Sentimental Bi-Partite Graph Of Political Blogs

by

Dananjayan Thirumalai

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved February 2012 by the
Graduate Supervisory Committee:

Hasan Davulcu, Chair
Hessam Sarjoughian
Arunabha Sen

ARIZONA STATE UNIVERSITY

May 2012

ABSTRACT

Analysis of political texts, which contains a huge amount of personal political opinions, sentiments, and emotions towards powerful individuals, leaders, organizations, and a large number of people, is an interesting task, which can lead to discover interesting interactions between the political parties and people. Recently, political blogosphere plays an increasingly important role in politics, as a forum for debating political issues. Most of the political weblogs are biased towards their political parties, and they generally express their sentiments towards their issues (i.e. leaders, topics etc.,) and also towards issues of the opposing parties. In this thesis, I have modeled the above interactions/debate as a sentimental bi-partite graph, a bi-partite graph with Blogs forming vertices of a disjoint set, and the issues (i.e. leaders, topics etc.,) forming the other disjoint set, and the edges between the two sets representing the sentiment of the blogs towards the issues.

I have used American Political blog data to model the sentimental bi-partite graph, in particular, a set of popular political liberal and conservative blogs that have clearly declared positions. These blogs contain discussion about social, political, economic issues and related key individuals in their conservative/liberal view. To be more focused and more polarized, 22 most popular liberal/conservative blogs of a particular time period, May 2008 - October 2008(because of high intensity of debate and discussions), just before the presidential elections, was considered, involving around 23,800 articles.

This thesis involves solving the questions: a) which is the most liberal/conservative blogs on the web? b) Who is on which side of debate and what are the issues? c) Who are the important leaders? d) How do you model the relationship between the participants of the debate and the underlying issues?

To

thirumalna, renamma, divya, jaithatha, sreenthatha, leelavva

ACKNOWLEDGMENTS

Dr. Hasan Davulcu is the strongest reason behind my successful graduate experience. My association with him dates back to my first semester here at ASU, recommending me to experience my first internship opportunity at Limelight Networks. I would like to thank him for his insightful guidance, help and support. His way of looking research problems have always been inspirational.

I would like to express my gratitude to Dr. Arunabha Sen and Dr. Hessam Sarjoughian and Dr. Tom Taylor for their timely help, valuable guidance and for being on my thesis committee. Dr.Sen's patience of explaining problems in a simple way and Dr.Sarjoughian's belief on students is highly commendable.

I would like to thank Dr. Robert Greenes and my colleagues at Bio-Medical Informatics, ASU for their support over the past one and half years.

I would like to extend my gratitude to members of the CIPS lab – Shreejay Nair, and Sedat Gokalp, who have been my friends and colleagues at CIPS and have always helped and supported me. I would also like to thank my colleagues at CIPS lab for giving me a wonderful experience to work with.

I would like to thank my friends – Iniyan, Adi, Yasar, Ganesh, Praveen, JP, Madhan, Prithvi for making my stay enjoyable at Tempe. I will always be thankful to Raj, Rohit, and Shiva for their invaluable comments. Most of the motivation of my life has come through form of Faheed, Lokesh, Babu, Hareesh, and MP. I am fortunate to have as my backbone, my entire family. I will always be indebted to my parents, sister, grandparents and all my relatives for their support throughout my life, past, present and beyond.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

INTRODUCTION

Blogosphere, made of blogs and their interconnections [1] forming a social network where any author can publish their opinions, is playing a very important role[12] as a forum of public debate with huge impact on the media, politics and policy. Especially the political blogs, they have become a major part of politics[9]. Reports show that lots of Americans are using online blogs/new sites to stay informed about politics. In fact, more than half [54%] of all American adults were online political users (includes users who get political news online, Go online to take part in political activities, use social networking site for political purposes) in 2010[7]. It is also seen that political blogs receive a huge amount of traffic just before the election. Analyzing the impact of blogs, even political influential people and political parties set up blogs during the 2004, 2008 U.S. Presidential elections.

