ISSN (Print): 2349–8919
ScieXplore: International Journal of Research in Science, *Vol 1(2), 98–109, July–December 2014* ISSN (Online): 2350–0999

Comprehensive Evaluation of Machine Learning Techniques and Novel Features for Web Link Spamdexing Detection

S. K. Jayanthi[1], S. Sasikala[2*] and J. P. Vishnupriya[3]

[1]Associate Professor, Department of Computer Science, Vellalar College for Women, Erode, India; jayanthiskp@gmail.com
[2]Assistant Professor, Department of Computer Science, KSR College of Arts and Science, Tiruchengode, India; sasi_sss123@rediff.com
[3]M.E. Student, Kumaraguru College of Technology, Coimbatore, India; vishnu18bvb@gmail.com

Abstract

World Wide Web (WWW) is a huge, dynamic, self-organized, and strongly interlinked source of information. Search engine became a vital IR (Information Retrieval) system to retrieve the required information. Results appearing in the first few pages gain more attraction and importance. Since users believe that they were more relevant because of its top positions. Spamdexing plays a key role in making high rank and top visibility for an undeserved page. This paper focus on two aspects: new features and new classifiers. First, 27 new features which are used to commercially boost the ranking and reputation are considered for classification. Along with them 17 new features were proposed and computed. Totally 44 features were combined with the existing WEBSPAM-UK 2007 dataset which is the baseline. With all these features, feature inclusion study is carried out to elevate the performance. Second aspect considered in this paper is exploring new suite of five different machine learners for the web spam classification problem. Results are discussed. New feature inclusion improves the classification accuracy of the publicly available WEBSPAM-UK 2007 features by 22%. SVM outperforms well than the other methods in terms of accuracy.

Keywords: Decision Table, HMM, Search Engine, SVM, Web Spam

1. Introduction

World Wide Web revolution has a profound impact in the past decade. Web growth is referred in exponential manner. The current size of web contains 2.18 billion pages as on Thursday, 14 November, 2013 [17]. Millions of web pages are added every day and, on the other hand millions of the web pages are modified or deleted from the web. The information available in web is diverse in nature. Since web is an open medium, there is no one monitoring the content published in web. As a consequence, there is no mechanism to control the quality or appropriateness of the content.

The manipulation of the content and link attributes brings the website to the top position in search engines visibility. This is called as spamdexing. There are two types of spamdexing: content and link. The interpretation of the link attributes of the website such as the incoming links, outgoing links and degree distribution to increase its ranking is known as the link spam. Symantec releases the following key findings in 2013 Internet Security Threat Report:

1. Web-based attacks increased 30% and 42% raised in targeted attacks in 2012.
2. 31% of all targeted attacks aimed at businesses with less than 250 employees.
3. One specific attack infected 500 organizations in a single day and a single threat infected 600,000 Macs in 2012.
4. The number of phishing sites spoofing social networking sites increased 125%.
5. Web attacks blocked in average per day in 2011 is 190,370 and in 2012 it increases to 247,350.
6. New unique web domains identified in 2010 is 43,000 and in 2011 is 57,000 and it is raised to 74,000.

Radicati Research Group Inc., a research firm based in Palo Alto, California, states that: Spam leads to decreased productivity as well as increase technical expenses in businesses $20.5 billion annually. The average loss per employee annually is approximately $1934 because of spam. 58 billion

*Author for correspondence

spam links will be sent every day within the next four years, it will cost businesses $198 billion annually. Current spam cost annually per spam action is $49 and the total cost of spam for businesses will increase to $257 billion per year if spam continues to flourish at its current rate [14].

Radicati also states that: Web threats continue to become more advanced and prevalent. Websites are becoming bloated with nested objects that most users pay little attention to. Each of these elements on a webpage can be pulled from a different domain, and one webpage can easily have dozens of domains that it pulls from. Furthermore, access to malware is becoming much easier with exploit kits.

Anyone can buy an exploit kit with relative ease that gives the buyer access to tools that can exploit machines via software flaws. These kits are easy to use and do not require any technical know-how. The threats out there have usually been focused on financial gain, but sometimes cyber criminals plays with disruptive content [15].

Symantec intelligence report released in August 2013 states that: The global spam rate is 65.2 % in August 2013. The top-level domain (TLD) for Poland, .pl, has topped the list of malicious. Sex/Dating spam continues to be the most common category, at 70.4 percent. Weight loss spam comes in second at 12.3 percent [16]. It also releases the top ten sources of Spam as depicted in Fig. 1. Addressing web spam is an important issue right now as witnessed from the reports. Many studies on web spam are carried out in previous works.

This paper is organized as follows: Section 2 discusses the related work in this problem. Section 3 discusses new features used for this work. It also gives the details of the feature inclusion experiment and enumerates the details of the dataset. Section 4 gives a brief about the suite of the classifiers used in this paper. Parameter settings of the classifiers are also briefed. Section 5 elevates the experimental setup of the paper. Section 6 briefs the evaluation metrics and presents the results. Section 7 concludes the paper.

