SVMs for the Blogosphere: Blog Identification and Splog Detection

Pranam Kolari*, **Tim Finin and Anupam Joshi**
University of Maryland, Baltimore County
Baltimore MD
{kolari1, finin, joshi}@umbc.edu

Abstract

Weblogs, or blogs have become an important new way to publish information, engage in discussions and form communities. The increasing popularity of blogs has given rise to search and analysis engines focusing on the "blogosphere". A key requirement of such systems is to identify blogs as they crawl the Web. While this ensures that only blogs are indexed, blog search engines are also often overwhelmed by spam blogs (splogs). Splogs not only incur computational overheads but also reduce user satisfaction. In this paper we first describe experimental results of blog identification using Support Vector Machines (SVM). We compare results of using different feature sets and introduce new features for blog identification. We then report preliminary results on splog detection and identify future work.

Introduction

The blogosphere is a subset of the Web consisting of weblogs. Weblogs or blogs are web sites consisting of dated entries typically listed in reverse chronological order on a single page. While there is no general agreement on the content genre of weblogs (Krishnamurthy 2002) (Herring *et al.* 2004), they are commonly viewed to be one of personal journals, market or product commentary and filter blogs. From the perspective of search engines, the blogosphere differs from the general Web in that it consists of highly dynamic and dated content that can benefit from specialized information retrieval techniques. These new techniques make use of structured data (syndication feeds) in addition to traditional information retrieval on HTML content. While traditional search engines continue to discover and index blogs, the blogosphere has required a number of specialized search and analysis engines.

As the blogosphere continues to grow, several capabilities are becoming increasingly important for blog search engines. The first is the ability to recognize blog sites, understand their structure, identify constituent parts and extract relevant metadata. A second key competency is being able to robustly detect and eliminate spam blogs (splogs).

Most blog search engines identify blogs and index content based on update pings received from *ping servers*[1] or directly from blogs, or through crawling blog directories and blog hosting services. To increase their coverage, blog search engines continue to crawl the Web to discover, identify and index blogs. This enables staying ahead of competition in a domain where "size does matter". Even if a web crawl is inessential for blog search engines, it is still possible that processed update pings are from non-blogs. This requires that the source of the pings need to be verified as a blog prior to indexing content.[2]

In the first part of this paper we address blog identification by experimenting with different feature sets. We report results for identification of both blog home pages and all blog pages (e.g. category page, user page, post page) using SVMs. Due to the popularity of blog hosting services, many blogs can be identified by simple URL pattern matching. Hence, we consider the identification problem for the subset of blogosphere consisting of self-hosted blogs and the many less popular hosting services and compare them against human baselines. We introduce certain new features and detail how they can be effective for blog identification.

In the next part of this paper we discuss how features that fared well on blog identification perform for splog detection. The blogosphere is prone to different kinds of spam which includes comment spam, trackback spam, ping spam and splogs (spam blogs) all of which are widely employed by spammers. We restrict our work to detecting splogs, which include link spam blogs and fake blogs built through unjustified copying of web content. Initial results are encouraging and suggest interesting future directions. Our results also show that traditional email spam detection techniques by themselves are insufficient for the blogosphere.

This work is part of a larger project "**MeMeta**" (Kolari, Finin, & Joshi) that aims to develop an advanced blog search engine that will understand and represent much more information about blogs, their authors and the relationships among them. In addition to discovering blogs, extracting their metadata and identifying spam, MeMeta seeks to identify blog communities, recommend blogs to people, model

*Partial support provided by a generous fellowship from the IBM Corporation

[1] One such ping server is Pingomatic, http://pingomatic.com/.

[2] We have observed many update pings from non-blogs in the dataset we collected over a five month period.

trust relationships in blog communities and spot trends in the blogosphere. MeMeta's blog database is driven by a custom blog crawler that currently collects information on over five million blogs.

We summarize our primary contributions as follows:

- We formalize the problem of blog identification and splog detection as they apply to the blogosphere.
- We report on blog identification results with different features and identify features where results show significant improvements over human baselines.
- We introduce novel features such as anchor text for all URL's on a page and tokenized local and outgoing URL's on a page and show how they can be effective for the blogosphere.
- We report on initial results and identify the need for complementary link analysis techniques for splog detection.