With such an impact of blogs on politics, it becomes interesting to analyze the political text, which contains huge amount of sentiments, opinions towards important issues. In my thesis, I have considered the U.S. political Blogosphere, because the influence of such blogs on political discourse is most prominent in the U.S. politics compared to other countries. One of the major topics for these U.S. blogs is between conservative and liberal view of politics. To be more focused, very popular political liberal and conservative blogs that have clearly declared positions are considered. These conservative/liberal blogs contain discussion about social, political and economic issues and related key individuals. In general,

they express positive sentiment towards individuals whom they share ideologies with, and negative sentiment towards others. And also it is very common to see criticism of people within the same camp, and also support for people from the other camp.

These opposing camps generally debate on political issues (i.e. topics, leaders, organizations etc.). Since these blogs are biased towards their political parties/views, they generally express their positive or negative views on the issues. For example, conservative blogs intuitively should express positive sentiment towards conservative leaders/people, and negative sentiment towards liberal leaders and liberal blogs should express positive sentiment towards liberal leaders and negative sentiment towards conservative leaders.

This kind of interactions between the blogs and their view about the issues (leaders) is very interesting and needs to be modeled in some form of a structure, on which further interesting analysis and research can be made. In general, these kinds of interactions, real time problems are generally modeled as a Graph, an abstract representation of a set of objects where some pairs of the objects are connected by links [6]. In the above interactions, we can notice that, the set of blogs is disjoint from the set of issues (leaders) that they mention. We can make use of this fact and represent it as a bi-partite graph[5], which is a graph whose vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V; that is U and V are independent sets.

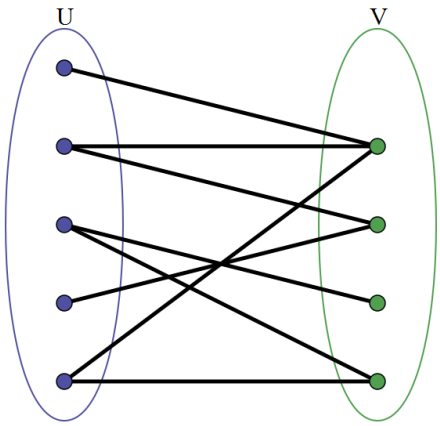

Figure 1. A bi-partite graph

In the above figure, the edges between the two disjoint sets represent the links or relationship between the vertices. I have modeled the debate as a sentimental bi-partite graph (signed bi-partite graph), a bi-partite graph with Blogs forming vertices of a disjoint set, and the issues (i.e. leaders, topics etc.,) forming the other disjoint set, and the edges between the two sets representing the sentiment of the blogs towards the issues. The edges can be either a green edge (positive sentiment or support) or a red edge (negative sentiment or opposition). In summary, my thesis involves the process of converting political blogs and their interactions with their issues to a sentimental bi-partite graph as shown in Fig 2.

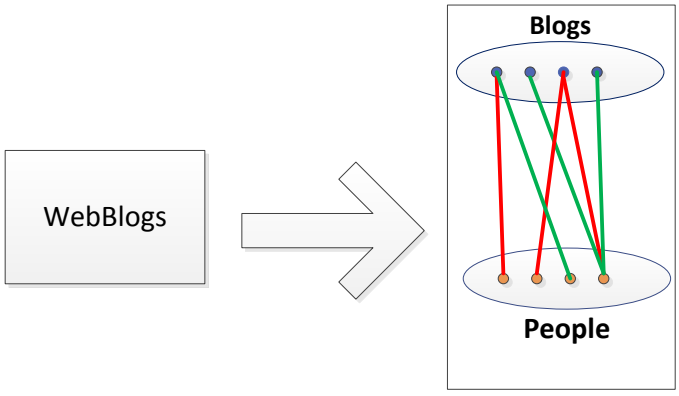

Figure 2. Summary of thesis – From blogs to sentimental-bi-partite graph.

MOTIVATION

With a huge impact of blogs on politics and enormous amount of political text on web, it becomes interesting to analyze the interactions using various data mining techniques. Especially, the interactions between the liberal/conservative blogs and the issues they mention are very interesting and needs to be modeled into a structure to investigate more on the interactions.

