, and using botnets, techniques that do not require Microsoft products and are not affected by blacklists [2]. We have recently seen an increase in more serious forms of spam such as phishing, or theft of personal data through fraudulent Internet sites [3]. This increase in phishing is worrisome for organizations with secrets since it can be a very cost-effective intelligence gathering [4].

Word clues to spam and phishing would seem the most robust method for detecting spam because they cannot be eliminated if the sender wants their message to influence someone, and influence is the goal of spam [5]. Words are central even in image-based spam. Management of word clues requires constant training processes for the software, however, and this imposes some burden on the user to approve or disapprove email. Word clues have also been criticized as being imperfect. So the question is whether their performance is sufficiently good to be preferable to a blacklisting or mail-correlation approach. This paper will investigate a range of methods for implementing word clues, and will compare them to other clues.

## II. EXPERIMENTS WITH HUMAN DETECTION OF SPAM AND PHISHING

Previous work of ours investigated to what extent people could detect documents (in particular, software requirements) that had been modified randomly [6]. Our subjects were able to detect surprisingly subtle modifications even though they were not familiar with the domain of the requirements. No correlation of performance with the number of unknown words (those not in a 58,000-word English wordlist) in a paragraph was observed, so lack of knowledge of technical terms did not affect performance. These results are consistent with automated attempts to detect manipulated text [7]. These experiments suggest that fake documents can be detected from analysis of their text even when no obvious clues are present. Note that none of the clues given for text by [8] applied here:

"offensive usage", "incomplete sentences", "ellipsis", "wordy", "sentence variety", "jargon usage", "run-on sentences", "second person", or "possessive form".

Recently we studied whether people could detect phishing email, email that directs users to a site where their personal information such as credit-card numbers is stolen, often for purposes of identity theft. Phishing is important because its rates are increasing while other spam is holding steady. Earlier work found that consumers were quite gullible about email and Web sites in 2000, but this is changing quickly as more media attention has focused on Internet fraud. [9] surveyed people to determine the clues they used to detect phishing Web sites, but this did not address detection of the earlier and more critical deception in the initial email.

So we conducted our own survey. We used 27 adult subjects that included both academics and nonacademics. For this we used ten examples presented on paper to the subjects, five of phishing and five of "false alarms", normal messages that looked close to phishing. Each example was a pair, an email printout and a view of the accompanying Website reached when the victim clicked on the link in the email. All these examples appeared equally suspicious to superficial inspection, so we expected the subjects to look more closely at them.

Our survey found 31% false positives and 14% false negatives, so people are clearly making errors of both types. That the false positive rate was significant should be important news for the organizations whose legitimate requests for updating of records looked so suspicious, and should suggest some immediate email and Website redesign. Examples that generated the most false positives were an overly vague bank message and a message asking for update of salary data by government employees.

## III. AUTOMATED-DETECTION EXPERIMENTAL SETUP

Spam is unsolicited commercial email sent in bulk mailings with typically low response rates. It is a major annoyance and reducing it is important on email servers. A defense is a trainable "spam filter" that rules out email with particular clues like the appearance of spam-associated words [10]. But comparisons of algorithms for filtering and clues to filter on have been so far inconclusive [11, 12] though Bayesian methods work well [13]. Opinions of spammers are amusing but contradictory [14]. We chose to compare trainable methods of automatically detecting spam so they could be personalized unlike general-purpose filters. Trainable filters can handle, for instance, someone who frequently purchases items online and receives legitimate mail from vendors concerning those purchases, while similar mail received by another person could be spam. All our programs were written in Java.

For experiments, we collected 4043 examples of spam email and 2656 examples of legitimate email, from which we randomly chose a training set of 2699 spam and 1817 "ham" or nonspam, and a test set of 1344 spam and 839 ham. Examples came from email archives of 12 individuals including the authors, plus archives from eight well-spaced days in www.spamarchive.org. Most of the training and test sets was in English, but some was in German and Spanish. The training set had 797,376 total words in the spam and 412,093 words in the ham.

