International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:8, No:10, 2014

Image Spam Detection Using Color Features and K-Nearest Neighbor Classification

T. Kumaresan, S. Sanjushree, C. Palanisamy

***Abstract*—**Image spam is a kind of email spam where the spam text is embedded with an image. It is a new spamming technique being used by spammers to send their messages to bulk of internet users. Spam email has become a big problem in the lives of internet users, causing time consumption and economic losses. The main objective of this paper is to detect the image spam by using histogram properties of an image. Though there are many techniques to automatically detect and avoid this problem, spammers employing new tricks to bypass those techniques, as a result those techniques are inefficient to detect the spam mails. In this paper we have proposed a new method to detect the image spam. Here the image features are extracted by using RGB histogram, HSV histogram and combination of both RGB and HSV histogram. Based on the optimized image feature set classification is done by using k- Nearest Neighbor(k-NN) algorithm. Experimental result shows that our method has achieved better accuracy. From the result it is known that combination of RGB and HSV histogram with k-NN algorithm gives the best accuracy in spam detection.

***Keywords*—**File Type, HSV Histogram, k-NN, RGB Histogram, Spam Detection.

I. Introduction

THE web has become an essential tool to most of the people. Many people are using email to send their messages world wild. Since email is one of the cheap and safe medium to send messages. Though there are many advantages in email, it has some disadvantages like spam mails. Spam is one the major issue in electronic mail. A research has shown that more than 80 percent of the emails are spam only, generally referred as web spamming [1]. Several problems are caused by web spamming; one of the main problems is nuisance to the users. Spam mail promotes unwanted materials among the internet users [2]. A spam not only gives nuisance to internet users but also degrades the performance of computers by installing malware or spyware software's [3] which comes as links in spam mail.

The web spamming techniques are constantly evolving, so there is a need of new technique that must be well suited to detect the spam mails. Some search engine ranking algorithms are detecting web spam successfully [4] by adopting some features. Sometimes search algorithms use machine learning algorithms to find the occurrence of spam message in a webpage. In general spam is of two types text based spam and image spam. Spam message comes along with an image will be called as image spam.

The Web has changed dramatically the way that people express themselves and communicate with others. They can post opinions of products at merchant sites (e.g., amazon.com) and express their views in blogs and forums. Nowadays it is well recognized that such user generated contents on the Web provide valuable information that can be exploited for many applications. They focus on customer reviews of products, which contain the information of consumer opinions on the products; also they are useful to both potential customers and product manufacturers [5], [6].

The objective of Web spam is to attract the people to market their product or services. Spam mails are mainly ads about a product or some services. Spam reviews are very different as they give false opinions, which are much more harder to detect even manually. Thus, many existing methods for detecting web spam and email spam are unsuitable for review spam [7]-[10].

Unlike the large amount of available approaches to deal with email spam's, there are few methods in the literature to automatically detect web spam [11]-[13]. In general, all the techniques employ one of the following strategies:

- To analyze only the relation of web links [14]
- To analyze only the content of the web pages [15]

To extract suitable features from both contents and web links.

II. Proposed Email Classification

A. Feature Extraction Using Their File Type

Consider four basic features that can quickly be derived from an image at an extremely low computational cost. These are the width and the height denoted in the header of the image file, image file type, and the file size. Based on these raw features, generate a small 6 dimensional feature vector f1 to f6. Dimensional feature vector f1 denotes the image width in header and f2 denotes the image height in the header, f3 denotes the aspect ratio and f4 denotes the GIF image, JPEG image is noted by f5 and f6 denotes the PNG image. The image file type features (f4, f5, and f6) are binary features those are set to 1 if the file is of the specified type and to 0 otherwise. Spam detection using file type focus on the three dominant file formats commonly seen in email, which are the Graphics Interchange Format (GIF), the Joint Photographic Experts Group (JPEG) format, and the Portable Network Graphics (PNG) format.

T. Kumaresan is working as an Assistant Professor in the Department of Computer Science and Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India (e-mail: speak.kumaresh@gmail.com)

S Sanjushree is a PG scholar studying ME Software Engineering at Bannari Amman Institute of Technology, Sathyamangalam.(e-mail: sampathsanjushree@gmail.com)

Dr. C. Palanisamy is working as a Professor and Head, Department of Information Technology, Bannari Amman Institute of Technology, Sathyamangalam (e-mail: hodit@bitsathy.ac.in).

A general idea of the image dimensions (i.e. width and height) can be gathered by parsing the image headers of the GIF, JPEG, or PNG files using a minimal parse. This is very fast since it doesn't decompress or decode any actual image data. Unfortunately we only get a general idea because obtaining the actual dimensions can be somewhat trickier and more time consuming in most cases.