One huge motivation of modeling the structure as a bi-partite graph is that the partitioning and scaling of such a structure gives interesting results as shown in [13]. Partitioning the blogs and the underlying issues mentioned in the blogs, partitions the blogs into two opposing camps. Scaling both the blogs and the underlying issues on a uni-variate scale can help identify moderate and extreme blogs within each camp, and polarizing vs. unifying issues. Major motivation for this thesis involves finding the answers for the questions:

a) Which are the most liberal/conservative blogs on the web
b) Who is on which side of the debate and what are the issues?
c) Who are the important leaders?
d) How do you model the relationship between the participants of the debate and the underlying issues?
e) How do you construct the signed bi-partite graph structure model from blogs?

RELATED WORK

There is a considerable amount of research going on about U.S. political blogs. Especially, there is a huge research going on what impact does the blogs have on politics such as [12], [9], and [7]. Some papers have also discussed about the impact of liberal/conservative bloggers and the discussions [15]. [8] discusses about the linking patterns, and degree of interaction and differences between liberal and conservative blogs.

Due to the content of high sentiments involved in the political blogs, there has been research in the sentiment analysis of political blogs. In [14], the authors focus on modeling the sentiments in the blogs centered on Barack Obama, during the presidential elections - 2008. They automate the sentiment analysis using a hybrid of machine learning and logic-based classification techniques. And they use Amazon's mechanical turk to find the sentiment. [17] discusses about using sentiment analysis to predict affiliations using a probabilistic classifier. [14] and [8] uses similar websites as this thesis, but different data.

Collection of the blog data involves information/data extraction techniques from the blog sites. The collection of the data can be challenging because blog-sites can follow a specific template or custom made. [23] extracts blogs independent of the templates using machine learning techniques. [10] extracts blogs based on DOM structure(template dependent).

Bi-partite graphs have been widely used to represent relationships between two sets of entities. Many data mining applications are modeled as bi-partite graphs such as terms and documents in the field of information retrieval,

customers and items purchased in recommendation systems. [11, 18 ,22] also represent entities using bi-partite graphs.

The main motivation of the thesis is to model the structure, to be used for partitioning and scaling [13], where the paper proposes algorithms to solve different problems on signed bi-partite graphs. They partition and scale the bi-partite graphs using spectral clustering and techniques similar to HITS (authority/hubs) ranking in Information retrieval.

Identifying the entities in a text needs NLP techniques, and by itself is a very interesting problem. [20] uses Hidden Markov model for NER(named entity recognition) and Stanford NER uses probabilistic techniques[16]. There are many Parts-of-speech taggers, involving NLP techniques, such as Stanford POS tagger [21]. There are furthermore mining research work involved in political blogosphere, as in [19], which clusters the blog posts.

OVERVIEW OF THE THESIS

Given the impact of blogs on politics, and the interaction of liberal/conservative blogs on various important issues, it becomes interesting to model the interactions of the blogs on various issues (topics, leaders, organizations etc.) I model the above interactions as a sentimental bi-partite graph, as this becomes an interesting structure to perform further analysis like scaling and partitioning [13]. I have primarily considered leaders/politically influential people as issues. So the bi-partite graph consists of one set of blogs and one set of politically influential people. These two sets are disjoint. The interaction between the blog and the

people/leaders is represented as edges. Again, the edges can be of two types. Green edges – represent support for the leader by the blog (positive sentiment). Red edges – represent opposition for the leader by the blog (negative sentiment). This structure is called Sentimental bi-partite graph or Signed bi-partite graph and is shown in Figure 2.

My thesis concentrates on converting the U.S. Political blogs to the above mentioned structure. This involves a sequential process of identifying the important blogs; extracting the text content from the blogs; cleaning the extracted data; identifying the named entities (organizations, cities etc.) and leaders, analyzing the sentiment of the leaders from the blogs, representing the sentiments as a bi-partite graph. The overall process can be depicted as shown in Figure 3. The first step involves finding the important political blogs. I considered finding the important liberal/conservative blogs which have very interesting links and interactions between them. (a) Initially I collected the important blogs from etalkinghead.com and politicalbloglistings.com which were tagged liberal or conservative.(around 150 blog sites – 350,000 articles from 2002 - 2011) (b) And then I considered most influential conservative and liberal blogs and filtered for the blogs which had high U.S. political ranking in Technorati website. (Around 28 blog sites – 530,000 articles from 2007 - 2011) (c) Due to the intensity of the debates I considered temporal articles – articles before the presidential elections 2008 – from May 2008 – October 2008(involves 22 sites – around 23,800 articles)