The rest of the paper is organized as follows. Section two gives preliminaries and section three scopes the problem. Section four briefly describes the blog collection procedure and section five describes the labeled corpus. In section six we detail our metrics for evaluation. In section seven we describe our baselines for comparison and in section eight we report on experimental results both for blog identification and splog detection. We discuss results in section nine before concluding the paper.

Preliminaries

We begin this section by defining terms which are prerequisites for this work. The blog identification and splog detection problems we address are based on these definitions. We also introduce the features used in classification problems and Support Vector Machines.

Definitions

While there is no general consensus on the definition of blog, we use the following from Wikipedia[3] which defines a **blog** as:

> *"A blog, originally (and still) known as a weblog, is a web-based publication consisting primarily of periodic articles (normally, but not always, in reverse chronological order)."*

These periodic articles are also known as blog posts. We use a stricter definition for "blog" through additional constraints that include: (i) a blog post be dated; (ii) a blog post either provide one of comments or a trackback, or be uniquely addressable through permalinks; and (iii) not be a forum or traditional news website (edited content). In addition to a home page listing the most recent blogs, a blog also consists of other pages for category, author specific and individual blog posts.

"Splog" is a fairly new term referring to spam blogs. In line with the fact that the blogosphere is a subset of the Web, we adopt from a definition of web spam. In a recent analysis on web spam taxonomy (Gyöngyi & Garcia-Molina 2005), **web spam** is defined as:

> *"Any deliberate human action meant to trigger an unjustifiably favorable relevance or importance for some page, considering the page's true value."*

Splog is any such web spam page that is a blog. Splogs can be broadly classified into link spam blogs and fake blogs. The former is quite common in the rest of the Web, while the latter is more frequent in the blogosphere. Fake blogs unjustifiably extract text from other sources on the Web and publish them as a post[4]. As a result of this they create favorable relevance to posts on diverse topics enabling placement of context sensitive advertisements which drive revenue.

The use of fake blogs as a spamming technique is driven by the very nature of blog content and blogging infrastructure. Blogs are usually commentary oriented; in which an excerpt from another page is included as part of the post. This makes it difficult (for humans and machines alike) to differentiate between fake and authentic blogs giving splogs the credibility they do not deserve. Further blog hosting services and applications provide programmatic interfaces for posting content. This eases the automated posting of large amounts of spam content. In addition, blog search engines return results by post timestamp making fake blogs an easy and effective spamming technique. It is these fake blogs that make the splog detection problem particularly challenging. It should be noted here that comment, trackback and (update) ping spam are different aspects of blog spam though they might be related to splogs.

In what follows we detail SVMs and their related concepts as they apply to this work.

Feature Vector

Most machine learning algorithms use a vector representation of documents (web pages) for classification. Such vectors are commonly termed as feature vectors and represented by a set of features as $\{ f_1, f_2 .. f_m \}$. Every document has a specific signature for these features represented by a weight as $\{ w(f_1), w(f_2) .. w(f_m) \}$. Traditionally web page classification has required using different types of features (feature types), selecting the best (feature selection) and encoding their values in different ways (feature representation).

Feature Types

One of the most common feature types is the bag-of-words where words appearing on a page are used as features, and is quite effective for topic classification. In addition, other features like content of specific tags on a page (anchor text, title) and customized features (most popular for email spam detection) are also used either independently or in conjunction with bag-of-words.

While these traditional features are quite effective for topic classification, their applicability to categorization problems in the blogosphere has not yet been completely analyzed. Due to the nature of its publishing infrastructure certain new features can also be effective in the blogosphere. For instance, most of the blogs have specific types of local and non-local links and associated anchor text. While

[3] http://en.wikipedia.org/wiki/Blog/

[4] Ironically a search for splog on popular search engine returns such definitions listed on splogs themselves

local links are for individual posts, comments, trackbacks, category pages and archives; global links are primarily for blog and feed directories (Technorati, Feedburner, Bloglines, etc.) and blogrolls (lists of regularly followed blogs and blogs of a close knit community).