Clues to spam follow a binomial model where the expected value is $e/(n+e)$ and the standard deviation is $\sqrt{ne/(n+e)}$ where e is the number of occurrences of the clue in spam and n is the number of occurrence of the clue in nonspam or "ham". We considered as significant all clues whose probability deviated from the mean by more than S times the standard deviation, where S is a key parameter.

## IV. EXPERIMENTS WITH AUTOMATED SPAM DETECTION

### *A. Clues from the message source*

Microsoft apparently correlates many mail messages to detect those sent in bulk. Those sites responsible for sending such mail are warned and then blacklisted so that mail from them is either flagged as suspicious or dropped. However, this tends to discriminate against sites without infrastructure to discipline an occasional abusive user, such as small and minimally funded sites. It also can unfairly hurt sites that are "innocent bystanders" to spoofing.

So an important question is how much spamming varies per source site. We extracted the sites names in our training set (3282 in spam and 1498 in ham) and computed conditional probabilities of spam (Table 1). While these statistics do show some obvious spam sites and obvious ham sites, high-spam sites do include innocent Internet service providers victimized by spam like freespace.to and sbcglobal.net. It appears unfair to blacklist or whitelist based on such counts.

### *B. Word clues for spam*

Spam text provides strong clues by particular rare words like "Viagra" and "Lagos" that are rarely associated with legitimate email. Thus we can obtain a high degree of recall and precision by focusing first on these words – although that is not the end of the story, since users expect very high recall and precision in spam filters.

From our training sets, we calculated conditional probabilities of words appearing in spam, ignoring capitalization. We excluded numbers and 3466 "stopwords" collected in our previous data-mining research [15], of which 602 were ordinary words like "we", "this", and "thing" plus the names of antispam sites and tools, and 2864 were personal names (excluding those with common English meanings like "Hall"). We included message-header information (so the words of the subject and originating address could be clues) except for lines starting with "X-" since those in www.spamarchive.com often represented forwarded messages

**Table 1: Representative conditional probabilities of spam based on source site in our training set.**

| Prob. | Spam | Ham | Site |
|---|---|---|---|
| 0.0 | 0 | 28 | pomona.edu |

| | | | |
|---|---|---|---|
| 0.0 | 0 | 26 | netflix.com |
| 0.0 | 0 | 24 | lists.dci.pomona.edu |
| 0.0 | 0 | 17 | nwdc.navy.mil |
| 0.0 | 0 | 17 | xtra.co.nz |
| 0.0 | 0 | 11 | wylelabs.com |
| 0.0 | 0 | 10 | dci.pomona.edu |
| 0.0 | 0 | 9 | fastmail.fm |
| 0.0 | 0 | 8 | cs.pomona.edu |
| 0.0 | 0 | 7 | baesystems.com |
| 0.0 | 0 | 7 | nsf.gov |
| 0.071 | 1 | 13 | sbcglobal.net |
| 0.114 | 4 | 31 | lifescapeinc.com |
| 0.222 | 2 | 7 | cox.net |
| 0.285 | 4 | 10 | mail.communications.sun.com |
| 0.461 | 6 | 7 | redshift.com |
| 0.466 | 14 | 16 | aol.com |
| 0.500 | 5 | 5 | JJILL.COM |
| 0.500 | 7 | 7 | comcast.net |
| 0.666 | 18 | 9 | gmail.com |
| 0.846 | 11 | 2 | msn.com |
| 0.892 | 50 | 6 | hotmail.com |
| 0.900 | 9 | 1 | earthlink.net |
| 0.917 | 78 | 7 | yahoo.com |
| 1.0 | 6 | 0 | arena.sci.univr.it |
| 1.0 | 6 | 0 | bellsouth.net |
| 1.0 | 6 | 0 | mweb.co.th |
| 1.0 | 6 | 0 | serifsoftware.com |
| 1.0 | 8 | 0 | excite.com |
| 1.0 | 8 | 0 | heavypockets.net |
| 1.0 | 10 | 0 | lycos.com |
| 1.0 | 12 | 0 | hotmail.ca |
| 1.0 | 13 | 0 | chase.com |
| 1.0 | 13 | 0 | paypal.com |
| 1.0 | 16 | 0 | eveningdate.net |
| 1.0 | 18 | 0 | freenet.de |
| 1.0 | 24 | 0 | clickta.com |
| 1.0 | 34 | 0 | newsletterabo.com |
| 1.0 | 53 | 0 | freespace.to |

from spam filters including spam likelihood ratings. Example positive word clues for spam were “astrology”, “porno”, “kitchenkraft” “phpmailer”, “offersonthenet”, and “oin01”. Example negative word clues for spam were “assert”, “humane”, “wessex”, “tomcat”, “algebra”, and “america”.

