Image Spam Hunter

Yan Gao, Ming Yang, Xiaonan Zhao
EECS Dept., Northwestern Univ.
2145 Sheridan Rd., Evanston, IL 60208
{y-gao2,m-yang4,xiaonan-zhao@northwestern.edu}

ABSTRACT

Spammers are constantly creating sophisticated new weapons in their arms race with anti-spam technology, the latest of which is image-based spam. The newest image-based spam uses simple image processing technologies to vary the content of individual messages e.g. by changing foreground colors, backgrounds, picture sizes, font types, or even rotating and adding artifacts to the images. Thus, they appear distinct from one another and pose great challenges to conventional spam filters. This paper aims to investigate this kind of image-based spam detection problem. Given the sample spam images collected from real spam emails and the random selected natural images, we propose to employ probabilistic boosting tree to give soft decision on whether an incoming image is a spam or not based on efficient global image features, i.e. color and gradient orientation histograms. The proposed method achieves fairly good performance in 5-fold cross-validation on the data set with 928 spam images and 810 natural images.

Keywords

Image spam, probabilistic boosting tree, color histogram, gradient orientation histogram

1. INTRODUCTION

Anti-spam is a very active area of research, and various forms of filters, such as white-lists, black-lists, and content-based lists are widely used to defend against spam. Content-based filters make estimations of spam likelihood based on the texts in that email message and filter messages based on a pre-selected threshold [2]. Since spam detection can be converted into text classification problem, many content-based filters utilize machine-learning algorithms for filtering spam. However, a number of spammers have been evading filters recently by encoding their messages as images and including some irrelevant good words. This implies the contents are hard to retrieve from the binary image encoding.

This type of image spam is not rare anymore and accounts for 30% of all global spam in 2006, compared with just 1% in late 2005. By sending emails that contain no text, only pictures, or along with irrelevant good words, spammers have found that they can evade many security systems. The messages often include image files that have a screen shot offering the same types of information advertised in traditional text-based spam. Spammers are also getting sneakier, using techniques like image tiling to vary the images slightly for each message. They do this easily by changing the shade of the border or background, changing the line spacing or margins, or even adding tiny specks to the background. These changes are unnoticeable to the eyes, but completely change the data's appearance to most anti-spam engines. The consequence is a huge quantity of image-based spams that contain random patterns with almost no repetitions. Some sample spam images are shown in Fig. 1 to illustrate the diversity of spam images.

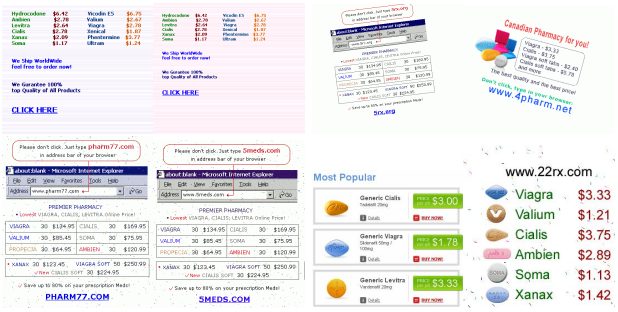


Figure 1: Sample spam images: image size changes and rotation (1st row), artifacts in the background and images with icons (2nd row).

In this paper, we explore a recurrent pattern detection system against image-based spam by using machine learning algorithms. We aim to find an intrinsic mechanism to match recurrent patterns across similar but not-identical images. In the real internet, mail service providers, e.g. hotmail and gmail, can provide the customers a *junk* button, which may help them to collect useless spam emails and further filter out huge number of similar spams received by other customers. It is an important requirement that spam email detection system can tolerate some false negatives (i.e. allowing some spams to get through), but cannot afford the false positives (i.e. filtering off the normal photos and image attachments).

