Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 70 (2015) 434 – 441

Procedia Computer Science

4th International Conference on Eco-friendly Computing and Communication Systems (ICECCS)

Comprehensive Literature Review on Machine Learning structures for Web Spam Classification

Kwang Leng Goh[a], Ashutosh Kumar Singh[b]

[a]Curtin University, Kent St, Bentley WA 6102, Australia
[b]National Institute of Technology, Thanesar, Kurukshetra, 136119, India

Abstract

Various Web spam features and machine learning structures were constantly proposed to classify Web spam in recent years. The aim of this paper was to provide a comprehensive machine learning algorithms comparison within the Web spam detection community. Several machine learning algorithms and ensemble meta-algorithms as classifiers, area under receiver operating characteristic as performance evaluation and two public available datasets (WEBSPAM-UK2006 and WEBSPAM-UK2007) were experimented in this study. The results have shown that random forest with variations of AdaBoost had achieved 0.937 in WEBSPAM-UK2006 and 0.852 in WEBSPAM-UK2007.

Keywords: Machine Learning ; Web Spam Classification ; Web Spamming

1. Introduction

In 2006, it was estimated that approximately one seventh of English webpages were spam, which became obstacles in users information-acquisition process[46]. In 2007, the cost of Web spam was estimated at US$ 100 billion globally and United States alone suffered an estimated cost of US$ 35 billion[4]. The intention of Web spam was to mislead search engines by boosting one page to undeserved rank. Consequently, it leaded Web user to irrelevant information. This kind of exploitation degraded the Web search engines by providing inappropriate or bias query results. Henzinger et al.[30] had identified Web spam as one of the most important challenges in Web search engine industries. Many people became frustrated by constantly finding spam sites when they were looking for legitimate content. In addition, Web spam had an economic impact since a high ranking provided large free advertising and so an increase in the Web traffic volume[3]. Even worse, at least 1.3% of all search queries directed to the Google search engine contain results that link to malicious pages[21]. In addition, one consultancy estimated that Russian spammers earned roughly

[*] Corresponding author: Kwang Leng Goh
E-mail address: alex.goh@curtin.edu.au

doi:10.1016/j.procs.2015.10.069

US$2M to US$3M per year and one IBM representative claimed that a single spamming botnet was earning close to $2M per day[31]. Search engine companies generally employed human experts who specialized in detecting Web spam, constantly scanning the Web looking for spamming activities. However, the spam detection process often time-consuming, expensive and difficult to automate.

The development of an automatic Web spam detection system was an interesting problem as it concerned massive amounts of data to be analysed, the involvement of multi-dimensional attribute space with potentially hundreds or thousands of dimensions, and the extremely dynamic nature for novel spamming techniques that emerged continuously[44]. Often, large amount of Web spam pages were generated using machines by stitching together grammatically from a large collection of sentences[23]. Thus, machine learning method provided an ideal solution due to its adaptive ability to learn the underlying patterns for classifying spam and non-spam[22]. Machine learning approach can be divided into two categories —features and structures. The former depicted as the input used for classification while the latter defined the machine learning algorithm that was used for learning.

In this paper, the machine learning algorithms for Web spam detection were focused. C4.5 decision tree[39] (DT) and support vector machine[19](SVM) were two commonly used machine learning approaches among the adversarial information retrieval community. However, there were some evidences showing that SVM actually outperforms DT. Despite of that, researchers had shown that the outcome of SVM is easily manipulated in adversarial classification tasks like spam filtering[10]. Furthermore, recent papers[9,48] indicated that by injecting contaminated training data, the accuracy of the SVM will be significantly degraded. Previous studies had shown that multilayer perceptrons (MLP) neural network as an alternative Web spam classification tool[28] over SVM. However, there were still other popular machine learning algorithms within Web spam literatures that were not compared. Closest to this paper was a Web spam study reported by Silva et al.[43] who reported precision, recall and F measure in their study. In this paper, the area under the receiver operating characteristic curve (AUC) is used to evaluate the performance in Web spam detection for the reason that it did not depend on any threshold[22] like precision, recall and F-measure, and it aimed at measuring the performance of the prediction of spamicity[18].