In view of this we introduce two new features for blogs: **bag-of-anchors** and **bag-of-urls**. In **bag-of-anchors**, features are extracted from the anchor text all URLs on a page. Correspondingly, in **bag-of-urls**, URLs are tokenized (split) on "/", ".", "?" and "=" and each resulting token is used as a feature. Such a tokenization will separate out specific features (e.g. dates, popular directories, hosting services, archives, comments, etc.) for blogs. While anchors have been used previously in classification tasks, they are used for anchor text on referring pages and not on the page being classified.

Blogs as a publishing mechanism have crossed international boundaries, and so have many of the blog hosting tools (e.g., MoveableType, Wordpress). Some blogs are even multi-lingual where posts in multiple languages are listed on the same page. In view of this we also employ N-grams (Darnashek 1995) which are known to be very effective in the presence of multi-lingual data. In the **bag-of-ngram** approach text on page is converted into tokens of character size "n" using a window of adjacent characters and used as features. For instance, the word "comment" in English would lead to the 4-grams (comm, omme, mme, **ment**) and a synonym "koment" in a different language would lead to the 4-grams (kome, omen, **ment**) providing improved results across languages.

We did not use either stop word elimination on our feature sets due to two primary reasons. First, SVMs are known to be quite robust even without such preprocessing and second, unedited blog content usually makes use of many such stop words and this can be a distinguishing feature for some classification tasks. Our use of N-grams made sure stemming was indirectly considered.

Feature Representation

While there are different forms of feature representation, we employed normalized term frequency (TF) and binary features. In normalized TF, weights are normalized to the closed real interval $[0, 1]$ whereas in binary, weights are limited to values from the $\{0, 1\}$ set based on their absence or presence in the document.

Feature selection

We used Mutual Information (Relative Entropy) (Fano 1961) as our primary feature selection mechanism. In simple terms it is a measure of how often a feature occurs with samples of a particular class. Relative entropy provides the information theory based association of a feature to a class and is given by:

$$I(t, c) = \log \frac{P(t, c)}{P(t) * P(c)}$$

where t is the feature for which mutual information is computed and c is class against which it is compared. $P(t, c)$ is the joint probability of t and c occurring together and $P(t)$ and $P(c)$ are their posterior probabilities. This value is approximated and evaluated as:

$$I(t, c) \approx \log \frac{A * N}{(A + C) * (A + B)}$$

Where A is the number of times t and c co-occur, B is the number of times that t occurs without c, C is the number of times c occurs without t and N is the total number of documents in the corpus. The metric for the average usefulness of a feature for the entire corpus as used by a binary classifier is given by:

$$I_{avg}(t) = P(c_1) * I(t, c_1) + P(c_0) * I(t, c_0)$$

where c_0 and c_1 are the two classes.

In addition to mutual information, we also employed frequency count of words (term frequency based feature selection) in the entire corpus. This is a naive approach for feature selection but performs well in practice for SVMs. We did not use any other feature selection techniques. Traditionally, many different techniques (Yang & Pedersen 1997) have worked well for document classification.

Support Vector Machines

Support Vector Machines (SVM) (Boser, Guyon, & Vapnik 1992) are widely used for text (and hyper-text) classification problems. They have been shown to be very effective in many domains including topic classification, sentiment classification (Pang, Lee, & Vaithyanathan 2002) and email spam classification (Drucker, Wu, & Vapnik 1999). Based on this universality of SVMs for categorization and its relative efficiency and effectiveness (Zhang, Zhu, & Yao 2004) we use SVMs for all our classification tasks.

SVMs treat the classification task as one involving two classes where labeled training examples are represented using feature vectors. Formally, two classes $y_i \in \{-1, 1\}$ are to be identified using N labeled training samples represented by $(\mathbf{x}_1, y_1), ..(\mathbf{x}_N, y_N)$, where $\mathbf{x_i}$ is the feature vector representing individual training samples. If y_i's are linearly separable, then SVM finds the optimal weight vector **w** such that

$$\|\mathbf{w}\|^2 \; is \; minimum$$

and

$$y_i * (\mathbf{w} * \mathbf{x_i} - b) \geq 1$$

SVMs are also quite effective when used in combination with kernel functions. We use only linear kernels in all our experiments.

