International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-1, Issue-3, July 2011

# Web Spam Detection Using Different Features

**Sumit Sahu, Bharti Dongre, Rajesh Vadhwani**

***Abstract*—Spamming is any deliberate action solely in order to boost a web page's position in search engine results, incommensurate with page's real value. Web Spam is the Web pages that are the result of spamming. Web spam is the deliberate manipulation of search engine indexes. It is one of the search engine optimization methods. Implementing web spam on a search engine reduces the redundant and non-desirable results. In our paper we discuss the features which are responsible for web page ranking. We also discuss the results of the different classification techniques on our dataset which we process from the WEBSPAM-UK2006 Dataset. We are also proposing a feature which will help in the web spam detection.**

***Index Terms*—web spam; feature selection; classification technique; N gram algorithm;**

## I. INTRODUCTION

Internet has become an indispensable method to communicate with each other, because of its popularization, low cost, and fast delivery of message. Internet is also widely used for search engines. Search engines make all the information in hand in few seconds. As the popularization of the search engines the problem also come in light which is Web Spam.Web Spam means spamdexing, when we search for a query in the search engines it gives results based on query. But in the results there are some links which will redirect us to the spam sites. Web spam can be very dangerous from the perspective of the user. As the spam site can attack the system of the victim by various way.

Spam site can contain malware, when user open the site the malware silently get installed on the system. The site can also affect the financial status by stilling the private information like bank account number, password and other financial information. Becchetti et al. [1], performs a statistical analysis of a large collection of Web pages. In particular, he computes statistics of the links in the vicinity of every Web page applying rank propagation and probabilistic counting over the entire Web graph in a scalable way. He builds several automatic web spam classifiers using different techniques. This paper presents a study of the performance of each of these classifiers alone, as well as their combined performance. Egele et al. [2] introduce an approach to detect web spam pages in the list of results that are returned by a search engine. In a first step, he determines the importance of different page features to the ranking in search engine results. Based on this information, he develops a classification technique that uses important features to successfully distinguish spam sites from legitimate entries. By removing spam sites from the results, more slots are available to links that point to pages with useful content. Additionally, and more importantly, the threat posed by malicious web sites can be mitigated, reducing the risk for users to get infected by malicious code that spreads via drive-by attacks. A feature is a property of a web page, such as the number of links pointing to other pages, the number of words in the text, or the presence of keywords in the title tag. To infer the importance of the individual features, he performs "black-box testing" of search engines. More precisely, he creates a set of different test pages with different combinations of features and observe their rankings. This allows us to deduce which features have a positive effect on the ranking and which contribute only a little.

**Manuscript received June 14, 2011**.

**Sumit Sahu**, Computer Science and Engineering , MANIT ,Bhopal, India, +91-9827788517, (email: sumitsahu59@gmail.com)

**Bharti Dongre**, Computer Science and Engineering , MANIT ,Bhopal, India , (emai: bharti_dongre167@yahoo.com)

**Rajesh vadhwani**, Computer Science and Engineering , MANIT ,Bhopal, India , +91-9893338992, (email: wadhwani_rejesh@yahoo.co.in).

## II. RELATIVE WORK

Wei Wang et al. [3] present use the notion of content trust for spam detection, and regard it as a ranking problem. Besides traditional text feature attributes, information quality based evidence is introduced to define the trust feature of spam information, and a novel content trust learning algorithm based on these evidence is proposed. Finally, a Web spam detection system is developed and the experiments on the real Web data are carried out, which show the proposed method performs very well in practice. Jun-Lin Lin et al. [4] Work presents three methods of using difference in tags to determine whether a URL is cloaked. Since the tags of a web page generally do not change as frequently and significantly as the terms and links of the web page, tag based cloaking detection methods can work more effectively than the term- or link-based methods. The Proposed methods are tested with a dataset of URLs covering short-, medium- and long-term users' interest.

