Tracking Influence and Opinions in Social Media

Akshay Java

November 6, 2006

Abstract

These days Social Media such as forums, wikis and blogs in particular are playing a notable role in influencing the buying patterns of consumers. Often a buyer looks for opinions, user experiences and reviews on such sources before purchasing a product. Detecting influential nodes and opinion leaders and understanding their role in how people perceive and adopt a product or service provides a powerful tool for marketing, advertising and business intelligence. This requires new algorithms that build on social network analysis, community detection and opinion extraction.

Studies on influence in social networks and collaboration graphs have considered a static view of the network and are based purely on link analysis. These techniques have been found to be effective in performing analysis at an aggregate level and to identify key individuals who play an important role in propagating information. However, influence on the Web is often a function of topic. A blog like Daily Kos that is influential in politics is less likely to have an impact on the technology related blogs. Similarly, Techcrunch, an extremely popular technology blog might not be influential when it comes to politics. We propose the notion of 'topical influence' and extend existing techniques to make them topic sensitive.

An important component in understanding influence is to detect sentiment and opinions. Changes in opinions, aggregated over many users, can be a predictor for an interesting trend in a community. Sufficient adoption of this trend could lead to a 'tipping point' and consequently influencing the rest of the community. Strong opinions over time indicate biases. It is important to model bias to target the right audience for marketing and advertising. A community of ipod fanatics for example, needs little or no convincing about the product. Influencing an opinion leader in such already positively biased communities is going to have less significant impact for the product. Here we also describe BlogVox, a blog analytics system that we developed for TREC opinion retrieval task. This system finds opinionated blog posts about a topic. We propose to extend this system to track aggregated opinions over many users.

Finally, we propose to model influence as a temporal phenomenon. The Blogosphere, being a buzzy and dynamic environment, has new topics emerging constantly and blogs rising and falling in popularity. Tracking these changes over time allows us to find blogs that are influential versus something that is just briefly popular. For example, many thousands of sites linking to a 'Coke Mentos' video in one day indicates popularity. But thousands of links accumulated consistently over time by a blog indicate that it is influential. Additionally, by studying how a meme spreads, we can identify key individuals that are either buzz generators or early adopters who create sufficient interest on the topic.

In this thesis we develop an improved influence model that includes components for community structure, topic categorization, representation of key beliefs and opinions, and a temporal analysis of how these change. We show that his model outperforms existing, simpler models by evaluating its precision and recall for appropriate tasks.

Contents

Chapter 1

Introduction

Social Media is a dynamic and growing area that includes a collection of blogs, wikis, forums, photos and videos sharing sites. According to wikipedia [1]

> "social media describes the online tools and platforms that people use to share opinions, insights, experiences, and perspectives with each other"

Opinions and views expressed in such medium are often instrumental in a customers buying decisions. In this work, we describe automated techniques to model, measure and identify influential individuals and their opinions.

Social Media sites are sometimes also referred to as Web 2.0. Web 2.0 was a term coined by O'Rielly Media [55] to describe the emerging genre of next-generation internet sites. A typical characteristic of these Social Media sites is the high level of user participation. Using comments, trackbacks, folksonomies and underlying social networks, these sites allow users to actively participate in conversations and easily share information with others.

A primary component in the Social Media landscape are blogs. Weblogs or blogs started initially as online diaries or journals. Today, blogs are used for a wide variety of purposes. Typically it is a means of content publishing and a place where people share their views, opinions or experiences. There has been a phenomenal growth of the Blogosphere. Presently there are close to 52 million blogs, they are doubling approximately every six months [61]. Blogs have become a means by which new ideas and information spreads rapidly on the web. They often discuss the latest trends and echo with reactions on different events in the world. The collective wisdom present on the blogosphere is invaluable for market researchers [17, 58] and companies launching new products.

In fact, advertisers are already realizing the potential of blogs in influencing buying decisions of their target audience. Often when a buyer is interested in purchasing a product, blogs offer free and frank customer reviews. As Robert Scooble puts it "blogging is one huge word-of-mouth engine" [60]. Forster Research [13] projects a sustained growth in advertising and marketing spending from around $14.7 billion in

[1] `http://en.wikipedia.org/wiki/Social_media`

2005, to about $26 billion by 2010 (in US alone). According to this survey, 64% of the advertisers are interested in advertising on blogs.

Many startups have pitched pre-launch and beta version of their products to various bloggers with the hope that their reviews would create a buzz on the blogosphere and bring more attention to their company. Microsoft recently launched the Origami tablet PC amongst speculations that the hype was generated by a targeted campaign to a selected set of bloggers. Companies like Wal-mart are now trying to win back public opinion on their corporate policies by providing bloggers with exclusive news and inviting them for visits [7]. Even the US Department of Defense is now reaching out to hundreds of bloggers with more information about its counter-terrorism initiatives [25]. Recently, it has also become a common business practice to maintain a corporate blog and a number of the Fortune 500 companies claim to have a corporate blog [5]. This trend is changing how companies engage with their clients.

The Blogosphere provides a unique resource for studying information flow in online Social Media. At the same time it is also possible to monitor the trends and opinions that are being expressed online. The dynamic nature of the Blogosphere, the abundant user-generated content and complex social structure pose interesting questions. The main focus of this thesis is to study and characterize the nature of influence and opinion formation in online social media.

Consider that your goal was to market Microsoft's new MP3 player Zune, the new rival to Apple's hugely popular Ipod. One of the first things to do would be to monitor the space of consumer electronics. This can be done at a meta-level by using some of the trend analysis available trend analysis tools. There are many online services like Blogpulse[2] and Google Trends[3] that allow users to track 'buzz' and generate alerts for different keywords. This gives an aggregate picture of the popularity for the term based on the volume of online chatter. Recently there has been an interest in mining such trends to predict the success of movies [16] and the sales of books [50, 19]. However, using trend analysis alone, gives a partial view of the market and one can not make a judgment about the sentiment of the customers. A large volume in online chatter about the latest blockbuster need not be an accurate predictor of sales. While there might be many people talking about a particular movie, the overall sentiment could be negative. This underscores the importance of opinion extraction in such situations. We describe BlogVox [27], a system for monitoring the Blogosphere to find opinionated posts.

Next, we would like to identify online communities of users that talk about the different portable music devices. On the Web, a community is usually defined as a set of pages that have a greater probability of linking within the set versus outside it. Detection of such communities allows the advertiser to market a product to the *right set of audience*. In comparison advertising programs such as Google Adsense work by matching documents with keyword specific advertisements. While such systems cast a wider net, it is also useful to have a targeted marketing strategy. To this effect, we describe two ways to model communities on the Blogosphere: readership based and network based. Using the readership information from a popular blog reader, we extract groups of blogs that are in the same community. In addition we use standard

[2] http://www.blogpulse.com

[3] http://www.google.com/trends

network analysis tools such as distance-based clustering and graph partitioning to extract communities of blogs. The extracted communities are used to perform further analysis to extract specific individuals who are *influential*.

While there is little consensus amongst researchers on how influence can be defined, we consider influence as:

> *The ability of a blogger to* persuade *another blogger or a reader to*
>
> - *take an action by means of writing a new post about the topic.*
> - *quote the blogers comments in his own post.*
> - *link to the originating blog by means of comments or trackbacks.*
> - *link to the blog post/blogger through other means such as delicious links, blogrolls, etc.*

Identifying the influence network and knowing who the key players are and who is already on board can be a powerful tool for marketing a product.

In this work we describe techniques to model influence based on direct measurable properties of a blog. We describe ways to automatically identify opinions and model bias of the blogger with respect to a product or a topic and measure its effect by identifying the community it influences.

The proposal is organizes as follows: Chapter 2 formalizes the research objectives and the proposed thesis. We also specify the key contributions that would be achieved and the scope of this work. Chapter 3 describes some of our preliminary study into the structure of the Blogosphere. Understanding the structure of the Blogosphere is essential to be able to leverage existing techniques from the research in Web graphs and also helps understand the differences that need to be addressed. In Chapter 4 we describe some of the existing approaches to modeling influence and provide a novel technique that uses readership information rather than graph approach. Next, Chapter 5 describes an opinion retrieval system that was used as part of the TREC conference. This system is being developed to identify opinions and biases of the bloggers. Finally, in Chapter 6 we describe some of the ideas for the proposed approach and evaluation criteria. Chapter 7 describes the proposed timeline and the possible milestones for this work. Finally, we conclude by providing a discussion of the broader implication of this work.

Chapter 2

Research Objectives

2.1 Thesis Statement

An accurate model of influence on the blogosphere must analyze and combine many contributing factors, including topic, social structure, opinions, biases and time. We will develop, implement and experimentally evaluate such a model to demonstrate its improved accuracy over models based on any one of these factors.

2.2 Contributions

The following are the main contributions of this work:

- First we aim to characterize influence and study what makes a blog or an individual influential. In effect, by understanding the constituents and dynamics of influence we aim to effectively *measure* it.
- We provide a new approach to model influence in real-world, large-scale scenarios using blog data as our testbed. Such a model would incorporate topics and opinions which would provide a better representation of influence.
- We develop novel techniques to model communities using readership information and compare it with link-based community extraction.
- We develop a system to detect and extract opinions from blog posts. We describe a novel technique to extend influence model to incorporate *link polarity*. By monitoring opinions over a period of time we can accurately map it to biases and beliefs of a blogger.
- Our hypothesis is that to model influence we need to combine many contributing factors like topic, social structure, opinions, biases and time. As part of this research we will aim to test this hypothesis and experimentally validate the effect of each of these.

2.3 Scope

The scope of this work overlaps with some areas of social sciences, where researchers worked towards understanding how human society functions. Some researchers in social science [40] as well as computer science have developed theories of individual behavior in real world organizations and that of agents in artificial societies and multi-agent scenarios[15]. While we draw and expand on the body of literature available in these fields, we limit the scope to understanding influence in online media.

With advances in opinion mining and social network analysis, in recent times there have been concerns about the negative consequences of *influencing the influencers*. While we understand that these are valid concerns, we feel that there are positive benefits of such applications can be helpful in understanding the markets, or sentiments of people in other countries a government policy. We limit our scope to the study of opinion extraction and influence propagation.

Chapter 3

Structure of the Blogosphere

This section describes the basic infrastructure of the Blogosphere and discusses certain key differences of blog content from Web documents.

3.1 Blogosphere Infrastructure

In terms of size, though it constitutes only for a portion of the whole Web, the Blogosphere is already quite significant and is getting increasingly bigger. Presently there are close to 52 million blogs and are rapidly doubling every six months and a large fraction of these blogs are active. Blogs are typically published through blog hosting sites or tools like wordpress that can be self-hosted. An entry made by a blogger appears in a reverse chronological order. Whenever a new post is published, a ping server is notified of the fresh content. Infrastructurally, this is one of the critical difference from the Web. While on the Web, search engines rely on crawlers to fetch and update the index with new content, the stream of pings provides information that new content has been published on a blog. This is done essentially to ensure that downstream services (like search engines and meme trackers) can quickly find new content, thus ensuring the freshness of their index.

The blog home page can contain various anchortext links that provide personal information, links to recent posts, photos, blogrolls (links to blogs frequently read), delicious bookmarks, FOAF descriptions etc. Each blog post contains a title, date, time and the content of the post. Additionally, posts can also be assigned tags or categories that provide information about the topic or keywords that are relevant to the post. This form of meta-data information or *folksonomies* have been popularized by sites like delicious[1] and flickr[2] and provide a way for users to conveniently organize content. Finally the blog itself can be subscribed via RSS (Really Simple Syndication) feeds. Through this simple XML formatted file, users can subscribe to blogs, news sites and also personalized content such as alerts and search results.

[1] http://del.icio.us
[2] http://www.flickr.com

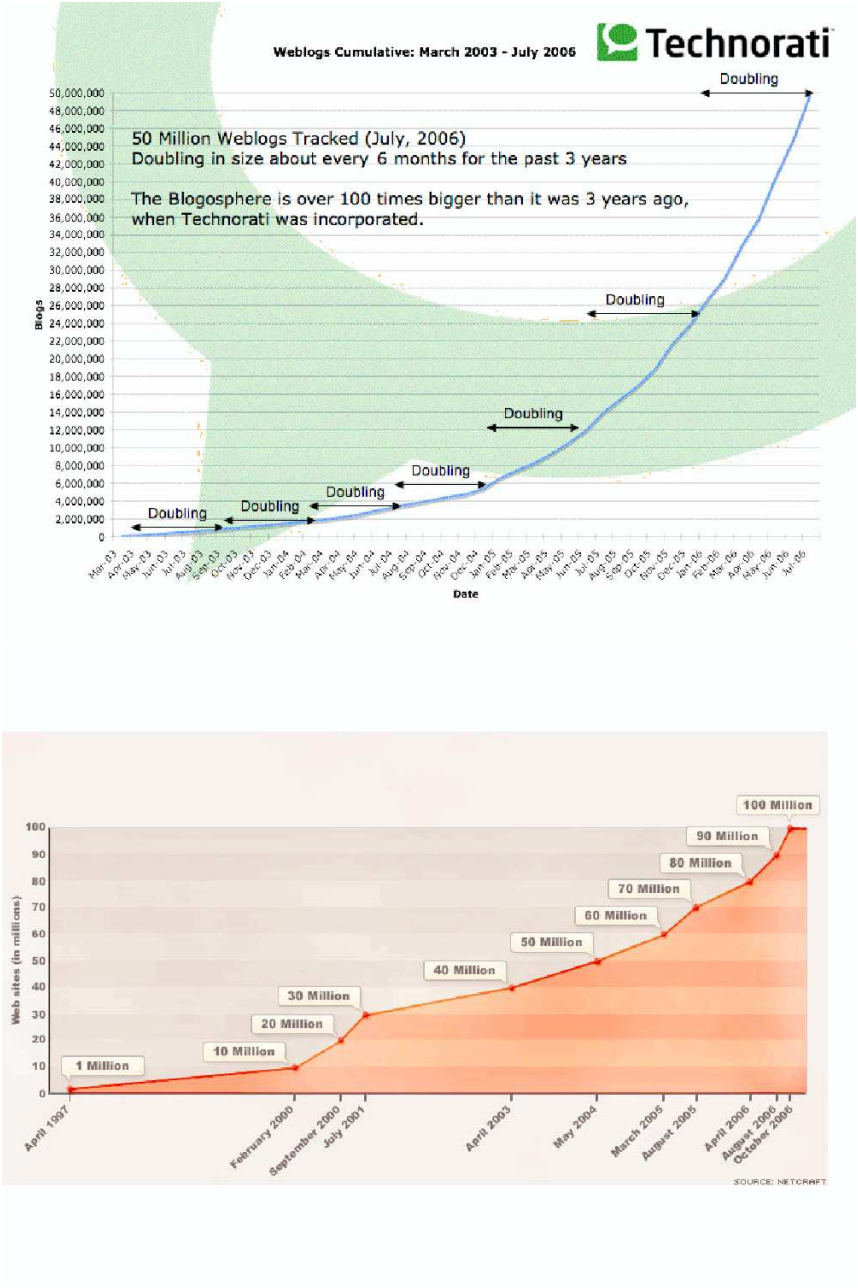


Figure 3.1: *The blogosphere continues to double every six months. This increase has also contributed to the growth of the Web in general (sources: Technorati, Netcraft, CNN)*

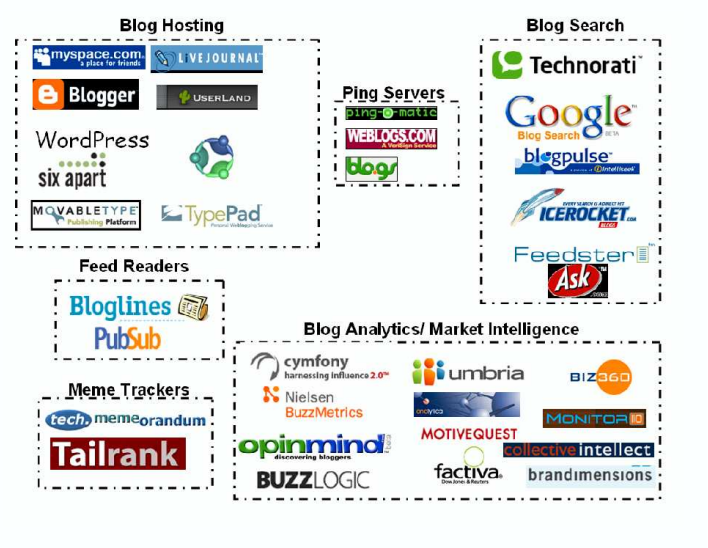
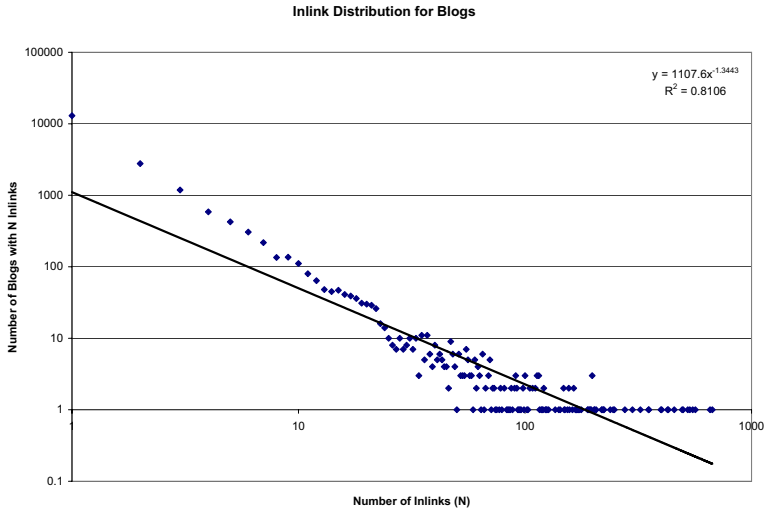


Figure 3.2: *A number of companies are now providing blog analytics tools and 'brand monitoring' tools.*

The current trends are only indicators of sustained growth in user-generated content. As this space becomes larger, there is an increasing need to develop tools and techniques that can help keep tabs on the 'pulse of the Blogosphere'. Figure 3.2 summarizes some of the commercial vendors in this space.