Table 2 shows precision values for given recall values in detecting spam with just word clues on our test set. (Recall is defined as the fraction of the spam in the test set that was identified by our program; precision is defined as the fraction of actual spam in the set of email identified as spam by the program.) We show only data for recall at least 90% because we figure no one wants a spam filter that fails to catch at least 90% of the spam. N is the number of word clues used, S is the minimum number of standard deviations of the deviation of the conditional probability of the word in spam from its binomial expected value, M is the minimum necessary count on the word in both spam and ham, and W is the number of successive words considered for clues. The Type is the evidence combination method used for multiple word clues: O is the geometric mean (antilog of the average of the logs) of the odds, G is the geometric mean of the probabilities, and A is the arithmetic mean of the probabilities. These three combination methods are simplifications of the most popular probabilistic evidence combination methods in the literature: Naïve Bayes on the odds, straight Naïve Bayes, and support vector machines.

Ignore the last four rows for now. Of the other rows, the parameters of the first appear to be the best. The fourth row shows best for 0.92 and 0.90 recall, but we figure that performance at higher levels of recall is more important. The arithmetic mean approach does not appear very good at all, so it does not appear that support-vector machines on the raw data are a good choice for word clues to spam; the geometric mean was better, but still consistently inferior to the odds mean. Using either more or fewer word clues decreased performance, so apparently 2559 clues is desirable. Finally, rows 10 and 11 show that inclusion of probabilities for pairs of words occurring more than 15 times like “computer science" (negative clue), "company website" (negative clue), and "actual results" (positive clue) did not help performance.

Only roughly 5% of this corpus was phishing, so we could not test if its clues differed from those of other spam. But we did test this further as reported in section 5.

**Table 2: Precision as a function of recall for word clues to identify spam email in our test set.**

| N | S | M | W | Type | 1.0 | .98 | .96 | .94 | .92 | .90 |
|---|---|---|---|---|---|---|---|---|---|---|
| 2559 | 3 | 30 | 1 | O | .616 | .810 | .883 | .926 | .932 | .944 |
| 2559 | 3 | 30 | 1 | G | .616 | .782 | .844 | .884 | .896 | .909 |
| 2559 | 3 | 30 | 1 | A | .616 | .616 | .630 | .650 | .715 | .745 |
| 4196 | 3 | 15 | 1 | O | .616 | .798 | .873 | .921 | .948 | .957 |
| 4196 | 3 | 15 | 1 | G | .616 | .749 | .853 | .892 | .908 | .918 |
| 1576 | 10 | 30 | 1 | O | .616 | .779 | .865 | .905 | .932 | .940 |
| 1576 | 10 | 30 | 1 | G | .616 | .774 | .835 | .872 | .894 | .904 |
| 3085 | 1 | 30 | 1 | O | .616 | .808 | .875 | .917 | .931 | .940 |
| 3085 | 1 | 30 | 1 | G | .616 | .808 | .858 | .882 | .902 | .912 |
| 2688 | 3 | 30 | 2 | O | .616 | .779 | .865 | .905 | .932 | .940 |
| 2699 | 3 | 30 | 2 | G | .616 | .774 | .835 | .872 | .894 | .904 |
| 4498 (+LIWC) | 3 | 30 | 1 | O | .616 | .823 | .892 | .921 | .934 | .944 |
| 2559 | 3 | 30 | 1 | O+fit1 | .616 | .823 | .897 | .924 | .932 | .945 |
| 2559 | 3 | 30 | 1 | O+fit2 | .616 | .832 | .874 | .907 | .927 | .939 |
| 2559 | 3 | 30 | 1 | O+fit3 | .616 | .834 | .892 | .924 | .936 | .946 |