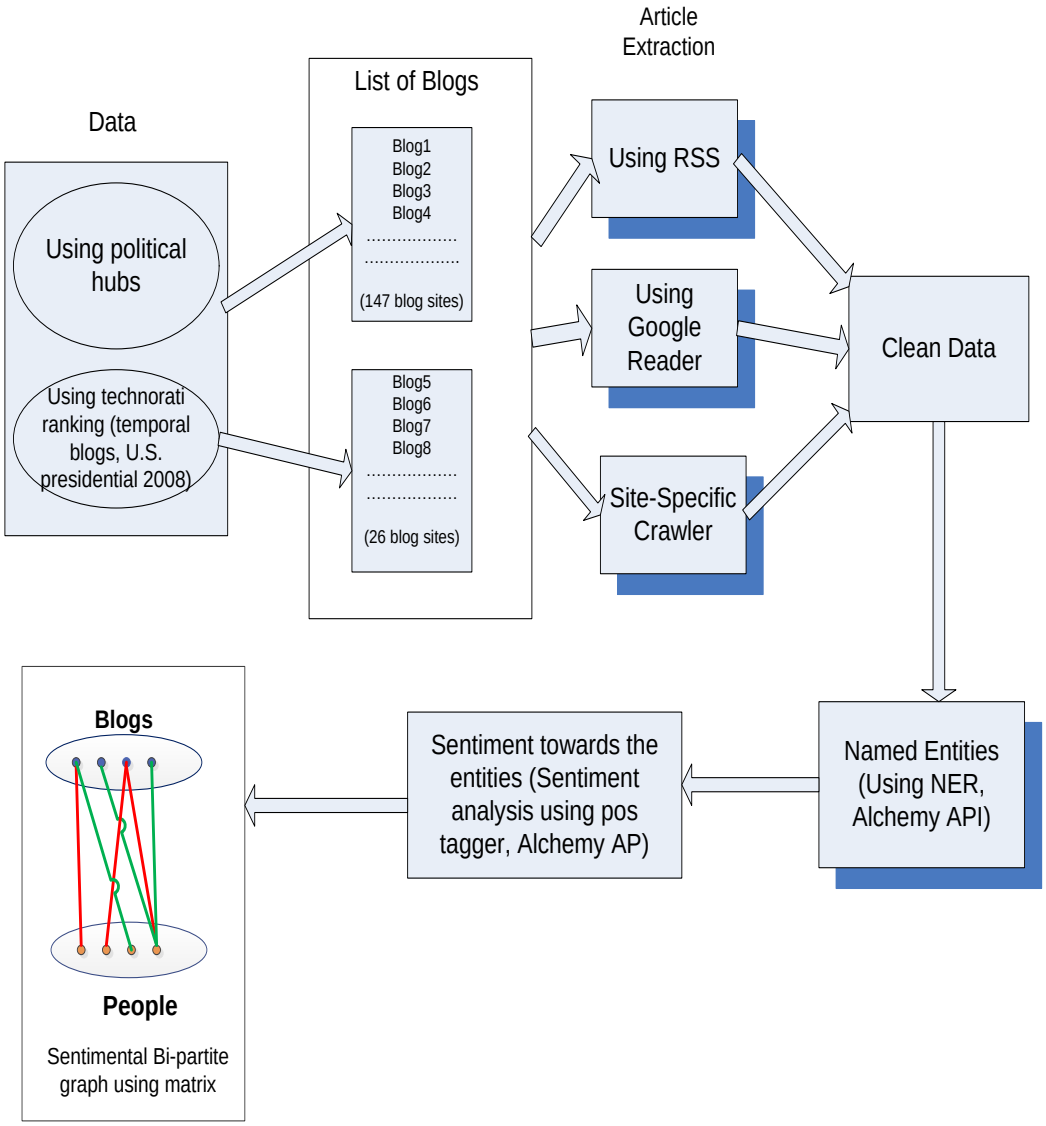

Figure 3. Summary of the thesis

The next step is data extraction process**,** which involves collecting the core blog content articles, without the headers, footers, menus, advertisements, comments, timestamp etc. This step was carried over by 3 techniques: (a) Using RSS feeds and blogger back up tools (b) Google Reader – which maintains an history of all the RSS feeds that users have subscribed for. Both Steps (a) and (b) involved writing xml parsers to get only content of the blog articles. (c) ***Site***

specific crawlers, crawlers which use blog extraction techniques such as regular expression matching, pattern matching etc. The site specific crawlers involved writing a specific regular expression blog extractor for each blog site and this was very efficient in time, compared to traditional crawlers. Most of the blog sites followed a certain template such as wordpress, blogspot, typepad etc. And moreover the blogs followed a specific structure and the posts were sorted by different categories such as date, topics, author name etc.

The next step involves the process of cleaning the collected HTML (CSS, Javascript etc.) data, so that we have only English sentences. This process is important to make sure that, NLP techniques such as ***Named Entity Recognition*** (NER) works properly. Again, the cleaning step involves regular expression matching techniques. The cleaned English articles are then analyzed using NLP techniques (finding the names with capitalization, using ascriptions etc.) to find named entities. I use (a) Stanford NER tagger which uses CRF – probabilistic model to tag the Named entities, such as organizations, persons, cities etc. Then I consider the person entities which was very interesting. (b) Alchemy API to tag the named entities. Alchemy api used xml request/responses to tag the names from a given text. To increase the efficiency in time, we processed the text, in batches like a bag of words, but we lost the accuracy of the sentiment.

The next step is the ***Sentiment analysis*** step, which analyses the sentiment of the blog towards the named entity that we found using the above process. This again used NLP techniques. I used two strategies: (a) Finding the snippets/words before and after the named entities (eliminating stop words) and look for