3.2 Blogosphere as a Graph

A number of researchers have studied the graph structure of the Web. According to the classic 'Bow Tie' model [9] the WWW exhibits a small world phenomenon with a relatively large portion of links constituting the core or Strongly Connected Component (SCC) of the graph. Ravi Kumar et. al. [38] have studied the evolution of the blog graph and find that the size of the Blogosphere grew drastically in 2001. They find that at a microscopic level there was also emergence of stronger community structure. Using a graph represented by the link structure of the blog post to blog post links from a collection of about 3 Million blogs we find power law distributions for both the inlink and outlinks in the graph (see Figure 3.3). Similar results were also discussed in [20] while studying how topics propagate on the Blogosphere. Herring et al. [23] performed an empirical study the interconnectivity of a sample of blogs and found conversations more concentrated to a subset of bloggers known as a-listers.



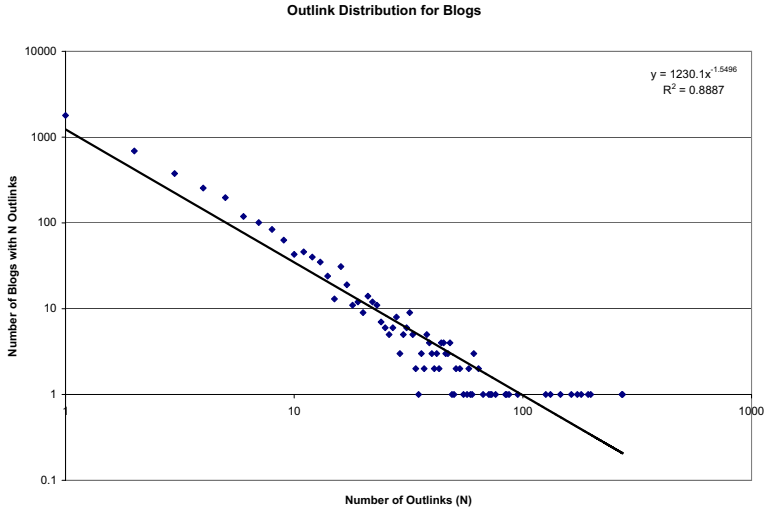


Figure 3.3: *Degree distributions of inlinks and outlinks in blogs indicate similar pattern as Web data. In particular existence of power law distributions indicate a typical phenomenon of a few sites getting most of the attention.*

3.3 Link Analysis on Blogs

3.3.1 Ranking Algorithms

While there are a number of ranking techniques for the Web, many blog search engines still use inlink counts for a post or a blog as a measure of 'authority'. Such metrics are at best a measure of popularity, nevertheless they remain in practice today. The PageRank algorithm works by simulating a random walk by a user who follows a link with probability *q* and jumps to a random section of the web graph with probability of (*1-q*). If $C(a)$ is a set of outlinks for page *a* and $P_i..P_n$ are the pages pointing to a. Then

$$PR(a) = q + (1 - q) \sum_{i=1}^{n} PR(p_i)/C(p_i)$$

One advantage of the pagerank algorithm is that it is relatively less expensive to compute and has fast convergence.

Another ranking scheme used is HITS [31] which assigns a hub and authority score. A good hub is one that points to a number of authoritative sources, while a good authority is one that is pointed to by many hubs. $H(p)$ = hub value of the page p and $A(p)$= authority value of a page p.

$$Authority(v) = \sum_{v \in S, v \to p} Hub(p)$$

And

$$Hub(p) = \sum_{v \in S, p \to v} Authority(v)$$

Similarly, for the Blogosphere, Adar et al. [3] have proposed a variation of PageRank, called iRank, described to rank blogs based on their *informativeness*. In this scheme, each directed edge is assigned a weight $W_{ij} = w(\Delta d_{ij})$ where Δd refers to the time difference between the blogs citing a URL and $w(\Delta)$ is the weight function that gives importance to URL citations which are closer in time. The edge weights are then normalized and PageRank computation follows. This weighted graph is called the *implicit information flow graph*. iRank makes use of the temporal nature of blogs by differentially weighing each citation in the graph by the time difference between when the blog mentions a URL and how soon it is referenced by other blogs.

3.3.2 Community Detection

Social structure in any society emerges from our desire to connect with others around us who share similar views and interest. Communities emerge in many types of networks. Starting with Milgram's experiments [48] that led to the popular anecdote on the 'six degrees of separation', the study of the underlying structure and properties has interested researchers for many years. Many real world networks like collaboration/coauthor [51], biological networks [64] and internet exhibit the small-world phenomenon.

Flake et. al. [1] describe a network flow based approach to partitioning the graph into communities. Recently, there has been renewed interest in community detection for blog data. Lin et. al. [43] identify a group of blogs that are mutually aware of each other. Post-to-post links, comments, trackbacks, all constitute to different types of actions that indicate awareness. Using an algorithm similar to PageRank each pair of blogs is weighted with an association score based on the different actions between the corresponding blogs. However, this technique requires a seed set of blogs to extract the community. Additionally, they provide a clustering algorithm to visualize such communities [62].

Some community detection techniques require computation of "betweenness centrality" which is an expensive calculation over very large graphs [53]. Betweenness centrality is a measure of the number of times a node is on the shortest path route amongst all other pairs of nodes. Newman provides a fast approximation [52] to this measure. Using their algorithm we have extracted some of the communities from the WWW 2006 dataset. Figure 3.4 shows the distribution of community size in this set. It can be observed that there are a number of small communities. Smaller communities are those that are extremely narrow in their topic or are a group of blogs of friends linked to each other. Most of the popular blogs tend to be in the large community of forming the core of the Blogosphere. By analyzing one such community we find that there are a large number of interconnected political blogs. Figure 3.5 shows a visualization of this graph. The size of the node is proportional to the degree of the blog.

3.4 Content Analysis on Blogs

Blogs, tend to be informally written, poorly structured, prone to spelling and grammatical errors, and feature non-traditional content. In addition, performing linguistic analysis on blogs is plagued by two new problems: (i) the presence of spam blogs and spam comments, and (ii) extraneous non-content including blog-rolls, link-rolls, and advertisements. In Chapter 5 we describe useful techniques designed to eliminate noisy blog data by sharing our experiences with BlogVox - a blog analytics engine we developed for TREC. Our findings underscores the importance of removing spurious content from blog collections.

3.4.1 Linguistic Analysis

Traditional natural language text processing systems are usually applied to tasks with high quality text. In practical environments, including online chat, SMS message, email messages, wiki pages and blog posts NLP systems are less effective. Blog posts are noisy, ungrammatical and poorly structured text and which make it difficult to process even using information extraction tools and shallow parsers. The main reason for this is the dependence of such tools on capitalizations and proper sentence boundary detection. In addition the extensive use of slangs and neologisms in blogs makes it difficult for NLP tools to extract useful information. In some of the related work, RSS feeds from news sources have been somewhat easier to process using OntoSem, a

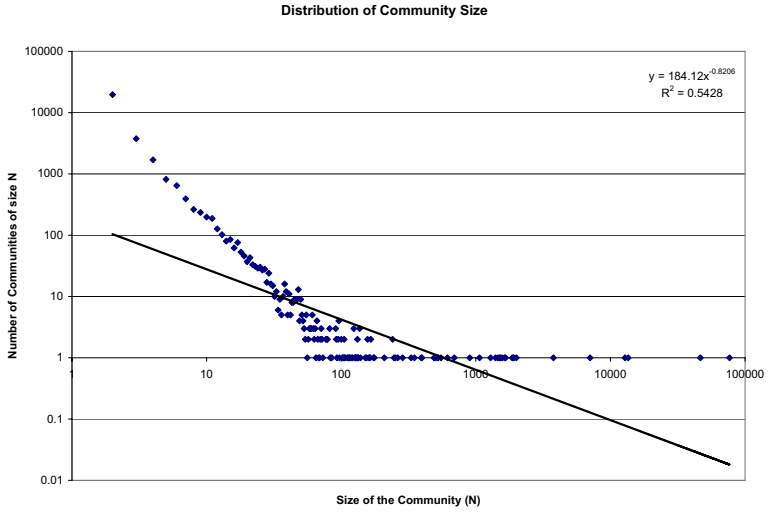


Figure 3.4: *The graph shows the distribution of the size of the communites extracted from a Blog graph. There are a few very large communities, while most of the graph contains smaller communities typical of a power law distribution.*

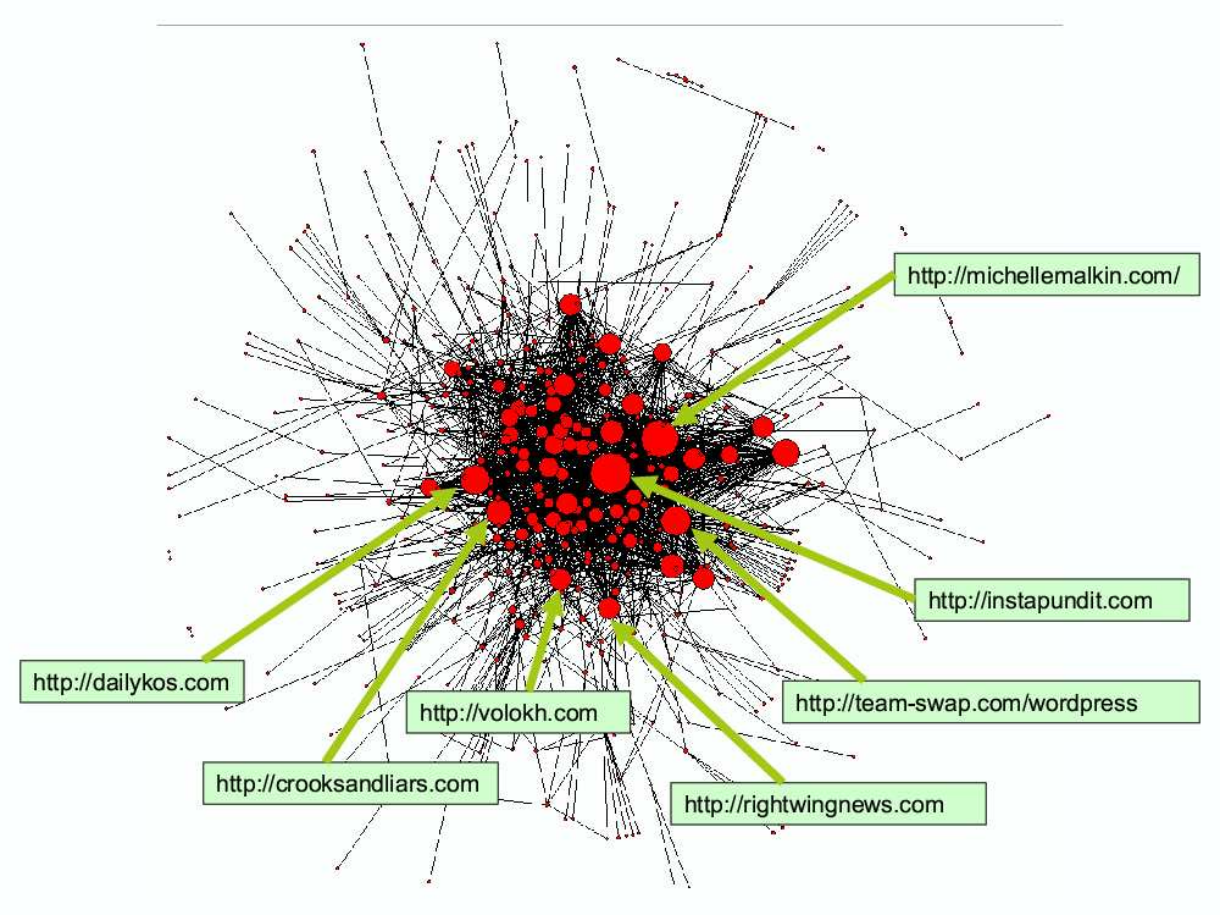


Figure 3.5: *A view of a sub-community containing a number of political blogs consisting about 13K vertices. The size of the node is proportionate to its degree.*

sophisticated ontological-semantics-based NLP system [26]. However, our experience is that large scale linguistic processing of blog data is more difficult.

3.4.2 Splog Removal

Recently, Spam blogs, or splogs have received significant attention, and techniques are being developed to detect them. [34] have recently discussed the use of machine learning techniques to identify blog pages (as opposed to other online resources) and to categorize them as authentic blogs or spam blogs (splogs). [36] extends this study by analyzing a special collection of blog posts released for the Third Annual Workshop on the Weblogging Ecosystem held at the 2006 World Wide Web Conference. Their findings on spam blogs confirms the seriousness of the problem. In our work we use some of their proposed techniques for data cleaning and minimizing the effect of splogs in our algorithms.

3.4.3 Post Content Analysis

The very nature of blogging platforms poses an important challenge. The blog hosting tools provide a number of structural and formatting elements that help provide information and share content. Blog owners promote friends, products, services and often their own posts by featuring them on blog-rolls and link-rolls that are often replicated across the entire blog. Such extraneous content makes it difficult for text analysis tools to analyze blogs. Spam blogs, spam comments and extraneous content indexed by a blog processing system put an unnecessary strain on the computational infrastructure, and ultimately skew results of blog analysis. To some extent this problem is alleviated by the use of RSS syndication. However, not all of the blogs use full syndication. Hence some of the search engines and other tools need to still parse or scrape content from homepages and permalinks. We describe novel techniques that use simple heuristics and and machine learning to filter extraneous links to identify the actual content of the post.

Chapter 4

Influence Models

4.1 Introduction

Research in the area of information propagation was inspired by a large body of work in disease and epidemic propagation. As described in [20] this model applies well in the Blogosphere where a blogger may have a certain level of interest in a topic and is thus *susceptible* to talking about it. By discussing the topic he/she may *infect* others and over time might *recover*. The authors use this approach in characterizing individuals into various phases of a topic in which they are more likely to become *infected*. They model individual propagation and use an expectation maximization algorithm to predict the likelihood of a blogger linking to another blogger. They also study the different types of topics present in the dataset and describe an approach to categorize topics into subtopics. Certain topics are more *infectious* than others and spread through the social network of bloggers. Automatically predicting such topics and developing models to accurately identify the propagation patterns on the Blogosphere is the main focus of this work.

Since bloggers are constantly keeping abreast of the latest news and often talk about new trends before they peak, recent research has focused on extracting opinions and identifying buzz from blogs [17]. Gruhl et al. [19] have found strong correlation between spikes in blog mentions to Amazon sales ranks of certain books. More recently, Lloyd et al. [45] found similar trends for named entities in blog mentions and RSS news feeds.

Blogs are often topical in nature and their link structures constantly evolve as new topics emerge. Ravi et al. [38] study the word burst models [32] and community structure on the Blogosphere [39]. They find a sustained and rapid increase in the size of the strongly connected component on the Blogosphere and explain that the community structure is due to the tendency of the bloggers to topically interlink with posts on other blogs.