This paper aims to provide a comprehensive machine learning approaches comparison within the Web Spam detection community using a standardized performance evaluation metric area under the receiver operating characteristic curve. In addition, several ensemble meta-learning algorithms such as boosting, bagging, rotation forest and stacking were included in the comparison to improve the classifier. Two well-known public available Web spam datasets WEBSPAM-UK2006[14] and WEBSPAM-UK2007[49] are used in this paper. Both datasets were downloaded from the Laboratory of Web Algorithmics, Universit degli Studi di Milano, with the support of the DELIS EU - FET research project. The former dataset was also used in part of a Web Spam Challenge in 2007[15,16] while the later dataset was used in Web Spam Challenge 2008[18].

The remainder of this paper is organized as follows. Related works available in the literatures are reported in Section 2, followed by descriptions of machine learning algorithms and meta-algorithms that are presented for comparison in Section 3. Section 4 describes the datasets, performance evaluation and parameters settings of the classifiers. Section 5 presents the results and discussion and lastly the conclusion in Section 6.

2. Related Work

In recent year, researchers in the adversarial information retrieval community had moved towards machine learning approach to detect Web spam. Actually the Web spam problem can be viewed as a classification problem. Machine learning constructed Web spam classifiers have shown positive results due to their adaptive ability to learn the underlying patterns for classifying spam and non-spam. The WEBSPAM-UK datasets have made a leap in Web spam community for using various machine learning models. In fact, previously there are few Web spam challenge series Web spam challenge track I[15], II[16] and III[18] which aim is to bring both machine learning and information retrieval community to solve the Web spam labelling problem.

Becchetti et al.[5] study several link-based metrics which include rank propagation for links and probabilistic counting to improve the Web spam detection techniques. Moreover, the authors conducted another similar research[7] which include more link-based metrics such as degree correlation and number of neighbours, and as a result the metrics achieve 80.4% detection rate with 1.1% false positive using DT with Boosting on WEBSPAM-UK2002 dataset. Besides link-based features, some researchers[37] propose several content-based features for Web spam detection. The content of Web pages can be modified in order to attract Web users, a technique known as keyword-stuffing. The authors experiment on 105 million Web pages and 86.2% spam pages detected using DT.

Stacked graphical learning[32], a meta-learning scheme, has shown positive results using DT in Web spam detection[17]. Some researchers[34] take advantage of stacked graphical learning by generating features by averaging known and predicted labels for similar nodes of the graph. The authors achieve improvement of 0.01% F-measure for small graph and 0.111% F-measure for large graph.

Gan and Suel[27] propose 8 content features, 14 link-based features and 3 additional features which include number of hosts in the domain, ratio of pages in this host to pages in this domain and number of hosts on the same IP address. The overall features achieved more than 90% F-measure for spam and non-spam detection in Swiss dataset from DT and SVM.

Castillo et al.[17] use the combination of link-based features from[7] and content-based features from[37] and experiment on WEBSPAM-UK2006 dataset and result in 88.4% of spam hosts detected with 6.3% false positive using DT.

A preliminary study on using linguistic features for Web spam detection is conducted by Piskorski et al.[38] and concluded by providing several discriminating Corleone and General Inquirer attributes that are promising enough to discriminate spam and non-spam.

Becchetti et al.[8] perform a detailed statistical analysis that only consider link structure of the Web for Web spam detection. Their experiments show that the performance of all combined features is comparable with that state-of-the-art spam classifier that use content attributes. Becchetti et al.[6] later use both link and content features to classify spam and non-spam. In addition, the authors use graph clustering algorithms, propagation of predicted labels and stacked graphical learning to improve the classification accuracy using DT with bagging. As a result, their proposed methodology manages to detect up to 88% of spam pages.