### *C. Generalizing word clues*

A list of 2559 words represents a relatively weak theory of spam, since new spam words occur all the time. One approach is to generalize the word clues into categories. We investigated this by analyzing our training sets with the LIWC software (www.liiwc.org) that categorizes text into 89 categories by counting words and punctuation of particular kinds. This software identifies 2319 distinct significant words and is good at identifying emotional appeals in text. We counted the occurrences of the LIWC words in our training set, and noted significant increases in spam over nonspam in the categories of question marks, negative emotions, anger, families, television, music, money, metaphors, religion, death, body words, sex, and food; there were significant decreases in pronouns, "we", and words concerning motion and occupation.

We were interested in whether the LIWC words could extend the coverage of our training sets to some related words that did not occur sufficiently often in our training set to generate reliable probabilities. To study this, we calculated a conditional probability of occurring in spam in our training set for each LIWC class. For LIWC words that did not occur sufficiently often in our training set, we used the probability for their LIWC class as their conditional probability. The fourth row from the bottom of Table 2 shows the results on our test set; clearly the additional words helped.

### *D. Lexical clues*

We considered five additional categories of clues to spam:

- Lexical: Counts of words in grammatical and other interesting categories in an email message.
- Header: Features of the message header, including author, path, time, and subject.
- Miscellaneous: Other message-body features such as punctuation patterns, links, image links, and attachments.
- Case-based reasoning: Measured similarity of the message to known spam and ham.
- Neural network: Measured mathematical fit to the parameters of known spam.

For lexical analysis, we first tried the probabilistic tagger QTag, but found that it placed too much weight on expected sentence structure in determining the most likely part of speech for a given word. Although this approach might work for typical English prose, it did not on spam messages, many of which included nonsensical sequences to confuse spam filters. So we tagged using a dictionary-based approach combining WordNet (www.cogsci.princeton.edu/~wn) and the "Moby Part-of-Speech II" database (prdownloads.sourceforge. net/wordlist/post1.tar.gz) that listed the most likely part of speech for each of about 218,000 words. This database had the standard grammatical categories to which we added categories of mixed case (both upper and lower case), stopword, initial-letter capitalized, number, symbols containing both letters and numbers, HTML, and unknown.

The 12 most useful word categories, in terms of minimizing the false negative plus false positive rate when used with an optimal threshold, were non-noun (0.59), letters and numbers (0.67), HTML (0.74), no initial capital (0.75), not in word list (0.76), not definite article (0.84), not conjunction (0.84), not mixed case (0.85), not stopword (0.86), not pronoun (0.87), not preposition (0.89), and not participle (0.89). By manual experimentation, the best joint indicators of spam in the training set were the percentage of non-nouns, the percentage of words containing both letters and numbers, the percentage of HTML words, the percentage of words that were not mixed-case, and the percentage of words that were not participles. We got a minimum sum of recall plus precision of 49%, so results for these clues alone were unimpressive, but they could supplement word clues.

### *E. Message-header clues*

Analysis of the headers of email was cited as especially useful by [16]. Most (but not all) of our sample email came with headers. Some information is straightforward to extract such as time of day the email was sent, the date as well as the day the email was sent, message content type, and domain of the sender's email address. "Received" headers are useful from every server the email traverses en route to the receiver. For the header in Figure 1, a spammer lied that he was sending the email from dial45.neoms.mail.us with an IP address of 245.15.75.158. However, the mail server knew the spammer was actually coming from kil-dial35.asysijd.cz with an IP address of 195.75.66.68. Getting these Received headers does require special software tools since most mail programs do not show them. Other useful information from the Received headers is their number, since this suggests distance, and chains of multiple Received listings.

```
Received: from [dial45.neoms.mail.us[245.15.75.158]]
(kol-dial35.asysijd.cz[195.75.66.68]) by
mail.domain.com (8.8.7/8.8.7) with SMTP…
```

**Figure 1: Example suspicious email header.**

To determine if an IP address or domain name is valid (working), a DNS (Domain Name Server) lookup was performed. Validity is important because many scam sites are only operate for a few days. We tested clues of (1) a legitimate received "from" entry, (2) an illegitimate received "from", (3) number of Received headers > 2, (4) number of Received headers ≤2, (5) a legitimate Received chain (if more than two Received header entries), (6) an illegitimate Received chain, (7) content type header, (8) day of week sent, (9) date sent, (10) email sent during normal hours (0800-2359), (11) email not sent during normal hours. Of these, only clues 1, 4, 6, 7 for text format, 10, and 11 were significant for the training set, but none of the clues were misleading. We obtained the recall-precision curve in Figure 2 when using all the clues.

