Combating against Web Spam through Content Features

Muhammad Iqbal[1], Malik Muneeb Abid[2]

[1] School of Information Sciences and Technology, Southwest Jiaotong University, Sichuan, Chengdu, PR China
[2] School of Transportation and Logistics, Southwest Jiaotong University, Sichuan, Chengdu, PR China

Abstract
Web spamming refers to use of unethical search engine optimization practices to gain better position on Search Engine Result Page (SERP). Making judgment on web-page to declare it as spam or ham is complicated issue because different search engines have different standards. Link-based spamming, cloaking and content spamming is main focus of different anti spam techniques. Even though these anti-spam techniques have had much success, however, these techniques still face problems when combating against a new kind of spamming techniques. This paper presents a usage of different machine learning methods which provides a solution for supervised classification problem. We have used WEBSPAM-UK-2007 public data set and in our experiments. The final results are compared and analyzed with well known classifiers. The results show that Jrip and J48 perform well compared to other two methods.
Keywords: *Content Spam, Spam detection, supervised algorithms.*

1. Introduction

The Internet has changed our world in so many ways. This is a platform to disseminate information, opportunity to expand business, a vital source for education, way of faster communication, easier way of exploration the world and increasing our productivity and bringing transparency in systems. In short, it offers a multiple benefits to everyone who is really willing to use it. Our web has experienced exponential growth for the last few decades and become the biggest repository of data ever built. Internet has become a major source for companies and individuals to become a take benefits from advanced applications such as e-commerce, teleconference, e-learning, telemedicine, video on demand and online gaming [1]. The growth of Internet can be described by several statistical factors like penetration rate and Internet users. According to Internet live stats [2], today around 40% of the world population has an internet connection, while this figure was below 1% in 1995. The end users of internet have increased tenfold from 1999 to 2013. The global growth rate of Internet users between 1/7/2000 - 1/7/2013 was 556% and currently this platform holds almost 3 billion users [2]. The statistics of same research group shows that around 75% (2.1 billion) of all internet population in the world (2.8 billion) belongs to top 20 countries. The rest of 25% (0.7 billion) users are from 178 other countries. Currently China is the world's largest Internet users market (642 million in 2014), representing nearly 22% of total internet users population.

According to the statistics of CNNIC [33], by the end of 2013, the Internet users in China were 618 million and the penetration rate of Internet was 45.8%. Among these users the number of mobile Internet were around 500 million and they are still growing. Currently around 81% mobile phone users are accessing internet. A vast majority of internet users rely on search engines to retrieve information every day. Search engines are the key to finding specific information on the vast expanse of the www, but most of the time users receive combination of ham and spam pages against their queries (see Figure 1).

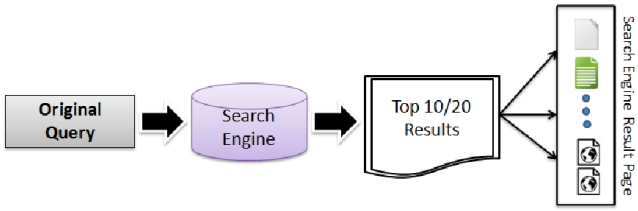

Fig. 1 Query to search engine return

It is interesting that the Web has dramatically changed the way of interaction among the people. They are now expressing their views on forums and blogs, which are now well recognized that such user generated contents on the Web provide valuable information that can be exploited for many applications. It is getting easy day by day to add information to the Web via HTML pages, wikis, blogs, and other documents. Meanwhile, it is getting tougher to differentiate between accurate or trustworthy information and inaccurate or untrustworthy information. In fact, there is lack of quality control which does not put constraints of

users to write only useful information and accurate data on the Web.

The immense growth of internet has also brought challenges to Information Retrieval (IR) systems. Due to the wide variety of data types (texts, pictures, audio, speech, video etc.), in which information is stored and communicated is additional challenging tasks for IR Systems to dig exact information to users [3].