describing words/ascriptions using POS tagger (Stanford pos tagger). These describing words would mention the sentiment of the blog towards the sentiment. (b) Then we used Alchemy API, to find the sentiments. Both the previous step of finding the person names and finding the sentiment towards the entities can be combined into one step using Alchemy API. As mentioned before, due to the huge amount of date, to increase the efficiency in time, we processed the text, in batches, but we lost the accuracy. The accuracy was improved when there were single article processing at time. So we considered each blog article at once, for the blog posts which had very heavy traffic and high intensity of issues. Then we performed ***Frequency analysis*** of the data, to analyze the quality of the data. I checked if the results contained in blog sites contained politically influential leaders from Liberal/Conservative leaders list from telegraph.co.uk.

Then the final step is the ***synthesis of the bi-partite graph*** from the sentiment-analyzed data. We combine all the sentiment values of all the articles in a single blog towards persons (in the most liberal/conservative people list) and sum it up to obtain an overall score of the entire blog towards persons. Similarly we perform this with all the blogs, to obtain the sentiment scores towards all the persons by all the blogs. And we represent the score values as an adjacency matrix, which is the sentimental bi-partite graph. Then we also perform ***Keyword extraction***, finding the important keywords in the articles along with the sentiment expressed by the blogs. Forming a bi-partite graph using keywords vs blogs is also an important structure, as scaling and partitioning it, gives highly extreme topics on which liberal/conservative people debate. As a summary, I have

first identified the important blogs. Then, for each blog, I collected the blog articles. For each blog, I performed the NER tagging to find the named entities (from all the blog articles in the blog) and perform sentiment analysis to find the sentiment of the blog with her named entities. Hence I have a list of names and their sentiments for each blog. This data is then synthesized as a sentimental bi-partite graph

OUTLINE

The rest of the thesis is organized sequentially according to steps involved. In chapter 2, we discuss about the importance of data, identifying and selecting the important blogs. In chapter 3, we discuss about the various techniques employed to extract the main crux of the blogs – the content of the blog articles. In chapter 4, we discuss about cleaning the data collected and identifying the named entities using different techniques. In chapter 5, sentiment analysis of the named entities using different techniques is discussed and how the data is analyzed using frequency analysis. Synthesizing the above data as a bi-partite graph is discussed in chapter 6, along with the intuitive polarization of the blogs for the keywords. This is followed by results in chapter 7 and finally we conclude in chapter 8 with directions for future work.