Linked latent Dirichlet allocation (LDA), an extension of LDA is used for Web spam classification[11]. The linked LDA technique consider linkage such as topics are propagated along links in such a way that the linked document directly influences the words in the linking Document. The authors concluded that linked LDA outperforms LDA and other baseline classifier about 3% to 8% in AUC performance.

Historical Web page information is important for Web spam classification. Dai et al.[20] propose 1270 temporal features to improve the performance of Web spam classifiers. The features are experimented using SVM on WEBSPAM-UK2007 and have shown that their approach improves the F-measure by 30% compared to the baseline classifier which only considers current page content.

Martinez-Romo and Araujo[36] presented 42 language model features to represent a Web document that calculate disagreement between two Web pages. The authors experiment using cost sensitive DT with Bagging on WEBSPAM-UK2006 and WEBSPAM-UK2007 and show that the language model features improve the F-measure of the former dataset by 6% and latter dataset by 2%. Later on, the authors combined their language model features with 12 qualified link analysis features [35] along with both content and link-based features, the overall features achieve 0.86 F-measure and 0.88 AUC performance in WEBSPAM-UK2006, and 0.40 F-measure and 0.76 AUC performance in WEBSPAM-UK2007 using DT.

Li et al.[33] generate 10 new features from link features based on genetic programming and show that the new features are well performed using SVM than 41 standardized link-based features and also 138 transformed link-based features.

Though DT is the most used machine learning algorithm when Web Spam classification first started, SVM has become state of the art machine learning model for Web spam classification in recent years as Abernethy et al.[1] obtained the best result in Web Spam Challenge 2007 with area under receiver operating characteristic (AUC) performance of 0.963 using SVM compare to C4.5 DT with AUC performance of 0.935. Yuchun et al.[50] obtained higher AUC results with less time and space using SVM than DT in spam senders behavior analysis. Jia et al.[51] did some simulation research on machine learning models for Web spam detection and their results showed that SVM outperformed both rule-based classifier and decision tree classifier in terms of precision, recall and F1-value.

Having said so, Goh et al.[28] have shown that MLP improve the AUC performance up to 14.02% over SVM in WEBSPAM-UK2006 and up to 3.53% over SVM in WEBSPAM-UK2007. Nevertheless, there are other machine learning algorithms available in literatures which will be useful to show comparison with current state-of-the-art classifier for Web Spam detection.

3. Methodology

In this section, several machine learning algorithms from top 10 data mining algorithms[47] are described and evaluated in this paper. Furthermore, several meta-algorithms are presented to enhance the AUC results of selected machine learning algorithms.

The machine learning classifiers for Web Spam detection are:

- Support Vector Machine (SVM) - SVM[19] discriminates a set of high-dimension features using a or sets of hyperplanes that gives the largest minimum distance to separates all data points among classes.
- Multilayer Perceptron Neural Network (MLP) - MLP[29] is a non-linear feed-forward network model which maps a set of inputs x onto a set of outputs y using multi weights connections.
- Bayesian Network (BN) - A BN[26] is a probabilistic graphical model for reasoning under uncertainty, where the nodes represent discrete or continuous variables and the links represent the relationships between them.
- C4.5 Decision Tree (DT) - DT[39] decides the target class of a new sample based on selected features from available data using the concept of information entropy. The nodes of the tree are the attributes, each branch of the tree represents a possible decision and the end nodes or leaves are the classes.
- Random Forest (RF) - RF[13] works by constructing multiple decision trees on various sub-samples of the datasets and output the class that appear most often or mean predictions of the decision trees.
- Nave Bayes (NB) - The NB[41] classifier is a classification algorithm based on Bayes theorem with strong independent assumptions between features.
- K-nearest Neighbour (KNN) - KNN[2] is an instance-based learning algorithm that store all available data points and classifies the new data points based on similarity measure such as distance.