4.2 Link-based Measures

The link-based analysis in this section is based on the problem posed by [58] and influence models proposed by [30, 29]. These models aim to mathematically simulate the spread of information in social networks. Kempe et al. proposed an approximation of the NP-Hard problem of identifying a set of influential nodes to target so that we can maximize the number of nodes that are activated or influenced. We use the basic *Linear Threshold Model* as proposed by Kempe et al. While the original models were validated on citation graphs, which are much smaller, we apply these algorithms on graphs derived from links between blogs. The citation graphs tend to be much cleaner and some of the techniques proposed do not apply well in the presence of splogs. We also discuss the applicability of simpler, PageRank-based heuristics for influence models on the Blogosphere and the web in general.

Bloggers generally tend to follow mainstream media, the Web, and also blog posts from people who may share similar interests. When an interesting *meme* emerges on some site, a blogger may choose to share it with his audience. Additionally, he may provide more insights and *trackback* to other sources of similar information. Other readers may comment on this post and thereby contribute to the conversation. Such an interactive process leads to the flow of information from one blogger to another. In approximating this interaction, we consider the presence of a link from site *u* to site *v* as evidence that the site *u* is *influenced by* site *v*. We consider only outlinks from posts and do not use comment links or trackback links for building the blog graphs. we take a rather simplistic view in the influence models and convert the *blog graph* to a directed *influence graph*. Figure 4.1 shows a hypothetical blog graph and its corresponding influence graph. An influence graph is a weighted, directed graph with edge weights indicating how much influence a particular source node has on its destination. Starting with the influence graph we aim to identify a set of nodes to target a piece of information such that it causes a large number of bloggers to be influenced by the idea.

4.2.1 Cascade Models

Different influence models have been proposed [29, 30]. The two general categories are *Linear Threshold Model* and *Cascade Model*. We describe some of these below:
In the basic *Linear Threshold Model* each node has a certain threshold for adopting an idea or being influenced. The node becomes activated if the sum of the weights of the active neighbors exceeds this threshold. Thus if node *v* has threshold θ_v and edge weight b_{wv} such that neighbor *w* influenced *v*, then *v* becomes active only if

$$\sum_{w\ active\ neighbors\ of\ v} b_{wv} \geq \theta_v$$

and

$$\sum b_{wv} \leq 1$$

Another model is the *Independent Cascade Model* in which a node gets a single chance to activate each of its neighboring nodes and it succeeds with a probability P_{vw} which is independent of the history.

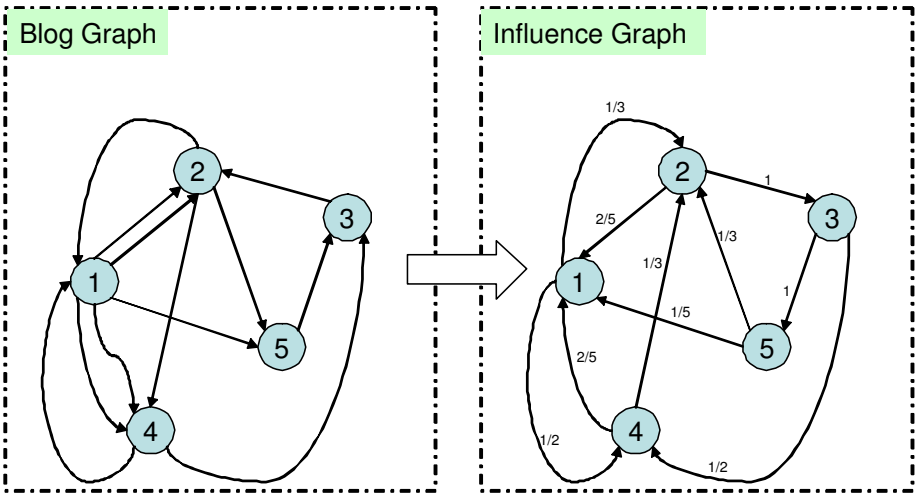


Figure 4.1: This diagram shows the conversion of a blog graph into an influence graph. A link from *u* to v indicates that *u* is influenced by *v*. The edges in the influence graph are reverse of the blog graph to indicate this influence. Multiple edges indicate stronger influence and are weighed higher

As described in the above model, we rank each directed edge between *u*, *v* in the *Influence Graph* such that the presence of multiple directed edges provides additional evidence that node *node u influences node v*. If $C_{u,v}$ is the number of parallel directed edges from *u* to *v* the edge weight

$$W_{u,v} = \frac{C_{u,v}}{d_v}$$

where d_v is the indegree of node *v* in the influence graph.

Since computing the optimal value of expected size of the influenced set, $\sigma(A)$, remains an open question, the algorithm runs the influence propagation model for pseudo-random threshold values and computes the approximate size of $\sigma(A)$.

In selecting the order of activation of nodes, the simplest ranking scheme is one using the number of inlinks (which corresponds to the outlinks in the influence graph). This represents how many other nodes can be influenced by activating the selected node. We also explored PageRank [56] as a heuristic in selecting the target set.

Finally we compare these heuristics with the greedy hill climbing algorithm. In the greedy approach nodes are incrementally added to the initial activation set without backtracking. At each time step, the influence model is run and a node is selected to be added to the initial target set. The node is selected such that adding it to the target set would provide the largest locally optimal increase in the size of the influenced node set.

Other methods such as “distance centrality” based heuristic are also widely used in many studies. This however could not be applied to the blog dataset since computing

the centrality scores over large graphs is expensive without partitioning or identifying subgraphs.

4.2.2 Evaluation

The following section describes some of the experiments and results.

Weblog Dataset The dataset released by Intelliseek/Blogpulse[1] for the 2006 Weblogging Ecosystems Workshop consists of posts from about 1.3 million unique blogs. The data spans over 20 days during the time period in which there were terrorist attacks in London. This time frame witnessed significant activity in the Blogosphere with a number of posts pertaining to this subject. The link graph that we extracted from this dataset consists of 1.2 million links among 300K blogs. However it was also observed that Livejournal [2] sites tend to be highly interlinked and hence for the purpose of the influence models presented in the following sections, we do not consider blogs from these sites for inclusion in the initial activation set. However, we do not discard the blogs from the link graph.

In addition to the Blogpulse dataset we have used the publicly listed feed subscriptions from 82,428 users which consisted of 2,786,687 feeds in all, of which about 496,893 are unique. Bloglines allows users to organize their subscriptions in folders. Although only 35% of Bloglines subscribers use this feature, it provides substantial data to categorize the feeds into different topics.

Effect of Splogs Firstly we investigate the influence of splogs on the different ranking schemes. Splogs are identified using a supervised learning with Support Vector Machines as described in [35]. Figure 4.2 shows the distribution of the splogs in the top results as ranked by PageRank, HITS and Indegree heuristic. From this graph we can observe that the indegree and HITS based heuristics were subject to being easily spammed. As described in [65] the HITS algorithm is susceptible to spamming and this is particularly true in presence of Tightly Knit Communities which have been proposed by [8] and also studied by [41]. Other variations of the HITS algorithm attempt to solve this problem, however it was interesting to note that this issue, though may have already been seen in the case of web spam, is also true for the blog domain. However, PageRank was found to be much resilient to splogs. In order to further investigate this, we analyze the link types. Table 4.1 shows the different types of links in the graph. From this, we can observe that splogs generally tend to link to other splogs. Thus indicating that there is a presence of a community structure in splogs. However splogs also link to legitimate blogs, which most often are high ranked or popular pages.

Node selection heuristics We run the influence models using different heuristics such as PageRank, indegre and greedy algorithm. In PageRank and indegree the nodes are added to the initial target set in the order of their rank. Once a node is selected, the influence model is run and nodes are activated depending on their threshold. The

[1] http://www.blogpulse.com
[2] http://livejournal.com

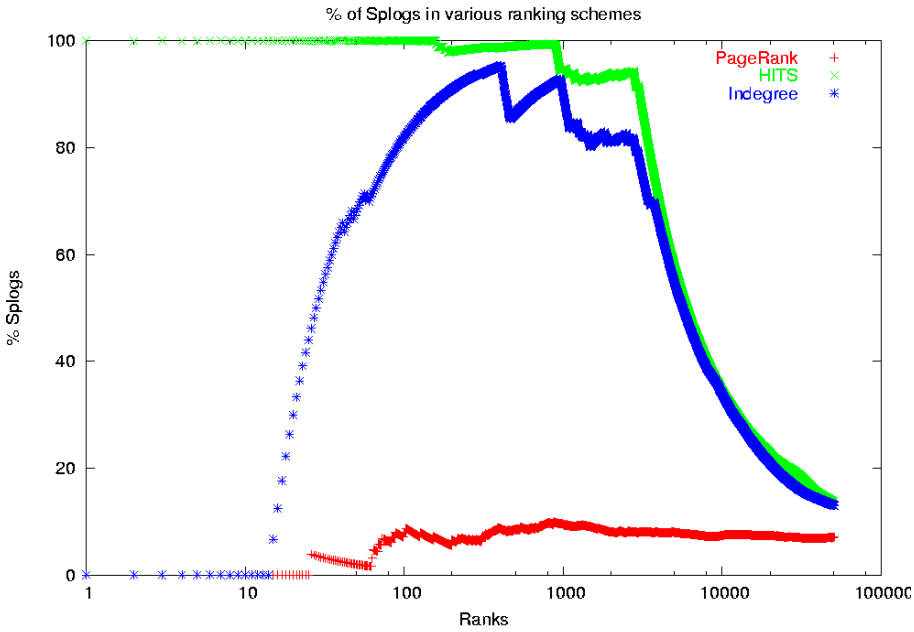


Figure 4.2: *Percentage Splogs in PageRank, HITS, Indegree. The above graph shows what percentage of the top N results in each of the ranking schemes are splogs.*

number of nodes influenced is estimated over multiple iterations. In greedy algorithm nodes are incrementally added to the target set if they locally maximize the size fo the influenced node set. Figure 4.3 shows that the indegree heuristic (which corresponds to the outdegree in the influence graph) at first seems to perform well, but fails to influence more nodes as the initial set grows. Looking further at the results of the top ranks, suggested that the spread of influence plateaus due to the presence of a community of splogs, quite likely from a tightly knit community, around the top ranks. Eliminating such spam using the algorithm previously described results in the 4.4. Thus, by eliminating splogs, the top results obtained from the indegree heuristics almost approximated PageRank. This was also due to the fact that about 70% of the blogs as ranked by PageRank and indegree match after splog elimination.

However, it was found that the PageRank and greedy heuristics seem to perform almost the same even after the elimination of roughly 103687 nodes which correspond to splogs (including failed URLs).

PageRank vs Greedy Heuristic The Greedy heuristic of node selection performs better than both PageRank or indegree. However one of the disadvantages of the greedy approach is that it is computationally quite expensive. PageRank on the other hand is an iterative algorithm that converges to the principal eigenvector of the adjacency matrix. While it is faster to compute, it requires knowledge of the structure of links which

Link type	count
splog+failed to splog+failed	704451
splog+failed to blogfailed	81846
blog to blog	452305
blog to splog+failed	33108
total	1271710

Table 4.1: *Table shows the link types from the link graph of 301700 nodes. 35.5% were from a blog to a blog while 55.4% were from splog+failed to splog+failed.*

might emerge only after the Blogpost has been read and linked to by other blogs over a period of time.

As observed in table 4.1, splogs often link to blogs and hence when a blog is activated it may happen that the influence propagation could lead to the activation of some splogs (which may be a part of a community). These splogs may in turn activate other splogs. Due to this reason it can be observed that after the elimination of splogs from the link graph, the number of activated nodes at target set size of 100 is actually slightly lower than the number of activated nodes when we considered the link graph with splogs in it.

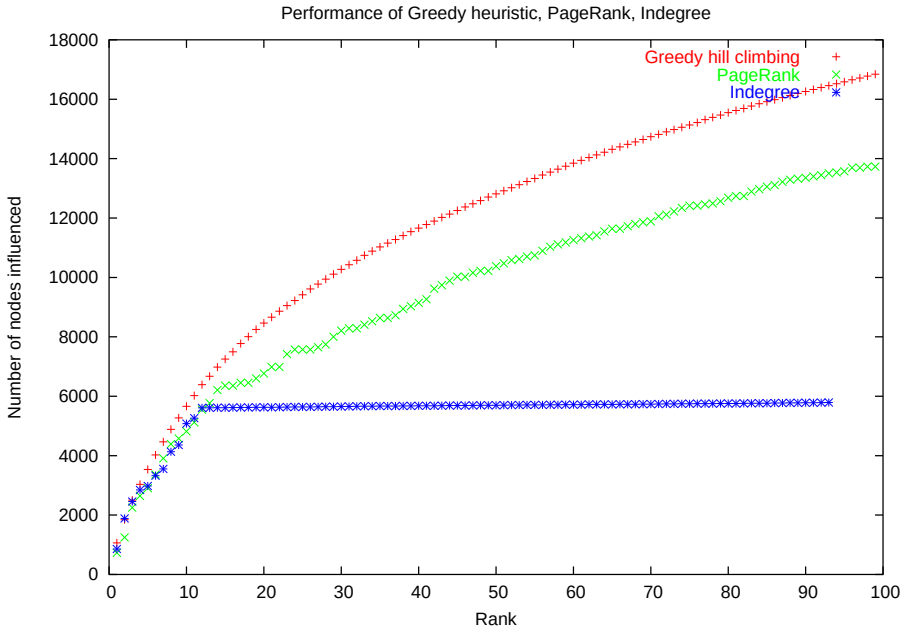


Figure 4.3: *The graph shows the performance of greedy heuristic vs PageRank heuristic vs indegree. The above results show the average influence after 10 iterations of each heuristic.*

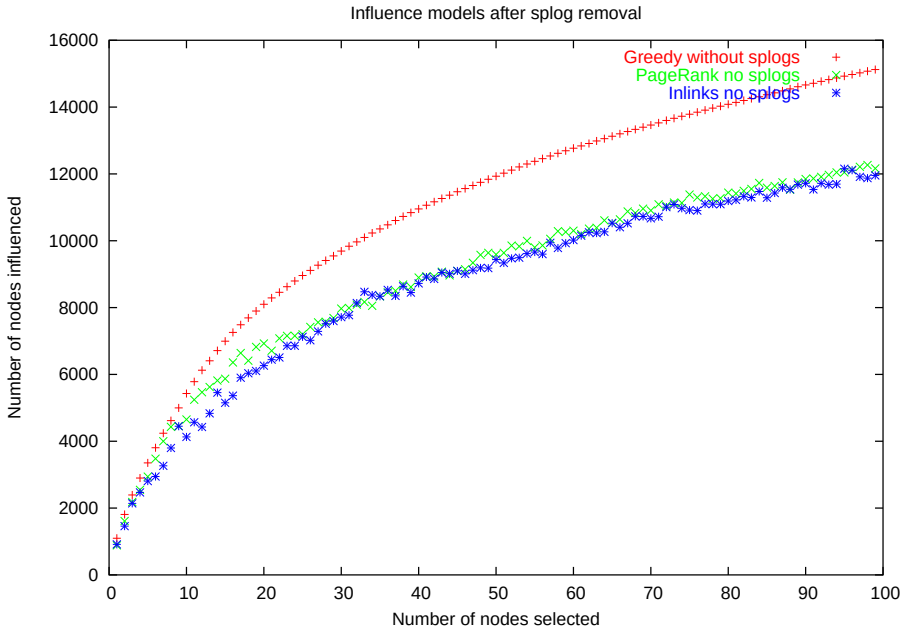


Figure 4.4: *The graph shows the performance of the indegree and pagerank heuristic after splog elimination. The above results show the average influence after 10 iterations of each heuristic.*

4.3 Readership based Influence Measures

Blogs have become a means by which new ideas and information spreads rapidly on the web. They often discuss the latest trends and echo with reactions to different events in the world. Protocols such as RSS, ATOM and OPML have made it much easier to share information online. Both RSS and ATOM are XML based file formats used for syndication. Outline Processor Markup Language (OPML) is a popular XML based format used to share an outline of the feed subscriptions.

Today feeds are being used to provide a wide variety of content online - blogs, wikis, main stream media, search results, etc. all support different forms of syndication. Users can subscribe to such feeds in an RSS reader such as Bloglines[3], Google Reader[4], News Gator[5], etc. Typically, a user adds a feed to an RSS reader when she came across it (perhaps, by chance) as a reference on another blog.

The emphasis in blog search results is on *freshness* and relevance may be measured in terms of how related and how recent the blog post is. Measures of authority are mostly based on the number of inlinks. As Michael Arrington points in a recent post

[3]http://www.bloglines.com
[4]http://www.google.com/reader
[5]http://www.newsgator.com

[6], this can be misleading since a single post from a popular blogger on any topic may make him the top most blog for that topic, even if his blog has little to do with it.