Chapter 2

DATA

I have considered the political blogosphere data, because of the huge impact of blogs on politics and the interestingness to analyze the political text. Identifying the blogs, by itself is a challenging process because the data from the blogs collected must be politically polarized and should express their sentiments towards politically influential issues (leaders, organizations, topics etc.) to make the problem of synthesizing the interactions more exciting. In specific, I have considered the U.S. politics blog data, because the influence of such blogs on political discourse is most prominent in the U.S. politics compared to other countries[2]. To be more focused, I have considered the liberal and conservative political blogs, because of the major topics for these U.S. blogs is between conservative and liberal view of politics and also because the interesting interactions and linkages between them [8, 15]. They generally express their sentiments towards their issues. I have considered people/leaders to be the entities which the blogs express their sentiment.

These conservative/liberal blogs contain discussion about social, political and economic issues and related key individuals. In general, they express positive sentiment towards individuals whom they share ideologies with, and negative sentiment towards others. And also it is very common to see criticism of people within the same camp, and also support for people from the other camp. These opposing camps generally debate on political issues (i.e. topics, leaders, organizations etc.). Since these blogs are biased towards their political

parties/views, they generally express their positive or negative views on the issues. For example, conservative blogs intuitively should express positive sentiment towards conservative leaders/people, and negative sentiment towards liberal leaders and liberal blogs should express positive sentiment towards liberal leaders and negative sentiment towards conservative leaders. So the aim of the first step is to find the most important political liberal and conservative blogs that have clearly declared positions. This step becomes important because, the rest of the process is based on the initial blogs that we collect. I use different strategies to collect the important liberal and conservative blogs list and they are mentioned as follows.

BLOGS FROM POLITICAL HUB SITES

The first strategy is to collect the list of blogs from important political hub sites. Due to the importance of the political hubs, it must be pointing to a list of important political blogs. We got the list of blogs from two websites namely – etalkinghead (http://directory.etalkinghead.com/) and political Blog listings (http://politicalbloglistings.blogspot.com/), which were labeled as conservative or liberal. These sites were basically political blog directories, with a list of blog sites, classified as Independent, liberal, moderate, religious, liberal, conservative, libertarian etc, and had a snippet with a small description of the blogs. We considered only conservative and liberal blogs. There were around 450 distinct liberal/conservative blogs from both the hubs. Many were outdated, or with very few articles. Eliminating them, we had around 147 blog sites, with around

350,000 articles. All the blogs were posted between 2002 and 2011. A small snippet of the blog list is shown in Figure. 4; the entire 147 blog list (with conservative/liberal label and the period which the blogs cover) is in appendix A.

	BLOG	URLS	Liberal/Conservative	Year
2				
3	The Abercrombie View	http://abercrombieview.blogspot.com	Conservative	2008-2010
4	apologiesdemanded	http://apologiesdemanded.blogspot.com	Conservative	2003-2011
5	The Absurd Canadian	http://absurd-canadian.blogspot.com/	Conservative	2004-2006
6	A Good Choice . . . for Ohio	http://agoodchoice.blogspot.com/	Conservative	2006-2011
7	Eschaton	http://eschatonblog.com/	Conservative	2002-2011
8	Belmont Club	http://belmontclub.blogspot.com/	Conservative	2003-2005
9	Junto Boyz	http://bernardmoon.blogspot.com	Conservative	2003-2011
10	Bill's Comments	http://billscomments.blogspot.com/	Conservative	2004-2011
11	Blog Hogger	http://bloghogger.blogspot.com/	Conservative	2003-2007

Figure 4. Snippet of list of blogs collected from political hub sites.