Scope

Some might argue that recognizing a Web page as a blog as opposed to a non-blog is a relatively easy problem. Since the majority of blogs are hosted by well known hosting services (e.g., blogspot, msn-spaces), a simple process of URL pattern matching should provide sufficiently good accuracy.

Hence we eliminate the top blogging host services from the sample set. Our sample consists of many self-hosted blogs and blogs from the many less popular blog hosting services (e.g. blogspot, msn-spaces etc are eliminated). Based on the way we collected our data, our samples also feature a significant number of blogs from languages other than English.

We limit the spam problem in the Blogosphere to that of splog detection. The problem of comment spam and trackback spam is not considered and is outside the scope of our work. It can be argued that the problem of update ping spam can be tackled indirectly using the combination of blog identification (for non-blog pings) and splog detection (for splog pings). We make use of only local information as used by email spam detection techniques in the past, and do not incorporate link analysis (Gyöngyi, Garcia-Molina, & Pedersen 2004) techniques on the web graph. Since blog search engines do not yet make use of complex page importance techniques (they use only recency of post) while returning results, fake blogs (without complex link spam techniques) are currently the most common splog technique. This also justifies our experimentation with traditional email spam categorization techniques for splog detection.

Data Collection

We have been collecting blogs data for use in our larger blog specific project - MeMeta. In this section we describe a subset of the corpus (relevant to this work) we have collected so far and give some basic statistics. In the next section we describe how we label a subset of these for the creation of training examples.

Typically, there are four primary sources for collecting blogs: (i) processing streams of pings from blog ping servers containing URLs of new posts; (ii) crawling directories which allow robot indexing of their site content; (iii) using developer API's provided by popular blog search engines; and (iv) regular monitoring existing ping update services and blog hosting services ("changes.xml") for republished pings. The latter two options are easier ways of collecting blogs, evident from how some of the new search engines[5] claim to index blogs. We followed a similar approach and use techniques (iii) and (iv) to collect new blog URLs. We briefly describe how we used these mechanisms in turn.

The Technorati[6] blog search engine provides an API[7] to access its data on blogs. We used this API (query limit of 500) over a period of four months (May-August 2005) to randomly sample Technorati's blog index by submitting queries for the freshest posts containing words picked randomly from an English dictionary. While we expected to collect only blogs in English, it was surprising that our collection also contained many non-English blogs. We believe this is due to some commonly occurring words in multiple languages. Based on this technique, we collected the URLS of approximately 500,000 unique blog home pages. Since we queried for the freshest posts our subset also included many splogs which are known to be very "live". A statistic on the blogs we collected and the top 10 hosting services (based on URL pattern match) where they are published is listed in table 1.

	Popularity
blogspot	44%
msn	23%
livejournal	8%
aol	1%
splinder	1%
20six	1%
typepad	1%
blog	1%
fc2	1%
hatena	1%

Table 1: *Hosting statistics for blogs collected using the Technorati API.*

	Percentage
blogspot	39%
msn	23%
e-nfo	2%
travel-and-cruise	1%
lle	1%
find247-365	1%
typepad	<1%
blog	<1%
fc2	<1%
hatena	<1%

Table 2: *Hosting statistics for blogs collected from the weblogs.com ping server.*

Blog ping update services (and some blog search engines) for accept pings from updated blogs and pass them on to blog search engines or other interested parties. This information is often republished in the blogosphere as the most recently updated blogs as an XML document (e.g., "changes.xml"). We monitored these published files by fetching them at regular intervals from one of the of the popular ping mediators (http://weblogs.com/). Based on this technique we collected around five million unique blog home pages over a period of five months (April-August 2005). Table 2 gives a statistic on blog homepages we collected and the top ten domains where they (pings) come from.

