# Improving Spam Filtering by Detecting Gray Mail

Wen-tau Yih
Microsoft Research
One Microsoft Way
Redmond, WA, USA
scottyih@microsoft.com

Robert McCann
Microsoft
One Microsoft Way
Redmond, WA, USA
robmccan@microsoft.com

Aleksander Kołcz
Microsoft Live Labs
One Microsoft Way
Redmond, WA, USA
ark@microsoft.com

## ABSTRACT

We address the problem of *gray mail* – messages that could reasonably be considered either spam or good. Email users often disagree on this mail, presenting serious challenges to spam filters in both model training and evaluation. In this paper, we propose four simple methods for detecting gray mail and compare their performance using recall-precision curves. Among them, we found that email campaigns that have messages labeled differently are the most reliable source for learning a gray mail detector.

Preliminary experiments also show that even when the gray mail detector is imperfect, a traditional statistical spam filter can still be improved consistently in different regions of the ROC curve by incorporating this new information.

## 1. INTRODUCTION

Spam filters are faced with the challenge of distinguishing messages that users wish to receive from those they do not. At first glance this seems like a clear objective, but in practice this is not straightforward. For example, it has been estimated that two-thirds of email users prefer to receive unsolicited commercial email from senders with whom they've already done business, while one-third consider it spam [2]. There are many similar types of mail that are not clearly spam or good mail, such as newsletters and legitimate advertisements. We call this mail *gray mail*.

The gray mail problem can be treated as a special kind of label noise. Instead of accidentally flipping the label from spam to good or vice versa by mistake, different users may simply have different email preferences, which are reflected on the inconsistent labels of gray mail. Another reason is that individual users change their own preferences over time. For example, it is common for a user who tires of a particular newsletter to begin reporting it as spam rather than unsubscribing [1]. Some companies also do not respect unsubscribe requests and continue sending mail that some users then consider spam. In all cases the effect is the same – senders send mail that is not clearly spam or good and spam filters are faced with the challenge of determining which subset of this mail should be delivered.

The presence of gray mail raises two major problems for global anti-spam systems. First, because gray mail is not clearly good or spam by definition, it makes accurate evaluation of a filter performance a challenge, when personalization or user preference is ignored. Second, labels assigned to gray mail for training are noisy, which can deteriorate learning and hinder overall filter performance. Thus it is important that we are able to detect this mail and handle it appropriately in the context of anti-spam systems. Gray mail detection can also bring benefits on the client side. Identifying these messages in the inbox can allow a system to prompt for user preferences on this difficult class of mail, increasing personalization and improving overall user satisfaction.

In this paper, we conduct a pioneering study on gray mail detection. We compare four simple methods and evaluate their performance in different recall-precision regions. Preliminary experiments show that by using even a coarse gray mail detector, we can consistently improve upon a standard statistical spam filter.

## 2. GRAY MAIL DETECTION METHODS

Although there are large spam corpora in both academia and industry, similar resources do not exist for gray mail. Instead of manually annotating messages as gray mail or not and then using them to train a classifier, we seek detection methods that use regular datasets of email labeled as spam or good. In particular, we explore four approaches: leveraging the output of a spam filter, building an ensemble of spam filters, creating an approximate dataset based on sender IP information and identifying email campaigns with mixed labels.

### 2.1 Leveraging the Output of a Spam Filter

Whether a gray mail message is spam or good is ambiguous by definition. Thus both humans and statistical filters should be *uncertain* on the true labels of gray mail. Although the final decision of a spam filter is binary, the model behind a statistical filter generally produces a real-valued number that indicates the confidence of this decision. For margin-based methods such as SVMs, this confidence measure is often the distance between the example and the decision hyperplane; for naive Bayes and logistic regression the estimated probabilities play this role.

