Web Spam Challenge 2007 Track II
Secure Computing Corporation Research

Yuchun Tang, Yuanchen He, Sven Krasser, and Paul Judge

Secure Computing Corporation
4800 North Point Parkway, Suite 300
Alpharetta, GA 30022, USA
{ytang,yhe,skrasser,pjudge}@securecomputing.com
http://www.trustedsource.org

Abstract. To discriminate spam Web hosts/pages from normal ones, text-based and link-based data are provided for Web Spam Challenge Track II. Given a small part of labeled nodes (about 10%) in a Web linkage graph, the challenge is to predict other nodes' class to be spam or normal. We extract features from link-based data, and then combine them with text-based features. After feature scaling, Support Vector Machines (SVM) and Random Forests (RF) are modeled in the extremely high dimensional space with about 5 million features. Stratified 3-fold cross validation for SVM and out-of-bag estimation for RF are used to tune the modeling parameters and estimate the generalization capability. On the small corpus for Web host classification, the best F-Measure value is 75.46% and the best AUC value is 95.11%. On the large corpus for Web page classification, the best F-Measure value is 90.20% and the best AUC value is 98.92%.

Key words: Web Spam, Web Linkage Graph, Support Vector Machine, Random Forest

1 Introduction

It is important to detect deliberate actions of deception to increase the ranking of targeted Web pages or Web hosts on search engines for internet search providers. Web can be naturally represented by a graph, where each node corresponds to a Web page/host and a directed edge from node A to node B denotes the number of hyper links from A to B. The goal of the Web Spam Challenge is to utilize machine learning methods for automatically labeling nodes to be "spam" or "normal" in such a graph. The challenge is labeling all nodes of a graph from a partial labeling of them. In the given datasets, only 10% nodes are manually labeled by human experts. For the Track II of the challenge, a single standard set of features has been provided for each node. Readers are suggested to refer http://webspam.lip6.fr/ for more details.

2 Experiments

The experiments are conducted with several steps including text-based feature preprocessing, link-based feature preprocessing, SVM modeling, dimensionality reduction, and RF modeling.

2.1 Text-based Feature Preprocessing

On the small corpus, the features are normalized Term Frequency-Inverse Document Frequency (TF-IDF) values, and we directly use them for classification modeling.

On the large corpus, the features are TF values. The maximal TF value in the corpus is 39041, so we simply divide each value by 39401 to scale each feature into [0,1].

2.2 Link-based Feature Preprocessing

Given a node A in a Web linkage graph, 9 features are extracted and listed in Table 1.

Table 1. Link-based Feature Extraction

id	name	meaning
l01	Od	the number of links from A to other nodes
l02	Odn	the number of links from A to other known normal nodes
l03	Ods	the number of links from A to other known spam nodes
l04	Id	the number of links from other nodes to A
l05	Idn	the number of links from other known normal nodes to A
l06	Ids	the number of links from other known spam nodes to A
l07	Bd	the number of other nodes that are connected with A in both directions
l08	Bdn	the number of other known normal nodes that are connected with A in both directions
l09	Bds	the number of other known spam nodes that are connected with A in both directions

Notice that we remove self-connection links, which otherwise induce noise when calculating the number of links to and/or from known nodes.

We also extract other 25 features as shown in Table 2.

The value of these 25 features is defined to be 0 if the corresponding denominator is 0.

These 34 link-based features are standardized into [0,1] before being fed into classification modeling. The standardization formula is $(x - min)/(max - min)$.

2.3 SVM Modeling

Table 3 lists characteristics of the datasets after preprocessing. The small dataset is for Web host classification while the large one for Web page classification.