In this section we present readership-based measures of the *topical influence* of a blog. The study presented here is based on the feed subscriptions of a large sample of Bloglines publicly listed users. Using this data, we first characterize the general feed usage patterns. Next, we try to identify the feeds that are popular for a given topic using the folders names as an approximation for a topic. By merging related folders we can create a more appropriate and compact set of topics. Finally we discuss some of the preliminary results in using this approach in support of a number of blog-related applications: feed browsing, feed recommendations, and searching for influential blogs in a different dataset.

By combining measures of topical authority with link-based algorithms we can provide better results for general or conceptual queries for which the goal is to find a resource of useful blogs on a topic.

4.3.1 Dataset Description

Bloglines provides a feature wherein a user may choose to share their subscriptions. We conduct a study of the publicly listed OPML feeds from 83,204 users consisting of a total of 2,786,687 subscriptions of which 496,879 are unique. Figure 4.5 shows the distribution of the top domains in the blogines dataset and compares this with domain distribution from the Blogpulse dataset that was released by Intelliseek for the Webloggling Ecosystems Workshop held at the 15th International World Wide Web Conference. This dataset consists of posts from about 1.3 million blogs over a span of 20 days. The charts indicate differences in blog search engines indexes and RSS monitoring services. In particular, while there are a number of Bloglines users who subscribe to Web 2.0 sites and dynamically generated RSS feeds over customized queries, most of the Blogpulse dataset consists of blogs from blog hosting companies such as livejournal, xanga etc.

4.3.2 General statistics

According to Bloglines/Ask in July 2005 there were about 1.12 Million feeds that really matter, which is based on the feeds subscribed by all the users on Bloglines. A study of the feeds on Bloglines by McEvoy [47] in April 2005 showed that there were about 32,415 public subscribers and their feeds accounted for 1,059,140 public feed subscriptions.

We collected similar data of the publicly listed users on Bloglines. From last year, the number of publicly listed subscribers had increased to 83,204 users (2.5 times that of last year) and there were 1,833,913 listed feeds (1.7 times) on the Bloglines site. Hence, even though the Blogosphere is almost doubling every six months, we found that the number of feeds that *"really matter"* doubles roughly every year. Inspite of it, it is still only a small fraction of the entire Blogosphere. Following sections describe some of the usage patterns and interesting statistics obtained from our analysis.

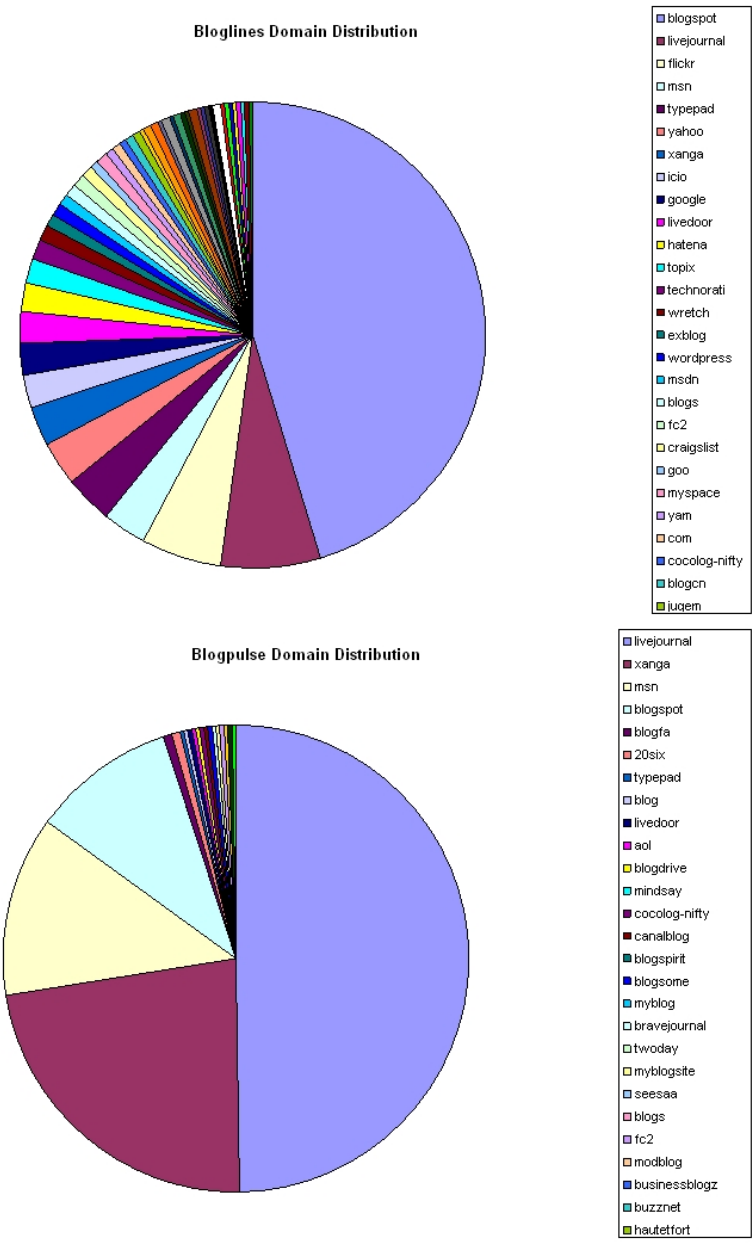


Figure 4.5: The distribution of domains in the Bloglines dataset. In comparison the second graph shows the data from Blogpulse.

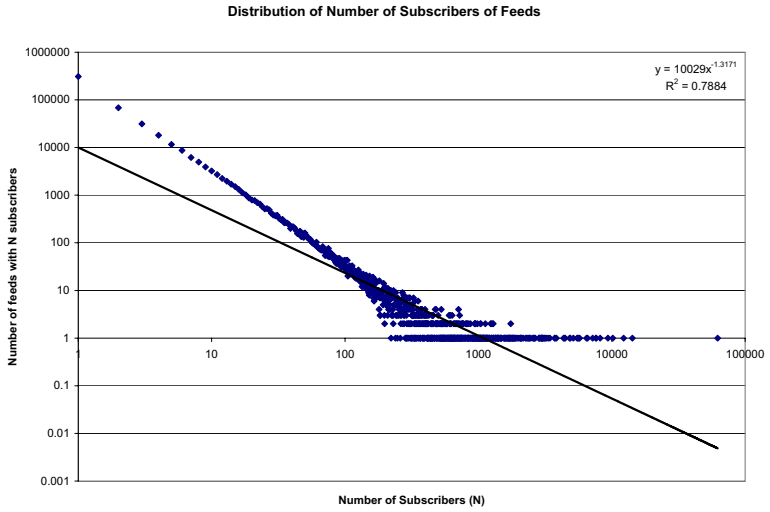


Figure 4.6: The number of subscribers for feeds follows a power law distribution. A few feeds are highly subscribed but most of the feed have a modest number of subscriptions. There are a number of feeds that have single digit subscribers.

User Statistics

Figure 4.6 shows the distribution of the number of subscribers for 496,879 unique feeds across 83,204 subscribers. This graph indicates a typical power law behavior with a few feeds having a large number of subscribers while most having a modest number of subscribers.

The number of subscribers for a feed is an indication of its authority and influence over its audience.

Next, we analyzed the number of feeds subscribed per user. Figure 4.7 indicates a slight deviation from a power law curve. We find that there are a few users, which are mostly automated tools posting to Bloglines, having very high number of feeds. Most of the users on the other hand have between 30 and 100 subscriptions. It is possible that for most users there is an inherent limit on the amount of information that they can keep track of at any given time. A new user who joins Bloglines might subscribe to a few feeds from the long tail (such as blogs belonging to friends or based on special interests) but is more likely to also subscribe to feeds that are already quite popular (such as Slashdot, Boingboing etc.).

Figure 4.8 shows a scatter plot of the number of folders compared to the number

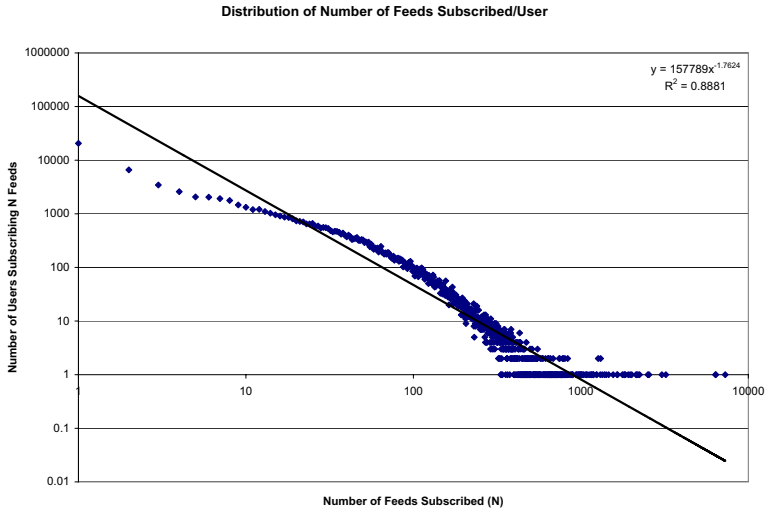


Figure 4.7: The total number of subscribed feeds across all users. Most users have about 80-100 feeds subscribed per user.

of feeds subscribed by a user. It can be observed from this graph that as the number of feeds subscribed increase, the users tend to organize them into greater number of folders.

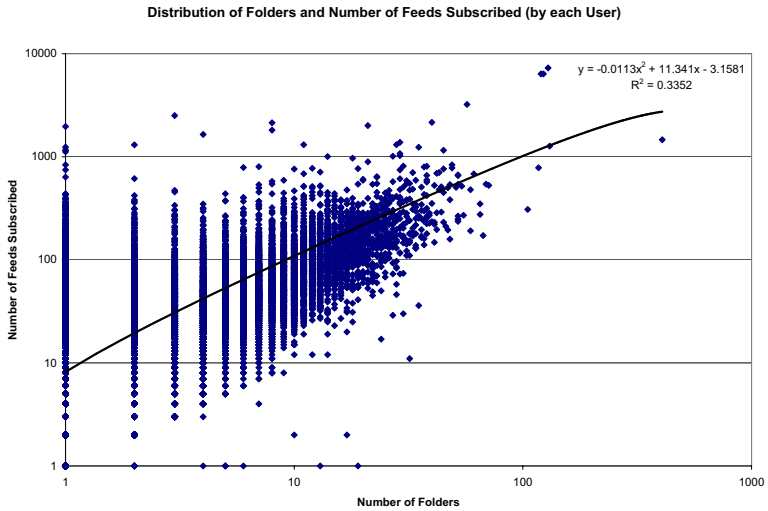


Figure 4.8: Scatter plot showing the relation between the number of folders and number of feeds subscribed. Note: This includes the feeds subscribed under the default folder. Inspite of the high variance there is a clear trend that as the number of feeds increase, users tend to organize them into more folders.

Folder Statistics

Bloglines has a feature by which a user may organize their feeds into different folders. While only some (26,2436 or about 35%) of the public subscribers use folders, it provides a user generated categorization of feeds. Other 4.9 shows the distribution of folder usage across these users and Figure 4.10 shows the number of folders per user.

4.3.3 Topic Representation and Clustering

Folder names can be treated as an approximation of a topic. Folder names in Bloglines are used in a way that is similar to Folksonomies on the web. As shown in Figure 4.11, by aggregating folders across all users, we can generate a tag cloud that shows the relative popularity and spread of various topics across Bloglines users. The tag cloud

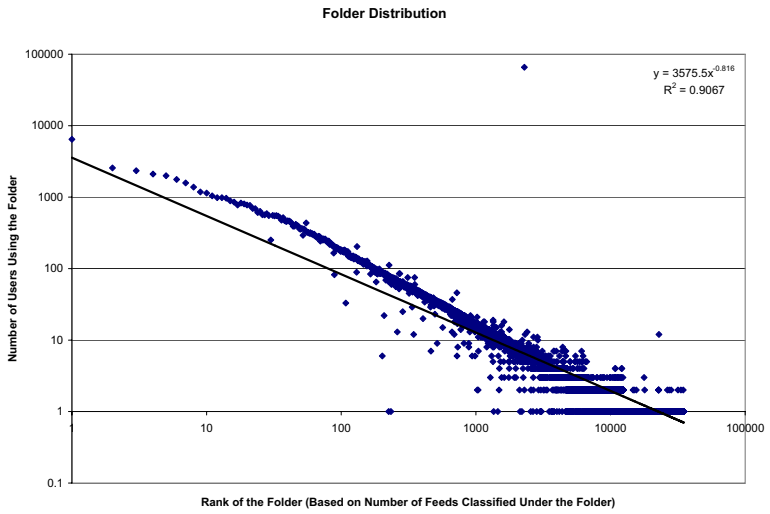


Figure 4.9: The number of users who use a folder

shown here is based on the top 200 folders. Note that the tag cloud contains terms such as 'humor' and 'humour', etc. These terms represent variations in which different users label feeds. By merging folder names that are *'related'* we can generate a more appropriate and compact representation of the tag cloud.

Next, we describe an approach used to merge related folders together. We were first tempted to use a morphological approach – merging the *blog* and blogs categories, for example. However, we soon discovered that folders with lexically similar names might actually represent different categorization needs of the users. For example, the folder 'Podcasting' consists of feeds that talk about how to podcast and provide tools. On the other hand 'Podcasts' refers to feeds containing actual podcasts. Other examples include 'Music' vs. 'Musica' (a topic with Spanish music blogs).

For each folder we construct a vector containing the feeds that have been categorized under that folder name and their corresponding counts. At this step we take only the top 100 most frequently occurring feeds per folder. Ideally this threshold would have to be determined statistically. Some folders, such as 'friends', were observed to consist of a large set of feeds for each of which there are only a handful of subscribers. On the other extremely popular folders like 'politics' contained a number of feeds that have many subscribers.

Two cases need to be considered for computing folder similarity: first is the case where feeds in one folder may either partially or completely subsume feeds present in

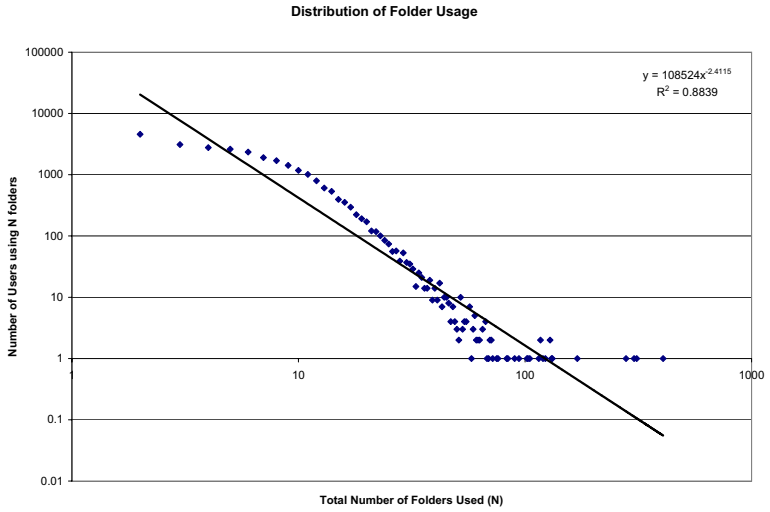


Figure 4.10: Distribution of the number of folders per user

another folder. Subsumption indicates that there is a broader category and the larger folder is more general. For example the folder 'news' subsumes a number of folders that are more specific, such as 'tech news','IT news', 'general news', etc. For detecting the topics, it suffices to put these into a single category titled 'news'. To compute subsumption we first find an overlap factor. For all folder pairs i,j we maintain a score of the overlap of feeds in folder j with feeds in folder i as follows:

$$overlap = \frac{matches}{size_j}$$

Folder similarity can be described in terms of the feeds that are contained in the folders. Two folder names are considered to be similar if they contain similar feeds in them. For each pair of folder names we compute the cosine similarity as follows:

$$cos(i,j) = \frac{\sum_k W_{i,k} W_{j,k}}{\sqrt{\sum_k W_{i,k}^2 * \sum_k W_{j,k}^2}}$$

The weight $W_{i,k}$ is determined by a TFIDF score for each feed in the vector. The weights are computed using the following formula:

$$W_{folder}(feed) = freq_{folder}(feed) * log(\frac{folderCount}{|foldersContainingFeed|})$$

Before Merge

After Merge

Figure 4.11: The tag cloud generated from the top 200 Folders before and after merging related folders. The size of the word is scaled to indicate how many users use the folder name.