Given an email message $x$, suppose the probability that $x$ is spam estimated by the filter is $p(x)$. Values close to 1 or 0 can then be interpreted as confident classifications, while values close to 0.5 can indicate high uncertainty. Under the assumption that gray mail are those messages for which the filter is uncertain, we can define a function $g(x) = 0.5 - |p(x) - 0.5|$ to represent the uncertainty or *grayness* of $x$. A gray mail detector can then be constructed using

*CEAS 2007 - Fourth Conference on Email and Anti-Spam,* August 2-3, 2007, Mountain View, California USA

this function along with a decision threshold selected using a held-out set. Note that since we only use the function to compare messages, any similar monotonic transformation can be used.

## 2.2 Comparing an Ensemble of Spam Filters

Another view of gray mail is that it is the messages users might *disagree* on their labels, which suggests another detection approach – the *ensemble* method. It mimics this behavior by learning multiple spam filters using different disjoint subsets of training data. An email message is then classified as gray or not based upon the level of disagreement between these models. Unlike the previous approach, the ensemble method works for both binary decisions and real-valued confidence output. When only the final decisions of the filters are available, a simple voting scheme can be used to judge the disagreement. Suppose $c_i(x) \in \{0, 1\}$ is the prediction of filter $i$. The degree of disagreement of $k$ models can then simply be defined as $g(x) = \sum_{1 \leq i < j \leq k} \mathbf{1}(c_i(x) \neq c_j(x))$, where $\mathbf{1}$ is the indicator function. It is not hard to see that a different function $g'(x) = k/2 - |\sum_{1 \leq i \leq k} c_i(x) - k/2|$ will have the same effect.

When the estimated probabilities given by the spam filters are available, a natural way to judge the disagreement is to use the *variance* of these real-valued scores directly. Note that this method completely ignores the prediction of each spam filter, instead measuring the proximity between their probability estimates. In this paper we form an ensemble using 10 filters trained by logistic regression and measure disagreement as the variance of their probability estimates. The ensemble approach is similar to [6], although the goal there is to detect adversarial label errors. It is also similar to boosting [3] or the two-stage framework [7]. The difference is that we do not change the distribution of the training data when learning the filters but use disjoint subsets of the original training data.

## 2.3 Creating Approximate Data Using Sender IP Information

One major obstacle for applying machine learning directly to train a gray mail detector is the lack of labeled (gray vs. spam/good) data. However, by leveraging existing labeled (spam vs. good) data and sender information, we can "simulate" a training set as follows. If most mail from a particular sender (e.g., an IP address) is labeled consistently (either spam or good), then we can naively assume that the sender only sends one class of mail. On the other hand, if the ratio of spam versus all messages is between $\alpha$ and $1 - \alpha$, then we can assume all of their messages are gray mail. Note that these assumptions are clearly unrealistic because many mixed senders, such as email forwarders, actually send both good and spam mail rather than strictly gray mail. Therefore, this scheme generates only an approximate data set for training a gray mail detector.

In this paper, we set $\alpha$ to 20%. In order to reliably judge the ratio of spam and good email, we only consider senders with some minimum number of messages in our training set (10 in our experiments). In addition, we use the first 24 bits of an IP address (i.e., class C) instead of the full IP address to identify senders. We do not create different features for training the gray mail detector; only the labels of those messages are changed.

## 2.4 Identifying Email Campaigns with Mixed Labels

Remember that gray mail are messages that could reasonably be considered either spam or good. If several users receive the same message but assign different labels to it, then this message fits the definition of gray mail perfectly. Following this rationale, we first find email campaigns in our data using the near-duplicate detection technique developed in [5]. By comparing the fingerprints or signatures extracted from each message, messages that have identical content are clustered reliably with high precision.

If most messages in the campaign are labeled as spam or good, then all the email in this campaign is considered *not* gray. Similarly, if the ratio of spam labels versus all messages is between $\alpha$ and $1 - \alpha$, then we can assume all messages in this campaign are gray mail. Following the same setting described in Sec. 2.3, we set $\alpha$ to 20% and consider only campaigns with more than 10 messages. Notice that although this method seems to match the definition of gray mail better, in practice it is only able to find messages in campaigns large enough to satisfy our detection criteria.

