Spam / Junk E-Mail Filter Technique

Hrishikesh P.
Research Scholar, MCA
Thakur Institute of Management Studies,
Career Development and Research (TIMSCDR)

ABSTRACT

Most e-mail readers spend a significant amount of time regularly deleting junk e-mail (spam) messages, which are a part of marketing campaigning efforts of various companies wherein users normally signed in and it also results in increasing volume of storage space and consumes network bandwidth. A challenge, therefore, rests with the developers and improvement of automatic classifiers that can differentiate authentic e-mail from spam. Spam detectors normally use Naïve Bayesian approach and large feature sets of binary attributes that determine the existence of common keywords in spam emails. Spammers/Marketers recognize these approaches to impede their messages and have developed tactics to bypass these filters, but these ambiguous tactics are themselves patterns that human readers can often identify quickly. The preliminary study tests an alternative approach using a neural network (NN) classifier to overcome drawbacks of Naïve Bayesian approach. This approach uses a feature set, which uses descriptive characteristics of words and messages similar in the way that users would use to identify spam

General Terms

Junk mail, HTML Component, human reader, NN methodology

Keywords

Spam, HTML, nSL, nLS, e-mail, URL, Neural Network

1. INTRODUCTION

The quantity of junk e-mail (spam) transmitted on the Internet is in huge proportions. The inconvenience of unwanted e-mail messages was identified as early as 1975 – the volume of spam's was relatively limited until the mid-1990s. Spam mails quantity was simply 8% of network e-mail traffic in 2001 but has swollen to about 40% of e-mail traffic today. One research firm has projected that the cost to combat spam across the U.S. was around $10 billion in 2003[1][3].

Many commercial and open-source products exist to suffice the increasing need for spam classifiers, and a variety of methodologies have been developed and applied toward the problem. The simplest and most common method is to use filters that display messages based upon the presence of common words or expressions common to junk e-mail (spam). Other tactics include blacklisting (rejection of messages received from the addresses of known spammers) and white listing (acceptance of message received from known and trusted recipient). Effective spam filtering technique uses a combination of these three methods. The primary defect in the first two methods is that it relies upon complacence by the spammers by considering that they are not likely to forge. White listing risks the possibility that the receiver will miss authentic e-mail from a known or expected recipient with an unknown address.

To overcome these drawbacks, Naïve Bayesian approach was proposed, that inspected manually-categorized messages for a set of words, expressions and non-textual characteristics (such as the time of initial transmission or the existence of attachments). These methodologies used binary attributes, where Xn = 1 if a property is represented, else Xn=0. In each case, the words were manually-derived selections. In addition to these methods, several solutions exist that claim high success rates (99.5%) with Naïve Bayesian filters.

While Naïve Bayesian technique performs effectively, it suffers from two inherent problems. The first is that they count upon a consistent terminology used by the spammers. Frequently used words must be identified as they appear in use of new spam, and, in the case of hash tables, any new word must be assigned an initial random (probability) value when it is created. Spammers use this drawback in generating spam with strings of random characters. The second problem is one of content. Binary word- characteristics, and expression-characteristics, do not classify the common outlines use in spams that humans can easily identify, such as unusual spellings, images and hyperlinks, and patterns such as HTML components, which are typically hidden from the recipient. In summary, Naïve Bayesian classifiers are indeed immature, and require considerable modifications for each e-mail classification. A human reader requires comparatively little calculation to infer if a given e-mail is a genuine message or spam. While spammers send messages that vary widely in structure, subject, and style, they typically include classifiable strategies that are designed to draw attention or to bypass spam filters. These ambiguous strategies are patterns that human readers can often identify quickly.

In this research work we apply a neural network (NN) approach to the classification of spam using characteristics comprised from expressive characteristics of the ambiguous patterns that spammers employ, rather than the content or frequency of keywords in the messages. This methodology produces similar results but with fewer attributes and is much more effective than the Naïve Bayesian approach[4][5].

2. METHODOLOGY

This project was carried out on 1654 e-mails over a period of several months. None of the mails contained attachments. Each e-mail message was saved as a text file, and then analyzed to identify each header element to differentiate them from the body of the message. Every substring within the subject header and the message body that was delineated by white space was considered to be a token by an alphabetic word of English alphabetic characters (A-Z, az) or apostrophes. The tokens were assessed to create a set of 17 features from each mail.