The machine learning ensemble meta-algorithms on the other hand are:

- Boosting algorithms - Boosting[12] works by combining a set of weak classifier to a single strong classifier. The weak classifiers are weighted in some way from the training data points or hypotheses into a final strong classifier, thus there are a varieties of boosting algorithms. Here, three boosting algorithms are introduced in this paper:
  - Adaptive Boosting (AdaBoost) - The weights of incorrectly labelled data points are adjusted in AdaBoost such that the following classifiers focus more on incorrectly labelled or difficult cases[24].
  - LogitBoost - LogitBoost[25] is actually an extension of AdaBoost where it applies the cost function logistic regression to AdaBoost, thus it classifies by using a regression scheme as base learner.
  - Real AdaBoost - Unlike most Boosting algorithms which returns binary valued classes (Discrete AdaBoost), Real AdaBoost[42] outputs a real valued probability of the class.
- Bagging - Bagging[12] is a method by generating several training sets of the same size and use the same machine learning algorithm to build model of them and combine the predictions by averaging. It is often improve the accuracy and stability of the classifier.
- Dagging - Dagging[45] generates a number of disjoint and stratified folds out of the data and feeds each chunk of data to a copy of the machine learning classifier. Majority vote is done for predictions since all the generated machine learning classifier are put into the voted Meta classifier. Dagging is useful for base classifiers that are quadratic or worse in time behaviour on the number of instances in the training data.
- Rotation Forest - The rotation forest[40] is constructed using a number of the same machine learning classifier typically decision tree independently and trained on a new set of trained features form by sub-sampling of the datasets with principal component analysis applied on each sub-sets.

4. Datasets, Performance Evaluation and Parameters Settings

In our experiments, two public available Web spam datasets WEBSPAM-UK2006 [18] and WEBSPAM-UK2007 [19] are used. Both datasets provide evaluated sets, SET 1 for training and SET 2 for testing as the motivation behind the Web Spam Challenge Series is to provide solution to combat Web spam from machine learning. The distribution of feature vectors is shown as:

Table 1. Distribution of Features Vectors

Notation	Feature Set	No. of Features
A	Content-based Features	24
B	Full Content-based Features	96
C	Link-based Features	41
D	Transformed Link-based Features	138

Feature A denotes the content-based features. Most of these features are extracted from Ntoulas et al. [24] and they comprise of the number of words in the page, number of words in the title, average word length, fraction of anchor text and visible text, compression rate, corpus precision and corpus recall, query precision and query recall, independent trigram likelihood, and entropy of trigrams. In total, there are 24 content-based features.

Feature B denotes the full content-based features. Since feature A are based on page feature, the authors [26] aggregate the content-based features for pages in order to obtain content-based features for hosts. Therefore, in total there are 96 content-based features (4 x feature A).

Feature C denotes the link-based features. Most are computed on the home page and also the page with the maximum PageRank in each host. The link-based features include degree-related measures like in-degree, out-degree, edge-reciprocity and assortativity coefficient. Besides this degree related features, PageRank, TrustRank, truncated PageRank and estimation of supporters are also included in this link-based features. In total there are 41 link-based features.

Feature D denotes the transformed link-based features. They are just simple numeric transformations and combinations of the link-based features. After transformation, there are 138 transformed link-based features.

Details on the standard feature vectors can be found in [26]. More details on the link-based features can be found in [22] while the content-based features can be found in [24].

Note that not all hosts provide content-based features, thus hosts that provide only link-based features are discarded in this study in order to maintain consistency and fairness for machine learning algorithms. Below is the distribution of spam and non-spam in WEBSPAM-UK2006 and WEBSPAM-UK2007

Table 2. Distributions of spam and non-spam in WEBSPAM-UK2006 and WEBSPAM-UK2007

	WEBSPAM-UK2006			WEBSPAM-UK2007		
	SET 1	SET 2	TOTAL	SET 1	SET 2	TOTAL
SPAM	553	1250	1803	208	113	321
NON-SPAM	3510	601	4111	3641	1835	5476