# 3. EXPERIMENTS

In this section we conduct an experimental study comparing the gray mail detection methods discussed in Sec. 2. We also show how a normal statistical spam filter can benefit from incorporating gray mail detection.

## 3.1 Detecting Gray Mail

We built and evaluated different gray mail detectors using mail from the Hotmail Feedback Loop. These are messages labeled as spam or good, obtained by polling over 100,000 Hotmail volunteers each day. Interested readers are referred to [7] for more details. We randomly selected 800,000 messages received between January 1, 2006, and August 31, 2006, as our dataset for constructing different gray mail detectors. As discussed in the previous section, these messages are labeled as good or spam, with no indication of whether or not they are gray mail. For evaluation we chose 418 messages received between September 1st, 2006 and November 30th, 2006. We carefully examined the content of each message and annotated it as gray mail or not. Among these 418 messages, 163 are considered gray mail and 255 messages are not. While we are currently labeling more messages for gray mail detection, experiments using this evaluation dataset have shown some interesting and encouraging results.

When training a **basic** spam filter (Sec. 2.1), an **ensemble** of spam filters (Sec. 2.2), or an approximate gray mail detector using **sender** (Sec. 2.3) or **campaign** (Sec. 2.4) information, we use logistic regression with the SCGIS optimization method [4]. Only the content features (i.e., the words in the body or subject) are used.

Figure 1 shows the recall-precision curves of these methods. The basic method is consistently worse than the ensemble method and only slightly better than training a gray mail detector using sender or campaign information when recall is very high. This indicates that the probability estimation given by the regular filter is not a reliable measure of gray mail. Uncertain predictions may instead indicate spam that manages to fool the filter. The ensemble method performs reasonably well and is the best when recall is above 0.82. In comparison, using the sender or campaign information to

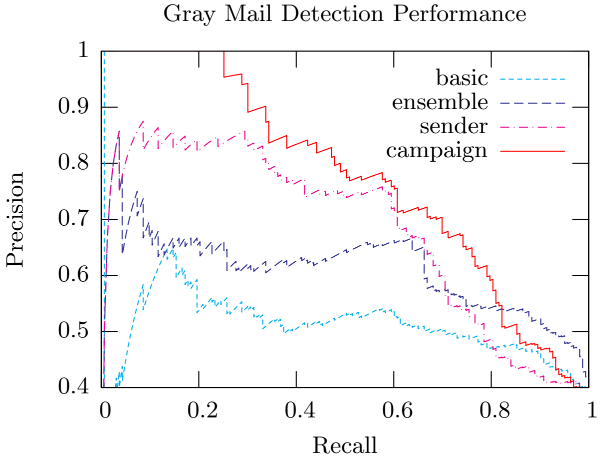

**Figure 1: The recall-precision curves of different gray mail detection methods**

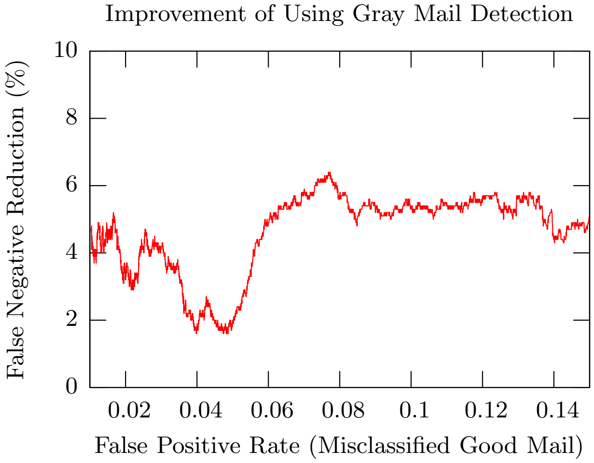

**Figure 2: The improvement of using the two-filter architecture compared to the regular spam filter in reducing false negative messages (i.e., unfiltered spam)**

form a dataset to train the gray mail detector seems better. In particular, the campaign information is more reliable and the corresponding detector has the highest precision except in the high-recall region.