To simulate the real spam detection process in the internet, in this paper, we propose a learning-based prototype system *Image Spam Hunter* to differentiate spam images from normal image attachments. We first cluster the collected disordered spams into groups based on image similarity measurement with as k-means or normalized cuts [6] on global color and gradient orientation histograms [3]. As a result, we can automatically obtain many groups of training spam images. Second, a recent machine learning algorithm probabilistic boosting tree (PBT) [7] is applied to distinguish image spams from good emails with image attachments. The proposed method achieves 0.86% false positive rates versus 89.44% true positive rates in 5-fold cross-validation on a database with 928 spam images and 810 normal images.

2. RELATED WORK

Many anti-spam techniques have been proposed and employed to counter email spam from different perspectives. Some people analyze the content of received messages and convert the text-based spam detection into the problem of text classification, so many content-based filters utilize machine learning algorithms for filtering spam. Among them, Bayesian-based approaches [2, 1] have achieved outstanding accuracy and have been widely used. As these filters can adapt their classification engines to the change of message content, they outperform heuristic filters.

However, a number of spammers have been evading text-based spam filters recently by encoding their messages as images. There are several organizations and companies designing a set of rules that are meant to combat image spams. For example, SpamAssassin (SA) [5] is the first to use Optical Character Recognition (OCR) software to pull words out of the images and then uses a blacklist of words to increase the SA scores. It was quickly realized that spammers are using similar obfuscation techniques in the images that they have long used in text emails, such as misspelling words, using characters that look like others. So a fuzzy matching was added to the plug-in. However, there are already reports of image spams that put a light background of random artifacts behind the text or rotate the image slightly. These practices do not affect the readability to humans, but do greatly affect the quality of the OCR output. Unfortunately, human pattern matching may be too good for the state of the art OCR to level up. In addition, OCR for every image with unknown fonts and sizes are computationally intensive.

3. OUR APPROACH

Text OCR from images is quite computationally demanding and vulnerable to image artifacts. Intuitively, people don't need to recognize the texts in the images to determine whether they are more likely to be spams. Since the spam images are artificially generated, we expect their image texture statistics are distinguishable from natural images such as sky, mountain, beach, buildings, and human. Therefore, we propose to employ a recent learning algorithm, probabilistic boosting tree(PBT) [7], to classify spam images from normal images based on efficient global image statistics, i.e. color and gradient orientation histograms.

3.1 Image feature extraction

We consider two cues, color histogram and gradient orientation histogram, as the feature vectors for learning. The observation is that most of spam images are converted from text spams, although they may contain some icons and artifacts. Thus, the color components may be quite limited compared with natural scenes. As shown in Fig. 2, the color histograms of natural scenes tend to be continuous, while the color histograms of artificial spam images tend to have some isolated peaks. Another observation is that the distribution of gradient orientation may reveal the characteristics of texts. Fig. 3 illustrates the comparison of 1D histograms of image gradient orientation of spam images and natural images. The distributions of gradient orientation for natural images appear more uniform and noisy than those of spam images. Gradient orientation histograms are particular effective to deal with gray-level images.

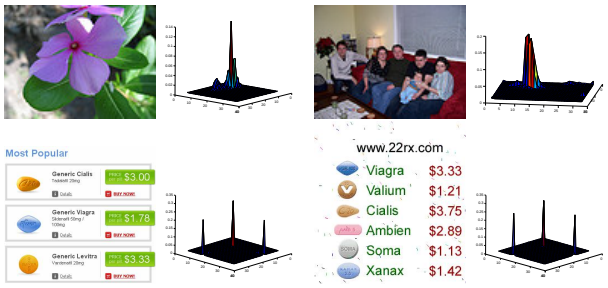


Figure 2: Color histograms comparison between natural images and spam images in 32×32 2D normalized RG plane.

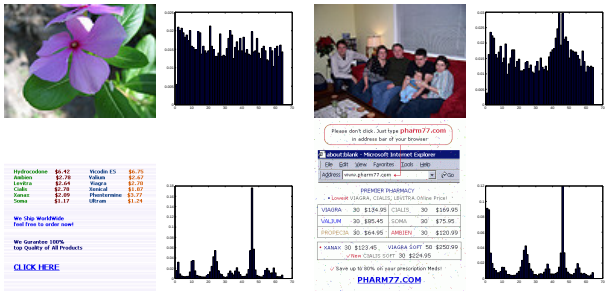


Figure 3: Gradient orientation histograms comparison between natural images and spam images.