Experimental results indicate that the tag-based methods outperform term- or link-based methods in both precision and recall. Moreover, a Weka J4.8 classifier using a combination of term and tag features yields an accuracy rate of 90.48%.Becchetti et al [5] presents a study of the performance of each of these classifiers alone, as well as their combined performance. Using this approach he is able to detect 80.4% of the Web spam in our sample, with only 1.1% of false positives. Castillo et al. [6] demonstrate three methods of incorporating the Web graph topology into the predictions obtained by our base classifier: (i) clustering the host graph, and assigning the label of all hosts in the cluster by majority vote, (ii) propagating the predicted labels to

neighboring hosts, and (iii) using the predicted labels of neighboring hosts as new features and retraining the classifier. Ntoulas et al. [7] considers some previously undescribed techniques for automatically detecting spam pages, examines the effectiveness of these techniques in isolation and when aggregated using classification algorithms. There is some paper which worked on the link spam. Mishne et al. [8] follow a language modeling approach for detecting link spam in blogs and similar pages. They examine the use of language in the blog post, a related comment, and the page linked from the comment. In the case of comment spam, these language models are likely to be substantially different. Benczúr et al. [9] propose method fights a combination of link, content and anchor text spam. He catches link spam by penalizing certain hyperlinks and compute modified PageRank values. Guang-Gang Geng et al. [10] focuses on how to take full advantage of the information contained in reputable websites (web pages). Manuel Egele et al. [11] determine the importance of different page features to the ranking in search engine results. Based on this information, he develops a classification technique that uses important features to successfully distinguish spam sites from legitimate entries. Lourdes Araujo et al. [12] present an efficient spam detection system based on a classifier that combines new link-based features with language-model (LM)-based ones. These features are not only related to quantitative data extracted from the Web pages, but also to qualitative properties, mainly of the page links. They consider, for instance, the ability of a search engine to find, using information provided by the page for a given link, the page that the link actually points at. Juan Martinez-Romo et al. [13] propose an algorithm based on information retrieval techniques to select the most relevant information and to rank the candidate pages provided for the search engine, in order to help the user to find the best replacement. Jacob Abernethy et al. [14] present an algorithm, witch, that learns to detect spam hosts or pages on the Web. Unlike most other approaches, it simultaneously exploits the structure of the Web graph as well as page contents and features. The method is efficient, scalable, and provides state-of-the-art accuracy on a standard Web spam benchmark. Benczúr et al. [15] proposed a novel method based on the concept of personalized PageRank that detects pages with an undeserved high PageRank value without the need of any kind of white or blacklists or other means of human intervention. He assumes that spammed pages have a biased distribution of pages that contribute to the undeserved high PageRank value. He define SpamRank by penalizing pages that originate a suspicious PageRank share and personalizing PageRank on the penalties. Jay M. Ponte et al. [16] proposes a approach significantly outperforms standard tf.idf weighting on two different collections and query sets. His component of a probabilistic retrieval model is the indexing model, i.e., a model of the assignment of indexing terms to documents. WEBSPAM-UK2006[17] collection, a large set of Web pages that have been manually annotated with labels indicating if the hosts are include Web spam aspects or not. This is the first publicly available Web spam collection that includes page contents and links, and that has been labeled by a large and diverse set of judges. We also used the WEBSPAM-UK2006 dataset.

## III. MATCH SCORE FEATURE

The features play very important role for the web pages to be selected by the search engines. On the basis of the features web pages get ranking in the search engines. The web page will get amount based on this ranking. So we propose a feature which will impact on the ranking of the web page. The feature is Match Score. The Match Score is matching score of the title of the page and the URL of that web page. The algorithm to find out the match score is based on the N-Gram computation.

N gram is technique of the matching two strings and gives the outcome. Here N can be 2 - bigram, 3-trigram and so on. In our case we use 2 gram matching two compare the title and URL of the page. The result of the algorithm is in the range 0 to 1 depends on the matching percentage. The Match score feature is very helpful to decide whether the web page is spam or ham (normal). Match score is accurate as it is generated by using the N gram algorithm. We implemented this algorithm in Java.

This algorithm convert the both titles and the URL into the strings. After converting the title and URL into strings the next step is to implement to execute the N gram algorithm.

The N gram algorithm is as follows:

Suppose we have two string s1 and s2, each having a length of 7 and 8 respectively.

Step 1: we split the string into groups of bi grams means two characters.

Step 2: we remove the common bi grams from both the strings.

Step 3: we start matching the bigrams of the first string to second string and make count of that.

Step 4: we apply a formula to calculate the N Gram Measure or similarity factor

$$S = \frac{2C}{A+B}$$

Here,

S = Matching Score

C = Number of Common Bigram between two strings

A = Number of unique Bigrams in first string

B = Number of Unique Bigrams in second string

## IV. EXPERIMENTS

For Experiments we use WEBSPAM-UK2006 dataset. It is based on a set of pages obtained from a crawl of the .uk domain. The data set was collected in May 2006 by the research group of the Laboratory of Web Algorithmic. At the Universit`a degli Studi di Milano.