In the case of GIF files (the current de facto standard for image spam) the presence of virtual frames, which can be either larger or smaller than the actual image width, is an issue that can only be detected while decoding the image data. Other embedded information such as alpha channel and multi frame images can require a full parse of the image data to detect.

Also, corrupted images can pose a problem. For corrupted images most typically some of the lines near the bottom of the image do not decode properly and no further image data can be decoded after that point. This is an issue for PNG and JPEG images as well, and seems to be one of the spammer's favorite tricks. Reverse engineering analysis performed by Secure Computing of versions of spam bot software responsible for generation and randomization of the images used in much of the image spam has uncovered memory leaks and other bugs in the image generation code that we speculate is occasionally introducing this corruption. As such, presence of corruption in the image currently happens to be a very good discriminator of spam and ham but if these bugs will eventually get discovered and addressed by the software authors, the feature may become less useful.

B. Feature Extraction Using RGB Color Histogram

The color histogram is a simple feature and can be calculated very efficiently by one simple pass of the whole image. We have used 64-dimensional color histogram based in the RGB color space. Values in each of the three color channels(R, G, and B) are divided into 4 bins of equal size, resulting in 4 × 4 × 4 = 64 bins in total. For each bin, the amount of color pixels that falls into that particular bin is counted.

Finally it is normalized so that the sum equals to one. In this paper we used L_1 distance to calculate the distance between two color histogram features. For images represented by D dimensional real-valued feature vectors, the L_1 distance of the pair of points $a = (a_1, \ldots, a_z)$ and $b = (b_1, \ldots, b_z)$ has the form

$$z(a,b) = \sum_{i=1}^{z} |a_i - b_i| \quad (1)$$

C. Feature Extraction Using HSV Histogram

A three dimensional representation of the HSV color space is a hexacone, where the central vertical axis represents the Intensity [9]. Hue is defined as an angle in the range [0,2S] relative to the Red axis with red at angle 0, green at 2S/3, blue at 4S/3 and red again at 2S. Saturation is the depth or purity of the color and is measured as a radial distance from the central axis with value between 0 at the center to 1 at the outer surface. For S=0, as one moves higher with the intensity axis, one goes from black to white through different shades of gray. On the other hand, for a given intensity value and Hue, if the saturation value is changed from 0 to 1, then the perceived color changes from a shade of gray to the most pure form of the color represented by its Hue. Looking at a different angle, any color in the HSV space can be transformed to a shade of gray by sufficiently lowering the saturation. Intensity value determines the particular gray shade to which this transformation converges. While the Saturation value is near 0, all pixels, even with different Hues look alike and as we increase the saturation as 1, they tend to get separated and are visually perceived as the true colors represented by their Hues. Thus, for low values of Saturation, a color can be approximated by a gray value specified by the Intensity level while for higher Saturation; the color can be approximated by its Hue. The Saturation threshold that determines this transition is once again dependent on the Intensity. For low intensities, even for a high saturation, a color is very close to the gray value and vice versa. Saturation value gives an idea about the depth of color and human eye is less sensitive to its variation compared to variation in Hue or Intensity. Therefore, use the Saturation value of a pixel to determine whether the Hue or the Intensity is more pertinent to human visual perception of the color of that pixel and ignore the actual value of the saturation. It is observed that for higher intensity values, a saturation value of 0.2 differentiates between Hue and Intensity dominance. Assuming that the maximum intensity value to be 255, we use the following threshold function to determine if a pixel should be represented by its Hue or its Intensity as its dominant feature.

$$th_{sat}(V) = 1.0 - \frac{0.8v}{255} \quad (2)$$

In the above equation, we see that for V=0, th(V) =1.0, meaning that all the colors are approximated as black whatever be the Hue or the Saturation. On the other hand, with increasing values of the Intensity, saturation threshold that separates Hue dominance from intensity dominance goes down.

D. Classification Using k-Nearest Neighbor (k-NN)

While using k-NN algorithm, after k nearest neighbors are found, several strategies could be taken to predict the category of a test document based on them. But a fixed k value is usually used for all classes in these methods, regardless of their different distributions. Equations (3) and (4) are the widely used strategies of this kind method.

$$y(d_i) = argmax_k \sum_{x_j \in KNN} y(x_j, c_k) \quad (3)$$

$$y(d_i) = argmax_k \sum_{x_j \in KNN} sim(d_i, x_j) y(x_j, c_k) \quad (4)$$

where d_i is a test image, x_j is one of the neighbors in the training set, $y(x_j, c_k) \in \{0,1\}$ indicates whether x_j belongs to class c_k, and $sim(d_i, x_j)$ is the similarity function for d_i and x_j. Equation (3) means that the predication will be the class that has the largest number of members in the k nearest neighbors; whereas (4) means the class with maximal sum of similarity will be the winner. The latter is thought to be better than the former and used more widely.