Specifically, we extract two D dimensional feature vectors $\mathbf{x^c} = \{\mathbf{x_1^c}, \cdots, \mathbf{x_D^c}\}$ and $\mathbf{x^g} = \{\mathbf{x_1^g}, \cdots, \mathbf{x_D^g}\}$ for each image. Each color can be represented by 2 independent components, so we build 2D color histograms in certain color space (normalized RG space in our experiments). Since we only care about the shape or color distribution rather than the exact meaning of color bins, we sort the bins in descent order and only keep the top dominant D bins as the feature vector $\mathbf{x^c}$. This approach also balances the need for high resolution of color histogram (otherwise similar color will be quantized to the same bin), and the need for efficient training and testing on feature vectors without too high dimensionality. In our experiments, we calculate $32 \times 32 = 1024$ 2D color histogram and test keeping different top $D = 32, 64, 128$ bins. To extract gradient orientation histogram $\mathbf{x^g}$, the image gradient for each pixel is calculated with Sobel operator, if the gradient magnitude is larger then a threshold $t_m = 50$, we

quantize its orientation angle $0^o - 360^o$ to one of D bins.

3.2 Training set generation

Since there are quite a lot of repetitive spam images and they can be categorized into several different classes, e.g. spams with texts and artifacts or spams with icons, the spam images in the same class appear very similar in feature spaces but quite different from other classes. To avoid only selecting specific groups of spam images in the training set, we need a rough clustering to generate the training set for learning rather than pure random selection. Therefore, for all spam images in the sample set we perform k-means to roughly cluster them to k groups ($k = 6$ empirically selected) to ensure the training set includes samples in each group. For example, for 5-fold cross-validation, training set is generated by selecting 4/5 samples from each group. Note we do not cluster the normal images.

3.3 PBT Classification

Image similarity measurement is an active and open research topic that is generally very difficult. In this paper, we aim to merely distinguish a specific group of images, i.e. the spam images, from normal images by inductive supervised learning. Thus, we collect training spam and normal image samples I_i and represent them with feature vectors with labels $(\mathbf{x}_i^c, \mathbf{x}_i^g, y_i)$, where $y_i = +1$ indicates spam image and $y_i = -1$ for normal image.

We implement a light version of probabilistic boosting tree method to classify the spam and natural images. Essentially, PBT is a decision tree trained with positive (spam images) and negative (normal images) samples, where each node in the tree is an Adaboost classifier. The Adaboost algorithm learns a strong classifier by combining a set of weak classifiers $H(\mathbf{x}) = \sum_{t=1}^{T} \alpha_t h_t(\mathbf{x})$. Denote the probabilities computed by each learned Adaboost classifier as

$$p(+1|\mathbf{x}) = \frac{exp\{2H(\mathbf{x})\}}{1 + exp\{2H(\mathbf{x})\}}, \; p(-1|\mathbf{x}) = \frac{exp\{-2H(\mathbf{x})\}}{1 + exp\{-2H(\mathbf{x})\}}.$$

At each node, the training samples are divided into two overlapped sets $S_{left} = \{(\mathbf{x}_i, y_i)|p(+1|\mathbf{x}_i) > 0.5 - \epsilon\}$ and $S_{right} = \{(\mathbf{x}_i, y_i)|p(-1|\mathbf{x}_i) > 0.5 - \epsilon\}$, then these two sets are passed to left and right sub-trees to further train Adaboost classifiers to separate them. To test a feature vector $\mathbf{x}$, the classification result combines the probability at every node in a probabilistic way, as