Table 2. More Link-based Feature Extraction

id	meaning
l10	Odn/Od
l11	Ods/Od
l12	Idn/Id
l13	Ids/Id
l14	$Odn/(Odn + Ods)$
l15	$Ods/(Odn + Ods)$
l16	$Idn/(Idn + Ids)$
l17	$Ids/(Idn + Ids)$
l18	$Od + Id$
l19	$Odn + Idn$
l20	$Ods + Ids$
l21	$(Odn + Idn)/(Od + Id)$
l22	$(Ods + Ids)/(Od + Id)$
l23	$(Odn + Idn)/(Odn + Idn + Ods + Ids)$
l24	$(Ods + Ids)/(Odn + Idn + Ods + Ids)$
l25	Bd/Od
l26	Bd/Id
l27	Bdn/Odn
l28	Bdn/Idn
l29	Bds/Ods
l30	Bds/Ids
l31	Bdn/Bd
l32	Bds/Bd
l33	$Bdn/(Bdn + Bds)$
l34	$Bds/(Bds + Bds)$

Table 3. Characteristics Of Datasets

dataset		#features	#samples	#normal : #spam
small	training	4,924,007	907	701:206
	validation		1,800	1,412:388
large	training	4,924,007	40,000	32,083:7,917
	validation		80,000	63,874:16,126

We select Support Vector Machines (SVM) [1] for classification because it demonstrates good performance on high dimensional datasets [2]. The LIBSVM software package, which is available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`), is used for SVM modeling.

We conduct SVM modeling with both linear and RBF kernels. 3-fold cross validation is conducted on the training dataset for parameter tuning. We tune the cost parameter c, and the gamma parameter γ if RBF kernel is used. We also tune the weight parameter w_1 while $w_0 = 1$ because there are much more normal nodes than spam nodes in the training datasets. A value of 0 indicates the normal class while a value of 1 denotes spam. All other parameters are fixed at default values in LIBSVM.

On the small corpus for Web host classification, the best F-Measure value is 73.22% and the best AUC value is 93.26%, which are achieved by modeling a RBF SVM with $c = 2^{10}$, $\gamma = 0.05$ and $w_1 = 2$.

On the large corpus for Web page classification, the best F-Measure value is 90.20% and the best AUC value is 98.92%, which are achieved by modeling a linear SVM with $c = 2^{14}$ and $w_1 = 4$. SVM modeling with RBF kernel cannot achieve better performance in our experiments.

The ROC curves are shown in Figures 1-2. The dotted and dashed curves denote 3-fold cross validation performance on the given training datsets while the solid curves denote prediction performance on the given validation datasets.

It is interesting to observe that linear SVMs have almost the same or even better performance than RBF SVMs. The reason is that the number of features is already much higher than the number of samples. Hence, it is not very helpful to conduct classification modeling in a even higher feature space with RBF transformation.

2.4 Dimensionality Reduction

With a linear SVM, features can be ranked based on their contribution to SVM classification. This is the basic idea of the Support Vector Machine - Recursive Feature Elimination (SVM-RFE) algorithm [3]. By applying SVM-RFE on the small dataset, 28,051 features are selected from the original 4,924,007 features for Web host classification. Most of link-based features contribute to classification as demonstrated in Table 4. Almost the same accuracy can be achieved before and after feature selection. It seems that dimensionality reduction can improve efficiency and may also generate more insights for Web spam detection. This is an interesting future work.

2.5 RF Modeling

We also implement another learning process similar to Random Forests modeling [4] on the small dataset with the 28,051 features. Firstly, we generate 100 datasets with 100% bootstrapping. Secondly, we randomly select 1000 features and remove all other features for each dataset. Thirdly, a C4.5 decision tree