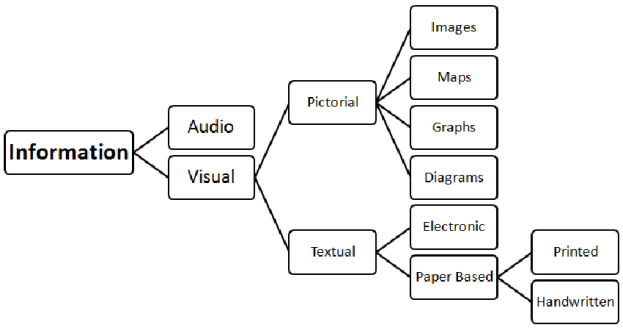

Fig. 2 Various forms of Information

Figure 2 attempts to broadly classify various sources of information. In IR studies, the information is discussed in terms of acquisition, organization, storage, retrieval, and distribution of information [4]. In order to retrieve information from internet, search engines are crucial for end users. An information retrieval process begins when an end user input a query into the system. According to Jansen et al. [5], approximately 80% of end users do not consider those entries that are placed after the third result page. Increase in the use of search engines (Google, Yahoo!, Bing, Baidu etc.) has made companies and web site developers to rank the web sites. Web Spam refers to the use of unethical search engine optimization techniques with the purpose of achieving better score against user query on search engines [6].

Apart from inaccurate or untrustworthy information, we also need to look for web spam problem. Currently, developing an effective spam detection solution is a challenging task for researchers and search engine companies.

By and large, web spamming has so many reasons. The most significant is to attract more users visit via search engines without focusing on the improvement of the quality of web page. The vital reasons for the increasing web spamming are the financial incentive as well as the dominant role of search engines. The aim of this paper is to analyze the efficiency of widely used machine learning algorithms. We selected several content features based and distinguished web spam from non-spam context. We also believe that such features could be common for the WEBSPAM‐UK2006 and WEBSPAM‐UK2007 data sets. To evaluate this hypothesis we created initially web spam detector by using WEBSPAM‐UK2007 dataset. We train and test our classifier though 10 cross validation scheme.

The remainder of this paper is organized as follows. Section 2 discusses the background & related work. Section 3 presents the introduction of widely used ranking algorithms. Section 4 presents the model for Content Spam detection and brief discussion on algorithms used for making content model. Section 5 provides the evaluation of our approach and Section 6 gives the final conclusion of this paper and the intended future work.

2. Related Work

The rise of Web spam was started in the mid-1990s and it has been growing in importance with the expansion of the internet. However, its study related to web spam in the academic domain is quite recent. The importance of spamdexing (derived from spam and indexing) and quality of results against users queries to the search engines was discussed by Henzinger et al. [7]. Gyongyi and Garcia-Molina [8] suggested taxonomy of Web Spam pages. Most of the research focuses on some of the main types of web Spam i.e. Content, Cloaking, Click and Link Spam.

Content spamming is believed to be the first web spam technique which was used to subvert the ranking of search engines. It was favorite spamming method for spammers because of the fact that most search engines apply the information retrieval models based on a page content to rank web pages, such as a vector space model [9], BM25 [10], or statistical language models [11]. Hence, spammers analyze the weaknesses of these models and exploit them. For example, spammers mislead search engines by forging of Term Frequency–Inverse Document Frequency (TF-IDF) score in their web sites [12].

Different studies [13, 14] have been done to analyze the importance of web page content and associated properties to detect web spam. Ntoulas et al. [15] introduced new features based on checksums and word weighting techniques. Ntoulas et al. [15] used a randomly generated dataset based on different domains (i.e., the.biz,.us,.com,.de,.net,.uk,.org and.edu), to show the working of their technique. Outcome of the study showed that about 70% of the pages of the.biz, 35% of the.us , and .com, .de , .net have between 15% and 20% of Web Spam pages. Although this amount is lower, it still remains high. On the other hand, their work also proved that the.edu domain is completely free from Web Spam. In the same work the relationship between the language of the pages and Web Spam was analyzed.

Fetterly et al. [16] analyzed the use of the “cut & paste” content between Web Pages in order to find Web Spam. A real-time system to detect spam pages by using HTTP response headers to extract several features was proposed

by Webb et al. [17]. Studies also exist that have combined the detection of different types of spam: Abernethy et al. [18] trained a Support Vector Machine (SVM) classifier with content and link data and Castillo et al.[19] combined content and topology information in a cost-sensitive tree.

Fetterly et al. [20] analyzed the prevalence of spam based on certain content-based properties of web pages. Their work found that features such as long host names, host names containing many dashes, dots and digits, little variation in the number of words in each page within a site, and frequent and extensive content revisions of pages between successive visits, are good indicators of spam web pages.

The selection of important features that depicts spam and web spam internet archives are discussed as sources for setups motivated by the needs of Internet preservation is discussed by Erdélyi et al. [21]. Ten features generated by the genetic programming were proposed to improve classification results for WEBSPAM - UK2006 by Shengen et al.[22].