Web Spam detection is also known as binary classification problem (spam or non-spam), thus the area under receiver operating characteristic curve (AUC) is used as evaluation metrics. The receiver characteristic curve is determined by plotting true positive rate vs the false positive rate in various threshold value. AUC is a measure for accuracy and also a performance metric for logistic regression. Unlike precision and recall that depends on particular threshold [16], AUC aims at measuring the performance of the prediction of spamicity [17]. Furthermore, AUC is a well evaluated performance evaluation in Web Spam community. A perfect model will score AUC of 1, while an area of 0.5 represent a chance of flipping a coin.

There are 7 machine learning algorithms and 6 meta-learning algorithms used for comparisons in this paper. SVM and MLP are computed in Matlab 2014a (Mathworks Inc, Natick MA, USA) and the rest of the algorithms including boosting algorithms are computed using WEKA [53]. In SVM network structure, radial basis function kernel is used for its promising performance as it non-linearly maps samples to a higher dimensional space. The sigma value of RBF is varied from 1 to 50 to obtain the optimal results. Besides RBF sigma, the scalar value are tweaked for soft margin [10] to find a hyper plane that splits the examples as clean as possible; the range of the scalar value is set between 1 to 50. For MLP, scaled conjugate gradient algorithm is incorporated as a supervised learning algorithm. The weights between the neurons are randomly set between 0 and 1. The model is executed based on 1000 epoch from 1 to the number of features. Since the weights between neurons are randomly generated, the process is executed 20 times to get the average for every epoch. The rest of machine learning algorithms are tweaked to use the default parameters in WEKA while the base learner in Boosting algorithms is Random Forest as it has the best performance among the machine learning algorithms (will explain in next section), the algorithms are trained in 10 parameters.

5. Results & Discussions

Table 3 illustrate the AUC results from various machine learning classifiers. The highlighted bold AUC results denote as the highest AUC result for the particular feature set. As shown in both tables, random forest has outperform other classifiers including SVM which widely used in Web spam community as much as 0.167 in WEBSPAM-UK2006 and 0.1092 in WEBSPAM-UK2007 AUC Difference (both results from feature set C). It also has shown that the combination of both full content features (feature set B) and transformed Link-based features (feature set D) produce the highest AUC results as both link and content features give more information to classify the classes more accurately. DT and KNN on the other hand produce the poorest results due to its simplicity design of the structure, the structures are not strong enough to learn the underlying patterns compare to other powerful machine learning algorithms.

Table 3. AUC Results on WEBSPAM-UK2006 and WEBSPAM-UK2007 from Machine Learning Classifier

	WEBSPAM-UK2006						WEBSPAM-UK2007					
Classifiers	A	B	C	D	A + C	B + D	A	B	C	D	A + C	B + D
SVM	0.751	0.808	0.728	0.799	0.805	0.839	0.678	0.742	0.622	0.661	0.723	0.752
MLP	0.799	0.870	0.830	0.828	0.869	0.887	0.703	0.747	0.624	0.669	0.735	0.769
BN	0.684	0.721	0.818	0.779	0.814	0.840	0.713	0.794	0.697	0.707	0.766	0.800
DT	0.706	0.694	0.693	0.723	0.685	0.701	0.590	0.570	0.500	0.652	0.599	0.510
RF	0.828	0.895	0.895	0.894	0.921	0.927	0.727	0.819	0.731	0.721	0.808	0.850
NB	0.687	0.721	0.726	0.736	0.743	0.763	0.622	0.653	0.641	0.683	0.646	0.668
KNN	0.691	0.729	0.678	0.692	0.730	0.725	0.622	0.627	0.539	0.550	0.619	0.564

To the extension of our results, ensemble meta-algorithms are employed to improve the AUC results of random forest as it has proved to be a powerful classifier for Web spam in previous tables. 3 boosting algorithms (AdaBoost, LogitBoost and Real AdaBoost), bagging, dagging and rotation forest are conducted here in this experiment. Table 4 illustrates the AUC results on WEBSPAM-UK2006 and WEBSPAM-UK2007 using random forest with ensemble meta-algorithms. The highlighted bold AUC results in the tables denote as the highest AUC result for the particular feature set.