Our motivation for this analysis was to confirm that the data collected through Technorati is indeed a good sampling of the blogosphere. Results generally matched in the relative order of blog hosting popularity but not in their exact position. However, it is evident from table 2 that update ping data is noisy. Hence we did not use ping data directly in our experiments, but rather indirectly during the creation of the negative training examples. We detail this and our labeling of samples in the next section.

[5]http://blogsearch.google.com/

[6]http://technorati.com/

[7]http://www.technorati.com/developers/

	percent
Legitimate	75%
Splog	25%
English	85%
Non-English	15%

Table 3: *Result of manual labeling of blogs sampled from Technorati.*

Labeled Corpus

Due to the noisy nature of data collected from ping update services we used the Technorati index as our primary blog dataset. We dropped the top 30 hosts (e.g. blogspot, msnspaces, etc.) from the dataset and uniformly sampled for around 3000 blog home pages. We then eliminated pages which were non-existent, whose size was less than 5KB or were written in Chinese, Korean or Japanese. We finally obtained 2600 blog homepages which we identified for use in the creation of our sample sets. Through a long manual process we tagged these blog homepages as one of legitimate or splog[8] and one of English or non-English. The distribution of these blogs is listed in table 3.

It is clear from the table that even a popular search engine like Technorati has a high percentage of splogs. In addition some of the pages identified as blogs by Technorati were actually forums (with syndication feeds). Also note that though our Technorati queries used English words, results still had a good number of non-English blogs. We then randomly sampled local links from these blogs which are non-blog home pages and post pages (blog subpages). We sampled to get a positive sample of around 2100 subpages. We did not manually verify these samples for correctness.

We then generated the negative subset for our samples. In training classifiers, generating negative samples has traditionally been both time consuming and erroneous. While a simple option is to sample a web search engine randomly, we employed a slightly more intricate approach. We extracted all external links from our positive dataset. This set consisted of links to other blogs (within the blogosphere) and the rest of the Web. The number of links we extracted was in the order of half a million. A completely manual process to identify non-blog pages among these links would have been next to impossible.

Hence, from these extracted outlinks we eliminated those URLs that we knew were blogs. We compared host names of extracted URLs against the host names of URLs we had collected through Technorati and update ping services. This was extremely useful since we had collected around five million unique blog homepages and consequently a high number of unique blog hosts (both blog hosting services and self hosted). For example, since we now know that pages with the domain "blogspot", "instapundit" etc. are blogs, we eliminated all such URLs from the negative set.

After the process of elimination of URLs in the negative set we were left with around 10,000 URLs out of which we uniformly sampled a smaller number. We then manually validated our process to generate a total of around 2600 negative samples. Our data creation process also made sure that negative samples had a sizeable number of non-English pages commensurate with the numbers in positive samples.

Towards generating positive samples for splog detection we picked those blogs that we had manually identified as splogs. This does imply that our splog dataset does not include blogs from the top hosting services, the ones which were eliminated them from our base dataset. We made this choice since it eased the process of manual labeling. Further investigation suggested that the nature of splogs were similar in top blog hosting services, and hence our choice should not significantly affect classification accuracy. Manual identification resulted in a positive sample of around 700 blogs (splogs). We randomly sampled for 700 authentic blogs from the remaining 1900 blog home pages to generative negative samples.

With all of the above techniques we created three data sets of samples:

1. **BHOME:** (blog home page, negative) consisting of (2600 +, 2600 -) samples for a total of 5200 labeled samples.
2. **BSUB:** (blog all pages, negative) of (2100 +, 2100 -) samples.
3. **SPLOG:** (blog home-page spam, blog home-page authentic) of (700 +, 700 -) samples.

Evaluation Metrics

We use the well-known metrics of evaluation – precision, recall and F1 measure – to compare the relative performance of using different features. Precision is a measure of the usefulness of retrieved documents and recall is a measure of the completeness of the retrieval process.

$$Precision\ P = \frac{Relevant\ Documents\ Retrieved}{Retrieved\ Documents}$$

$$Recall\ R = \frac{Relevant\ Documents\ Retrieved}{Relevant\ Documents}$$

The F1 measure is the harmonic mean between precision and recall and is defined as

$$F1 = \frac{2 * P * R}{P + R}$$

F1 measure provides a single metric for comparison across different experiments. All our results are based on Leave-One-Out Cross-Validation.