Mahmoudi et al [23] have discussed different feature selection methods based on Information Gain were proposed. The original hybrid spamicity score approach was used by Algur and Pendari [24]. The importance of various classes of web spam features add to classification accuracy is addressed by Erdélyi et al [25]. Applied Latent Dirichlet allocation language model to generate input for the classifier is done by Bíró et al [26].

Different researchers [13, 35, 36] use the WEBSPAM-UK2006 and WEBSPAM - UK2007 datasets [27] to obtain better results to classify web spam. We have compared results obtained by those works with our results.

3. Famed Ranking Algorithms

In order to get better position on search results, spammers usually use different spamming techniques to deceive search engines.

Most of these spamming methods (Click, Cloaking, link farming, and keyword stuffing) succeed in lots of cases to betray the ranking algorithms adopted by different search engines. The success of spamming techniques to betray a search engine yields non-relevant results to the query, and this hurts the reputation of search engine. This also frustrates the users and in many cases majority switches to another search engines.

This discussion section presents three significant ranking algorithms: i)Term Frequency-Inverse Document Frequency, ii)Page Rank, and iii) Hyperlink-Induced Topic Search. We also establish how spammers attempt to deceive these three algorithms to obtain the best possible rank for the spammed Web pages in the SERP.

3.1 The Term Frequency-Inverse Document Frequency (TF-IDF)

This is a numerical statistical method which is often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document or in a collection of documents (corpus)[28].

In particular, the TF-IDF weight is composed by two terms: i) the normalized Term Frequency (TF), which is the frequency of a word in a document, divided by the total number of words in that document; ii) the Inverse Document Frequency (IDF), which is the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears.

TF(t) = (Number of times term *t* appears in a document *d*) / (Total number of terms in the document).

i.e. number of times that term *t* occurs in document *d*. If we denote the raw frequency of t by f(t,d), then the simple tf scheme is tf(t,d) = f(t,d).

- For Boolean "frequencies":

$$tf(t, d) = 1 \quad (1)$$

- For logarithmically scaled frequency:

$$tf(t, d) = 1 + \log f(t, d). \quad 0 \text{ if } f(t, d) \text{ is zero} \quad (2)$$

- augmented frequency, to prevent a bias towards longer documents, e.g. raw frequency divided by the maximum raw frequency of any term in the document:

$$tf(t, d) = 0.5 + \frac{0.5 * f(t,d)}{max\{f(w,d) : w \in d\}} \quad (3)$$

The inverse document frequency is a measure of amount of information provided by word, that is, whether the term is common or rare across all documents. It is the logarithmically scaled fraction of the documents that contain the specific word. It is obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient:

$$idf(t, D) = \log(N/dft) \quad (4)$$

"Where N is total number of documents in the corpus and df_t **is** number of documents where the term t appears. Mathematically the base of the log function does not

matter and constitutes a constant multiplicative factor towards the overall result".

Then TF–IDF is calculated as

$$TF - IDF(t, d, D) = tf(t, d) * idf(t, D) \quad (5)$$

Spammers try to increase the TF-IDF scores in their desired content-based spam Web pages. For example Spammers use many repeated and unrelated words in tags of an HTML such as: the <body> tag, Anchor text, URL, Headers (<h1> ⋯ <h6> tags), <meta> tags, and the Web page <title>, with many repeated and unrelated words in order to obtain a higher TF-IDF score [29].

3.2 Hyperlink-Induced Topic Search (HITS) Algorithm

HITS algorithm, is a long-familiar method to find the Hubs and Authoritative Web pages, and is introduced by Jon Kleinberg in 1999, as a link analysis algorithm. It is aimed before the PageRank algorithm used for ranking Web pages [29]. HITS computes hub and authority score for each of the node in graph. Hub score indicates the Web pages that work as large directories, that do not actually hold the information. Rather it points to many authoritative Web pages, which actually store the information. So a good hub represented a Web page that points to many other Web pages. The second type is called authority Web page which holds the existent information, and a good authority is represented as a Web page which was pointed to by several hubs [29-30].

HITS calculate two values for each Web page: the first value is for the authority which represents the score of the content-based Web page, and the second value is for the hub, which estimates the score of its links to other Web pages [29].

∀p, we compute $A(p)$ using equation (6)

$$A(p) = \sum_{i=1}^{n} H(i) \quad (6)$$

Where A(p) is the Authority for *p* Web page; *n* is the total number of Web pages that are linked to *p*; and the H(i) is the hub value for the Web page that points to *p*.