In general, the image distribution of different classes in the training set is uneven. Some classes may contains more samples than others. Therefore, it is very likely that a fixed k value will result in a bias on large classes. For example, when using the strategy indicated by (2), many tiny similarity values would accumulate to a large one, which may improperly make a large class the final decision. To overcome this problem, we propose a different strategy as follows,

When we get the original k nearest neighbors, we compute the probability that one image belongs to a class by using only some top n nearest neighbors for that class, where n is derived from k according to the size of a class c_m in the training set. In other words, we use different numbers of nearest neighbors for different classes in our method. For larger classes, we use more nearest neighbors. The dynamic selection is based on the class distribution in the training set. To make the comparison between classes reasonable, we derive the probabilities from the proportion of the similarity sum of neighbors belonging to a class to the total sum of similarities of all selected neighbors for that class. Equation (5) gives the decision function in our improved kNN algorithm:

$$y(d_i) = argmax_m \frac{\sum_{x_j \epsilon top\_n\_KNN(c_m)} Sim(d_i,x_j) y(x_j,c_m)}{\sum_{x_j \epsilon top\_n\_KNN(c_m)} Sim(d_i,x_j)} \quad (5)$$

where

$$top_{n_{KNN(c_m)}} = \left\{ \begin{matrix} top\ n\ neighbor\ in\ the \\ original\ k\ nearest\ neighbors\ KNN \end{matrix} \middle| \left[\frac{n = K \times N(c_m)}{max\{N(c_j)|j = 1 \ldots . Nc\}} \right] \right\}$$

Note that $N(c_m)$ denotes the size of the class cm in the training set, and $max\{N(c_j)|j = 1 \ldots . Nc\}$ is the size of the largest class in the same set.

III. Results and Discussion

To evaluate the performance of our proposed method we have taken Spam Archive data set. The Spam Archive data set is provided by Giorgio Fumera's group. This spam archive data set contains combination of both spam and legitimate mails. In total, the images considered to this proposed work is about 5087 images combined of 3209 spam and 1878 ham images. Images in the dataset are JPEG, GIF, PNG and BMP images. Using the combination of RGB and HSV histogram best feature set were extracted. Images were classified by k-NN algorithm using best features. For comparative analysis we have compared our results with the existing BPN method results.

A. Performance Metrics

Accuracy (A), Precision (P), and Recall (R), are some of the well-known performance measures are used in this paper.

$$Accuracy\ (A) = \frac{TP + TN}{TP + TN + FN + FP}$$

$$Precision(p) = \frac{TP}{TP + FP}$$

$$Recall(R) = \frac{TP}{TP + FN}$$

where TP is the number of e-mail that is spam and correctly predicted as spam;

FP is the number of e-mail that is legitimate but predicted as spam;

TN is the number of e-mail that is legitimate and is truly predicted as legitimate (ham); and

FN is the number of e-mail that is spam but predicted as legitimate.

TABLE I
Feature Vectors for File Type

Proposed feature vector	GIF file	JPEG file	PNG file
f1	358	789	564
f2	400	600	760
f3	890	876	467
f4	743	891	789
f5	567	567	965
f6	789	705	650

Table I shows the detection of the image spam for GIF, JPEG and PNG format file type. This demonstrates file type feature vectors. Six features are present in this table. The feature vector for RGB is calculated by L_1 distance. Depend upon hue and saturation, HSV features are selected. Then these features are given to the k-NN classifier as an input to detect the spam

TABLE II
Spam Detection Accuracy Using Proposed Method and Existing BPN

Approach	k-NN	BPN
File properties	89%	86.6%
RGB histogram	92.5%	92.1%
HSV Histogram	93.9%	93.4%
combination of RGB and HSV	94.5%	94.1%

Table II gives the accuracy comparison of the proposed method of spam detection using k-NN. Here the four features are taken as for classification of k-NN and BPN. They are file properties, RGB histogram, HSV histogram and combination of both RGB and HSV. The spam detection by combination of both RGB and HSV gives 94.5 % accuracy for proposed method of k-NN. From this table clearly observed that the combination of both RGB and HSV histogram gives the best result followed by HSV histogram.

TABLE III
Precision and Recall Comparison

Approach	Comparison	
	Precision	Recall
File properties	81.8%	86.7%
RGB histogram	84.9%	90%
HSV Histogram	89%	91.8%
combination of RGB and HSV	92.9%	93%

Table III gives the precision and recall comparison of the proposed method of spam detection using k-NN. Precision and recall for combination of both RGB and HSV gives 92.9% and 93% for proposed method of k-NN. Form this table clearly observed that the combination of both RGB and HSV gives

the better result than other approaches.