Table 1. Features extracted from each e-mail

Feature	Features From the Message Subject
	Header
1	Number of alphabetic words that did not contain any vowels
2	Number of alphabetic words that contained at least two of the following letters (upper or lower case): J, K, Q, X, Z
3	Number of alphabetic words that were at least 15 characters long
4	Number of tokens that contained non-English characters, special characters such as punctuation, or numeric digits at the beginning or middle of the token.
5	Number of words with all alphabetic characters in upper case
6	Binary feature indicating occurrence of a character (including spaces) that is repeated at least three times in succession: yes = 1, no = 0
	Features From the Priority and Content-Type Headers
7	Binary feature indicating whether a priority header appeared within the message headers (X-Priority and/or X- MSMail-priority) or whether the priority had been set to any level besides normal or medium: yes = 1, no =0
8	Binary feature indicating whether a content-type header appeared within the message headers or whether the content type of the message has been set to "text/html": yes = 1, no = 0
	Features From the Message Bod
9	Proportion (fraction) of alphabetic words with no vowels and at leastseven characters
10	Proportion of alphabetic words that contained at least two of the following letters in upper or lower case: J, K, Q, X, Z
11	Proportion of alphabetic words that were at least 15 characters long
12	Binary feature indicating whether the white-space-delimited strings "From:" and "To:" were both present: 1 = yes, 0= no
13	Number of HTML opening comment tags
14	Number of hyperlinks ("href=")
15	Number of clickable images represented in the HTML
16	Binary feature indicating whether a color of any text within the body message was set to white: 1 = yes, 0 =no
17	Number of URLs within hyperlinks that contain any numeric digits or any of three special characters ("&", "%" or "@") in the domain or subdomain(s) of the link

The e-mails were manually classified into 800 authentic e-mails and 854 junk e-mails. Half of each category was randomly selected to embrace a training set (n = 827) and the remaining e-mails were used as a testing set. All feature values were scaled to range from 0 to 1.

The training data were used to train a 3-layer, back proliferation neural network with the number of hidden nodules ranging from 4 to 14 and the number of periods from 100 to 500. After training, the messages of the testing set were classified to obtain precise outcomes[10].

3. RESULTS

The relative success of spam filtering procedures is determined by classic processes of correctness and recall on the testing subsets of authentic and spam messages. Spam precision (SP) is defined as the percentage of messages considered as spam that actually are spam. Legitimate precision (LP) is the percentage of messages considered as genuine that are certainly authentic. Spam recall (SR) is defined as the share of the number of correctly-classified spam messages to the number of messages originally classified as spam. Similarly, legitimate recall (LR) is the amount of correctly-classified authentic messages to the number of messages originally classified as authentic[7]. Thus, we outline the counts:

nSS = the number of spam messages correctly categorized as spam.

nSL = the number of spam messages wrongly categorized as legitimate.

nLL = the number of legitimate messages correctly categorized as legitimate.

nLS = the number of legitimate messages wrongly classified as spam.

And the precision and recall formulas:

(1) Spam Precision (SP) = ___________

(2) Legitimate precision (LP) =___________

(3) Spam recall (SR) = ___________

(4) Legitimate recall (LR) =___________

Table 2 gives the results on the testing set by hidden nodule count and training periods. The trial with 12 hidden nodules and 500 periods produced the lowest number of misclassifications, with 35 of the 427 spam messages (8.20%) classified as valid (nSL), and 32 of the 400 authentic messages (8.00%) classified as spam (nLS), for a total of 67 misclassifications[8].

Hidden Nodes	**Training Epochs**	**Spam**		**Legtimate**	
		Precision (%)	**Recall (%)**	**Precision (%)**	**Recall (%)**
8	300 400 500	91.81 90.95 93.73	96.65 89.46 97.59	86.56 88.94 87.62	91.75 90.50 93.75
10	300 400 500	92.11 91.09 92.48	90.16 86.18 86.42	89.73 86.05 86.45	91.75 91.00 92.50
12	300 400 500	93.52 91.73 92.45	87.82 88.29 91.80	87.79 87.98 91.32	93.50 91.50 92.00
14	300 400 500	91.58 92.04 91.28	84.07 86.65 88.29	84.37 86.59 87.92	91.75 92.00 91.00

Out of the 35 misclassified spams, 30 were short in length, including HTML tags. Remaining 5 messages: 1 had many “comments” without comment delimiters; 2 were written completely in ASCII codes; 1 followed 4 image files with English words, and 1 creatively used an off-white color for fonts to mask the random characters added at the end of the e-mail.