First we start by ranking the folders based on the number of users using the folder. Next we go through this ranked list and merge related folders together. A lower ranked folder is merged into a higher ranked folder if $overlap > \theta$ or $cosine > \delta$. These thresholds were empirically set to 0.4. The table 4.2 shows the list of top 200 folders with the corresponding merged folders.

4.3.4 Applications

Our original motivation for this work was the need to classify blogs with respect to a set of topics for a study of influence on the Blogosphere [28]. We hoped that the folders common to many Bloglines users would provide an intuitive set of blog topics. Moreover, the feeds that Bloglines users assigned to the folders could be used to collect training data for the categories. This is in fact the case, but after working with the data, we recognized that it supports the needs of many other studies and applications.

Feed Recommender Folder similarity allows us to compare how related two folder vectors are based on the feeds that occur in them. Feed similarity can be defined in a similar manner: two feeds are similar if they often co-occur under similar folders. Note that this definition of feed similarity does not use the textual content of the feed but is entirely based on the subscription data. This gives us an ability to compare two feeds and recommend new feeds that are like a given feed. For each feed there is a folder vector that maintains a count of the number of times the feed has been categorized under a folder name. For a pair of feeds i,j feed similarity is defined as:

$$cos(i,j) = \frac{\sum_k W_{i,k} W_{j,k}}{\sqrt{\sum_k W_{i,k}^2 * \sum_k W_{j,k}^2}}$$

The weight $W_{i,k}$ is determined by a TFIDF score for each folder in the feed vector. The weights are computed using the following formula:

$$W_{feed}(folder) = freq_{feed}(folder) * log(\frac{feedCount}{|feedsLabeledFolder|})$$

Identifying Influential Feeds Consider a scenario where a user has a few blogs that she subscribes to or is familiar with a couple of extremely popular blogs for a topic. Now, she wishes to find other blogs that are also opinion leaders in this area. In this section we present a simple heuristic that is based on an influence propagation model using linear threshold that was described in the previous section.

Algorithm 1 can be used to identify a set of nodes that are influential for a given topic. As shown in figure 4.12, starting with some seed blogs for a given topic, we can induce a set of blogs that are termed as the *followers*. Followers are those blogs that are often influenced by the seed set. The goal is to infer other authoritative blogs or *leaders* for the topic. The pseudocode of the Leaders Influence Followers Algorithm 1 describes the various steps involved in identifying topical influential nodes.

Rank	Folder	Merged Folders
1	news	technology, tech news, it news, general news, technews, world news, news feeds, newspapers
2	blogs	friends, misc, personal, people, other, general personal blogs, blogroll, daily, weblogs, tech blogs, other blogs, favorites, bloggers
3	tech	friends, technology, tech news, gadgets, other, general, it, geek, blogroll, daily, english, it news, tech blogs, computer, technews, technology news, computers, favorites, technical, public, gadget, tech stuff, geek news, computing, techie, geek stuff
4	comics	fun, humor, funny, humour, cartoons, fun stuff, webcomics, comix, comic strips
5	music	mp3, mp3 blogs
6	politics	political, political blogs
7	podcasts	podcast
8	design	web design, web, web development, webdesign, webdev, css, web dev,web standards
9	sports	
10	science	science news
11	business	biz, business news
12	software	downloads
13	entertainment	fun, movies
14	linux	
15	mac	apple, macintosh, mac stuff, mac news
16	flickr	
17	food	cooking, food blogs
18	marketing	
19	games	gaming, video games
20	weather	
21	photography	photos, photo, fotografia
22	security	
23	java	
24	blog	blogger
25	programming	development, dev, technical, software development, code

Table 4.2: The top 25 folders with corresponding merged folders

Seed Blogs
http://www.dailykos.com
http://www.talkingpointsmemo.com
Top Leader Blogs
http://www.huffingtonpost.com/theblog
http://americablog.blogspot.com
http://thinkprogress.org
http://www.tpmcafe.com
http://www.crooksandliars.com
http://atrios.blogspot.com
http://www.washingtonmonthly.com
http://billmon.org
http://www.juancole.com
http://capitolbuzz.blogspot.com
http://instapundit.com
http://www.opinionjournal.com
http://digbysblog.blogspot.com
http://michellemalkin.com
http://www.powerlineblog.com
http://theleftcoaster.com
http://www.andrewsullivan.com
http://www.thismodernworld.com

Table 4.3: The influential feeds found starting with a small seed set for topic **'Politics'**

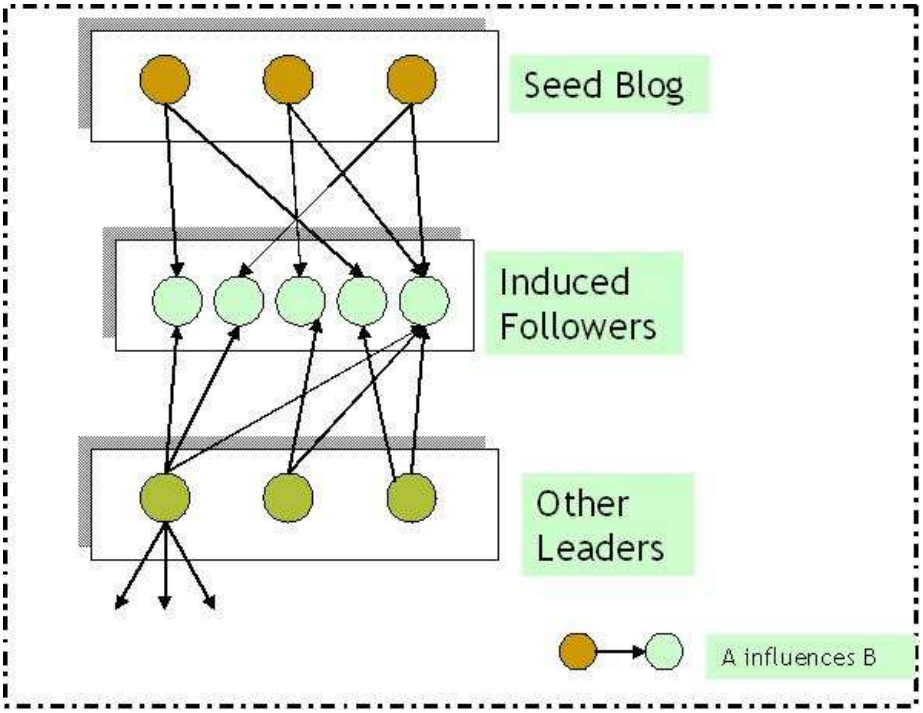


Figure 4.12: Identifying leaders: Starting with a few seed blogs on a topic a set of followers are induced and other leaders for this set are identified using the influence graph.

Starting with a few top ranked feeds from the Bloglines dataset for the folders 'Politics', 'Tech','Business' and 'Knitting' we use the LIFT algorithm to find other leaders in the Blogpulse dataset. Table 4.3 to 4.6 show some of the results.

FTM! Feeds That Matter FTM![6] is a site that was implemented out of a need to find a high quality listing or index of *topical* blogs and feeds. This site is based on the Bloglines dataset described in this paper and implements the algorithms presented here for merging folders and providing recommendations. For example if the user was interested in a topic, say photography, she could look at the tag cloud and quickly find feeds that are most often categorized under the folder name "photography". Next, the system allows users to subscribe to the popular feeds directly in their Bloglines or Yahoo RSS readers. Alternatively, one could start a known feed and FTM! would provide recommendations based on the subscription information. By monitoring the number of subscriptions for such recommendations we hope to further evaluate the recommendation algorithm itself.

We evaluate if folder similarity results in grouping related feeds together. We do

[6] http://ftm.umbc.edu/

Seed Blogs
http://blog.fastcompany.com
http://business2.blogs.com/business2blog
Top Leader Blogs
http://headrush.typepad.com/creating_passionate_users
http://blog.fastcompany.com
http://www.ipodlounge.com
http://www.fastcompany.com/homepage
http://brandautopsy.typepad.com/brandautopsy
http://gigaom.com
http://www.mobileread.com
http://blogs.salon.com/0002007
http://sethgodin.typepad.com/seths_blog
http://ad-rag.com
http://www.warrenellis.com
http://www.micropersuasion.com
http://battellemedia.com
http://www.adrants.com
http://customerevangelists.typepad.com/blog

Table 4.4: The influential leaders found starting with a small seed set for topic **'Business'**

Seed Blogs
http://slashdot.org
http://www.kuro5hin.org
Top Leader Blogs
http://www.boingboing.net
http://www.engadget.com
http://www.metafilter.com
http://www.c10n.info
http://www.makezine.com/blog
http://radio.weblogs.com/0001011
http://mnm.uib.es/gallir
http://www.mozillazine.org
http://weblogs.mozillazine.org/asa
http://www.gizmodo.com

Table 4.5: The influential feeds found starting with a small seed set for the topic **'Technology'**

Algorithm 1 Leaders Influence Followers Algorithm

$S \leftarrow SeedSet$
$F \leftarrow InfluencedFollowersSet$
$IG \leftarrow InfluenceGraph$
for all i such that $0 \leq i \leq max\_iterations$ **do**
  Activate S
  for all $v \in IG$ **do**
    θ_v = random score
  end for
  for all $v \in IG$ **do**
    if $\sum_{w\ active\ neighbors\ of\ v} b_{wv} \geq \theta_v$ **then**
      Activate v
      add v to F_i
    end if
  end for
end for
$F = F_i \bigcup F_{i+1} \bigcup \cdots F_{max\_iterations}$
for all k has inlinks to F **do**
  o_k = outlink count of k
  n_k = number of nodes linked from k to F
  $leader\_score = \frac{n_k}{o_k} * log(o_k)$
end for

this by comparing the folder similarity based on co-citations in URL vectors to text similarity of text obtained form the homepages of the feeds.

Figure 4.13 shows a comparison of average text similarity of feeds in the top 20 folders. For all the folders it was found that the feeds shared a greater similarity within the folder rather than across other folders. While the scores may seem low, studies on Technorati data by Brooks [10] show cosine similarity of posts sharing the same tag to be around 0.3. According to their study, when the same posts were clustered using the high scoring TFIDF terms the average text similarity was around 0.7.

Table 4.7 shows some of the recommendations for a few blogs. The feed recommendations are obtained by comparing the feeds to find how often they co-occur in the same folder. To evaluate the effectiveness of this system, we use the text based cosine similarity as a measure of how related the feeds are. We find that many of the recommended feeds have a high similarity score with the feed submitted.

4.4 Discussion

As described in this section, there are different possible ways in which we can use direct measurable attributes of the network to analyze influence. In the first case we used a simple model for tracking information cascades in the network. The second technique used a different approach through readership information. The assumption here was that the readers were a representative set of the people who follow blogs

Seed Blogs
http://www.yarnharlot.ca/blog
http://wendyknits.net
Top Leader Blogs
http://booshay.blogspot.com
http://mamacate.typepad.com/mamacate
http://www.thejonblog.com/knit
http://alison.knitsmiths.us
http://www.dioramarama.com/kmel
http://knittersofdoom.blogspirit.com
http://tonigirl.blogdrive.com
http://www.crazyauntpurl.com
http://www.januaryone.com
http://nathaniaapple.typepad.com/knit_quilt_stitch
http://www.knittygritty.net
http://www.katwithak.com
http://www.myblog.de/evelynsbreiwerk
http://nepenthe.blog-city.com
http://zardra.blogspot.com

Table 4.6: The influential feeds found starting with a small seed set for the topic **'Knitting'**

and their intuitive classifications into folders gave indication of the category to which the feed belonged. Both these approaches look promising and we plan to use them in the following manner: First, we plan to refine the cascade models to include polarity, topical sensitivity and temporal effect. Next, using either simulated graphs or data from the Blogosphere, we will test our models with the readership information as the benchmark. Such an approach would allow us to model and test *topical influence*.

http://www.dailykos.com	**Similarity**
http://www.andrewsullivan.com	**0.496**
http://www.talkingpointsmemo.com	**0.45**
http://atrios.blogspot.com	0.399
http://jameswolcott.com	**0.466**
http://mediamatters.org	0.262
http://yglesias.typepad.com/matthew/	0.285
http://billmon.org/	0.343
http://digbysblog.blogspot.com	**0.555**
http://instapundit.com/	0.397
http://www.washingtonmonthly.com/	**0.446**
http://blog.fastcompany.com	
http://business2.blogs.com/business2blog	0.303
http://www.fastcompany.com	0.454
http://sethgodin.typepad.com/seths_blog/	0.374
http://www.ducttapemarketing.com/	0.028
http://customerevangelists.typepad.com	0.399
http://blog.guykawasaki.com/	**0.441**
http://www.tompeters.com	**0.457**
http://www.paidcontent.org/	0.351
http://slashdot.org	
http://www.techdirt.com/	**0.516**
http://www.theregister.co.uk/	0.1
http://www.geeknewscentral.com/	0.286
http://www.theInquirer.net	0.2
http://news.com.com/	0.24
http://www.kuro5hin.org/	0.332
http://www.pbs.org/cringely/	0.087
http://backword.me.uk/	-
http://digg.com/	0.165
http://www.infoworld.com/news/index.html	0.203
http://www.yarnharlot.ca/blog/	
http://wendyknits.net/	**0.419**
http://www.woolflowers.net/	0.139
http://zeneedle.typepad.com/ zeneedle_process_of_art/	0.383
http://WWW.markarkleiman.com/	-
http://www.keyboardbiologist.net/knitblog/	0.297
http://alison.knitsmiths.us/	0.284
http://knitandtonic.typepad.com/knitandtonic/	**0.542**
http://www.crazyauntpurl.com/	**0.521**
http://www.lollygirl.com/blog/	**0.4**
http://ma2ut.blogspot.com	**0.423**

Table 4.7: Example recommendations and corresponding text similarity scores.

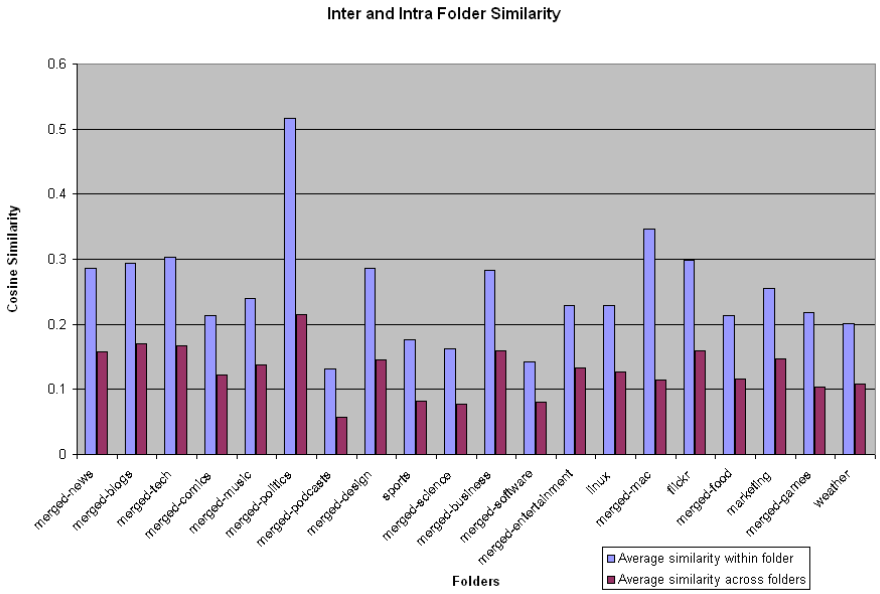


Figure 4.13: The average text similarity of the top 20 Folders. The chart shows the average similarity for the top 15 feeds within and across all the folders.

Chapter 5

Opinion Extraction

5.1 Introduction

Each person has a distinct set of likes and dislikes. Some of these are based on judgment and assessment of a past experience, while others may look seemingly random or subjective. Personal preferences, social and cultural factors shape our view and opinions. Social interactions play a major role in sharing our views and hence our opinions with others. Some individuals may be more vocal and expressive about what they think or believe while others may do so in a subtle way. Blogs started as online journals and diaries, which by nature tend to be informal, expressive and often opinionated. With the low barrier for creating content, social media, especially blogs have made it easier for people to share their views with others. Often such opinions manifest in subtle ways while talking about everyday experiences. The realization that such opinions have the power to influence others in the social network, has led to increased interest in "opinion/ sentiment" extraction. One such effort has been through the annual Text Extraction and Retrieval Conference (TREC) held by NIST.

Here, we describe our approach to the 2006 TREC Blog track opinion retrieval task. For the opinion retrieval task, we used a number of approaches to score the retrieved posts based on different features. The scores were then combined using machine learning. We also found that a challenging part of the opinion retrieval task was to effectively eliminate spam blog posts (splogs) and cleaning the blog posts to eliminate extraneous links.