Below equation (7) expresses the Hub Update Rule:

∀p, we compute $H(p)$ using equation(7)

$$H(p) = \sum_{i=1}^{n} A(i) \quad (7)$$

where $H(p)$ is the Hub for *p* Web page; *n* is the total number of Web pages *p* connected to; and the $A(i)$ is the Authority values for page *p*.

The Web page is considered to be as a good hub if it points to many good authoritative, and the Web page is assorted as a good authority if it is referred to by many good hubs. The hub values can be spammed through the use of link farms by adding the spam outgoing links to the reputable Web pages. So in this fashion spammers attempt to increase the hub values, and attract several incoming links from the spammed hubs to point to the target spam Web pages [29].

3.3 PageRank Algorithm

This famous algorithm was proposed and developed in 1998 by Google's founders (Larry Page and Sergey Brin [34]) to create a new kind of search engine as a part of their research project. It defines a numeric score which measures the closeness of specific Web pages relevance to particular queries. It is important due to the high score value of PageRank that determines the list of SEPR for corresponding queries. Lourdes and Juan [30] reported that impact on the ranking provided by a search engines is also influenced by internal and external links in a web sites.

The PageRank algorithm is believed to be one of the main factors in Google's popularity. Hence, this algorithm and how it works is considered as a top secret from company. The last disclosed about this algorithm from Google indicates that the PageRank algorithm is a link ranking one, which takes the number of internal links as an important factor in page popularity. PageRank gives each page a numeric score that determines the popularity of that page. The overall score of a page *p* is determined by the importance (PageRank scores) of pages which have out links to that page *p* [31].

According to Michal et al [32] now PageRank has been frequently used for citation analysis but now it also been applied on the publication citation network.

It is important to note that algorithm does not rank the whole website, but it's determined for each page individually. The generic formula which appears in the literature for calculating PageRank score for a page *p* is shown in the below equation:

$$PR(p) = \frac{1-d}{N} + d\left(\frac{PR(T1)}{C(T1)} + \cdots \frac{PR(Tn)}{C(Tn)}\right) \quad (8)$$

Where PR(p) is the PageRank value for a Web page p; C(T) is the number of forward links on the page T_n; N is the total number of Web pages in the Web; PR(T) is the PageRank of page T; d is the damping factor.

A Web page with a eminent PageRank score will appear at the top of the list of SEPR as a answer to a particular query. Despite this achievement for those search engines that use PageRank as a ranking method, spammers and malicious Web administrators use some of PageRank algorithm problems to boost the rank of their Web pages illegally by using techniques that violate the SEO tips, in order to gain more visits from Web surfers to their Website. As we know PageRank is based on the link

structure of the Web, it is therefore useful to understand how addition or deletion of hyperlinks influences its score.

4. Content Spam detection model

Now detecting a spam web page is viewed as supervised text classification problem. In the supervised classification scheme, the web spam classifier needs to be trained with a set of web pages that are previously classified into two
categories, spam and ham (legitimate page). Afterwards, spam is a relative concept, conceived spam for one user may not be the same for other users. Moreover, definition of spam varies for a specific user with respect to time, then, depending only on the capabilities of the trained classifier. Our spam detection system consists of mainly two phases which include training phase and testing phase. Figure 3 depicts the working details of our working model to distinguish spam and ham pages.

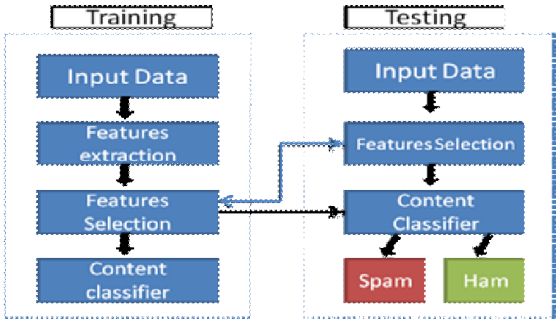

Fig. 3 Model to classify spam and ham

4.1 Experiment Methods

In this section, we present a brief description of the classification techniques used in this paper: Naive Bayes, J48, OneR and JRIP .

4.1.1 Naïve Bayes

The Naive Bayes (NB) Classifier method is based on the Bayesian theorem and is especially suited when the dimensionality of the training data (input) is high. In NB approach the classifier produces probability estimates rather than prediction. The use of NB classifier is attractive for large dataset because the creation of its model is easy due to elimination of complicated iterative parameter estimation.