Classifier	**Num feat**	**Num msgs**	**Spam %**	**SP %**
NN (12 nodes, 500 epochs)	17	827	51.6	92.5
Naïve Bayesian from				
Words	500	1789	88.2	97.1
Words+Phrases	500	1789	88.2	97.6
Words+Phrases+Non-textual	500	1789	88.2	100
Naïve Bayesian from				
Bare	50	1099	43.8	95.1
List	50	1099	43.8	96.8
Lemmatized	100	1099	43.8	98.3
Lemmatized + Stop List	100	1099	43.8	98.0

The 32 valid messages were misclassified due to characteristics that are unfamiliar for personal email. 22 had the features normally activated by spam: 6 were from a known recipient that prefers to write in white typeface on a colored background, 10 were responses that quoted HTML that triggered several features, 5 were commercial e-mail from known vendors, and 1 was ranked as “low” priority from a known recipient. The remaining 10 messages: 4 included special characters or vowel-less words in the subject header, 3 had several words of rare English characters, and 3 had a rare number of hyperlinks[13].

The NN accuracy is similar to the Naïve Bayesian filters and Table 3 presents evaluation.

SR %	**Num msgs**	**Spam %**
91.8	827	51.6
94.3	87.7	88.2
84.3	N/A	N/A

To test correctness of valid and spam messages are labeled by the blacklist databases, IP addresses of the messages is entered that were incorrectly labeled by the NN classifier into a site that sends IP addresses to 173 spam database and returns the number of hits. We entered both the original IP address as well as second IP address (mail server or ISP), if present[3][5][6].

Because we measured single-list hits to be anomalies since they aren't confirmed as blacklists, we calculated only hit greater than 1 as spam that would be blacklisted. The debarred outcomes are presented in Table 4. While the percentages of genuine messages considered spam are lower than the percentages of spam appropriately recognized as spam [3].

Classification	**Blacklisting (% Considered Spam**		
	1st IP Address (Original Address)	**2nd IP Address (E-mail Server/ISP)**	**Either 1st or 2nd IP Address**
nLS (32 E-mails)	53.1	25.0	53.1
nSL (35 E-mails)	40.0	60.0	97.1

4. CONCLUSION

Although the NN methodology is precise and useful, its spam correctness performance is not high enough to be used without administration. For this method to be more useful, the feature set would require modifications. However, NN required fewer features to attain outcomes similar to the Naïve Bayesian method, representing that characteristics of words and messages can be used efficiently to differentiate spam by a filters. An arrangement of keywords and characteristics may provide more accurate arrangement. A NN classifier using these expressive characteristics, however, may not reduce over time as swiftly as classifiers that depend on fairly static vocabulary. Approaches that apply a combination of procedures, such as a NN would likely produce better results.

5. REFERENCES

[1] Androutsopoulos, I; Koutsias, J; Chandrinos, K. V. and Spyropoulos, C. D. 2000. An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-mail Messages. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (Athens, Greece, 2000), pp. 160-167.

[2] Burton, B. 2002. SpamProbe – Bayesian Spam Filtering Tweaks. ttp://spamprobe.sourceforge.net/paper.html; last accessed November 17, 2003.

[3] Cranor, L. F. and LaMacchia, B. A. 1998. Spam! Communications of the ACM, 41(8): pp. 74-83.

[4] Declude, IP Lookup Against a List of All Known DNS-based Spam Databases.

[5] http://www.declude.com/junkmail/support/ip4r.htm; last accessed January 27, 2004.

[6] Graham, P. 2002. A Plan for Spam. http://www.paulgraham.com/spam.html; last accessed November 17, 2003.

[7] Graham, P. 2003. Better Bayesian Filtering. In Proceedings of the 2003 Spam Conference (Cambridge, Massachusetts, 2003). See http://spamconference.org/proceedings2003.html.

[8] Hauser, S. 2002. Statistical Spam Filter Works forMe. http://www.sofbot.com/article/Statistical_spam_filter. html; last accessed November 17, 2003.

[9] Hauser. S. 2003. Statistical Spam Filter Review. http://www.sofbot.com/article/Spam_review.html; last accessed November 17, 2003.

[10] Sahami, M.; Dumais, S.; Heckerman, D. and Horvitz, E. 1998. A Bayesian Approach to Filtering Junk E-mail. In Learning for Text Categorization—Papers from the AAAI Workshop (Madison, Wisconsin, 1998), AAAI Technical Report WS-98-05, pp. 55-62.

[11] Sebastiani, F. 2002. Machine Learning in Automatic Text Categorization. ACM Computing Surveys (CSUR), 34(1): pp. 1-47.

[12] Spertus, E. 1997. Smokey: Automatic Recognition of Hostile Messages. In Proceedings of the 14th National Conference on AI and the 9th Conference on Innovative Applications of AI (Providence, Rhode Island, 1997), pp. 10581065.

[13] Weiss, A. 2003. Ending Spam's Free Ride. NetWorker, 7(2): pp. 18-24.