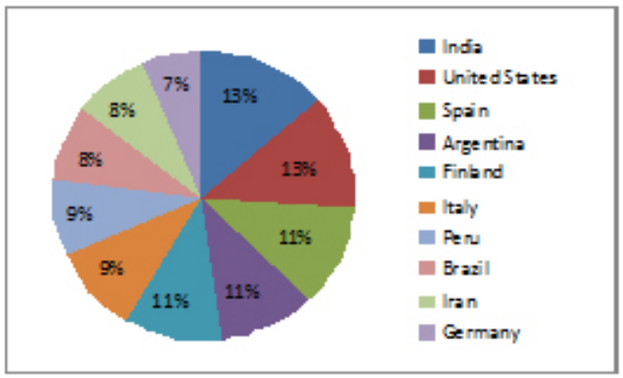


Figure 1. Top ten Spam Sources in August 2013 (Source: Symantec 2013)

2. Related Work

Egele et al. and Benczur et al. proposed new features for spam classification. Egele proposed features including number of inlinks, outlinks of a Website [1]. Delany et al. [2] and Erdelyi et al. [4] proposed classification models with different experimental setups. Chung et al. presented novel set of features including white score, spam score, relative trust, outgoing and incoming link related features, PageRank and hijacked score [3]. Kariampor et al used feature selection for the web spam classification. They used WEBSPAM-UK2007 dataset. Feature selection hikes the classification accuracy [5]. Geng et al. focused on re-extracted features for spam classification [6]. Benczur proposed the features based on the Online Commercial Intent (OCI) value of a Website including the Google Adwords value, OCI value from Microsoft Adcenter labs, Yahoo Mindset classification and Google Adsense values [7]. Proposed features are merged with WEBSPAM-UK2006 dataset. Performance is enhanced in considerable manner. Gan and Suel proposed a strategy for spamdexing detection [8]. Jayanthi and Sasikala used genetic algorithm in the implied in decision tree for Web spam classification [9]. Later they utilized the reduced error pruning logic to enhance the decision tree with regression logic for the same problem [10]. They also applied the Artificial Immune Recognition System based classifiers for the web spam problem. They proved that AIRS1 and AIRS2Parallel are two methods which give best results when compared with pioneered literature [11]. Naive bayes classifier is with principal components analysis is proposed by the same authors for the problem [12]. Tian et al. used combinatorial feature fusion method to attain optimized results [13].

3. New Features

In this work, Search Engine Optimization (SEO) features are proposed for web spamdexing detection. Spamdexing is the form of the black hat SEO. Interpreting the SEO features can help much better in discriminating the web spam. A set of 27 features belonging to SEO task is introduced in this work. Subsequently, a set of 17 computed features are introduced to improve the performance of the WLS classification.

- F1 Authority score of Domain
- F2 Authority score of the webpage

- F3 RD_Number of linking domains
- F4 Total number of anchor texts in website
- F5 SEOrank of the webpage
- F6 SEOtrust score of the webpage
- F7 Internal Links excluding 'Nofollow'
- F8 External Links excluding 'Nofollow'
- F9 Total number of internal links
- F10 Total number of external links
- F11 Cumulative total of the links in webpages
- F12 Linking RD excluding 'Nofollow'
- F13 Total number of linking RD
- F14 SD_SEOrank
- F15 SD_SEOTrust
- F16 SD_External Links excluding 'No-Follow'
- F17 SD_Total number of external links to SD
- F18 SD_Cumulative Total Links
- F19 SD_Linking RD excluding 'No-Follow'
- F20 SD_Total Linking Root Domains
- F21 RD_SEOrank
- F22 RD_SEOtrust
- F23 RD_External Links excluding 'No-Follow'
- F24 RD_Total number of external links
- F25 RD_Cumulative total links
- F26 RD_Linking Root Domains excluding 'No-Follow'
- F27 RD_Total Linking Root Domains

RD stands for the Root Domain and SD stands for the Sub Domain. Base WEBSPAM-UK 2007 dataset is processed as said in Table 1. After BCC, a sample collection sheet is obtained with 200 instances of equal samples. Corresponding