The data set was obtained using the UbiCrawler [18] software using breadth-first search. The crawl started from a large set of seed pages listed in the Open Directory Project. The seed set contained over 190,000 URLs in about 150,000 hosts. As a result, 77.9 million pages were collected, corresponding to roughly 11,400 hosts.

A group a volunteers, coordinated by the Universit`a diRoma “La Sapienza”, was asked to label each host as “nor-mal”, “borderline” or “spam”.

At the end of the process, 2,725 hosts were evaluated by at least two assessors and were classified as

"normal" or "spam".

Now we process data out of this dataset for our experiment. This is shown in Table 4.1.

**[Table 4.1 Train and Test Dataset]**

| Datasets | Training | | Testing | | Spam: Ham ratio |
|---|---|---|---|---|---|
| | Spam | Ham | Spam | Ham | |
| Dataset1 | 437 | 305 | 110 | 79 | 3:2 |
| Dataset2 | 545 | 311 | 189 | 92 | 2:1 |

Now we apply various classification schemes on these two datasets and the results are shown in the Table 4.2 and 4.3 for dataset1 and dataset2 respectively.

**Table 4.2**

**Experiment 1:** In this dataset1 having a spam: ham ratio of 3:2 is used.

| Classification Techniques | Recall | Precision | Accuracy |
|---|---|---|---|
| Tress.J48 | 52.4 | 49.5 | 52.38 |
| Functions.SMO | 53.4 | 45.7 | 53.44 |
| Bayes.Naivebayes | 56.1 | 43.9 | 56.09 |
| Meta.bagging | 54.0 | 50.8 | 53.96 |
| Meta.Logitboost | 55.0 | 52.7 | 55.03 |
| Meta.bagging+ZeroR | 58.2 | 33.9 | 58.20 |
| Meta.bagging+J48 | 54.0 | 51.3 | 53.98 |
| Trees.RepTree | 54.5 | 47.8 | 54.50 |
| Trees.LMT | 57.1 | 55.7 | 57.14 |
| Meta.Multischeme | 58.2 | 33.9 | 58.20 |
| Meta.MulticlassClassifier | 58.7 | 57.2 | 58.73 |
| Functions.SMO + NormalizedPolyKernel | 59.3 | 57.5 | 59.26 |
| Lazy.LWL | 57.1 | 55.2 | 57.14 |
| Meta.CVParameterSelection | 58.2 | 33.9 | 58.20 |

**Table 4.3**

**Experiment 2:** In this dataset2 having a spam: ham ratio of 2:1 is used.

| Classification Techniques | Recall | Precision | Accuracy |
|---|---|---|---|
| Tress.J48 | 66.2 | 51.6 | 66.18 |
| Functions.SMO | 67.3 | 45.3 | 67.27 |
| Bayes.Naivebayes | 67.3 | 45.3 | 67.27 |
| Meta.bagging | 65.5 | 52.3 | 65.46 |
| Meta.Logitboost | 65.1 | 44.8 | 65.09 |
| Meta.bagging+ZeroR | 67.3 | 45.3 | 67.27 |
| Meta.bagging+J48 | 67.3 | 62.1 | 65.46 |
| Trees.RepTree | 65.8 | 59.5 | 65.82 |
| Trees.LMT | 67.3 | 45.3 | 67.27 |
| Meta.Multischeme | 67.3 | 45.3 | 67.27 |
| Meta.MulticlassClassifier | 67.3 | 45.3 | 67.27 |
| Functions.SMO + NormalizedPolyKernel | 67.3 | 45.3 | 67.27 |
| Lazy.LWL | 67.3 | 45.3 | 67.27 |

## V. CONCLUSION AND FUTURE SCOPE

The search engine makes it possible to search for any kind of data or page at finger tip. But many web pages contain the spam content and spam links. This kind of the pages will cause the loose of the time and kind of precious personal data. To solve this kind of problem a phenomenon called web spam detection is used. Various work has been done in the field of web spam as described in literature survey.

For finding web spam, features play very important role. Some features are content based and some are on link based. We chose content based feature, match score which is proved to be good feature for web spam detection. We used the naïve bayes and SVM. And we also compare different classification techniques among which the meta.multiclassClassifier perform the best.