NB algorithm provides a way of calculating the posterior probability, $P(c \mid x)$, from $P(c)$, $P(x)$, and $P(c \mid x)$ NB classifier assume that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This kind of assumption is called class conditional independence.

$$P(c \mid x) = P(x \mid c)P(c)/P(x) \quad (9)$$

Where $P(c \mid x)$ is the Posterior Probability of class (target) class given predictor (attribute).Where $P(x \mid c)$ is representing likelihood, which is the probability of predictor given class

Where $P(c)$ is class prior probability

Where $P(x)$ is predictor prior probability

$$P(c \mid x) = P(x_1 \mid c) \times P(x_2 \mid c) \times P(x_n \mid c) \times P(c) \quad (10)$$

Algorithm 1: Naïve Bayes

Step1: Formulate prior probability of objects { in our case two objects Spam and Ham}
Step 2: Prior Probability of object1
Step 3: Prior Probability of object2
Step 4: Classify new object through calculating likelihood of object1 and object2

4.1.2 OneR

OneR stands for "One Rule", is a simple classification algorithm that generates one rule for each predictor in the data, then selects the rule with the smallest total error as its "one rule

Algorithm 2: OneR

Step1: For each predictor *P*.
  Step1.1: For each value *V* of that predictor *P*, make a rule *R* as follows
  Step 1.2: Count how often each value *V* of target class appears
  Step 1.3: Find the most frequent class
  Step 1.4: Make the rule assign that class to this value of the predictor
Step 2: Calculate the total error of the rules of each predictor.
Step 3: Choose the predictor with the smallest total error

4.1.3 JRIP

JRIP(RIPPER) is one of the basic and widely used algorithms in machine learning. Classes are examined in increasing size and an initial set of rules for the class is generated using incremental reduced error.

Algorithm 3:

Step 1: Initialize RuleSet (RS) = Ø, and for each class, DO:
Step 2: Partitioned training set in to growing and pruning set
Step 3: Construct initial RS from examples in the growing set.
Step 4: Rrules are added incrementally to the RS until no negative examples are covered.
Step 5: Improve accuracy of rules by using reduced error pruning { the error rate >= 50%.}
Step 6: To optimize the algorithm takes in account only a final sequence of conditions from the rule and sorts the deletion that maximizes the function, ENDDO

4.1.3 J48

J48 is an implementation of the Quinlan algorithm (C4.5).By using this classifier the algorithm builds a decision tree for the available dataset, whose nodes represent discrimination rules acting on selective features by recursive partitioning of data, using depth-first strategy. This algorithm uses the concept that each attribute of the data can be used to make a decision by splitting the data into smaller subsets.

Algorithm 4: J48

Step1: Generate a decision tree based on the attribute values of the available training data.
Step 2: Select features by using highest Information Gain Value to discriminate instances
Step 3: Find data instances falling within its category have the same value for the target variable or class then we terminate that branch and assign to it the target value that we have obtained..
Step 4: Terminate that branch and assign to it the target value that we have obtained.

5. Experimental framework and the data set

We used WepSpam-uk-2007 dataset for our experiment work. This dataset is a collection of 105,896,555 web pages from 114,529 hosts in the .uk domain and is created by Yahoo!. The percentage of Spam is 6%. A team of volunteers have manually labeled (spam/non-spam/undecided) 6,479 pages only.

In order to test the accuracy of our ML algorithms we have used WEKA toolkit, a tool for automatic learning and data mining. It includes different types of classifiers and different algorithms for each classifier. Results obtained from different classification algorithms are compared to evaluate their performances. For the evaluation purpose of our method we used k cross validation technique that consists in building k data subsets(Figure 4). In each iteration, a new model is built and assessed, using one of the sets as "test set" and the rest as 'training set". We have used 10 as the value for k ("ten-fold cross validation"), since it is a widely used number. Famous accuracy measure in the context of Information Retrieval (Precision ,Recall and AUC) , are used to estimate the accuracy.

$$\text{Precision} = True\ Positive \left(True \frac{positive}{Flse} Positive \right) \quad (11)$$

$$\text{Recall} = True\ Positive \left(True \frac{positive}{Flse} Nagative \right) \quad (12)$$

Table 1 show that j-48 algorithm can distinguish the spam and non-spam pages through the used content features, and obtain better recall and AUC values.