Table 1. Steps involved in feature inclusion experiment

Feature Inclusion Experiment
Step 1: SEOx is the base dataset arranged in balanced sequence with equalized spam and non-spam samples (WEBSPAM-UK 2007) Step 2: SEOy is the dataset with 27 base features Step 3: SEOxy is the combination of SEOx and SEOy which contains additional computed features listed below:
Page Trust Score PTS = (HP_SEO Rank)/((HP_SEO Rank + HP_SEO Trust))
Sub domain Trust Score SDTS = (SD_SEO Rank)/((SD_SEO Rank +SD_SEO Trust))
Root domain Trust Score RDTS = (RD_SEO Rank)/((RD_SEO Rank +RD_SEO Trust))
Cumulative Average Trust Score for Website CTW = (PTS+SDTS+RDTS)/3
Page Trust over Rank PTR = (HP_SEOTrust)/(HP_SEORank)
Subdomain Trust over Rank SDTR = (SD_SEOTrust)/(SD_SEORank)
Root Domain Trust over Rank RDTR = (RD_SEOTrust)
Page Valid Links HP_V_Links = HP_Tot_Links – (HP_Int_FL + HP_Ext_FL)
Page Valid Linking Rootdomain HP_V_LRD = HP_TLRD - HP_FLRD
Final Authority Score fAScore = P_{ASCORE}/D_{ASCORE}
Final SEO Rank for a Website f_SEORank = (HP_SEORank + SD_SEORank + RD_SEORank)
Final SEO Trust for a Website f_SEOTrust = (HP_SEOTrust + SD_SEOTrust + RD_SEOTrust)
Final SEO Spam Mass SEO _ SpamMass = (f_SEORank – f_SEOTrust)/(f_SEORank)
Home Page Spam Mass HP_SM = (HP_SEORank – HP_SEOTrust)/(HP_SEORank)
Subdomain Spam Mass SD_SM = (SD_SEORank – SD_SEOTrust)/(SD_SEORank)
Root Domain Spam Mass RD-SM= (RD_ SEORank – RD_SEOTrust)/(RD_SEORank)
Final Spam Mass value for a Website f_SM = HP_SM + SD_SM + RD_SM
Step 4: Merge Baseline with SEOx with removal of redundant features. It is referred as SEOxy. Step 5: Apply classifiers on the four datasets (Baseline, SEOx, SEOy and SEOxy) to verify the performance of the proposed features.

values of these features for the websites listed in Base dataset is collected from various sources on web [20] [21] [22]. Data values are collected between Mar'2013 and Apr'2013.

Totally four datasets are obtained from Feature inclusion experiment and they are Baseline, SEOx, SEOy and SEOxy. Performance of the new feature inclusion is tested against the base dataset with machine learning techniques. Results of experiments are discussed in the next Section.

4. Proposed Classifiers and its Specifications

4.1 Classifier

An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm, which maps input data to a category.

Figure 2 depicts the working method of the proposed work. Classifier performance depends greatly on the characteristics of the data to be classified. There is no single classifier that works best on all given problems. Various empirical tests have to be performed to compare classifier performance and to find the characteristics of data that determine classifier performance.

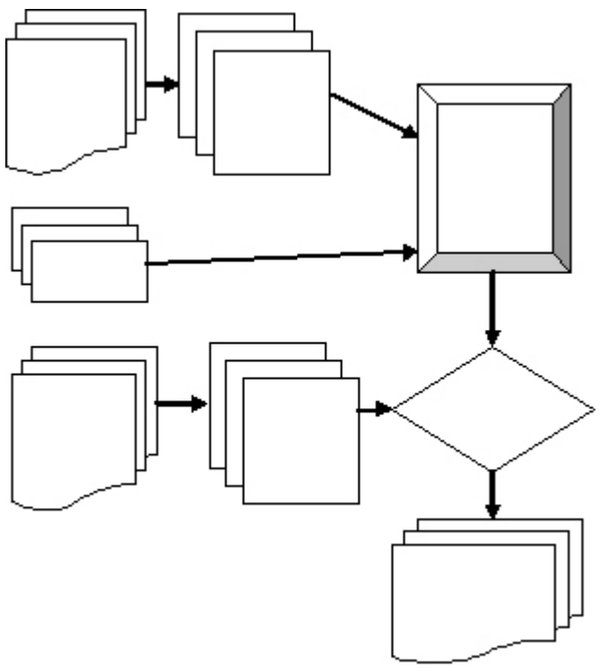


Figure 2. Machine Learning Scenario.

Determining a suitable classifier for a given problem is however still challenging. When considering a new application, the researcher can compare multiple learning algorithms and experimentally determine which one works best on the problem at hand [18]. In this paper, the following classifiers were applied:

- HMM – Bayesian network based classifier
- SVM – Statistical function based classifier
- Decision Table – Rule based learner
- RepTrees – Regression Tree based classifier
- Ensemble selection – Bagging

4.2 Hidden Markov Model (HMM)

An HMM is a stochastic finite automaton, where each state emits an observation. Xt is used to denote the hidden state and Yt to denote the observation. If there are K possible states, then Xt {1….k}. Yt is a feature-vector, Yt IRL. HMM is a state space model. HMM for this application can be defined as: HMM for spamdexing = Website Topology + Website Statistical parameters. The following are the notations used for this work and the pseudo code are:

N - number of states: Q = {q1; q2; : : : ;qT} - set of states

M - number of observations: O = {o1; o2; : : : ;oT} - set of observations

A - the state transition probability: aij = P(qt+1 = j|qt = i)

B- observation probability distribution: bj(k) = P (ot = k|qt = j) $i \le k \le M$

π - the initial state distribution

Full HMM is specified as a triplet: λ = (A,B,π)

Covariance type is set to full matrix and Iteration cutoff is 0.01with Number of states:2 and Random Seed set to 1.

4.3 Support Vector Machines (SVM)

Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. SVM is a classifier method that performs classification tasks by constructing hyper planes in a multidimensional space that separates cases of different class labels. SVM supports both regression and classification tasks and can handle multiple continuous and categorical variables. Here $SVM_{classify}$ {spam, nonspam} acts as a categorical value for classification. Linear kernel is used with epsilon 0.01, gamma 0.0, loss 0.1, nu 0.5. Probability estimate and normalize set to false and shrinking based on function set to true.