Table 4. AUC Results on WEBSPAM-UK2006 and WEBSPAM-UK2007 from Random Forest with Ensemble meta-algorithms

Ensemble	WEBSPAM-UK2006						WEBSPAM-UK2007					
Meta-algorithms	A	B	C	D	A + C	B + D	A	B	C	D	A + C	B + D
AdaBoost	0.813	0.896	0.870	0.873	0.919	0.930	0.737	0.802	0.693	0.705	0.827	0.852
LogitBoost	0.835	0.898	0.900	0.823	0.924	0.931	0.759	0.828	0.732	0.740	0.830	0.847
Real AdaBoost	0.840	0.903	0.904	0.904	0.929	0.937	0.758	0.828	0.737	0.740	0.827	0.850
Bagging	0.829	0.896	0.895	0.892	0.919	0.930	0.775	0.839	0.730	0.744	0.829	0.849
Dagging	0.800	0.862	0.866	0.859	0.886	0.901	0.762	0.820	0.730	0.731	0.805	0.831
Rotation Forest	0.835	0.895	0.900	0.893	0.920	0.929	0.771	0.838	0.746	0.734	0.831	0.839

Real AdaBoost has shown to improve the AUC results in WEBSPAM-UK2006 across all feature sets. However, other meta-algorithms such as AdaBoost, bagging and rotation forest have shown better results than Real AdaBoost in WEBSPAM-UK2007. Having said so, the ensemble meta-algorithms have shown to slightly improve the AUC results of random forest. The highest AUC result in WEBSPAM-UK2006 comes from Real AdaBoost with both full-content and transformed link-based features achieving 0.937 and the highest AUC result in WEBSPAM-UK2007 comes from AdaBoost with both full-content and transformed link-based features achieving 0.852. The results in this paper have also outperform previous work using MLP (0.89 and 0.77 respectively)[28]. Abernethy et al.[1] achieved 0.963 AUC using their proposed Web spam features while Li et al.[33] have developed 10 new features generated by genetic programming that work better than 41 link-based features and 138 transformed link features. The authors results are obtained on WEBSPAM-UK2006 using support vector machines. As it is indicated earlier in this chapter, the outcome of SVM is easily manipulated filtering[10]. Other features such as language models and qualified links achieved 0.88 and 0.76 for WEBSPAM-UK2006 and WEBSPAM-UK2007 using C4.5 Decision Tree[3]. Furthermore, also have outperform recent literature such as waged averaged AUC 0.895 for WEBSPAM-UK2006 and 0.745 for WEBSPAM-UK2007 dataset[35].

6. Conclusion & Future Work

Random Forest has proven to be a powerful classifier than most top data mining tools including SVM and MLP in Web spam detection with AUC results of 0.927 in WEBSPAM-UK2006 and 0.850 in WEBSPAM-UK2007 using both full content and transformed link-based features. With ensemble meta-algorithm such as Real AdaBoost and Discrete AdaBoost, the performance is slightly improve with 0.937 in WEBSPAM-UK2006 and 0.852 in WEBSPAM-UK2007.

This paper though only focuses on the structure of the machine learning classifiers used for Web spam classification. For future work, the features for Web spam detection are intended to comprehensively compared and studied. Furthermore, the structures in this study are intended to test on other Web Spam datasets.