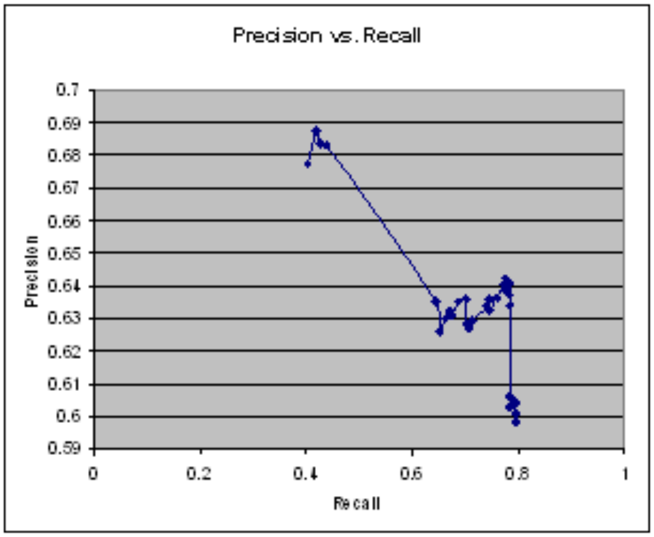

**Figure 2: Precision version recall for header analysis alone.**

### *F. Miscellaneous message-body clues*

We also investigated clues in HTML codes (e.g. "<strong>" and "FONT-FAMILY"), text formatting codes (e.g. "&reg"), unusual punctuation marks (e.g. "!" and "\"), and string names of encoding algorithms used in email (e.g. "JavaMail" and "Unsubscribe"). Clues were considered if they occurred at least 10 times in the training set and the conditional probability of spam when they appeared was greater than 0.7, the fraction of spam in the full training set. Despite our hopes, total size of the email was not a useful clue, and most of our clues were ineffective on short emails. We tested on a set of 200 emails of which 113 (0.61) were spam, and got 0.68 recall with 0.71 precision, so the clues were helpful. Increasing the threshold quickly worsened performance.

*G. Case-based reasoning*

In preliminary experiments with a smaller test set, we got good performance by a case-based reasoning approach. (Good results were also obtained by [17].) The K nearest neighbors in the training set to the given example in the test set were calculated, and if the majority of them were spam examples, the test example was labeled as spam. 32 boolean and scalar features of the email were used that experiments showed could not be omitted without decreasing performance:

- Number of body addresses
- Body length
- Unusual punctuation in the subject line
- Money words in the subject
- Sex words in the subject
- Money words in the body
- Sex words in the body
- Unusual punctuation in the body
- Exclamation marks in the subject
- Sender is .com
- Sender is .gov or .mil
- Sender is .org or .edu
- Sender is non-U.S.
- Message is a response
- Message is a forward
- HTML in the body
- Links in the body
- Secure links in the body
- Words referring to email
- Words referring to reports
- Words referring to user actions
- “From” phrases
- Words referring to the Internet
- Words referring to subscriptions
- Announcement phrases
- “Opt out” phrases
- “Instruction” phrases
- Insurance phrases
- Ordering phrases
- “Too good to be true” phrases
- Pronouns
- Mime content

Distance was computed as the weighted sum of feature differences, calculated as the absolute value of the numerical difference for boolean or fractional features, and otherwise $1-(1/(1+C_i\left|(\Delta f)\right|))$ where C is a scaling factor specific to the feature. The formula was necessary to keep factors in the range of 0 to 1.