4.4 Decision Table (DT)

Simple rule based classifier. Set the number of folds for cross validation (1 = leave one out) and best-first search is used which searches the space of attribute subsets by greedy hill climbing augmented with a backtracking facility. The direction is forward and search termination is set to 5 iterations.

4.5 REPTree (RT)

REPTree stands for Reduced Error Pruning and it use the logic of information gain with variance reduction for building the tree. Missing values are dealt with by splitting the corresponding instances into pieces. The algorithm uses the error pruning for the back fitting. Initial Count – Initial class value count is set to 0.0 and maxDepth – The maximum tree depth is set to -1 for no restriction. The minNum - minimum total weight of the instances in a leaf is 2. The minVarianceProp, minimum proportion of the variance on all the data that needs to be present at a node in order for splitting to be performed in regression trees is set to 0.001. Pruning – true to perform pruning. numFolds – Determines the amount of data used for pruning. One fold is used for pruning; the rest for growing the rules is 3. seed – The seed used for randomizing the data is 1. spreadInitialCount – Spread initial count across all values instead of using the count per value.

4.6 Ensemble Selection

Ensemble Selection combines several classifiers using the meta class logic. Greedy Sort Initialization is set to true for sort initialization greedily stops adding models when performance degrades. HillclimbIterations is the number of hillclimbing iterations for the ensemble selection algorithm and it is set as100. hillclimbMetric is the metric that will be used to optimizer the chosen ensemble and optimize to ROC is used in experiments.

Tree based classifiers are combined to form the ensemble. modelRatio is the ratio of library models that will be randomly chosen to be used for each iteration and set to 0.5. numModelBags is the number of "model bags" used in the ensemble selection algorithm and set to 10. Replacement value is set to true and it checks whether models in the library can be included more than once in an ensemble. Seed is 1 and it is the random number seed to be used. sortInitializationRatio is the ratio of library models to be used for sort initialization and set to 1.0. validationRatio is the ratio of the training data set that will be reserved for validation and assigned as 0.25.

5. Experimental Setup and Evaluation

This section evaluates the performance of the proposed classifiers in identifying the spamdexing. Specifically the following aspects are analyzed:

- Will SEO features incorporation lead to more accurate classification?
- Which machine learning model suits well for the problem?

5.1 Experimental Setup

Experiments are carried out with the classifiers and stipulated datasets. Classification is carried out over 10 fold cross-validation where the entire data is utilized for training and testing. Decision threshold is the assessment score as follows.

$$\text{Assessment score} \rightarrow \begin{cases} >0.5 \text{ then Class: Spam} \\ \leq 0.5 \text{ then Class: NonSpam} \end{cases} \quad (6)$$

An overview of the classification problem is given followed by a discussion of datasets and list of classifiers used in this paper. Spamdexing detection is considered as a binary classification problem. Samples are classified according to the assessment score as follows.

Task of the classifier is to examine the samples given and predict the feature vector as either one of the aforesaid class. The classifier can succeed if the distributions of the spam values are different from nonspam values in the dataset. Labels assigned to the feature vectors should be correct in training dataset in order to get clear inference for machine learning. Distribution amount of spam and nonspam samples should be equal in training and test dataset (e.g. 60/40, 70/30 or 50/50). In general, the approach adopted in this paper can be steered as follows:

1. Gather spamdexing training set. The training set needs to be representative of the real-world use of the function. Standard WEBSPAM-UK link based dataset is used as baseline in this paper. Thus a set of input objects is gathered and corresponding outputs are also gathered from human experts.
2. Determine the input feature representation of the learned function. The accuracy of the learned function depends strongly on how the input object is represented. Typically, the input object is transformed into a feature vector, which contains a number of features that are descriptive of the object.

3. Determine the structure of the learned function and corresponding learning algorithm.
4. Complete the design. Run the learning algorithm on the gathered training set. Some learning algorithms require the user to determine certain control parameters. These parameters are adjusted by optimizing performance on a subset (called a validation set) of the training set and cross-validation.
5. Evaluate the accuracy of the learned function. After parameter adjustment and learning, the performance of the resulting function should be measured on a test set that is separate from the training set. Present the results.

This paper classifies the spam based on the link features and SEO features which characterize the samples. These features encode samples in very high dimensional feature vectors. The high dimensionality of these feature vectors poses certain challenges for classification. Though only a subset of the generated features may correlate with spamdexing detection, it is not known in advance which features are relevant. Feature selection is applied in order to resolve this [19]. Explanation of feature selection techniques are out of the scope of the paper. So, they are excluded. Five different machine learning techniques were experimented. A standard 10-fold cross validation is used. Dataset is subject to the classifier and results are recorded.

Performance study was carried out. Mann-whitney rank sum test is conducted to propose most effective feature. The methods are selected based on evaluation metrics explained in Section 4.3. Performance comparison results and Model analysis are presented. These are documented in subsequent sections. Individual classifiers differ in their details but the protocol adopted is same to all the models considered in this paper.