Baselines

Some might further argue that even for the subset that we choose (mainly consisting of self-hosted blogs) simple heuristics would suffice for blog identification. In this section we make a case for using SVMs for blog identification by comparing with simple heuristics. These features include the HTML meta tag with name "generator", the HTML link tag for RSS or Atom feed or the occurrence of one of the

[8]We made sure we incorporate some of the common splog techniques listed by a blog search engine - http://blog.blogpulse.com/archives/000424.html

	P	R	F1
META	1	0.75	0.85
RSS/Atom	0.96	0.90	**0.93**
blog	0.88	0.79	0.83
comment	0.83	0.87	0.85
trackback	0.99	0.18	0.30
2005	0.56	0.97	0.71

Table 4: *Human Blog Identification Baselines measured using simple heuristics.*

substrings "blog, 2005, comment, weblog" on a web page. The accuracy by using these heuristics individually are summarized in table 4. To arrive at these results we used the **BHOME** sample set.
These numbers suggest that the existence of HTML link tag about RSS/Atom feed on a page is a good heuristic for blog identification. However two points should be noted here. First, the wide adoption of RSS and Atom feeds was initiated on the blogosphere, and is now seeing adoption elsewhere. This implies that the precision rates observed here will see a significant drop as feed links are published elsewhere on the Web. Second, we believe that our data collection process is slightly biased to collecting blogs which have RSS or Atom feeds. In fact, many blogs (more than what our recall suggests here) do not publish these syndication feeds. A recent statistic[9] by an influential blog search engine suggests that RSS and Atom enablement, atleast among the influential bloggers, is not all that high. Even with these factors we believe that results from blog identification using SVMs (without using HTML link element) should atleast match the one seen from RSS/Atom heuristic if not better them.

Splog detection is a problem that has been tackled in many other domains in the past and is becoming a serious problem for the blogosphere. We have not yet come across any splog detection technique which employs text categorization. One technique based on simple heuristics, and available for public use (http://antisplog.net/) gave a precision of 66% and a recall of 60% on the **SPLOG** dataset. This is only slightly better than a random baseline (Precision 50%, Recall 50%).

Results

For all of our experiments we used the SVM toolkit (Joachims 1998) using linear kernels with the margin parameter "c" set to the default value. The feature types we experimented with and the number of top features (selected using mutual information) used for different sets are listed in table 5. Each of these features was used in a bag-of-{tokens} where "token" stands for different feature types. Feature values where either of type binary or TF normalized. Most of the feature names listed in the table is self-explanatory. "meta" was based on the HTML meta tag (name="generator"), page title and page URL. The "link" feature type used contents of HTML link element on a page (with rel='alternate'). The last feature type "4grams" used

[9]http://www.sifry.com/alerts/archives/000310.html

Feature	**#BHOME**	**#BSUB**	**#SPLOG**
words	19000	19000	19000
urls	10000	10000	7000
anchors	10000	10000	8000
meta	2000	2000	3500
words+urls	29000	29000	26000
meta+link	2500	2500	4000
urls+anchors	20000	20000	15000
urls+anchors+meta	22000	22000	18500
4grams	25000	25000	25000

Table 5: *Feature types and their feature vector sizes for the three datasets used in experiments. Features were ranked using Mutual Information.*

Feature	P	R	F1
words	.945	.943	.944
urls	.971	.934	.952
anchors	.971	.901	.934
meta	.985	.767	.862
words+urls	.970	.969	**.969**
meta+link	.973	.940	.956
urls+anchors	.980	.955	.967
urls+anchors+meta	.984	.962	**.973**
4grams	.970	.943	.956

Table 6: *Results for the BHOME dataset using TF Normalized features. Blog Identification was most effective using features consisting of a combination of urls, anchors and meta tags.*

4gram sequences on a combination of urls, anchors and meta information on a single page. Note also that we did not use "link" in all but one of the features to show that classifiers are accurate even without this information.