BLOGS WITH HIGH TECHNORATI RANKING

The previously obtained blog list was huge involving around 147 websites. Few of the sites were not very influential. To obtain a better quality of data, with very polarized sentiments, we considered most influential blog sites mentioned in few political blogs and checked the importance of the blog site using Technorati – a blog search engine.

The technorati search engine ranks the blog sites based on importance (the number of links pointing to this page – link analysis) and similarity of the query. Technorati looks at tags that authors have placed on their websites. These tags help categorize search results, with recent results coming first. It rates each blog's "authority," the number of unique blogs linking to the blog over the previous six months. Such a data set got using technorati is very interesting because it contains

text-based discussions on politics and hyperlinks between blog posts. The hyperlink can be thought of as a social network through which different bloggers get aware of each other and shows how information passes in the political blogosphere. While in the previous method, we considered in the sites mentioned in the important political hubs, here we consider blog sites which are important political authorities.

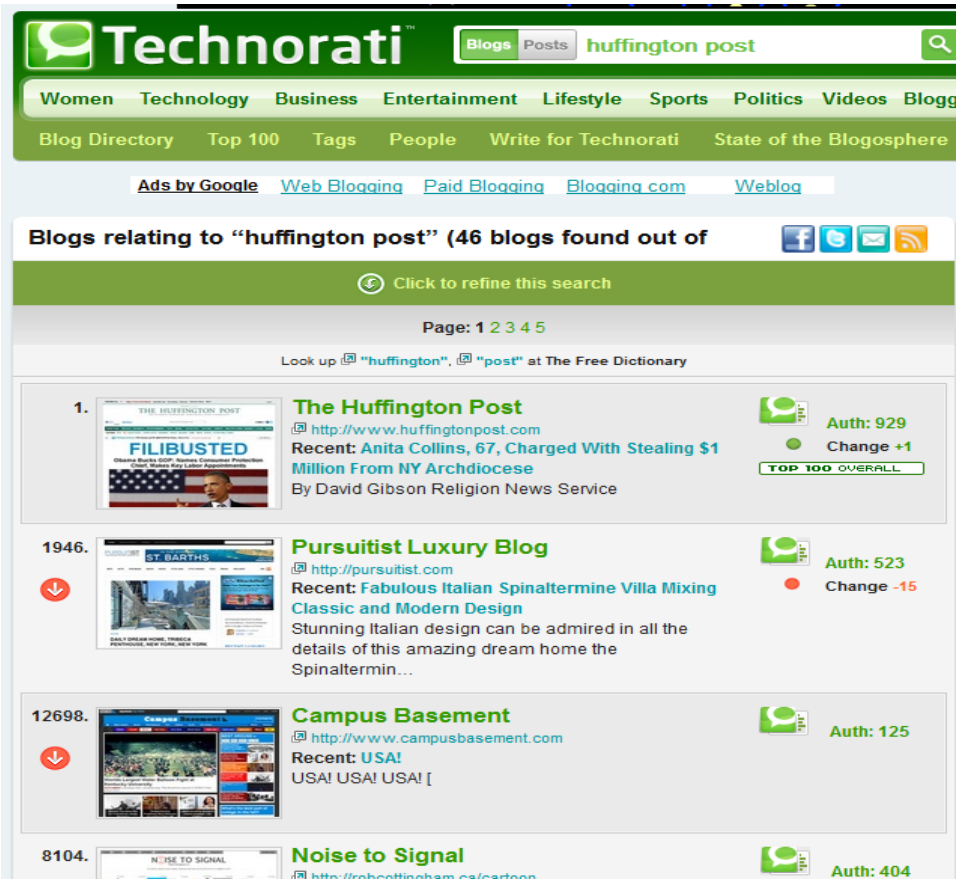

Figure 5. Search results of the blogs in technorati.com

The auth on the right of Figure 5 shows the authority value - the number of unique blogs linking to the blog over the previous six months. And each site has ranking based on what kind of sites point to this blog site. In figure 6, we can see the www.huffingtonpost.com website has different ranking for different topics – Entertainment, politics, living, U.S. Politics etc. We took the list of influential blogs from http://bengrivno.wordpress.com/2009/05/06/top-20-most-influential-conservative-blogs/ and http://technorati.com/blogs/directory/politics/uspolitics/ and only considered the blogs which had high U.S Politics ranking (above U.S. politics rank: 500). This was around 26 blogs – around 14 were conservative and 12 were liberal.