5.2 Hardware and Software Requirements

Experiments are carried out on a machine with 2 dual-core 2.33 GHz Pentium IV processors with 4 GB memory. Methods are implemented using the WEKA data mining toolkit [23]. The main objective of this experiment is to test the efficacy of the proposed features. Mann-Whitney Rank Sum test is applied to determine the best methods among the listed ones.

6. Evaluation Metrics and Test Results

The evaluation metrics used for the experiment is listed in Table 2. The total samples are divided into True Positives (A), False Positives (B), False Negatives (C) and True Negatives (D). The evaluation metrics considered in this experiment are: PPV or Recall, NPV, Sensitivity or Precision, Specificity, Accuracy and F-Measure. The formulas for all the metrics were listed in the table.

6.1 Features Interpretation and New SEO Features Introduction Results

This paper utilizes the commercial SEO features to detect the black hat SEO. Feature values are collected from the web and incorporated with the existing dataset. This paper is able to provide evident that SEO features inclusion improves the efficiency of the learning technique. WEBSPAM-UK 2007 dataset is used as the baseline. Feature inclusion improves the classification accuracy of the publicly available WEBSPAM-UK 2007 features by 22% (Figure 3).

6.2 Classifier Results

Acronyms used in the Table 3 are explained in section 4.1 and 4.2. The classifiers used and the descriptions are explained in section 4.2. The set of five classifiers are executed with the four datasets: B-BASE, SEOx, SEOy and SEOxy.

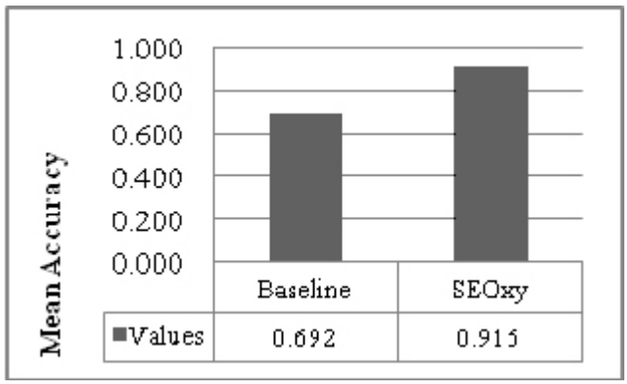


Figure 3. Mean Accuracy Comparison for Baseline and SEOxy (Considered and proposed features).

Table 2. Evaluation Metrics

		Actual outcome			
		P	N		
Test outcome	P	**A**	**B**	PPV	**A/(A+B)**
	N	**C**	**D**	NPV	**D/(C+D)**
		α	β		
		A/(A+C)	**D/(B+D)**		

P – Positive N – Negative
PPV – Positive Predictive Value
NPV – Negative Predictive Value
α – Sensitivity β – Specificity

Table 3. Classifier Results Summary

Classifier	α	B	PPV	NPV	FV	R
			B-BASE			
HMM	0.5	0	1	0	0.667	3
SVM	1	0	0.985	1	**0.992**	**1**
DT	0.462	0.462	0.462	0.462	0.462	4
RT	0.833	0	0.923	0.815	0.876	2
ES	0.462	0.462	0.462	0.462	0.462	4
			SEOx			
HMM	0.5	0	1	0	0.667	3
SVM	0.897	0	0.538	0.938	0.673	2
DT	0.712	0.641	0.569	0.769	0.632	5
RT	0.717	0	0.662	0.738	**0.688**	**1**
ES	0.709	0.653	0.6	0.754	0.65	4
Classifier	**α**	**B**	**PPV**	**NPV**	**FV**	**R**
			SEOxy			
HMM	0.5	0	1	0	0.667	2
SVM	0.796	0	0.662	0.831	**0.723**	**1**
DT	0.667	0.643	0.615	0.692	0.64	4
RT	0.685	0	0.569	0.738	0.622	5
ES	0.672	0.652	0.631	0.692	0.651	3
			B_SEOxy			
HMM	0.5	0	1	0	0.667	4
SVM	1	0	0.985	1	**0.992**	**1**
DT	0.956	1	1	0.954	0.977	2
RT	0.97	0	0.985	0.969	0.977	2
ES	0.929	0	1	0.923	0.963	3

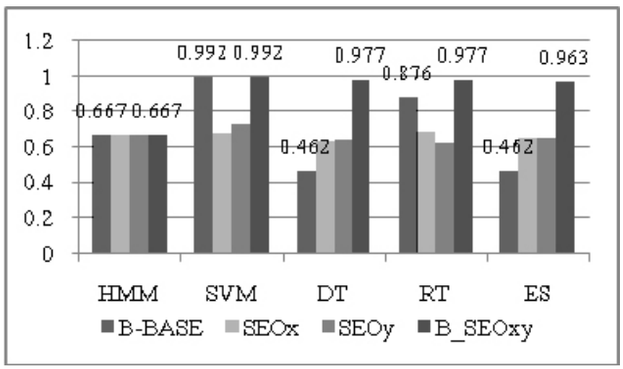


Figure 4. Accuracy Comparison of the five Classifiers for the Four datasets.