$$\begin{aligned} p(y|\mathbf{x}) &= \sum_{l_1} p(y|l_1, \mathbf{x}) p(l_1|\mathbf{x}) \qquad (1) \\ &= \sum_{l_1, l_2} p(y|l_2, l_1, \mathbf{x}) p(l_2|l1, \mathbf{x}) p(l_1|\mathbf{x}) \\ &= \sum_{l_1, \cdots, l_n} p(y|l_n, \cdots, l_1, \mathbf{x}), \cdots, p(l_2|l1, \mathbf{x}) p(l_1|\mathbf{x}), \end{aligned}$$

where the tree level l_i is an augmented variable and $p(l_i|\mathbf{x})$ denotes classification probability of the Adaboost classifier for this testing feature $\mathbf{x}$ at level l_i. An illustration of the hierarchical PBT is shown in Fig. 4, and we refer the reader to [7] for more details.

We train two PBTs for $\mathbf{x^c}$ and $\mathbf{x^g}$ independently. The classification probabilities for these two trees are denoted as

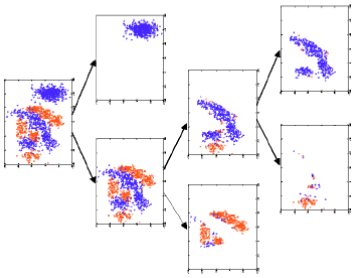


Figure 4: Illustration of PBT courtesy by Dr. Z. Tu.

$p(\pm 1|\mathbf{x}^c)$ and $p(\pm 1|\mathbf{x}^g)$ for a testing image. The testing image is marked as spam if and only if both PBTs make the same decision, so as to avoid false negative as much as possible.

$$y = \begin{cases} +1 & p(+1|\mathbf{x}^c) > 0.5 + \delta \text{ AND } p(+1|\mathbf{x}^g) > 0.5 + \delta \\ -1 & \text{otherwise.} \end{cases} \qquad (2)$$

where δ is a parameter to adjust the false positive and true positive rate.

4. EXPERIMENT

4.1 Implementations

We employ normalized RG space in color histogram calculation, which is insensitive to lightings (we also tested hue-saturation plane, the performance is similar). The color histograms are sorted in decreasing order and truncated to keep top D bins. In our PBT implementation, we employ Gentle Adaboost classifier in OpenCV library [4] at each node which consists of 100 decision stumps as weak classifiers. The ϵ is set to 0.1 to spilt the tree.

4.2 Dataset

We collect 928 spam images from real spam emails as the spam sample set. These image are subset of image spams we received in the last 6 months where image spams with animation are excluded. The normal images set includes 810 images randomly downloaded from Flickr.com along with 20 scanned documents.

4.3 Cross-validation

We test the spam detection by 5-fold cross-validation on the aforementioned database. There is no overlap of the training and testing sets. The performance is measured with the average false positive (FP) rate, i.e. the misclassification rate of normal images, and true positive (TP) rate, i.e. the detection rate of spam images, as follows:

$$\text{FP Rate} = \frac{\text{\# of normal images classified as spams}}{\text{\# of total normal images}}$$

$$\text{TP Rate} = \frac{\text{\# of spam images classified as spams}}{\text{\# of total spam images}}.$$

By testing different D values as in Tab. 1, we find $D = 64$ is sufficient to detect the spams in our current sample set. Classification accuracies over both training spam and normal images are listed in the table as well.

Table 1: Comparison of 5-fold cross-validation performance of different D dimensional vectors.

	Accuracy	FP Rate	TP Rate
32D	0.5631	0.0136	0.1944
64D	0.9494	0.0420	0.9417
128D	0.9471	0.0469	0.9426

By varying the δ, we obtain the Receiver Operating Characteristic (ROC) curve for $D = 64$ feature vectors as shown in Fig. 5. Without any learning method, the expectation of random guess will generate a 45^o line in the ROC curve. From the curve, we can see our approach achieves 89.44% detection rate of spams at FP rate 0.86%. This preliminary result is quite acceptable in real email systems, while large amount of samples are preferred to further validate the results.