Recall and precision are shown in Figure 3 as a function of K, the number of nearest neighbors considered. Small values seemed to work best, from which we conclude that similarity to a single known spam was usually sufficient. Surprisingly, performance was not improved by intuitively assigned weights to the features, from which we conclude that all features are roughly equal in importance. A subsequent experiment with a smaller subset of features did not perform as well (e.g., recall of 0.826 for precision of 0.613 on an initial cluster size of 50).

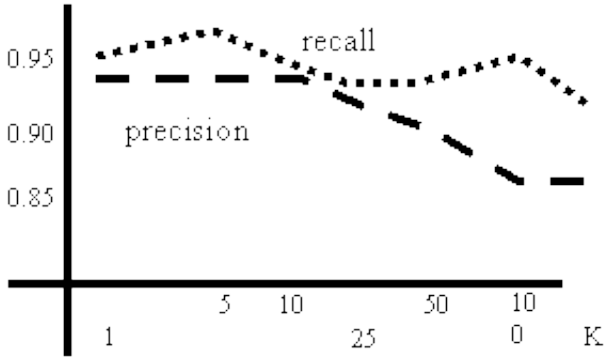

**Figure 1: Recall and precision as a function of average cluster size for case-based reasoning.**

*H. An artificial neural network*

We also built an artificial neural network to detect spam. The inputs were strong word clues from both the body and header, counts of unusual punctuation, categories of the sender's site, and the presence of images and other attachments in the message. The network had three layers, with three neurons at the middle level, one neuron at the output, and a neuron at the input layer for each of several intuitively-selected words as well as for other factors (for 43 inputs in all). Training tried to alternate spam examples with non-spam examples, and a "momentum" factor was used to keep weights changing in a consistent direction. Convergence was better with initial random weights than with uniform weights. Performance on the test set improved with up to 300 runs through the training examples, and then started to decrease with overfitting. After 300 runs we had 0.89 recall with 0.93 precision, 0.73 recall with 0.80 precision, and 0.33 recall with 0.93 precision on a smaller training set. Since performance was inferior to the case-based approach on similar clues, we did not explore this approach further.

*I. Combining clues from all modules*

To assess the relative importance of clue categories, we did a linear regression on the total assessments of email from all but the neural network approach, five in all, where the target value was 1 for spam and 0 for nonspam. (While a fit to the logarithms would be more consistent with Naïve Bayes, the difference in the calculation is negligible when a single factor predominates.) The weights of the best linear fit were negative for the header clues and case-based similarity, so we discarded their data and refit. (Most of the test set did not have full header information, and case-based reasoning used clues covered by other modules.) We then obtained weights of 0.896 for the word clues, 0.029 for the lexical clues, and 0.075 for the miscellaneous message-body clues. The third-to-last row of Table 2 shows the results for recall and precision with a weighted average using these weights, with the parameters of the first row of the table for the word clues. The fit definitely improved upon the word clues alone for the desirable high-recall situations, despite the small size of the weights on the other clues.

Figure 4 plots typical behavior of recall and precision as a function of the threshold. While the curves are not smooth, irregularity is not pronounced, so our results are probably similar over a wide range of email examples.

We were surprised that the header clues performed so poorly. So we separately examined just one of them, the time of the day of time expressions in an email. This should help since the probability of spam was 95% for email having a time from midnight to 1AM, but only 44% for email having a time from 10AM to 11AM. The recalculated regression fit weights were 0.757 for word clues, 0.017 for lexical clues, 0.070 for miscellaneous message-body clues, and 0.168 for the new time clue. The results of using these weights are shown in the second-to-last line of Table 2. This improved precision for recall of 0.98, but decreased it for recall of 0.96 and 0.94, so there is a tradeoff. We tried halving the weight on the new factor and obtained the results in the last line of the table; this only decreased precision for 0.96 recall, so it looks better. It may be that least-squares fitting tends to work hard to minimize extreme cases which are not important to worry about with spam.

So we have justified the approach endorsed in [5] of focusing heavily on word clues, though some other clues will help a little as a supplement. Our results contradict those of [18] which found advantages in using over 50 kinds of clues. Note that we have not investigated context-dependent clues such as associations between email [19], which can provide a "guilt-by-association" factor for characterizing email.