Among the dataset it is evident that the SEOxy dataset which is the merged dataset of WEBSPAM 2007 and 27 proposed new SEO features performs well.

In order to find the classifier which gives the optimal performance in terms of performance efficiency Mann-whitney Rank sum test is carried out. Results summary of the test is given in the Table 4. RS represents ranksum and U represents the value of the (RS - (Number of datasets)). FR represents the Final Rank based on the logic "least of U is high in performance".

Table 4. Mann-whitney ranksum test results

Method	BASE	SEOx	SEOy	B_SE Oxy	Rank
HMM	3	3	2	4	3
SVM	1	2	1	1	1
DT	4	5	4	2	5
RT	2	1	5	2	2
ES	4	4	3	3	4

The final order of Table 5 ensemble selection classifier performs well compared with tree based, rule based classifiers. Statistical function based HMM equally

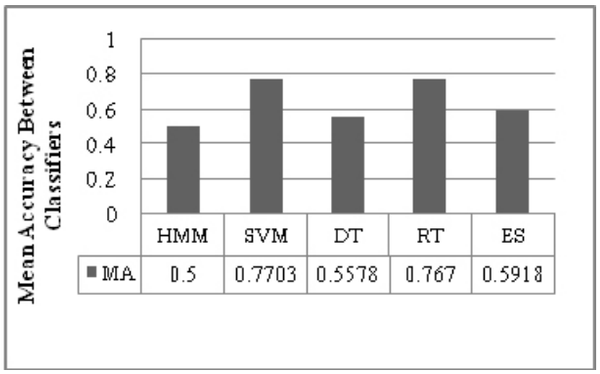


Figure 5. Overall Mean Accuracy Comparison of the classifiers.

Table 5. Rank for classifier based on accuracy

Rank	Classification Models
1	Support Vector Machine
2	REPTree
3	Hidden Markov Model
4	Ensemble Selection
5	Decision Table

performs well to Ensemble based learners. Comparative analysis shows that SVM performs well followed by Ensemble selection and HMM. Comparison of the accuracy for the classifiers is given in Figure 4.

Performance comparison of the methods on dataset is given in Figure 5 and Figure 6 shows the overall mean accuracy comparison of the classifiers in which SVM and Reptree are almost closer. SVM leads by 1% higher accuracy than the Reptree.

7. Conclusion

This paper addresses the problem of detecting spamdexing using machine learning techniques over website features. The challenge is to achieve higher efficiency in discrimination of the spam and non-spam. To this end, the contributions of this paper are:

1. Introduced new set of 44 unique features for the spamdexing classification.
2. Utilized machine learning techniques which were not explored to the problem yet.
3. Evident to show that the performance is improved by utilizing new features to the existing one.

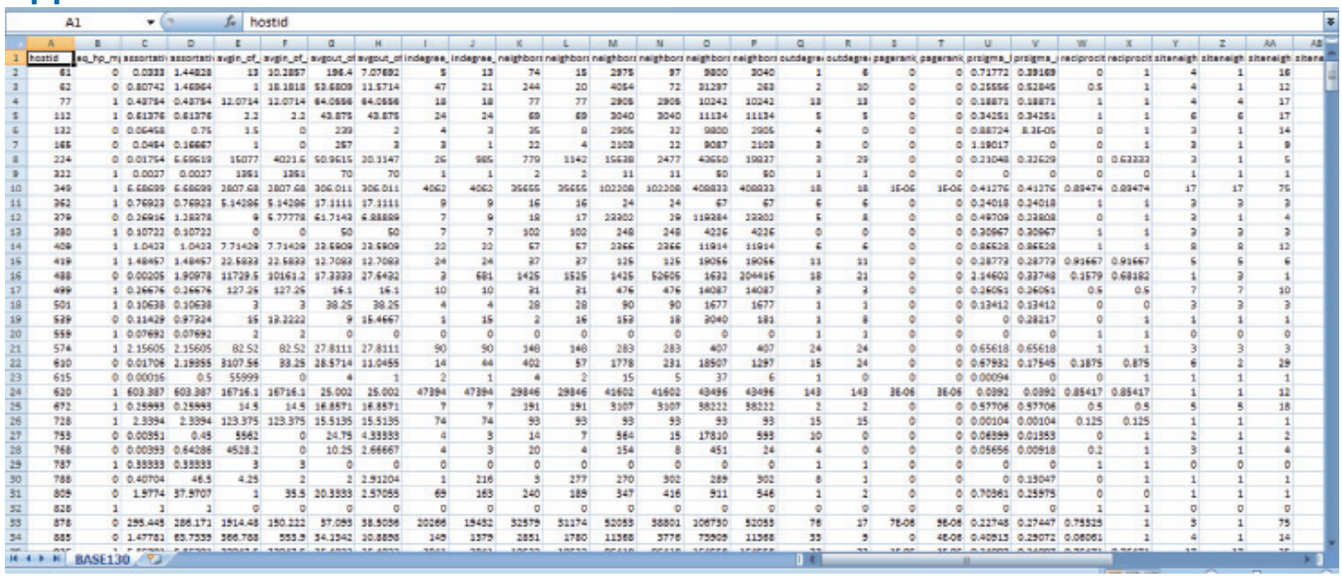
Link related features play a vital role in spam discrimination. In this paper, only link based features are considered and hence it cannot detect the content based spam. When both features are combined then it could be possible to achieve more accurate results and this will be the future scope of the research.