In practice, the choice of detection method depends on its application. For example, if our task is to remove as much gray mail as possible from a large set of regular training data (spam vs. good), then the ensemble method can operate in the high-recall region to remove most gray mail. However, if our task is to select a set of gray mail messages for further study (e.g., for feature engineering), then learning a detector using the campaign information can provide a high-accuracy gray mail sample and therefore is our best choice.

### 3.2 Improving Spam Filtering by Separating Gray Mail

Gray mail detection has several applications and here we test an important one – improving spam filtering. To evaluate this potential we use the following simple system architecture. Before training, we first apply a gray mail detector to separate the regular training data into two disjoint subsets – **gray** and **b&w** (i.e., spam/good). We then train two spam filters, one on each of these datasets. When classifying a new message as spam or good, we apply the gray mail detector and, according to its prediction, classify the message using the gray or b&w spam filter.

To test this scheme we randomly selected 300,000 messages for training from those received between September 1, 2006, and November 30, 2006. Similarly, 100,000 messages received between December 1, 2006, and December 31, 2006 were selected for testing. To build the gray mail detector we used the dataset constructed using the campaign information (Sec. 2.4) with a threshold that provides a coarse coverage with reasonably high accuracy.

Even with the gray mail detector that is far from perfect, we clearly see an advantage with the two-filter architecture. Figure 2 shows the improvement brought by gray mail detection in reducing false negative messages (unfiltered spam). The x-axis shows the false-positive (misclassified good mail) rate of the filters and the y-axis shows the percentage of false negative reduction. All these filters were trained using logistic regression with the same learning parameter setting. Only content features were included and identical training examples were used in these two different frameworks. From the figure, we can see that gray mail detection helps reduce the false negative rate by roughly 2% to 6% in the low false-positive region. Although the improvement may seem limited, the fact that even an imperfect gray mail detector consistently helps over different regions of the ROC curve is truly encouraging. We suspect that the performance difference of these two frameworks will increase after the gray mail detector is further improved.

## 4. CONCLUSIONS

Gray mail – messages which are not clearly spam or good mail – present significant obstacles to training and evaluating global spam filters. It also indicates key points where more personalized filtering is needed to handle different user preferences. In this paper we highlight this important class of mail and compare four simple methods for detecting gray mail. Among them, we found that treating email campaigns with different labels as gray mail is the most reliable one. Moreover, we show how a global spam filter can benefit from gray mail detection, even when the detector is imperfect.

In the future we plan to investigate more effective methods for gray mail detection, such as better ways to collect training data, different machine learning frameworks and features related to sender information (e.g., statistics on how long the sender has been active and the sending pattern). We also plan to pursue new methods for leveraging gray mail detectors to further improve spam filtering and email personalization.

## 5. ACKNOWLEDGMENTS

We would like to thank Xin Huang, Geoff Hulten, Kris Iverson, Harry Katz, Eriola Kruja, and Suresh Velagapudi for valuable discussion around this work. They were instrumental in motivating this problem and its potential value to more comprehensive anti-spam frameworks.

## 6. REFERENCES

[1] Email Sender and Provider Coalition. Consumers savvy about managing email according to ESPC survey results; embrace numerous tools and methods to manage spam reporting and unsubscribing. Email Sender and Provider Coalition (ESPC) press release, http://www.espcoalition.org/032707consumer.php, March 2007.

[2] D. Fallows. Spam: How it is hurting email and degrading life on the Internet. *Pew Internet and American Life Project*, October 2003.

[3] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[4] J. Goodman. Sequential conditional generalized iterative scaling. In *ACL '02*, 2002.

[5] A. Kolcz and A. Chowdhury. Hardening fingerprinting by context. In *Proceedings of the 4th Conference on Email and Anti-Spam*, 2007.

[6] S. Laxman. Error correction problem in learning svms. Personal Communications, 2007.

[7] W. Yih, J. Goodman, and G. Hulten. Learning at low false positive rates. In *Proceedings of the 3rd Conference on Email and Anti-Spam*, 2006.