UMBC and JHU/APL collaborated as a team for athe 2006 TREC Blog track sponsored by NIST. This track asked participants to implement and evaluate a system to do "opinion retrieval" from blog posts. Specifically, the task was defined as follows: build a system that will take a query string describing a topic, e.g., "March of the Penguins", and return a ranked list of blog posts that express an opinion, positive or negative, about the topic. For evaluation, NIST provided a dataset of over three million blogs drawn from about 100,000 blogs. Participants built and trained their systems to work on this dataset. Contestants do an automatic evaluation by downloading and running, without further modification to their systems, a set of fifty test queries.

5.2 BlogVox: Seperating Blog Wheat from Blog Chaff

This section describes BlogVox [27], an opinion extraction system.

5.2.1 The TREC Blog Track

Opinion extraction has been studied for mining sentiments and reviews in specific domains such as consumer products [12] and movies [57, 16]. More recently, blogs have become a new medium though which users express sentiments. Opinion extraction has thus become important for understanding consumer biases and is being used as a new tool for market intelligence [18] [54][44]. Different sentiment classification techniques have been applied in movies and product domains. Many of these techniques use a combination of machine learning, NLP and heuristic techniques. While some of the work looks at identifying opinions at a document level, others have tried to classify sentences and summarize opinions. Sentiment classification techniques often rely on identifying subjective terms and phrases using NLP, machine learning or wordlists. Nigam and Hurst use a combination of shallow parsing and machine learning to identify polarity and topical classification of sentences. The sentence level models are used in making document level classifications. Turney [63] proposes a simple unsupervised technique using search engine hits to determine the *semantic orientation* of phrases containing adjectives or adverbs in the reviews.

Minqing Hu and Bing Liu [24] propose using WordNet to determine the polarity of different adjectives. Their goal is to identify sentiment at a sentence level. The overall polarity score for a sentence is determined by combining the weights contributed by each of the adjectives near a feature word. The Opinion Observer system [44] extends this work to summarizing the pros and cons of various features of a product.

For TREC our team developed a system based upon the Lucene information retrieval system for the basic retrieval task. Compared to domain-specific opinion extraction, identifying opinionated documents about a randomly chosen topic from a pool of documents that are potentially unrelated to the topic is a much more difficult task. Our goal for this project was to create a system that could dynamically learn topic sensitive sentiment words to better find blog posts expressing an opinion about a specified topic. We use a meta-learning approach and designed an architecture where a set of scorers would each evaluate every relevant document and produce a score representing how opinionated it is. These scores would then be used as a feature vector for an SVM [11] to classify our documents. Following is a description of the BlogVox system which utilized machine learning techniques for both pre-indexing data preparation and post-retrieval ranking for opinionatedness.

5.2.2 Pre-indexing Processing

The TREC dataset consisted of a set of XML formatted files, each containing blog posts crawled on a given date. The entire collection consisted of over 3.2M posts from 100K feeds [46]. These posts were parsed and stored separately for convenient indexing,

using the HTML parser tool [1]. Non-English blogs were ignored, as was any page that failed to parse due to encoding issues.

To make the challenge realistic NIST explicitly included 17,969 feeds from splogs, contributing to 15.8% of the documents. There were 83,307 distinct homepage URLs present in the collection, of which 81,014 could be processed. The collection contained a total of 3,214,727 permalinks from all these blogs. Our automated splog detection technique identified 13,542 blogs as splogs. This accounts for about 16% of the identified homepages. The total number of permalinks from these splogs is 543,086 or around 16% of the collection. While the actual list of splogs is currently not available for comparison , the current estimate seem to be close. To prevent the possibility of splogs skewing our results permalinks associated with splogs were not indexed.

To improve the quality of opinion extraction results, it is important to identify the title and content of the blog post because the scoring functions and Lucene indexing engine can not differentiate between text present in the links and sidebars of the blog post. Thus, a post which has a link to a recent post titled 'Why I love my Ipod' would be retrieved as an opinionated post even if the actual post is actually about some other topic.

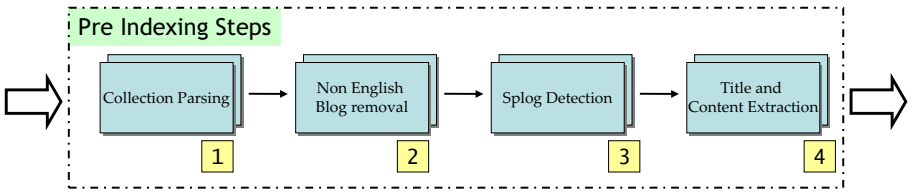


Figure 5.1: BlogVox text Preparation steps: (i) parsing the TREC corpus (ii) removing non English posts (iii) Eliminating splogs from the collection (iv) removing spurious material from the DOM tree.

5.2.3 Post-retrieval Processing

After pre-indexing blog posts are indexed using Lucene, an open-source search engine. Lucene internally constructs an inverted index of the documents by representing each document as a vector of terms. Given a query term Lucene uses standard Term Frequency (TF) and Inverse Document Frequency (IDF) normalization to compute similarity. In addition, the scoring formula can also be tuned to perform document length normalization and term specific boosting [2]. We used the default parameters while searching the index. However, in order to handle phrasal queries such as "United States of America" we reformulate the original query to boost the value of exact matches or proximity-based matches for the phrase.

Given a TREC query a set of relevant posts are retrieved from the Lucene index and sent to the scorers. As shown in figure 5.2, a number of heuristics are employed

[1] `http://htmlparser.sourceforge.net/`
[2] `http://lucene.apache.org/java/docs/scoring.html`

to score the results based on the likelihood that it contains an opinion about the query terms. These scorers work by using both document level and individual sentence level features. Some of the scoring heuristics were supported by a hand-crafted list of sentiment words.

The following is a brief description of each scoring function:

Query Word Proximity Scorer finds the average number of sentiment terms occurring in the *vicinity* of the query terms using a window size of 15 words before and after the query terms. If the query is a phrasal query, the presence of sentiment terms around the query was weighted twice.

Parametrized Proximity Scorer was similar to the Query Word Proximity Scorer. However, we used a much smaller dictionary which was divided into two subsets: highly polar sentiment words, and the relatively less polar words. We used parameters to specify the window of text to search for sentiment words (five and fifteen), and to boost sentiment terms around phrase queries (one and three). This resulted in a total of eight scorers.

Positive and Negative Scorers counted the number of sentiment words (positive, negative) in the entire post.

We also experimented with other scoring functions, such as adjective word count scorer. This scorer used an NLP tool to extract the adjectives around the query terms. However, this tool did not perform well mainly due to the noisy and ungrammatical sentences present in blogs. Once the results were scored by these scoring modules, we used a meta-learning approach to combine the scores using a supervised learning approach.

We used Support Vector Machines, trained using a set of 670 samples of which 238 were positive (showed a sentiment) and the rest of negative. Polynomial kernel with degree gave the best results with precision of 80% and recall of 30%. The model was trained to predict the probability of a document expressing opinion. This value was then combined with the Lucene relevance score to produce final runs.

5.2.4 Data Cleaning

Identifying and Removing Spam

Two kinds of spam are common in the Blogosphere (i) spam blogs or splogs, and (ii) spam comments. We first discuss spam blogs, approaches on detecting them, and how they were employed for BlogVox.

Problem of Spam Blogs Splogs are blogs created for the sole purpose of hosting ads, promoting affiliate sites (including themselves) and getting new pages indexed. Content in splogs is often auto-generated and/or plagiarized, such software sells for less than 100 dollars and now inundates the Blogosphere both at ping servers (around 75% [33]) that monitor blog updates, and at blog search engines (around 20%, [37]) that index them. Spam comments pose an equally serious problem, where authentic blog posts feature auto-generated comments that target ranking algorithms of popular search engines. A popular spam comment filter [3] estimates the amount of spam detected to be

[3] http://akismet.com

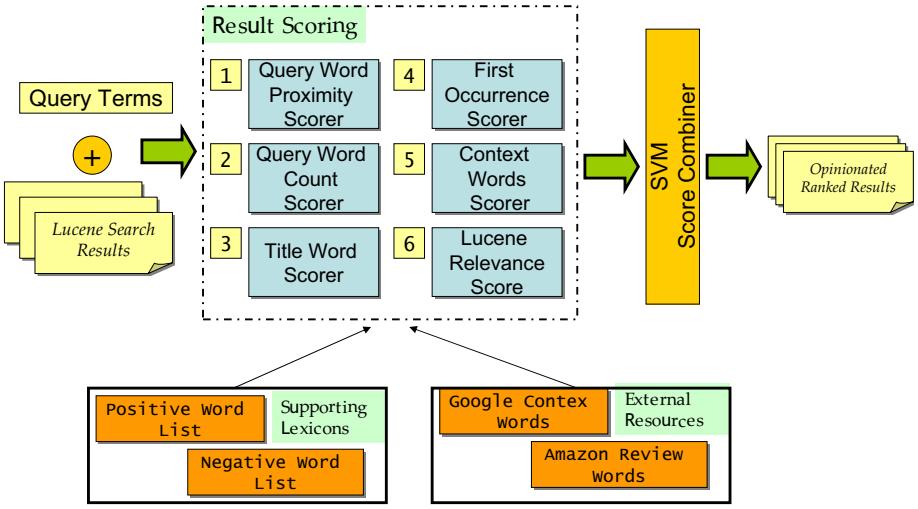


Figure 5.2: After relevant posts are retrieved, they are scored by various heuristics and an overall measure of opinionatedness computed by a SVM.

around 93%.

Figure 5.3 shows a splog post indexed by a popular blog search engine. As depicted, it features content plagiarized from other blogs (ii), displays ads in high paying contexts (i), and hosts hyperlinks (iii) that create link farms. Scores of such pages now pollute the Blogosphere, with new ones springing up every moment. Splogs continue to be a problem for web search engines, however they present a new set of challenges for blog analytics. This paper stresses the latter.

Detecting Splogs Splogs are well understood to be a specific instance of the more general spam web-pages [21]. Though offline graph based mechanisms like TrustRank [22] are sufficiently effective for the Web, the Blogosphere demands new techniques. The quality of blog analytics engines is judged not just by content coverage, but also by their ability to index and analyze recent (non-spam) posts. This requires that fast online splog detection/filtering [34][59] be used prior to indexing new content.

We employ statistical models to detecting splogs as described by [37], based on supervised machine learning techniques, using content local to a page, enabling fast splog detection. These models are based solely on blog home-pages, and are based on a training set of 700 blogs and 700 splogs. Statistical models based on local blog

Figure 5.3: A typical splog, plagiarizes content (ii), promotes other spam pages (iii), and (i) hosts high paying contextual advertisements

Feature	*Precision*	*Recall*	*F1*
words	.887	.864	.875
urls	.804	.827	.815
anchors	.854	.807	.830

Table 5.1: *SVMs with 19000 word features and 10000 each of URL and anchor text features ranked using Mutual Information.*

features perform well on spam blog detection. See Table 5.1. The bag-of-words based features slightly outperforms bag-of-outgoingurls (URL's tokenized on '/') and bag-of-outgoinganchors. Additional results using link based features are slightly lower that local features, but effective nonetheless. Interested readers are referred to [37] for further details. Therefore, BlogVox used only local features to detect splogs.

Comment spam Comment spam occurs when a user posts spam inside a blog comment. Comment spam is typically managed by individual bloggers, through moderating comments and/or using comment spam detection tools (e.g. Akismet) on blogging platforms. Comment spam and splogs share a common purpose. They enable indexing new web pages, and promoting their page rank, with each such page selling online merchandise or hosting context specific advertisements. Detecting and eliminating comment spam [49] depends largely on the quality of identifying comments on a blog post, part of which is addressed in the next section.

Algorithm 2 Blog post cleaning heuristic

```
Nodes[] tags = tags in the order of the depth first traversal of the DOM tree
for all i such that 0 ≤ i ≤ |tags| do
  dist = nearestLinkTag(tags, i);
  if dist ≤ θ_dist then
    eliminate tags[i]
  end if
end for
```

5.2.5 Identifying Post Content

Most extraneous features in blog post are links. We describe two techniques to automatically classify the links into content-links and extra-links. Content links are part of either the title or the text of the post. Extra links are not directly related to the post, but provide additional information such as: navigational links, recent entries, advertisements, and blog rolls. Differentiating the blog content from its chaff is further complicated by blog hosting services using different templates and formats. Additionally, users host their own blogs and sometimes customize existing templates to suit their needs.

Web page cleaning techniques work by detecting common structural elements from the HTML Document Object Model (DOM) [67, 68]. By mining for both frequently repeated presentational components and content in web pages, a site style tree is constructed. This tree structure can be used for data cleaning and improved feature weighting. Finding repeated structural components requires sampling many web pages from a domain. Although blogs from the same domain can share similar structural components, they can differ due to blogger customization. Our proposed technique does not require sampling and works independently on each blog permalink.

Instead of mining, we used a simple general heuristic. Intuitively extraneous links tend to be tightly grouped containing relatively small amounts of text. Note that a typical blog post has a complex DOM tree with many parts, only one of which is the content of interest in most applications.

After creating the DOM tree we traverse it attempting to eliminate any extraneous links and their corresponding anchor text, based upon the preceding and following tags. A link **a** is eliminated if another link **b** within a θ_{dist} tag distance exists such that:

- No title tags (H1, H2...) exist in a θ_{dist} tag window of **a**.
- Average length of the text bearing nodes between **a** and **b** is less than some threshold.
- **b** is the nearest link node to **a**.

The average text ratio between the links, $\alpha_{avgText}$ was heuristically set to 120 characters and a window size, θ_{dist} of 10 tags was chosen. The Algorithm 2 provides a detailed description of this heuristic.

Next we present a machine learning approach to the link classification problem. From a large collection of blog posts, a random sample of 125 posts was selected.

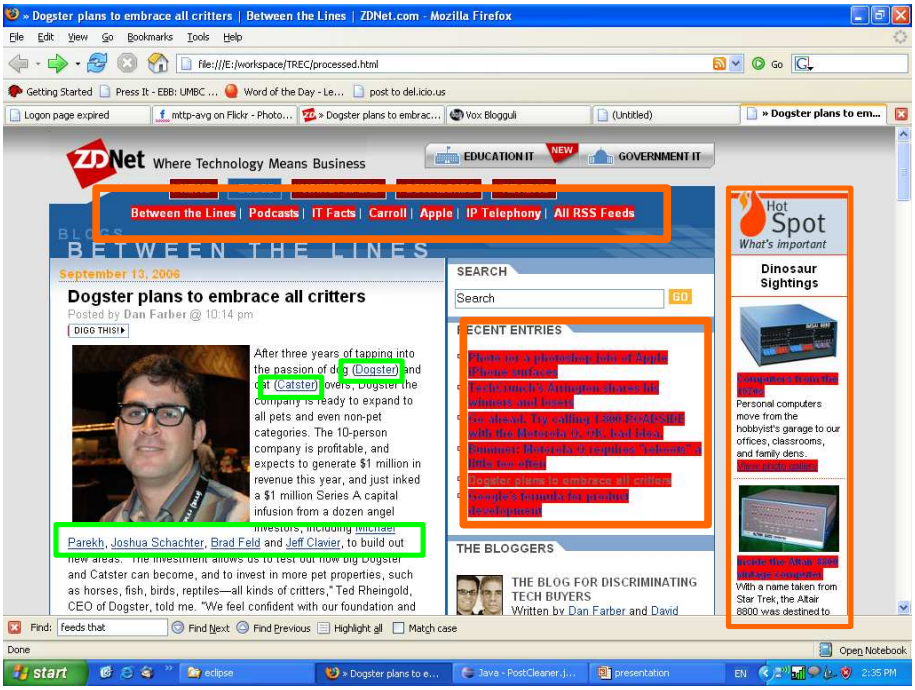


Figure 5.4: A typical blog post containing navigational links, recent posts, advertisements, and post content with additional links in it. Highlighted links are eliminated by the approximation heuristic.

A human evaluator judged a subset of links (approximately 400) from these posts. The links were manually tagged either content-links or extra-links. Each link was associated with a set of features. Table 5.2 summarizes the main features used. Using this feature set an SVM model was trained [4] to recognize links to eliminate. The first set of features (1-7) was based on the tag information. The next set of features (8-9) was based on position information and the final set of features (10-13) consisted of word-based features. Using features (1-7) yields a precision of 79.4% and recall of 78.39%, using all our features (1-13) yields a precision of 86.25% and recall of 94.31% under 10-fold cross validation.