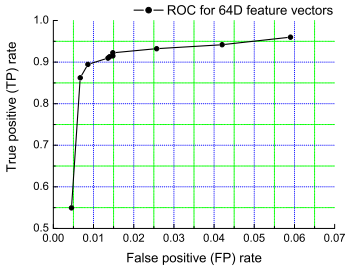


Figure 5: ROC curve for 64D feature vectors.

4.4 Error analysis

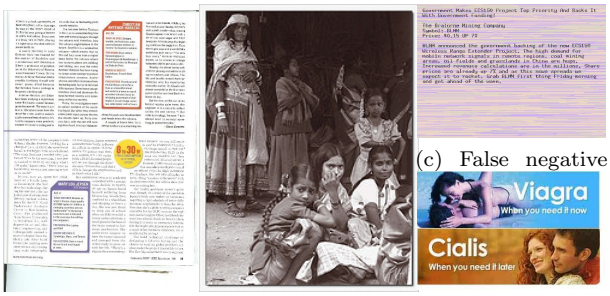
By analyzing the incorrectly classified images, we find out that they fall into 4 typical categories:

1. Scanned documents with icons may be wrongly blocked, e.g. in Fig. 6(a). This can easily be mistaken for spam even by human, if the text content is not taken into account. In our model, its color and gradient histograms are indistinguishable from those of spams. Correctly classifying these images can only rely on the help of OCR or other content detection method, which will considerably increase the computational complexity.

2. Photos with relatively less color components and sharp contrast may be blocked, e.g. in Fig. 6(b). Our model tends to classify images with sparse color histogram peaks and pulse-like gradient feature as spams. Photos with less colors and certain contrast patterns may have a chance to be blocked as spam.

3. Spams with colorful gradually-changing background may survive, e.g. in Fig. 6(c). This kind of spams may fool the system with real photo-like color and gradient features. Their histograms do not have outstanding single pulse because of the gradually changing color in the background.

4. Spams with a large portion of real life photos may survive, e.g. in Fig. 6(d). It is highly possible for spams with only a few texts on real life photos to cheat the spam hunter and get through. We should be very cautious about this kind of images because many of normal photos may have some texts on them (date on the photo, banner in the photo, etc) and we do not want to miss them.



(a) False positive (b) False positive (c) False negative (d) False negative

Figure 6: Sample false positives and false negatives.

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed a machine learning method to detect the spam images from images in normal emails. The proposed method extracts efficient global image features to train an advanced binary classifier to distinguish the spam images, which achieves promising preliminary results on our limited sample database.

Normal image is a very broad concept which may not be represented by finite sample images, and there is still a huge gap between the global image features and the semantic meaning of the email contents. Therefore, our future work will examine some high level human vision models and more sophisticated analysis methods to improve the performance, e.g. structure pattern analysis to take the spatial information into consideration, and texture analysis in frequency domain with wavelet. In addition, we will collect more spam and normal image samples to verify our results.

6. REFERENCES

[1] J. Blosser and D. Josephsen. Scalable centralized bayesian spam mitigation with bogofilter. In *Proc. USENIX LISA*, 2004.

[2] K. Li and Z. Zhong. Fast statistical spam filter by approximate classifications. In *ACM SIGMETRICS*, pages 347 – 358, St. Malo, France, June 2006.

[3] A. Maccato and R. deFigueiredo. The image gradient histogram and associated orientation signatures. In *IEEE Int'l Symposium on Circuits and Systems*, volume 1, pages 239– 242, Seattle, WA, 1995.

[4] Open Source Computer Vision Library. http://www.intel.com/technology/computing/opencv/.

[5] SA. http://spamassassin.apache.org/.

[6] J. Shi and J. Malik. Normalized cuts and image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 22, pages 888– 905, 2000.

[7] Z. Tu. Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering. In *IEEE Int'l Conf. on Computer Vision (ICCV'05)*, volume 2, pages 1589– 1596, Beijing, China, Oct.17 - 21, 2005.