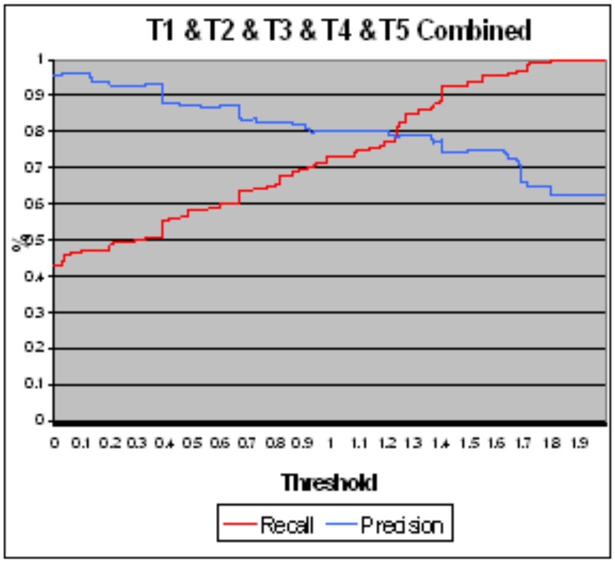

**Figure 4: Recall and precision on all clues for a smaller (250-example) test set.**

## V. DETECTING PHISHING

Because of our success with word clues for spam, we also examined their usefulness for phishing [20]. We used 200 examples from our own email and from www.antisphishing.org, and used our spam database for nonexamples since so little of it was phishing. Example positive clues for phishing (more than two standard deviations from the expected value) were “renew”, “verify”, “bank”, “alert”, “notify”, and “immediate.” Example negative phishing clues were “course”, “student”, “week”, “data”, “question”, “technology”, “width”, and “style.” Two-word phrases like “renew immediately” were clues but did not add much, as with spam detection. As with spam, we used odds multiplication to calculate a cumulative probability with multiple word clues. Performance was not as good as for spam. We got for instance 0.115 precision with 0.352 recall, 0.206 precision for 0.242 recall, and 0.562 precision for 0.180 recall. Still, this was good as people did in our phishing survey in section 2.

Some additional clues beyond those of spam are possible for phishing to compensate for the decreased value of word clues. Since phishing requires a link to a Web page, any email not containing such a link is innocent; since the link is often deceptive, any email with the link identical to the visible text is very likely innocent. Table 3 shows the testing of some clue candidates on the 100 most recent examples of phishing at www.antiphishing.org on March 14, 2007 and using our spam training set for nonexamples. It can be seen that the presence of a traditional text link and the word “bank” are good clues, but other seemingly good candidates are not.

**Table 3: Testing of phishing clues on examples from www.antiphishing.org.**

| **Clue** | **Prob. in 100 examples** | **Prob. in non-examples** |
|---|---|---|
| **Message has a link** | 1.00 | 0.20 |
| **Link different from visible text for it** | 0.90 | 0.98 |
| **Digits in URL** | 0.55 | 0.68 |
| **Executable in URL** | 0.17 | 0.19 |
| **Link is an image** | 0.03 | 0.41 |
| **“%” or “@” in URL** | 0.07 | 0.06 |

| | | |
|---|---|---|
| **"Bank" in link text or URL** | 0.28 | 0.00 |
| **"PayPal" in link text or URL** | 0.08 | 0.06 |
| **"EBay" in link text or URL** | 0.13 | 0.06 |

Other more ambitious clues for phishing are also possible. Since phishing imitates very specific Web sites and email, known spam targets such as banks and other online financial services can store their legitimate email and Web pages in a database, and new email and Web sites can be compared to them; strong coincidences may be phishing. A "whois" lookup to determine whether a site is recent will often provide a good clue for phishing.

## VI. CONCLUSIONS

Detection of spam and phishing can be automated, but obtaining the high performance necessary for useful systems is a challenge. Our experiments confirm that word clues are by far the best, though a few other clues can also contribute a few additional percentage points of accuracy. This should not change much in the future although image-based spam is becoming more common, because effective spam images needs text to motivate people to do rather specific things, and such text can be extracted from images by character-recognition methods.