We compared the original baseline heuristic against human evaluators. The average accuracy for the baseline heuristic is about 83% with a recall of 87%.

[4] `http://svmlight.joachims.org/`

Procedure 3 int nearestLinkTag(Nodes[] tags, int pos)

```
minDist = |tags|
textNodes = 0
textLength = 0
title = false;
for all j such that pos − θ_dist ≤ j ≤ pos + θ_dist do
  node = tags[j]
  if j = 0||j = pos||j > (|tags| − 1) then
    continue
  end if
  if node instanceOf TextNode then
    textNodes++;
    textLength += node.getTextLength();
  end if
  dist = |pos − j|
  if node instanceOf LinkNode && dist < minDist then
    minDist = dist
  end if
  if node instanceOf TitleNode then
    title = true
  end if
end for
ratio = textLength / textCount
if ratio > α_avgText||title == true then
  return tags.size()
end if
return minDist
```

5.3 Evaluation

The opinion extraction system provides a testbed application for which we evaluate different data cleaning methods. There are three criteria for evaluation: i) improvements in opinion extraction task with and without data cleaning ii) performance evaluation for splog detection iii) performance of the post content identification.

While we are still awaiting the official TREC evaluation, this section evaluates our splog detection and blog post cleaning performance. In the final version of this paper we will include the precision/recall statistics from the official TREC runs with and without the use of data cleaning.

5.3.1 Splog Detection Evaluation

For now, we evaluate the influence of splogs and post cleaning in the context of search engine retrieval. Given a search query, we would like to estimate the impact splogs have on search result precision. Figure 5.5 shows the distribution of splogs across the 50 TREC queries. The quantity of splogs present varies across the queries since

ID	Features
1	Previous Node
2	Next Node
3	Parent Node
4	Previous N Tags
5	Next N Tags
6	Sibling Nodes
7	Child Nodes
8	Depth in DOM Tree
9	Char offset from page start
10	links outside the blog?
11	Anchor text words
12	Previous N words
13	Next N words

Table 5.2: Features used for training an SVM for classifying links as content links and extra links.

splogs are query dependent. For example, the topmost spammed query terms were 'cholesterol' and 'hybrid cars'. Such queries attract a target market, which advertisers can exploit.

The description of the TREC data [46] provides an analysis of the posts from splogs that were added to the collection. Top informative terms include 'insurance', 'weight', 'credit' and such. Figure 5.6 shows the distribution of splogs identified by our system across such spam terms. In stark contrast from Figure 5.5 there is a very high percentage of splogs in the top 100 results.

5.3.2 Post Cleaning Evaluation

In BlogVox data cleaning improved results for opinion extraction. Figure 5.7 highlights the significance of identifying and removing extraneous content from blog posts. For 50 TREC queries, we fetched the first 500 matches from a Lucene index and used the baseline data cleaning heuristic. Some documents were selected only due to the presence of query terms in sidebars. Sometimes these are links to recent posts containing the query terms, but can often be links to advertisements, reading lists or link rolls, etc. Reducing the impact of sidebar on opinion rank through link elimination or feature weighing can improve search results.

Table 5.3 shows the performance of the baseline heuristic and the SVM based data cleaner on a hand-tagged set of 400 links. The SVM model outperforms the baseline heuristic. The current data cleaning approach works by making a decision at the individual HTML tag level; we are currently working on automatically identifying the DOM subtrees that correspond to the sidebar elements.

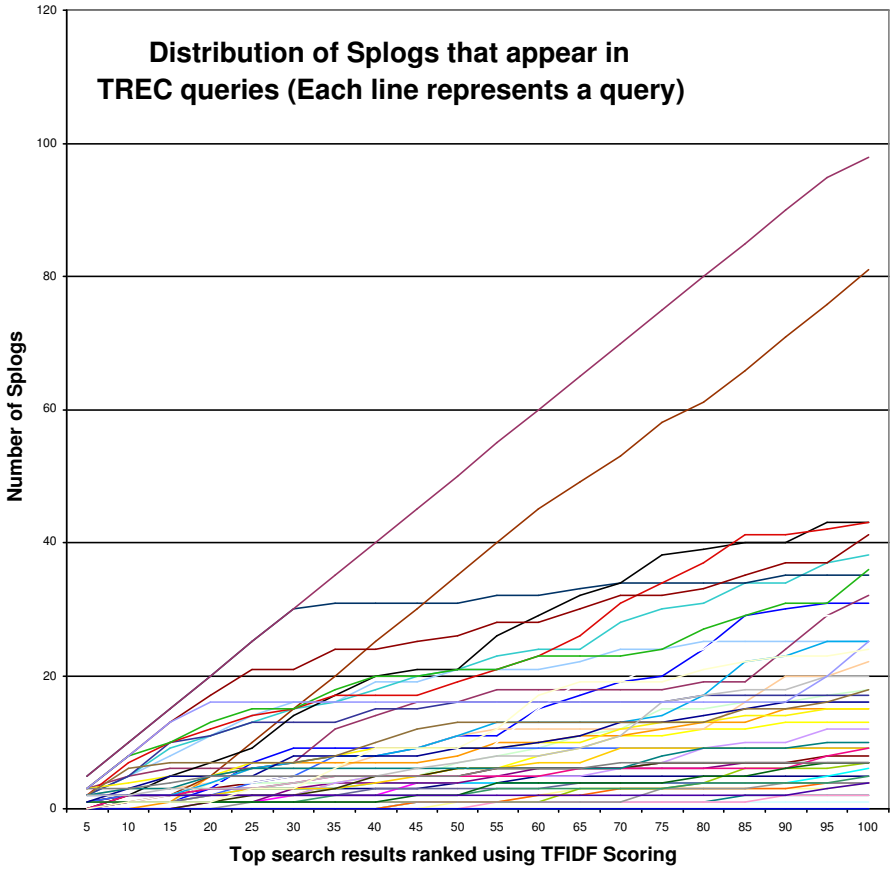


Figure 5.5: The number of splogs in the top x results for 50 TREC queries. Top splog queries include “cholesterol” and “hybrid cars”

Method	*Precision*	*Recall*	*F1*
baseline heuristic	0.83	0.87	0.849
svm cleaner (tag features)	0.79	0.78	0.784
svm cleaner (all features)	0.86	0.94	0.898

Table 5.3: *Data cleaning with DOM features on a training set of 400 HTML Links.*

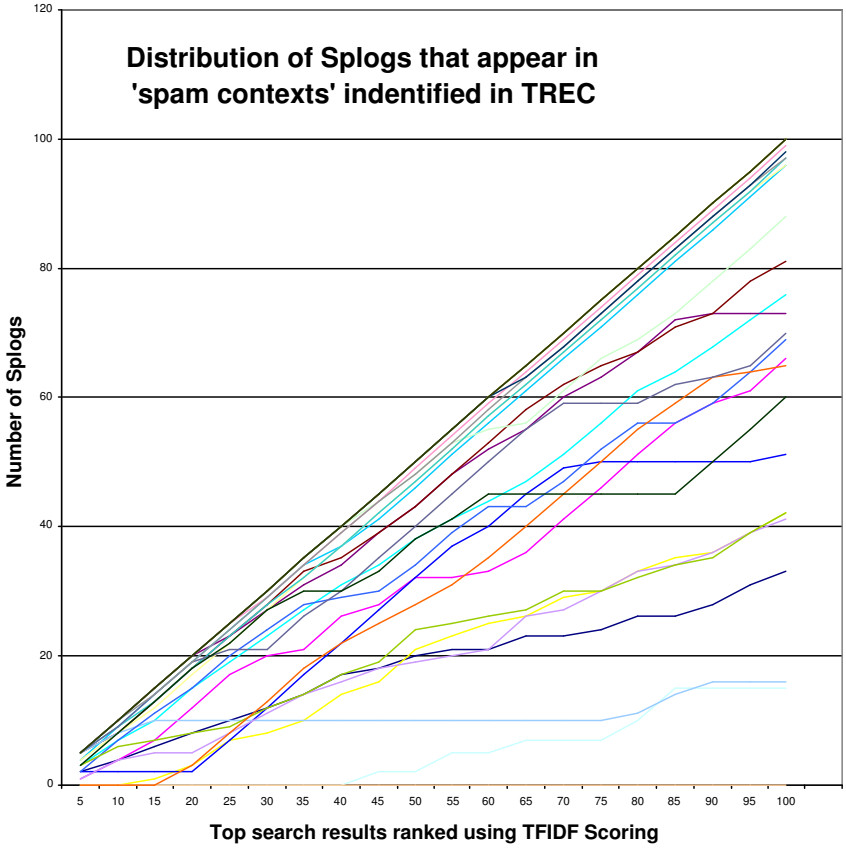


Figure 5.6: The number of splogs in the top x results of the TREC collection for 28 highly spammed query terms. Top splog queries include 'pregnancy', 'insurance', 'discount'

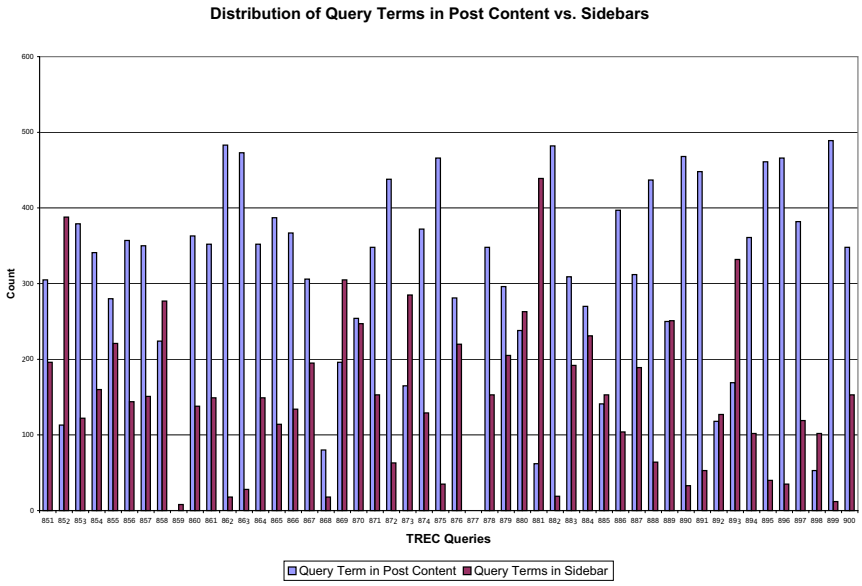


Figure 5.7: Documents containing query terms in the post title or content vs. exclusively in the sidebars, for 50 TREC queries, using 500 results fetched from the Lucene index.

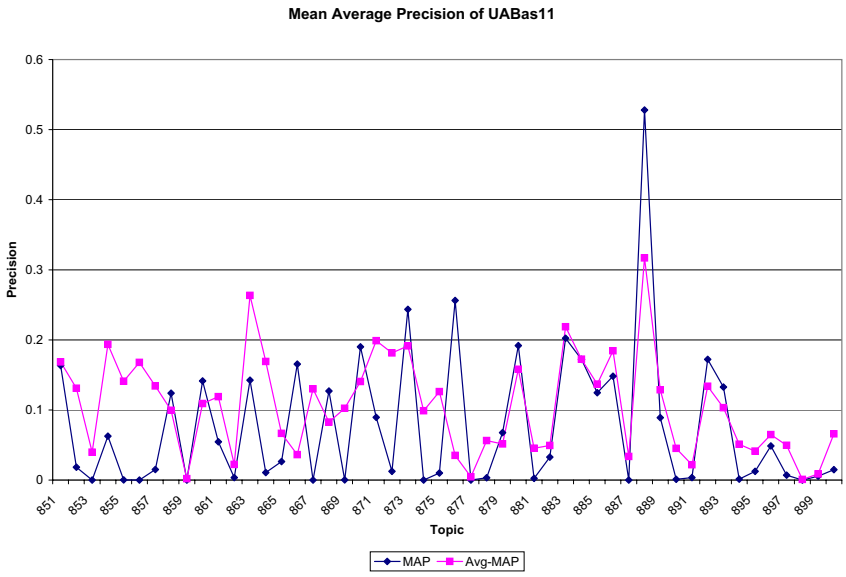


Figure 5.8: Mean average precision of submission UABas11

5.3.3 Trec Submissions

Figure 5.8 shows the results of the best run. We found that while some queries performed well, there were many queries for which the average precision was close to zero. As shown in Figure 5.9, further investigation into the retrieval results for these queries showed that for many of these queries there were problems with the retrieval itself. We think that this is due to some of the following reasons:

- using Lucene in with the default parameters.
- lack of query expansion modules.
- parsing errors encountered for some of the posts while indexing.

5.4 Discussion

This chapter summarized the BlogVox opinion extraction system. We plan to develop and extend the infrastructure to monitor opinions and model bias. Using such a system we would be able to incorporate link polarity. Finally, by monitoring the Blogosphere over time, we will be able to provide a system to monitor the overall sentiment of a community or that of an individual blogger.

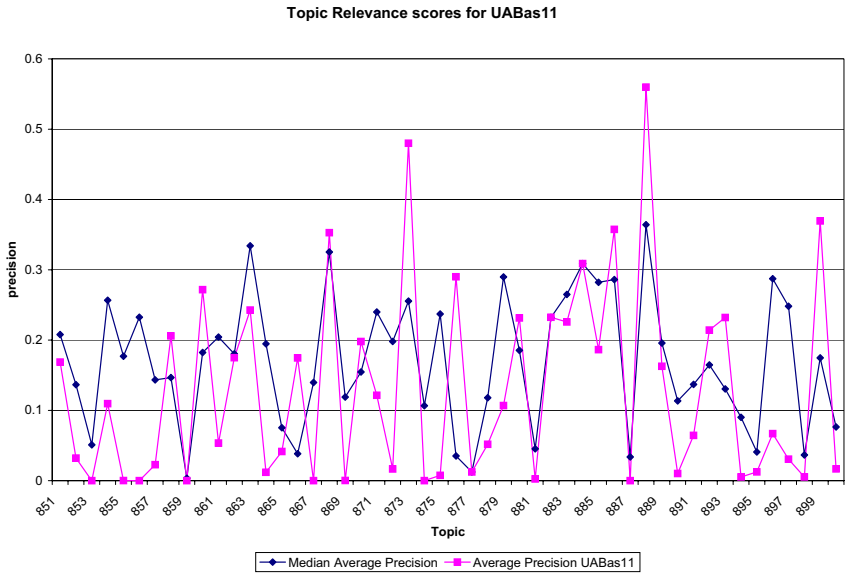


Figure 5.9: Topic relevance of submission UABas11

run	opinion		topic relevance	
	map	r-prec	map	r-prec
UABas11	**0.0764**	**0.1307**	**0.1288**	**0.1805**
UAEx11	0.0586	0.0971	0.0994	0.1367
UAEx12	0.0582	0.0934	0.0985	0.1355
UAEx13	0.0581	0.0923	0.0978	0.1360
UAEx21	0.0590	0.0962	0.0998	0.1366

Table 5.4: The results for the opinion and topic relevance performance of different runs

Chapter 6

Proposed Model and Evaluation Methodology

6.1 Proposed Approach

As part of the background work, in this thesis, we have identified some important components of influence. Still, there are more factors that need to be discovered and understood. There are multiple ways to approach this problem and while it may be too early to pick one over the other, in this section, we discuss a few possibilities for both the model and the evaluation methodology.

6.1.1 Influence Model

Stochastic models have been used for studying various properties of random graphs and scale-free networks. Erdos Renyi [14] proposed a generative, stochastic model for random graphs. On the other hand, most of the real-world networks are not random. One of the typical properties of real-world networks is that any two nodes can be reached in a relatively short number of steps. Duncan Watts and Steven Strogatz [64] first proposed the small-world network model. Using such a model they were able to describe a number of biological and computer networks. Small-world networks are a special case of random graphs and have properties that are in between regular networks and random graphs.

When Barabási et. al. studied the structure of the WWW, they found an interesting property [4] that explains the structure of the Web. They show that as new nodes are added in a network they tend to connect preferentially to existing nodes which have a high degree. Such a property is known as “preferential attachment”. In this approach we treat the Blogosphere as a graph. From our studies in Chapter 3, we know some of the statistical properties of a blog graph. By using computer-generated graphs with similar indegree, outdegree, community structre and preferential attatchment property, we can build an appoximate or simlulated Blogosphere. The advantage of using such an approach is that we can experiment different strategies using a graph of managible

size and known properties.