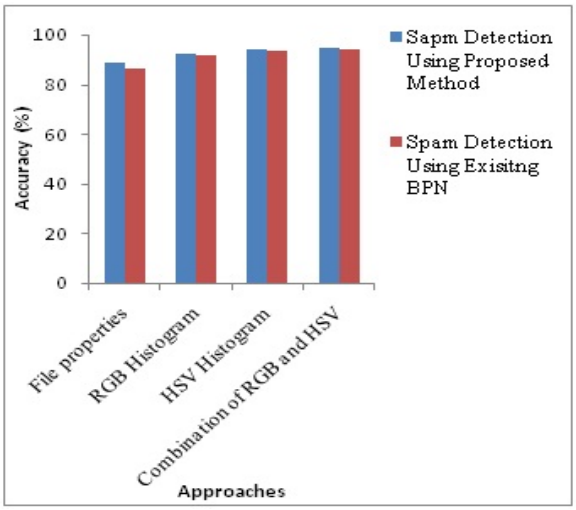

Fig. 1 Comparison of Accuracy

Fig. 1 shows the accuracy comparison of proposed spam detection method and existing BPN. From the figure it is clear that the proposed method of combination of RGB and HSV gives better result than other approaches and also it proves that the proposed method with k-NN classification of spam detection gives the better result.

IV. Conclusion

This paper reveals a general study on image spam, classification of image spam on the basis of text properties and content properties, and some of the methodologies in detecting the image spam. We have proposed a method to detect image spam using four approaches which are file properties, RGB histogram, HSV histogram and combination of both RGB and HSV histogram. Combination of RGB and HSV histogram gives the best feature set for classification. Using the optimized feature set classification is done by k-NN algorithm. Experimental results clearly shows that the combination of RGB and HSV histogram with k-NN approach obtained best results when compare to others. This is an effective method to detect the image spam. Future work may be incorporated with both image spam and text spam.

References

[1] K.M. Svore, Q. Wu, and C. J. Burges, "Improving web spam classification using rank-time features", Proceedings of the 3rd International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'07), Banff, Alberta, Canada, pp. 9–16, 2007.
[2] G. Shen, B. Gao, T. Liu, G. Feng, S. Song, and H. Li, "Detecting link spam using temporal information", Proceedings of the 6th IEEE International Conference on Data Mining (ICDM'06), Hong Kong, China, pp. 1049–1053, 2006.
[3] M. Egele, C. Kolbitsch, and C. Platzer, "Removing web spam links from search engine results", Journal in Computer Virology, vol. 7, pp. 51–62, 2011.
[4] Marc Najork, Web Spam Detection. Microsoft Research, Mountain View, CA, USA.
[5] M. Hu & B. Liu, "Mining and summarizing customer reviews", KDD' 2004.
[6] B. Liu, "Web Data Mining", Springer, 2007.
[7] Z. Gyongyi& H. Garcia-Molina, "Web Spam Taxonomy. Technical Report" Stanford University, 2004.
[8] K. Li, & Z. Zhong, "Fast statistical spam filter by approximate classifications", SIGMETRICS, 2006.
[9] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting spam web pages through content analysis", Proceedings of the World Wide Web conference (WWW'06), Edinburgh, Scotland, pp. 83–92, 2006.
[10] B. Wu, V. Goel& B.D. Davison, "Topical Trust Rank: using topicality to combat Web spam", WWW'2006.
[11] T. Almeida, A. Yamakami, and J. Almeida, " Evaluation of Approaches for Dimensionality Reduction Applied with Naive Bayes Anti-Spam Filters", Proceedings of the 8th IEEE International Conference on Machine Learning and Applications, Miami, FL, USA, pp. 517–522,2009.
[12] T. Almeida and A. Yamakami, " Content-Based Spam Filtering", Proceedings of the 23rd IEEE International Joint Conference on Neural Networks, Barcelona, Spain, pp. 1–7.2010.
[13] T. Almeida, J. Almeida, and A. Yamakami, "Spam Filtering: How the Dimensionality Reduction Affects the Accuracy of Naive Bayes Classifiers", Journal of Internet Services and Applications, vol. 1, no. 3, pp. 183–200, 2011.
[14] Q. Gan and T. Suel, "Improving web spam classifiers using link structure", Proceedings of the 3rd international Workshop on Adversarial Information Retrieval on the Web (AIRWeb'07), Banff, Alberta, Canada, pp. 17–20, 2007.
[15] T. Urvoy, E. Chauveau, and P. Filoche, "Tracking web spam with html style similarities", ACM Transactions on the Web, vol. 2, no. 1, pp.1–3, 2008.