Table 1: Experiment results

No of Features	Precision	Recall	AUC	Algorithms
10	0.895	0.946	0.496	J-48
10	0.901	0.919	0.53	Naïve Bayes
10	0.895	0.946	0.496	Jrip
10	0.913	0.946	0.502	OneR
20	0.895	0.945	0.514	J-48
20	0.9	0.918	0.526	Naïve Bayes
20	0.895	0.946	0.496	Jrip
20	0.913	0.946	0.502	OneR
30	0.913	0.945	0.528	J-48
30	0.899	0.916	0.556	Naïve Bayes
30	0.895	0.945	0.499	Jrip
30	0.913	0.946	0.502	OneR
40	0.909	0.945	0.516	J-48
40	0.9	0.923	0.553	Naïve Bayes
40	0.901	0.944	0.503	Jrip
40	0.913	0.946	0.502	OneR
50	0.921	0.936	0.643	J-48
50	0.901	0.919	0.586	Naïve Bayes
50	0.924	0.945	0.546	Jrip
50	0.913	0.946	0.502	OneR

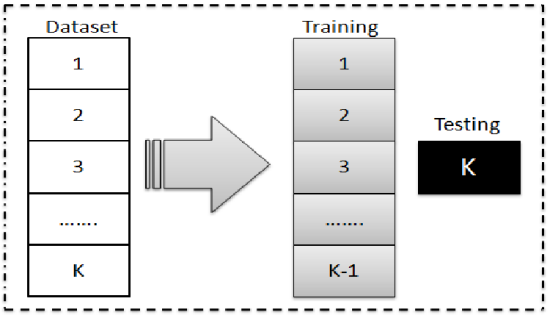

Fig. 4: k-cross validation method

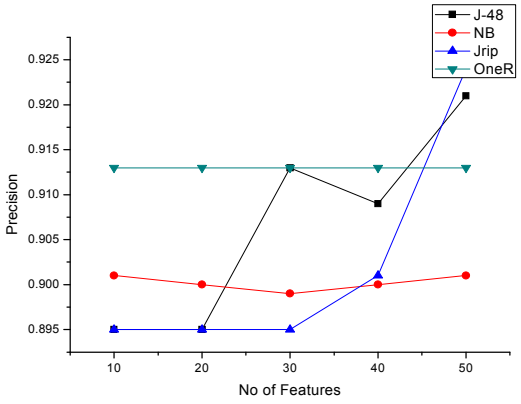

Fig. 5 Precision result

Figure 5 shows that J-48 improves precision score as the number of features to classifier increase. Naive Byes and Jrip did not depict good score, while the OneR results are steady.

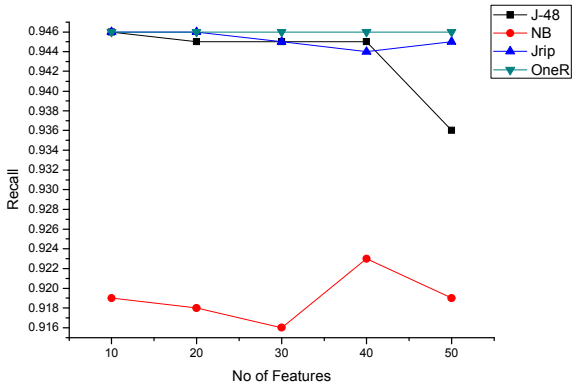

Fig. 6 Recall result

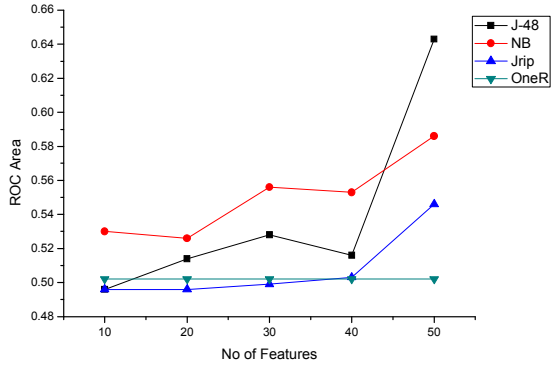

Fig. 7 AUC or ROC result

From Figure 6 we can observe the J-48 and J-rip and OneR results are much better than Naïve Byes. The results of J-48 and Naïve Byes algorithms in figure 7 demonstrate a significant increase of accuracy in terms of AUC value. Interestingly both these algorithms are performing well with the high dimensional data. Moreover, It is observed (Figure 5, 6 and 7) that OneR is producing same steady results for all variations of features. This shows that OneR didn't get any influence from changing the features.