8. References

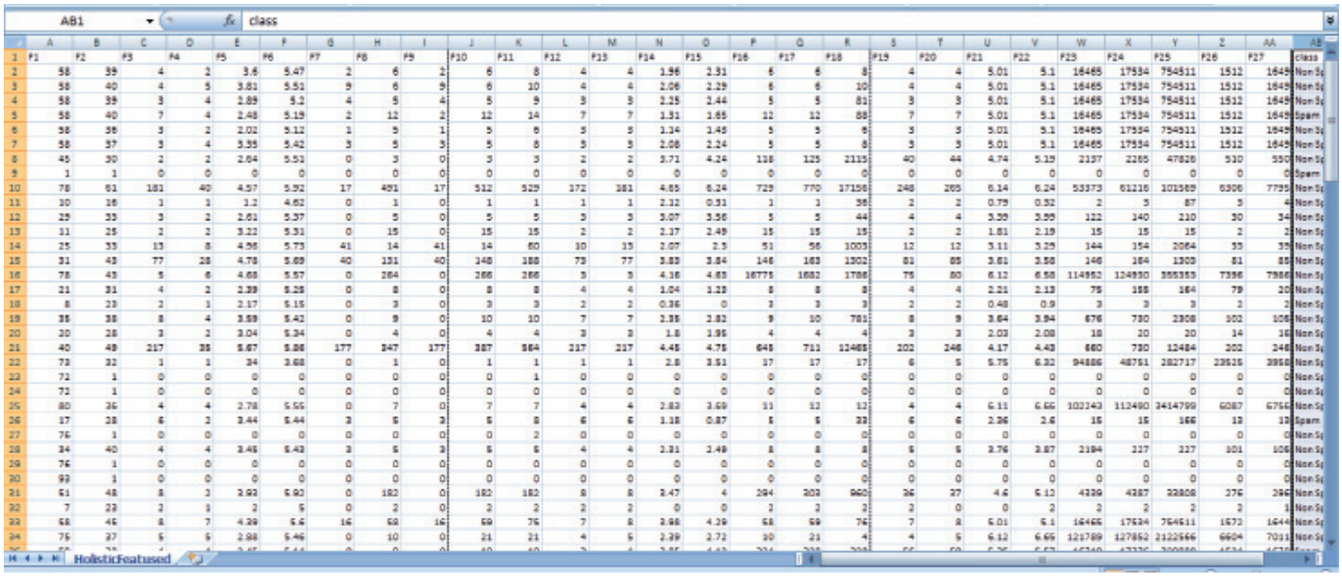
1. Egele M., Kolbitsch C., and Platzer C., "Removing Web Spam Links from Search Engine Results", *Journal of Computational Virology,* Springer-Verlag, France, 2009.
2. Delany S.J., Cunningham P., and Coyle L., "An Assessment of Case-Based Reasoning for Spam Filtering", *Springer Artificial Intelligence Review,* p. 359–378, 2005.
3. Chung Y., Toyoda M., and Kitsuregawa M., "Identifying Spam Link Generators for Monitoring Emerging Web Spam", WICOW'10, North Carolina, USA, 2010. p. 51–58.
4. Erdelyi M., Garzo A., and Benczur A., "Web spam classification: a few features worth more", WICOW/AIRWeb Workshop on Web Quality, India, 2011. p. 27–34.
5. Karimpour J., Noroozi A., and Abadi A., "The Impact of Feature Selection on Web Spam Detection", *I.J. Intelligent Systems and Applications*, p. 61–67, 2012.
6. Geng G., Wang C.H., and Dan Li Q., "Improving Web Spam Detection with Re-Extracted Features", WWW 2008, Beijing, China. 2008. ACM, p. 1119–1120.
7. Benczur A., Bıro I., Csalogany K., and Sarlos T., "Web spam detection via commercial intent analysis", 3rd International Workshop on Adversarial Information Retrieval on the Web, AIRWeb'07. 2007.
8. Gan Q., and Suel T., "Improving Web Spam Classifiers Using Link Structure", AIRWeb '07, Canada. 2007.
9. Jayanthi S.K., Sasikala S., "WESPACT: Detection of Web Spamdexing with Decision Trees in GA Perspective", International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012), Periyar University, Salem, IEEE Xplore, Listed in SCOPUS, 2012 Mar 21–23. p. 381–386.
10. Jayanthi S.K., and Sasikala S., "REPTree Classifier for Identifying Link Spam in Web Search Engines", *Ictact Journal On Soft Computing*, vol. 3(2), p. 498–505, 2007
11. Jayanthi S.K., Sasikala S., "Web Link Spam Identification Inspired By Artificial Immune System and the Impact of TPP-FCA Feature Selection on Spam Classification", *Ictact Journal On Soft Computing*, vol. 4(1), p. 633–644, 2013 Oct.
12. Jayanthi S.K., Sasikala S., "Naïve Bayesian Classifier and PCA for Web Link Spam Classification", *Georgian Electronic and Scientific Journal, GESJ: Computer Science and Telecommunications*, vol. 1(41), 2014 Mar.
13. Tian Y., Weiss G.M., and Ma Q., "A Semi-Supervised Approach for Web Spam Detection using Combinatorial Feature-Fusion", Graph labeling workshop and web spam challenge in European Conference on Machine Learning and Principles and Practice of Knowledge Discovery, 2010. p. 16–23.
14. Radicati01. Available: www.radicati.com, Accessed on Nov 2010.