## REFERENCES

[1] MessageLabs, "2005 Annual Security Report", retrieved January 26, 2006 from www.messagelabs.com/ Threat_Watch/Intelligence_Reports.

[2] J. Goodman, G. Cormack, & D. Heckerman, "Spam and the Ongoing Battle for the Inbox," *Communications of the ACM*, Vol. 50, No. 2, February 2007, 24-33.

[3] Fraudwatch.com, "Internet Fraud", retrieved March 11, 2005 from www.fraudwatchinternational.com/internetfraud/internet.htm.

[4] H. Berghel, "Phishing Mongers and Posers", *Communications of the ACM, 49* (4), April 2006, 21-25.

[5] J. Zdziarski, *Ending Spam: Bayesian Content Filtering and the Art of Statistical Language Classification,* No Starch Press, San Francisco, CA, 2005.

[6] N. Rowe, B. Duong, and E. Custy, "Fake Honeypots: A Defensive Tactic for Cyberspace," 7th IEEE Workshop on Information Assurance, West Point, NY, June 2006, pp. 223-230.

[7] S. Kaza, S. Murthy, & G. Hu, "Identification of Deliberately Doctored Text Documents Using Frequent Keyword Chain (FKC) Model, IEEE Intl. Conf. on Information Reuse and Integration, October 2003, pp. 398-405.

[8] A. Vrij, *Detecting Lies and Deceit: The Psychology of Lying and the Implications for Professional Practice*. Wiley, Chichester, UK, 2000.

[9] R. Dhamija, J. Tygar, & M. Hearst, "Why Phishing Works", Proceedings of the Conference on Computers and Human Interaction, April 2006, Montréal, QB, Canada, pp. 581-590.

[10] K. Yoshida, F. Adachi, T. Washio, H. Motoda, T., Homma, A. Nakashima, H. Fujikawa, & K. Yamazaki, "Density-Based Spam Detector", Proceedings of Conference on Knowledge and Data Discovery, August 2004, Seattle, WA, 2004, pp. 486-493.

[11] C.-C. Lai & M.-C. Tsai, "An Empirical Performance Comparison of Machine Learning Methods for Spam E-Mail Categorization", Proceedings of Conference on Hybrid Intelligent Systems, December 2004, pp. 44-48.

[12] R. Matsumoto, D. Zhang, & M. Lu, "Some Empirical Results on Two Spam Detection Methods", Proceedings of Conference on Information Reuse and Integration, November 2004, pp. 198-203.

[13] J. Hildalgo, G. Bringas, & E. Sanz, "Content Based SMS Spam Filtering," Proc. Intl. Conf. on Document Engineering, Amsterdam, 2006, 107-114.

[14] "Spammer X", *Inside the Spam Cartel*, Syngress, Rockland, MA, 2004.

[15] N. Rowe, "MARIE-4: A High-Recall, Self-Improving Web Crawler that Finds Images Using Captions," *IEEE Intelligent Systems*, 17 (4), July/August 2002, 8-14.

[16] Anonymous, "Reading Email Headers", retrieved March 21, 2006 from www.stopspam.org/email/headers.html.

[17] M. Sasaki & H. Shinnou, "Spam Detection Using Text Clustering", Proceedings of Conference on Cyberworlds, November 2005, pp. 316-319.

[18] T. Lynam, G. Cormack, & D. Cheriton, "On-Line Spam Filter Fusion," Proc. of 29th ACM SIGIR Conference, Seattle, WA, 2006, 123-130.

[19] S. Macskassy & F. Provost, Suspicion scoring based on guilt-by-association, collective inference, and focused data access. Proc. of 2005 Intelligence Analysis Conference, McLean, VA, May 2005, http://analysis.mitre.org.

[20] D. Barnes, "A Defense-in-Depth Approach to Phishing", M.S. thesis, Naval Postgraduate School, September 2006.

Manuscript received March 7, 2007. This work was supported in part by the National Science Foundation under the Cyber Trust program. The authors are with the Naval Postgraduate School, Code CS/Rp, 1411 Cunningham Road, NPGS, Monterey CA 93943 USA, ncrowe at nps.edu.