6. Conclusion and recommendations

Nowadays detecting spam web pages is one of the major challenges for search engines in their queries results. Supervised spam filters are effective tools for attenuating spam. Many search engines already deployed different ML techniques for elimination of spam traffic. This work compares efficiency of four ML methods. The experimental results showed that Naïve Bayes and J48 perform well compared to other two methods.

We compared the results of four discussed algorithms from the WEKA toolkit on WEBSPAM–2007 presented in several works with the results obtained by the same classification tool on the same data set described by our features. Limitation of our work is that still we did not get better AUC coefficient. Future extension of this work will consider the effect of each feature in large dataset to remove the unwanted instances. Moreover, we also look forward to combine results from different feature sets so as to increase AUC rate.

Future work will also consider the development of an algorithm based on Artificial Immune System to optimize the performance of our Classifier by using content and link features of web pages.

Acknowledgments

This work is made possible through the help and support from my Professor Dr.Zhu Yan. Additionally, I sincerely thank to my parents, family, and friends, who provide the advice and financial support. The product of this research paper would not be possible without all of them.

References

1. I.Neokosmidis,N.Avaritsiotis ,Z.Ventoura ,D.Varoutas, "Assessment of the gap and (non-) Internet users evolution based on population biology dynamics", Telecommunications Policy ,Vol. 39, No. 1, 2015,pp. 14–37
2. Internet live stats. Available: http://www.internetlivestats.com/internet-users
3. M. MITRA,B.B. CHAUDHUR, "Information Retrieval from Documents: A Survey", Information Retrieval Vol 2, 2000, pp 141–163
4. B. Liu, Web data mining: Exploring hyperlinks, contents, and usage data, Berlin, Heidelberg , Springer-Verlag, 2007
5. B.J. Jansen, A. Spink, "An Analysis of Web Documents Retrieved and Viewed", in 4th International conference on Internet Computing,Las Vegas, Naveda ,2003,pp. 65-69
6. Z. Gyongyi, H. Garcia-Molina,J. Pedersen, " Combating Web Spam with TrustRank", in 30th VLDB ,Toronto,Canada, 2004
7. M.R. Henzinger, R. Motwani, C. Silverstein, " Challenges in web search engines", SIGIR Forum, Vol. 36,2002,pp.11-22
8. Z.Gyongyi,H. Gracia-Molina, " Web Spam Taxonomy", Technical report 2004-25, Stanford InfoLab, 2004
9. G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing", Commun. ACM, Vol.18, Nov. 1975.
10. S. Robertson, H. Zaragoza, and M. Taylor, "Simple bm25 extension to multiple weighted fields", In Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management,CIKM'04, Washington, D.C., 2004
11. C.Zhai, Statistical Language Models for Information Retrieval. Now Publishers Inc., Hanover, MA, 2008.