Feature selection Algorithms can be used to reduce redundancy of dataset. Common words which are same in ham and spam differing by some threshold can be eliminated. A method can be proposed to reduce the training data by choosing appropriate instance which are support vectors. More features can also be selected to give more specific results

REFERENCES

[1] LUCA BECCHETTI, CARLOS CASTILLO, DEBORA DONATO, RICARDO BAEZA YATES, STEFANO LEONARDI "Link Analysis for Web Spam Detection"
[2] Manuel Egele , Clemens Kolbitsch , Christian Platzer, "Removing web spam links from search engine results" in Springer-Verlag France 2009 J Comput Virol (2011) 7:51–62
[3] Wei Wang , Guosun Zeng , Daizhong Tang "Using evidence based content trust model for spam detection" in Expert Systems with Applications 37 (2010) 5599–5606, Science Direct.
[4] Jun-Lin Lin "Detection of cloaked web spam by using tag-based methods" in Expert Systems with Applications 36 (2009) 7493–7499, Science Direct.
[5] Luca Becchetti, Carlos Castillo, Debora Donato, Stefano Leonardi, and Ricardo Baeza Yates. Link-based characterization and detection of web spam. In Proceedings of the 2nd International Workshop on Adversarial Information Retrieval on the Web (AIRWeb), 2006
[6] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: web spam detection using the web topology. Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pages 423–430, 2007.
[7] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In

Proceedings of the 15th International World Wide Web Conference (WWW), pages 83–92, Edinburgh, Scotland, 2006.

[8] Gilad Mishne, David Carmel, and Ronny Lempel. Blocking blog spam with language model disagreement. In Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb), Chiba, Japan, 2005.

[9] A. A. Benczúr, I. Bíró, and K. Csalogány. Detecting nepotistic links by language model disagreement. In Proceedings of the 15th International World Wide Web Conference (WWW), 2006.

[10] Guang-Gang Geng, Chun-Heng Wang, Qiu-Dan Li, Lei Xu and Xiao-Bo Jin." Boosting the Performance of Web Spam Detection with Ensemble Under-Sampling Classification".

[11] Manuel Egele, Christopher Kruegel, Engin Kirda "Removing Web Spam Links from Search Engine Results".

[12] Lourdes Araujo and Juan Martinez-Romo "Web Spam Detection: New Classification Features Based on Qualified Link Analysis and Language Models" in IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 5, NO. 3, SEPTEMBER 2010.

[13] Juan Martinez-Romo, Lourdes Araujo. "Retrieving Broken Web Links using an Approach based on Contextual Information".

[14] J. Abernethy, O. Chapelle, and C. Castillo, "Webspam identification through content and hyperlinks," in Proc. Fourth Int. Workshop on Adversarial Information Retrieval on the Web (AIRWeb), Beijing, China, 2008, pp. 41–44.

[15] András A. Benczúr, Károly Csalogány, Tamás Sarlós, Máté Uher "SpamRank – Fully Automatic Link Spam Detection Work in progress" in Proc. First Int. Workshop on Adversarial Information Retrieval on the Web (AIRWeb, Chiba, Japan, 2005, pp. 25–38

[16] Jay M. Ponte and W. Bruce Croft, "A Language Modeling Approach to Information Retrieval" in Proc. 21st Annu. Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'98), New York, 1998, pp. 275–281, ACM.

[17] C. Castillo, D. Donato, L. Becchetti, P. Boldi, S. Leonardi, M. Santini, and S. Vigna, "A reference collection for web spam," SIGIR Forum, vol. 40, no. 2, pp. 11–24, 2006.

[18] P. Boldi, B. Codenotti, M. Santini, and S. Vigna "UbiCrawler: a scalable fully distributed web crawler. Software, Practice and Experience", 34(8):711–726, 2004.

[19] (Basic Book/Monograph Online Sources) J. K. Author. (year, month, day). *Title* (edition) [Type of medium]. Volume(issue). Available: http://www.(URL)

[20] J. Jones. (1991, May 10). Networks (2nd ed.) [Online]. Available: http://www.atm.com

[21] (Journal Online Sources style) K. Author. (year, month). Title. *Journal* [Type of medium]. Volume(issue), paging if given. Available: http://www.(URL)

[22] R. J. Vidmar. (1992, August). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3).* pp. 876—880. Available: http://www.halcyon.com/pub/journals/21ps03-vidmar