15. Radicati02. Available: http://www.radicati.com/wp/wp-content/uploads/2013/05/Corporate-Web-Security-Market-2013-2017-Executive-Summary.pdf, Accessed on Oct 2013.
16. Symantec, Symantec Intelligence Report, b-intelligence_report_08-2013.en-us, Accessed on Aug 2013.
17. WWWsize. Available: http://www.worldwidewebsize.com/, Accessed on Nov 2013.
18. Wiki02. Available: http://en.wikipedia.org/wiki/Machine_learning, Accessed on 2013.
19. Wiki03. Available: http://en.wikipedia.org/wiki/Feature_selection, Accessed on 2013.
20. Dmoz open directory
21. Available: www.google.com
22. iwebtool, Available: http://www.iwebtool.com/pagerank_prediction, Accessed on 2012.
23. WEKA, Available: www.cs.waikato.ac.nz/ml/weka/

Appendix-A



A.1 Base and Proposed Dataset – CSV files



A.2 New FeaturesValues Exemplification for NLSDF Dataset

Computed Feature ID	Sample Computation	Resultant Values
PTS	$\frac{2.99}{2.99+5.13}=\frac{2.99}{8.12}$	0.368
SDTS	$\frac{1.75}{1.75+2.18}=\frac{1.75}{3.93}$	0.445
RDTS	$\frac{1.07}{1.07+1.24}=\frac{1.07}{2.31}$	0.463
CTW	$\frac{0.368+0.445+0.463}{3}$	0.425
PTR	$\frac{5.13}{2.99}$	1.71
SDTR	$\frac{2.18}{1.75}$	1.24
RDTR	$\frac{1.24}{1.07}$	1.15
HP_V_Links	$13-6+5=13-11=2$	2
HP_V_LRD	5-4	1
f_Ascore	$\frac{13}{27}$	0.46
f_SEORank	$(2.99+1.75+1.07)/3$	1.93
f_SEOTrust	$(5.13+2.18+1.24)/3=\frac{8.55}{3}$	2.25
SEO_SpamMass	$\frac{(1.93-2.85)}{1.93}=\frac{0.92}{1.93}$	0.48
HP_SM	$\frac{5.13-2.99}{2.99}=\frac{2.14}{2.99}$	0.715
SD_SM	$\frac{2.18-1.75}{1.75}$	0.245
RD_SM	$\frac{1.24-1.07}{1.07}$	0.16
f_SM	1.12	1.12

A.3 Hypothesis – Psried t-test results

H0 : SEO features ($NLSDF_{SEOBASE}$ *and* $NLSDF_{SEOCOMP}$) inclusion has no improvement in performance of *MLT*

H1 : SEO features ($NLSDF_{SEOBASE}$ *and* $NLSDF_{SEOCOMP}$) inclusion show improvement in performance of *MLT*

Two sample t test with paired samples				
Method	Without NLSDF Feature Inclusion	With NLSDF Feature Inclusion	Difference	
HMM	0.67	0.50	0.167	
SVM	0.99	0.99	0	
DT	0.46	0.98	-0.515	
RT	0.88	0.98	-0.101	
ES	0.46	0.96	-0.5	
	Min		-0.515	
	Q1-Min		0.1017	
	Med-Q1		0.2022	
	Q3-Med		0.2225	
	Max-Q3		0.1556	
			Mean : -0.202587	
			Standard Deviation (Std Dev) : 0.255485	
			Sample size: 10	
			Standard Error (Std Err) = Standard Deviation/sqrt(n) : 0.0807917	
			T = mean/s.e = -2.5075233	
			Degrees of Freedom (d = n-1 = 9	
			p-value: 0.0334464	
			t-crit: 2.26215	

T Test: Two Paired Samples								
							α : 0.05	
Groups	***Count***	***Mean***	***Std Dev***	***Std Err***	***T***	***df***	***Cohen d***	***Effect r***
Without NLSDF Feature Inclusion	10	0.49715	0.391343					
With NLSDF Feature Inclusion	10	0.699737	0.296055					
Difference	10	-0.20259	0.227449	0.071926	-2.81662	9	0.890693	0.684474

T TEST					
	p-value	*t-crit*	*lower*	*upper*	*Sig*
One Tail	0.010079	1.833113			Yes
Two Tail	0.020158	2.262157	-0.36529	-0.03988	Yes