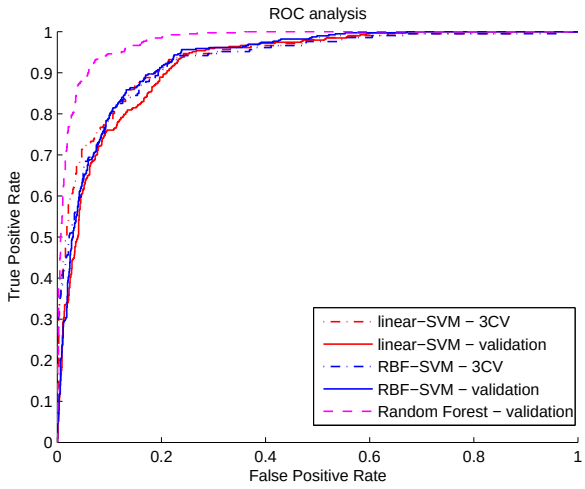

Fig. 1. ROC curves for Web host classification on the small dataset

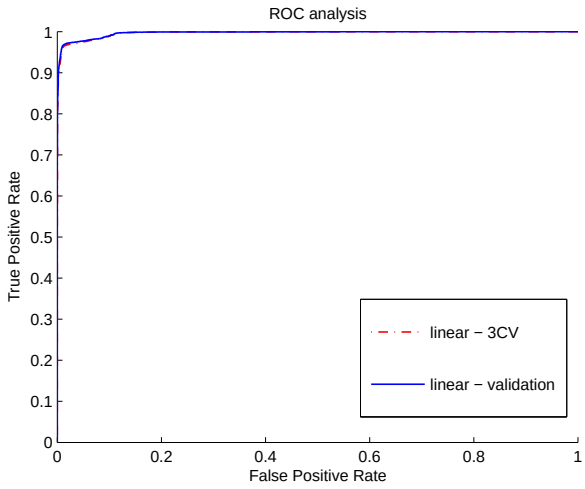

Fig. 2. ROC curves for Web page classification on the large dataset

Table 4. Link-based Features are highly ranked in SVM-RFE

id	ranking	id	ranking	id	ranking
l05	5	l34	490	l31	14020
l04	26	l01	869	l22	16252
l02	88	l06	950	l13	16706
l30	108	l21	1855	l28	17954
l19	116	l12	1876	l26	18122
l20	131	l27	3593	l23	21806
l03	151	l33	3604	l17	21911
l32	239	l15	5380	l14	24149
l11	242	l10	8017	l24	removed
l29	323	l09	10818	l08	removed
l07	326	l25	13970	l16	removed
l18	428				

[5] is modeled on each dataset in Weka with default values. Weka is available at `http://www.cs.waikato.ac.nz/ml/weka/`. Lastly, the decision values of the 100 decision trees are summed up and averaged for the final decision. Instead of 3-fold cross validation for SVM modeling, out-of-bag estimation on the training dataset is used to estimate generalization capability. This modeling method generates a even better performance with 75.46% F-Measure value and 95.11% AUC value. In Fig 1, this method demonstrates higher ROC curve than SVM counterparts. Due to limited computing power, we cannot run SVM-RFE feature selection and random forest modeling on the large dataset. But a quick look on the linear SVM shows that only 83,926 features contribute to the SVM classification.

Table 5 summarizes experiment results. For SVM methods, both 3-fold cross validation performance on the training dataset and prediction performance on the validation dataset are reported. For tree and forest methods, we report out-of-bag performance on the training dataset and prediction performance on the validation dataset.

Table 5. Experiment Results

Task	Method	F-Measure			AUC		
		CV/OB	validation	testing	CV/OB	validation	testing
host classification	RF with C4.5	78.74%	75.71%	75.46%	96.24%	95.84%	**95.11%**
	RBF SVM	72.50%	72.95%	73.22%	92.74%	93.12%	**93.26%**
page classification	Linear SVM	94.53%	94.96%	90.20%	99.64%	99.66%	**98.92%**

3 Conclusion

The data preprocessing and machine learning methods described in this paper demonstrate high accuracy on Web Spam Challenge corpora. On the small corpus for Web host classification, the best F-Measure value is 75.46% and the best AUC value is 95.11%. On the large corpus for Web page classification, the best F-Measure value is 90.20% and the best AUC value is 98.92%.

References

1. V. N. Vapnik. *Statistical Learning Theory.* John Wiley and Sons, New York, 1998.
2. C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
3. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.
4. L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
5. R. J. Quinlan, *C4.5: Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann, 1993.