In the first set of experiments we used feature count (number of times the feature appears in all documents) as the feature selection technique for **BHOME** with TF normalized feature values. The results were either similar or slightly less accurate than those we obtained using Mutual Information, which we report in table 6. Consequently, we report on only experiments using Mutual Information based feature selection. In the final set of experiments for **BHOME** we used binary feature values as opposed to the TF normalized. Results shown in table 7 were the best for blog identification providing close to 98% accuracy. Based on the success of the choice of features for identifying blog home pages, we next ran the same experiments on **BSUB** using both TF-normalized (results in table 8) and binary (results in table 9) features. It was clear from these results that the same features which fared well for the blog home page identification problem performed well in identifying all blog pages.
Our last set of experiments were to check how well the features we used in blog identification did for splog detection using the SPLOG dataset. Results for this set of experiments for TF-normalized features are listed in table 10 and binary

Feature	P	R	F1
words	.976	.941	.958
urls	.982	.962	.972
anchors	.975	.926	.950
meta	.981	.774	.865
words+urls	.985	.966	**.975**
meta+link	.973	.939	.956
urls+anchors	.985	.961	.973
urls+anchors+meta	.986	.964	**.975**
4grams	.982	.964	.973

Table 7: *Results for the BHOME dataset using Binary Features. Like TF-normalized features, blog identification was most effective using a combination of urls, anchors and meta tags.*

Feature	P	R	F1
words	.944	.903	.923
urls	.953	.896	.924
anchors	.966	.878	.920
meta	.977	.919	.947
words+urls	.966	.935	**.950**
meta+link	.971	.939	.955
urls+anchors	.973	.916	.944
urls+anchors+meta	.986	.952	**.969**
4grams	.969	.905	.936

Table 8: *Results from blog sub page identification using the BSUB dataset for TF Normalized feature values.*

Feature	P	R	F1
words	.976	.930	.952
urls	.966	.904	.934
anchors	.962	.897	.923
meta	.981	.919	.945
words+urls	.979	.932	**.955**
meta+link	.919	.942	.930
urls+anchors	.977	.919	.947
urls+anchors+meta	.989	.940	**.964**
4grams	.976	.930	.952

Table 9: *Results from blog sub page identification using the BSUB dataset for TF Normalized feature values.*

Feature	P	R	F1
words	.888	.870	**.879**
urls	.753	.729	.741
anchors	.818	.835	.826
meta	.738	.765	.751
words+urls	.894	.861	**.877**
meta+link	.723	.765	.743
urls+anchors	.849	.827	.838
urls+anchors+meta	.857	.823	.840
4grams	.877	.773	.822

Table 10: *Results from splog identification using the SPLOG dataset for TF Normalized feature values.*

Feature	P	R	F1
words	.887	.864	**.875**
urls	.804	.827	.815
anchors	.854	.807	.830
meta	.741	.747	.744
words+urls	.893	.869	**.881**
meta+link	.736	.755	.745
urls+anchors	.858	.833	.845
urls+anchors+meta	.866	.841	.853
4grams	.867	.844	.855

Table 11: *Results from splog identification using the SPLOG dataset for binary feature values.*

features are listed in table 11. From these results, it is quite clear that splog detection is a challenging task in the blogosphere. The best accuracy was around 88% for binary features which is encouraging. This is still significantly higher than existing baselines. In the next section we analyze results on blog identification and identify future work for the splog detection problem.

Discussion

Our results show that SVMs perform fairly well on blog identification as expected. In addition certain new feature types (e.g. URLs, anchors, etc.) give comparable or better results and lead to smaller feature space and hence complexity. Against our expectations, N-gram based measure did not significantly improve results on our multi-lingual dataset.