References

1. Abernethy J, Chapelle O, Castillo C (2010) Graph regularization methods for web spam detection. Mach Learn 81(2):207–225, DOI 10.1007/s10994-010-5171-1
2. Altman NS (1992) An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician 46(3):175–185
3. Araujo L, Martinez-Romo J (2010) Web spam detection: new classification features based on qualified link analysis and language models. Trans Info For Sec 5(3):581–590, DOI 10.1109/tifs.2010.2050767
4. Bauer JM, Eeten MJGv, Wu Y (2008) Itu study on the financial aspects of network security: Malware and spam. URL http://www.itu.int/ITU-D/cyb/cybersecurity/docs/itu-study-financial-aspects-of-malware-and-spam.pdf.
5. Becchetti L, Castillo C, Donato D, Leonardi S, Baeza-Yates R (2006) Using rank propagation and probabilistic counting for link-based spam detection. In: Proceedings of the Workshop on Web Mining and Web Usage Analysis (WebKDD 2006), ACM Press, vol 6
6. Becchetti L, Castillo C, Donato D, Leonardi S, Baeza-Yates R (2008) Web spam detection: Link-based and content-based techniques. In: The European Integrated Project Dynamically Evolving, Large Scale Information Systems (DELIS): proceedings of the final workshop, vol 222, pp 99–113
7. Becchetti L, Castillo C, Donato D, Leonardi S, Baeza-Yates RA (2006) Link-based characterization and detection of web spam. In: AIRWeb, pp 1–8
8. Becchetti L, Castillo C, Donato D, Baeza-YATES R, Leonardi S (2008) Link analysis for web spam detection. ACM Trans Web 2(1):1–42, DOI 10.1145/1326561.1326563
9. Biggio B, Nelson B, Laskov P (2012) Poisoning attacks against support vector machines. In: Langford J, Pineau J (eds) Proceeding of the 29th International Conference on Machine Learning (ICML 2012), Omnipress, pp 1807–1814
10. Biggio B, Nelson B, Laskov P (2011) Support vector machines under adversarial label noise. In: JMLR: Workshop and Conference Proceed- ings 20, MIT Press, p 97112
11. Br I, Siklsi D, Szab J, Benczr AA (2009) Linked latent dirichlet allocation in web spam filtering. DOI 10.1145/1531914.1531922
12. Breiman L (1996) Bagging predictors. Machine learning 24(2):123–140
13. Breiman L (2001) Random forests. Machine learning 45(1):5–32
14. Castillo C, Donato D, Becchetti L, Boldi P, Santini M, Vigna S (2006) A reference collection for web spam. SIGIR Forum 40(2)
15. Castillo C, Chellapilla K, Davison BD (2007) Web spam challenge track i
16. Castillo C, Davison BD, Denoyer L, Gallinari P (2007) Web spam challenge track ii
17. Castillo C, Donato D, Gionis A, Murdock V, Silvestri F (2007) Know your neighbors: web spam detection using the web topology. DOI 10.1145/1277741.1277814
18. Castillo C, Chellapilla K, Denoyer L (2008) Web spam challenge 2008
19. Cortes C, Vapnik V (1995) Support-vector networks. Machine Learning 20(3):273–297, DOI 10.1007/bf00994018, URL http://dx.doi.org/10.1007/BF00994018
20. Dai N, Davison BD, Qi X (2009) Looking into the past to better classify web spam. DOI 10.1145/1531914.1531916
21. Egele M, Kolbitsch C, Platzer C (2011) Removing web spam links from search engine results. Journal in Computer Virology 7(1):51–62
22. Erdlyi M, Garz A, Benczr AA (2011) Web spam classification: a few features worth more. DOI 10.1145/1964114.1964121
23. Fetterly D, Manasse M, Najork M (2005) Detecting phrase-level duplication on the world wide web. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp 170–177
24. Freund Y, Schapire RE (1996) Experiments with a new boosting algorithm. In: ICML, vol 96, pp 148–156