As seen from Chapter 4, existing epidemic based models of influence make some simplistic assumptions on how influence works. In cascade models, at every timestep, each of the node has a *single* chance of influencing its neighbors. Additionally, the probablity that is neighbor would adopt the opinion or idea is based on a pseudo-random function. In reality however there are a few problems with this assumption. In our proposed stochastic model, we would incorporate the following desired properties that are not described by cascade models:

- Each node should have multiple opportunities to influence its neighbors. Also, such influence propagation can happen *asynchronously*.
- Each node selects a topic vector T indicating its interests. Each of the node can select a positive or negative polarity for its neighbors. Thus incorporating the sentiment of the opinion and the node‘s own bias.
- Every node i is allowed to pick a pair $< w_{i,j}, t_k >$ such that t_k indicates a topic and $w_{i,j}$ is the weight indicating the strength of trust of i on its neighbor j for topic t_k.
- A node is preferentially more likely to be influenced by another node that has influenced it in the past.
- A node is more likely to be influenced by someone more authoritative than a random neighbor.
- A node is more likely to be influenced by other nodes that have picked a similar topic vector.
- Opinions are often localized within the network.

A related approach is the Latané model of social impact [40, 15]. In this model the impact of all nodes on an agent i is given by

$$I_i = (\sum_{j=1,n} \frac{s_j \sigma_j}{d_{i,j}}) + h$$

Where s_i is the strength of the opinion, $\sigma_i \in -1, 1$. $d_{i,j}$ models the node i's physical distance from node j and h is the external influence. The model is further extended to account for social noise. However, this model does not incorporate the social structure of the network. By restricting the node's influence network to its nearest neighbors, based on the structure of the network, [15] extend the latané model for multi-agent systems.

A study by Fang Wu and Bernardo Huberman [66] described a formal theory to model opinions and its evolution through the social network. They show by simulations that the opinions tend to be localized. An interesting observation in this work is that under certain conditions “the expected weighted fraction of the population that holds a given opinion is constant in time” [66]. In this model opinions are modeled as binary values and the influence network is restricted to the nearest neighbors.

Building on some of the related work in the fields of both Computer Science and Social Sciences, we plan to describe and test our models on both simulated graphs that

resemble the Blogosphere and actual blog data using available corpora such as TREC, WWW and ICWSM datasets.

6.2 Evaluation Methodology

Described below are some evaluation techniques and criteria. In some sense, both opinion and influence are subjective. One may or may not trust someone's opinions in food but would consider highly his views on books. In other cases, my tendency to be influenced on politics might be minimally based on my peers, but more directly influenced by main-stream-media. Such individual variations in behavior are difficult to model in an accurate manner and any mathematical model would fall short of predicting correctly the topics on which someone is likely to be influenced. These difficulties make it challenging to evaluate the effectiveness of the proposed approach in the real-world, unless we resort to some human evaluations and empirical results.

One form of evaluation could be to ask human judgements to share what they are influenced by. This could be done by utilizing their blogrolls or their feed subscription OPML information. By taking the user ratings as the ground truth, we can evaluate the effectiveness of our models. An approximation of influence is the information available from the readership data in "Feeds that Matter". The ranking for different topics was obtained purely on the basis of the readership information and no link structure was considered. Now, we can use our proposed model to 'predict' the influence of a given node for a topic. We can now compare this with by treating the readership information as the ground truth. Thus, finding blogs that are influential for different topics. This way we can measure the effectiveness at an individual level.

The next form of evaluation deals with communities. From the experiments in Chapter 3 it has been observed that blogs form communities. Such communities are often around a particular topic. Bloggers in the same community are more likely to link to similar documents. Their perceived values, likes and dislikes can be very related. An analysis of the political Blogosphere by Lada Adamic and Natalie Glance [2] shows that conservative blogs are more tightly connected to each other. They find a distinct divide amongst bloggers based on their political affiliations. By abstraction at a community level, we can track mainstream media sources and their influence on different communities. Links across two polar communities like "Republicans" and "Democrats" might be explained by citations with negative opinions. A pro-republican blogger might link to a pro-democrat but might do so with disagreement. Studying the effects at a community level allows us to in effect measure the overall performance of our models.

Finally, by tracking the blogs over a period of time we can generate a profile of each blog being monitored for the list of topics that generated the most conversations, which are an indicator of its influence. Frequently blog posts are about topics that are of current interest. Combined with blogs, we can monitor News streams to maintain a model of the current event topics. Given the blog graph and the learnt models of topics for wich a blog generates the most conversations, we can model the influence task as a link prediction problem [42]. The influence model can be used to effectively predict if a new post will be linked to by other bloggers. And if so, who are the most likely blogs

that would link to this post.

Chapter 7

Research Timeline

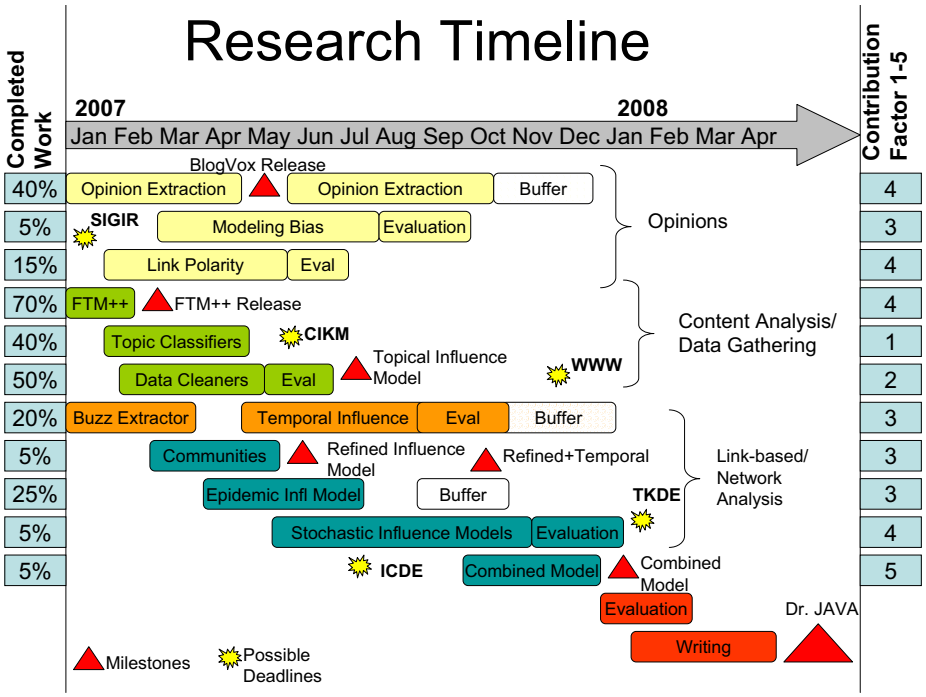


Figure 7.1: *Research Timeline*

The Figure 7 describes the research plan and the corresponding timeline. One of the most immediate milestone is the release of the next version of "Feeds that Matter". This site has created a lot of interest in the blogging community and has had a steady flow of visitors. The next version of FTM++ would have enhanced features like searching and a larger collection of recommended feeds. One possibility is to collab-

orate with Bloglines to obtain more user data. This dataset created would be used in future evaluations and hence is a critical component.

BlogVox was the system developed as part of the TREC conference. We plan to release a beta version of this system on a subset of the blogs, perhaps monitoring only opinions that are expressed on feeds listed in FTM++. Opinion extraction is the core module for modeling both link polarity and bias.

As BlogVox matures, we plan to concentrate our efforts on building the stochastic influence models by incorporating the temporal and structural analysis of the blog graph. The combined model would include each of the subcomponents and the focus starting from the later half of Fall 2007 would be on evaluation and refinement of the model.

While some of the final evaluation is Dec 2007 onwards, we hope that as each of the sub components are ready these would be evaluated in part. Listed in the diagram are only some of the target deadlines, in particular:

- SIGIR: Special Interest Group on Information Retrieval
- CIKM: Conference on Information and Knowledge Management
- WWW: World Wide Web Conference
- TKDE: IEEE Transactions on Knowledge and Data Engineering

Chapter 8

Conclusion

In this work, we have described few techniques to effectively model influence. As Social Media has becomes more prolific, we find that user-generated media is playing an important role in our online experience. It is also a storehouse of information, opinions and reviews. It is possible to extract useful information from such sources. This information can lead to a better understanding of the marke, customers and key players. However such implications are not only limited to marketing and business. The benefits of these techniques for spread social messages are equally promising. As the Blogosphere continues to grow, it becomes important to understand its impact on human communication and role in information propagation. Using techniques proposed in this work we can study influence propagation and opinion formation in such media.

Bibliography

[1] *Efficient Identification of Web Communities*, 2000.

[2] L. A. Adamic and N. Glance. The political blogosphere and the 2004 u.s. election: Divided they blog. In *Workshop on the Weblogging Ecosystem*, May 2005.

[3] E. Adar, L. Zhang, L. A. Adamic, and R. M. Lukose. Implicit structure and the dynamics of blogspace. In *Workshop on the Weblogging Ecosystem*, New York, NY, USA, May 2004.

[4] R. Albert, H. Jeong, and A.-L. Barabasi. The diameter of the world wide web. *Nature*, 401:130, 1999.

[5] C. Anderson. `http://www.thelongtail.com/the_long_tail/2005/12/announcing_the_.html`, 2005.

[6] M. Arrington. Finally! bloglines blog search. `http://www.techcrunch.com/2006/05/31/askcombloglines-launch-blog-search/`.

[7] M. Barbaro. Wal-mart enlists bloggers in p.r. campaign. `http://www.nytimes.com/2006/03/07/technology/07blog.html`.

[8] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *SIGIR*, pages 104–111, 1998.

[9] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. *Computer Networks*, 33(1):309–320, June 2000.

[10] C. H. Brooks and N. Montanez. Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *WWW*, 2006.

[11] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[12] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW*, pages 519–528, 2003.

[13] T. Elkin. Just an online minute... online forecast. http://publications.mediapost.com/index.cfm?fuseaction=Articles.showArticle&art_aid=29803.

[14] P. Erdos and A. Rényi, 1960.

[15] M. E. Gaston and M. desJardins. Social network structures and their impact on multi-agent system dynamics. In *FLAIRS Conference*, pages 32–37, 2005.

[16] N. G. Gilad Mishne. Predicting movie sales from blogger sentiment. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW 2006)*, 2006.

[17] N. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T. Tomokiyo. Deriving marketing intelligence from online discussion. In *KDD*, pages 419–428, 2005.

[18] N. S. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T. Tomokiyo. Deriving marketing intelligence from online discussion. In *KDD*, pages 419–428, 2005.

[19] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins. The predictive power of online chatter. In *KDD*, pages 78–87, 2005.

[20] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. Information diffusion through blogspace. In *WWW*, pages 491–501, 2004.

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

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

[23] S. C. Herring, I. Kouper, J. C. Paolillo, L. A. Scheidt, M. Tyworth, P. Welsch, E. Wright, and N. Yu. Conversations in the blogosphere: An analysis "from the bottom up". In *HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 4*, page 107.2, Washington, DC, USA, 2005. IEEE Computer Society.

[24] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, New York, NY, USA, 2004. ACM Press.

[25] U. P. International. Pentagon reaches out to bloggers. http://www.upi.com/SecurityTerrorism/view.php?StoryID=20060306-020621-7264r.

[26] A. Java, T. Finin, and S. Nirenburg. SemNews: A Semantic News Framework. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06)*, pages 1939–1940, Menlo Park, CA, February 2006. American Association of Artificial Intelligence. AAAI Student Abstract Program.

[27] A. Java, P. Kolari, T. Finin, J. Mayfield, A. Joshi, and J. Martineau. BlogVox: Separating Blog Wheat from Blog Chaff. In *Proceedings of the Workshop on Analytics for Noisy Unstructured Text Data, 20th International Joint Conference on Artificial Intelligence (IJCAI-2007)*, September 2006.

[28] A. Java, P. Kolari, T. Finin, and T. Oates. Modeling the Spread of Influence on the Blogosphere. Technical report, University of Maryland, Baltimore County, March 2006.

[29] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD*, pages 137–146, 2003.

[30] D. Kempe, J. M. Kleinberg, and É. Tardos. Influential nodes in a diffusion model for social networks. In *ICALP*, pages 1127–1138, 2005.

[31] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[32] J. M. Kleinberg. Bursty and hierarchical structure in streams. *Data Min. Knowl. Discov.*, 7(4):373–397, 2003.

[33] P. Kolari. Welcome to the splogosphere: 75% of new pings are spings(splogs), 2005. [Online; accessed 22-December-2005; `http://ebiquity.umbc.edu/blogger/?p=429`].

[34] P. Kolari, T. Finin, and A. Joshi. SVMs for the Blogosphere: Blog Identification and Splog Detection. In *Proceedings of the AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*. AAAI Press, March 2006.

[35] P. Kolari, T. Finin, and A. Joshi. SVMs for the blogosphere: Blog identification and splog detection. In *AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006. to appear.

[36] P. Kolari, A. Java, and T. Finin. Characterizing the Splogosphere. In *Proceedings of the 3rd Annual Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, 15th World Wid Web Conference*, May 2006.

[37] P. Kolari, A. Java, T. Finin, T. Oates, and A. Joshi. Detecting Spam Blogs: A Machine Learning Approach. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006)*, July 2006.

[38] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. On the bursty evolution of blogspace. In *WWW*, pages 568–576, 2003.

[39] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Trawling the web for emerging cyber-communities. *Computer Networks*, 31(11-16):1481–1493, 1999.

[40] B. Latané. The psychology of social impact, 1981.

[41] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (salsa) and the tkc effect. *Computer Networks*, 33(1-6):387–401, 2000.

[42] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 556–559, New York, NY, USA, 2003. ACM Press.

[43] Y.-R. Lin, H. Sundaram, Y. Chi, J. Tatemura, and B. Tseng. Discovery of Blog Communities based on Mutual Awareness. In *Proceedings of the 3rd Annual Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, 15th World Wid Web Conference*, May 2006.

[44] B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the web. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 342–351, New York, NY, USA, 2005. ACM Press.

[45] L. Lloyd, P. Kaulgud, and S. Skiena. Newspapers vs blogs: Who gets the scoop? In *AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006. to appear.

[46] C. Macdonald and I. Ounis. The trec blogs06 collection: Creating and analyzing a blog test collection. Technical report, 2006. Department of Computer Science, University of Glasgow Tech Report TR-2006-224.

[47] C. McEvoy. Bloglines users are a load of knitters. `http://usability.typepad.com/confusability/2005/04/bloglines_user_.html`.

[48] S. Milgram. The small-world problem, 1967.

[49] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *AIRWeb '05 - 1st International Workshop on Adversarial Information Retrieval on the Web, at WWW 2005*, 2005.

[50] G. Mishne and M. de Rijke. Deriving wishlists from blogs show us your blog, and we'll tell you what books to buy. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 925–926, New York, NY, USA, 2006. ACM Press.

[51] M. E. Newman. Scientific collaboration networks. i. network construction and fundamental results. *Phys Rev E Stat Nonlin Soft Matter Phys*, 64(1 Pt 2), July 2001.

[52] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133, 2004.

[53] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69:026113, 2004.

[54] K. Nigam and M. Hurst. Towards a robust metric of opinion. In *Exploring Attitude and Affect in Text: Theories and Applications, AAAI-EAAT 2004*, 2004.

[55] T. O'Reilly. What is web 2.0. `http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html`, 2005.

[56] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[57] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP 2002*, 2002.

[58] M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *KDD*, pages 61–70, 2002.

[59] F. Salvetti and N. Nicolov. Weblog classification for fast splog filtering: A url language model segmentation approach. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 137–140, New York City, USA, June 2006. Association for Computational Linguistics.

[60] R. Scooble and S. Israel. *Naked Conversations: How Blogs are changing the way businesses talk to their customers*. 2006.

[61] D. Sifry. State of the blogosphere august 2006. `http://www.sifry.com/alerts/archives/000436.html`, 2006.

[62] B. Tseng, J. Tatemura, and Y. Wu. Tomographic Clustering To Visualize Blog Communities as Mountain Views. In *Proceedings of the 2rd Annual Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, 15th World Wid Web Conference*, May 2005.

[63] P. D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424, 2002.

[64] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.

[65] B. Wu and B. D. Davison. Identifying link farm spam pages. In *WWW (Special interest tracks and posters)*, pages 820–829, 2005.

[66] F. Wu and B. A. Huberman. Social structure and opinion formation, July 2004.

[67] L. Yi and B. Liu. Web page cleaning for web mining through feature weighting. In *Proceedings of Eighteenth International Joint Conference on Artificial Intelligence, IJCAI-03*, 2003.

[68] L. Yi, B. Liu, and X. Li. Eliminating noisy information in web pages for data mining. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD-2003*, 2003.