Though our results are fairly good, our classifier still misclassified some samples. We analyzed some of these misclassified samples and report on them with possible explanations. Positive samples which were not identified included blogs in languages other than English (e.g. http://www.starfrosch.ch/starfrosch/), those not allow direct commenting on posts (only using permalink), ones that appeared to be only fuzzily classifiable (e.g., http://www.standeatherage.com/) and blogs with no posts. Some of these misclassified blogs were hosted using popular blogging software with significant changes to site design and template (e.g. http://digitalextremist.com/). Precision values of the learned classifiers were quite high (close to

99%). For experiments with **BSUB** dataset we observed a slightly lesser precision. Closer examination revealed that the misclassified pages included non-post pages (such as author bios), post pages without any content, and RSS or Atom feed pages. Some others were forum pages or page-redirects to other non-blog pages. We attribute the relative lower precision to the fact that we did not manually verify the **BSUB** dataset.

Overall, accuracy of the learned classifiers for blog identification was quite good. In addition, the use of non-typical features is effective for blog identification. We obtained slightly higher accuracy with a much lesser number of features. Our results confirm that certain new features can be effective for blog identification. Further investigation on analyzing the scale of feature size reduction is an interesting direction for future study. Though initial results are encouraging, SVMs with our current features did not perform as well as we hoped for the splog detection problem. In view of this we are pursuing many future directions.

Our current work is on modeling the blogosphere so as to highlight its relational characteristics. Towards this end, we are working on capturing explicit and implicit metadata in blogs which will enable the creation of a semantic graph for the Blogosphere. In addition, we believe classifiers should be trained separately on different classes of splogs based on a splog taxonomy. We are currently working towards developing such a splog taxonomy. We believe that the application of the above two techniques will improve the effectiveness of splog detection.

Conclusion

In this paper we have described our work on blog identification and preliminary work on splog detection in the context of our larger MeMeta project. We experimented and employed novel features for the blog identification problem. These new features provide better accuracy with a lesser number of features. Splog detection is a challenging problem that is well beyond the approaches successful for traditional email spam detection. Through our continuing work, we are identifying better techniques for splog detection, making use of specialized features, metadata, splog taxonomy and the Web Graph.

Acknowledgments

We thank Jim Mayfield at JHU APL for suggesting the use of Technorati to collect blogs and Tim Oates at UMBC for his comments on SVMs and margin parameters. We thank Technorati for making available their developer API and its open index. Partial support was provided through a generous fellowship from IBM and by NSF awards NSF-ITR-IIS-0326460 and NSF-ITR-IDM-0219649.

References

Boser, B. E.; Guyon, I. M.; and Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. In *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, 144–152. New York, NY, USA: ACM Press.

Darnashek, M. 1995. Gauging similarity with n-grams: language independent categorization of text. *Science* 267:838–848.

Drucker, H.; Wu, D.; and Vapnik, V. 1999. Support vector machines for Spam categorization. *IEEE-NN* 10(5):1048–1054.

Fano, R. M. 1961. Transmission of information.

Gyöngyi, Z., and Garcia-Molina, H. 2005. Web spam taxonomy. In *First International Workshop on Adversarial Information Retrieval on the Web*.

Gyöngyi, Z.; Garcia-Molina, H.; and Pedersen, J. 2004. Combating web spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Databases*, 576–587. Morgan Kaufmann.

Herring, S. C.; Scheidt, L. A.; Bonus, S.; and Wright, E. 2004. Bridging the gap: A genre analysis of weblogs. In *HICSS '04: Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 4*, 40101.2. Washington, DC, USA: IEEE Computer Society.

Joachims, T. 1998. Text categorization with suport vector machines: Learning with many relevant features. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, 137–142. London, UK: Springer-Verlag.

Kolari, P.; Finin, T.; and Joshi, A. The memeta project. urlhttp://ebiquity.umbc.edu/project/html/id/68/.

Krishnamurthy, S. 2002. The multidimensionality of blog conversations: The virtual enactment of sept. 11. In *Internet Research 3.0, Maastricht, Netherlands*.

Pang, B.; Lee, L.; and Vaithyanathan, S. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yang, Y., and Pedersen, J. O. 1997. A comparative study on feature selection in text categorization. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, 412–420. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Zhang, L.; Zhu, J.; and Yao, T. 2004. An evaluation of statistical spam filtering techniques. *ACM Transactions on Asian Language Information Processing (TALIP)* 3(4):243–269.