25. Friedman J, Hastie T, Tibshirani R (2000) Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). The annals of statistics 28(2):337–407
26. Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. Machine learning 29(2-3):131–163
27. Gan Q, Suel T (2007) Improving web spam classifiers using link structure. DOI 10.1145/1244408.1244412
28. Goh KL, Singh AK, Lim KH (2013) Multilayer perceptrons neural network based web spam detection application. In: Signal and Information Processing (ChinaSIP), 2013 IEEE China Summit & International Conference on, IEEE, pp 636–640
29. Haykin S (1998) Neural Networks: A Comprehensive Foundation. Prentice Hall PTR
30. Henzinger MR, Motwani R, Silverstein C (2002) Challenges in web search engines. SIGIR Forum 36(2):11–22, DOI 10.1145/792550.792553
31. Kanich C, Weaver N, McCoy D, Halvorson T, Kreibich C, Levchenko K, Paxson V, Voelker GM, Savage S (2011) Show me the money: Characterizing spam-advertised revenue. In: USENIX Security Symposium
32. Kou Z (2007) Stacked graphical learning. Thesis
33. Li S, Niu X, Li P, Wang L (2011) Generating new features using genetic programming to detect link spam. In: 2011 International Conference on Intelligent Computation Technology and Automation (ICICTA), vol 1, pp 135–138, DOI 10.1109/icicta.2011.41
34. Lszl A, Siklsi L (2007) Semi-supervised learning: A comparative study for web spam and telephone user churn. In: Graph Labelling Workshop and Web Spam Challenge, p 1
35. Luckner M, Gad M, Sobkowiak P (2014) Stable web spam detection using features based on lexical items. Computers & Security 46:79–93
36. Martinez-Romo J, Araujo L (2009) Web spam identification through language model analysis. DOI 10.1145/1531914.1531920
37. Ntoulas A, Najork M, Manasse M, Fetterly D (2006) Detecting spam web pages through content analysis. DOI 10.1145/1135777.1135794
38. Piskorski J, Sydow M, Weiss D (2008) Exploring linguistic features for web spam detection: a preliminary study. DOI 10.1145/1451983.1451990
39. Quinlan JR (1993) C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc.
40. Rodriguez JJ, Kuncheva LI, Alonso CJ (2006) Rotation forest: A new classifier ensemble method. Pattern Analysis and Machine Intelligence, IEEE Transactions on 28(10):1619–1630
41. Russell S, Norvig P (1995) Artificial intelligence: a modern approach
42. Schapire RE, Singer Y, Singhal A (1998) Boosting and rocchio applied to text filtering. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp 215–223
43. Silva RM, Yamakami A, Almeida T (2012) An analysis of machine learning methods for spam host detection. In: Machine Learning and Applications (ICMLA), 2012 11th International Conference on, IEEE, vol 2, pp 227–232
44. Sydow M, Piskorski J, Weiss D, Castillo C (2007) Application of machine learning in combating web spam
45. Ting KM, Witten IH (1997) Stacking bagged and dagged models. In: ICML, Citeseer, pp 367–375
46. Wang YM, Ma M, Niu Y, Chen H (2007) Spam double-funnel: Connecting web spammers with advertisers. In: Proceedings of the 16th international conference on World Wide Web, ACM, pp 291–300
47. Wu X, Kumar V, Quinlan JR, Ghosh J, Yang Q, Motoda H, McLachlan GJ, Ng A, Liu B, Philip SY (2008) Top 10 algorithms in data mining. Knowledge and Information Systems 14(1):1–37
48. Xiao H, Xiao H, Eckert C (2012) Adversarial label flips attack on support vector machines
49. Yahoo! (2007) Web spam collections. URL http://barcelona.research.yahoo.net/webspam/datasets/
50. Yuchun T, Krasser S, Yuanchen H, Weilai Y, Alperovitch D (2008) Support vector machines and random forests modeling for spam senders behavior analysis. In: Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE, pp 1–5, DOI 10.1109/glocom.2008.ecp.419
51. Zhiyang J, Weiwei L, Wei G, Youming X (2012) Research on web spam detection based on support vector machine. In: Communication Systems and Network Technologies (CSNT), 2012 International Conference on, pp 517–520, DOI 10.1109/csnt.2012.117