12. N. Spirin, J. Han, "Survey on Web Spam Detection: Principles and Algorithms", SIGKDD Explorations ,Vol. 13,No. 2,2011,pp. 50-64
13. V. M. Prieto,M. Alvarez,F. Cacheda, "SAAD, a content based Web Spam Analyzer and Detector", Journal of Systems & Software ,Vol. 86, No.11,2013, pp. 2906–2918
14. D. Fetterly, M. Manasse, M. Najork, " Spam, damn spam, and statistics: using statistical analysis to locate spam web pages", in Proceedings of the 7th International Workshop on the Web and Databases: Colocated with ACM SIGMOD/PODS 2004, WebDB'04, ACM, New York, NY, USA (2004), pp. 1–6
15. A. Ntoulas, M. Manasse Detecting spam web pages through content Analysis Proceedings of the World Wide Web conference, ACM Press,2006,pp.83-92
16. D. Fetterly, M. Manasse, M. Najork, " Detecting phrase-level duplication on the world wide web", in Proceedings of the 28th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval ,ACM Press,2005, pp. 170-177
17. S. Webb, J. Caverlee, and C. Pu, "Predicting web spam with http session information," in Proc. 17th ACM Conf. Information and Knowledge Management (CIKM'08), New York, 2008, pp 339-348
18. J. Abernethy, O. Chapelle, and C. Castillo, "Web spam identification through content and hyperlinks," in Proc. Fourth Int. Workshop on Adversarial Information Retrieval on the Web (AIRWeb), Beijing ,China ,2008, pp.41-44
19. C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri, Know your neighbors: Web spam detection using the web topology," in Proc. 30th Annul Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'07), New York, 2007, pp.423-430
20. M. Egele · C. Kolbitsch,C. Platzer, "Removing web spam links from search engine results", J Comput Virol (2011)
21. M. Erdélyi, A.A. Benczúr, J. Masanés, D. Siklósi, " Web spam filtering", in internet archives Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb '09 ACM, New York, NY, USA ,2009
22. L. Shengen, N. Xiaofei, L. Peiqi, W. Lin, "Generating new features using genetic programming to detect link spam", Proceedings of the 2011 Fourth International Conference on Intelligent Computation, ICICTA '11, vol. 01, IEEE Computer Society, Washington, DC, USA,2011
23. M. Mahmoudi, A. Yari, & S.Khadivi , "Web spam detection based on discriminative content and link features", In 5th International Symposium on Telecommunications (IST), 2010, pp. 542-546
24. S. Algur,N. Pendari," Hybrid spamicity score approach to web spam detection",In International Conference on Pattern Recognition Informatics and Medical Engineering (PRIME), 2012, pp. 36-40
25. M. Erdélyi, A. Garzó, A.A. Benczúr, "Web spam classification: a few features worth more", Proceedings of the 2011 Joint WICOW AIRWebWorkshop on Web Quality, WebQuality '11 ACM, New York, NY, USA (2011), pp. 27–34
26. D. Bíró, J. Siklósi,A. Szabó, A. Benczúr , "Linked latent dirichlet allocation in web spam filtering Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb '09, ACM, New York, NY, USA (2009), pp. 37–40
27. C. Castillo, D. Donato, L. Becchetti and P. Boldi , "A reference collection for Web spam", SIGIR Forum, Vol. 40,No.2 ,2006,pp.11-24
28. Q. Kuang; X. Xu, "Improvement and Application of TF•IDF Method Based on Text Classification," International Conference on Internet Technology and Applications, Wuhan, 2010, pp. 1-4
29. N. Mohammed,M. Al-Kabi,M. Izzat, Alsmadi,Heider A. Wahsheh," Evaluation of Spam Impact on Arabic Websites Popularity", Journal of King Saud University - Computer and Information Sciences Vol. 27, No 2, 2015, Pages 222–229
30. L. Araujo,J. Martinez-Romo, "Web Spam Detection: New Classification Features Based on Qualified Link Analysis and Language Models," Information Forensics and Security, Transactions on IEEE , vol.5, no.3, 2010, pp.581,590
31. A.Naga Venkata Sunil, Anjali Sardana, "A PageRank Based Detection Technique for Phishing Web Sites", IEEE Symposium on Computers & Informatics,2012
32. M. Nykl, K. Ježek ,D. Fiala, M. Dostal," PageRank variants in the evaluation of citation networks, Journal of Informatics, Vol. 8, No.3,2014, Pages 683–692
33. CNNIC Released the 33rd Statistical Report on Internet Development in China [online] Available: http://www1.cnnic.cn/AU/MediaC/rdxw/hotnews/201401/t20140117_43849.htm
34. S. Brin, L. Page, "The anatomy of a large scale hyper textual web search engine", Computer Networks and ISDN Systems, Vol. 30 ,1998, pp. 107-117
35. M. Luckner, M. Gad,P. Sobkowiak, "Stable web spam detection using features based on lexical items", Computers & Security ,Vol. 46, October 2014, Pages 79–93
36. M.Najork, "Web Spam Detection", in Encyclopedia of Database Systems, Springer Verlag, September 2009.

Muhammad iqbal was born in 1972 in Pakistan. He received B.Sc(Hons) and M.Sc degree in Computer Technology from Sindh University, Pakistan and MS in computer Science from SZABIST, Karachi, Pakistan. Since 2012, he is a PhD student in School of Information Sciences & Technology (SIST), Southwest Jiaotong University, Sichuan, Chengdu, PR China. His research interests are Network Security, Data Mining, Supervised Machine Learning algorithms and high speed data networks.

Malik Muneeb Abid was born in 1987 in Pakistan. He received B.Sc degree in Civil Engineering from U.E.T Taxila, Pakistan and MS degree in Transportation Engineering from NUST, Pakistan. Since 2013, he is a PhD student at School of Transportation and Logistics, Southwest Jiaotong University, Sichuan, Chengdu, PR China. His research interests are Network Robustness, Transportation network modeling and simulation, Data Mining, Supervised Machine Learning algorithms.
He is member of IAROR and PEC.