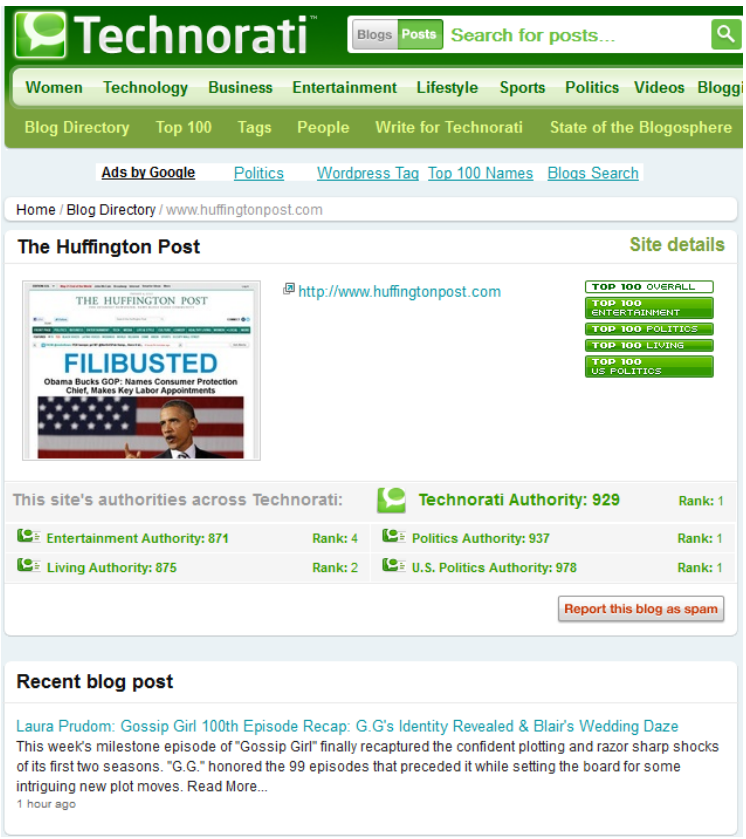

Figure 6. Description of blog "huffington post" with different category rankings

TEMPORAL BLOG DATA

For the above collected data, to consider the most recent blog articles, we picked the sites which have blog articles from 2007 to 2011. This included all the 26 blogs with more than 500,000 articles, which was again huge. The list obtained is shown in Table 1.

Table1

List of blogs collected using technorati ranking within time range 2007 - 2011

Blog name	view	Technorati rank
Huffington Post	liberal	1
Think Progress	liberal	3
Daily Kos	liberal	6
Crooks and Liars	liberal	20
Digby's Hullabaloo	liberal	25
Balloon Juice	liberal	41
Firedoglake	liberal	43
AMERICABlog	liberal	45
Informed Comment	liberal	55
Boing Boing	liberal	79
Truthdig	liberal	120
Talking Points Memo (TPM)	liberal	145
Wonkette	liberal	341
Red State	conservative	5
Michelle Malkin	conservative	10
Hot Air	conservative	19
The volokh conspiracy	conservative	23
News Busters	conservative	29
Reason magazine/Hit and run	conservative	31
Ann Althouse	conservative	70
Atlas shrugs	conservative	109
Stop the ACLU	conservative	130
American Thinker	conservative	142
Pajamas media	conservative	213
Little green footballs	conservative	299
Wizbangblog	conservative	421

U.S. PRESIDENTIAL ELECTION DATA – 2008

In spite of reducing the blog corpus within time range 2007 – 2011, we had a huge amount of blogs. During the NER phase and Sentiment analysis phase, the time taken to find the named entities and the sentiment was very high. Due to high volume of blogs, we tagged a chunk of articles together to find the names and the sentiments, instead of sending one blog article together. This reduced the accuracy of the NER tagger and sentiment analysis API. We wanted to still reduce the time frame to decrease the time taken, but the polarization of the articles should also be present. After experiments of partitioning and scaling the final bi-partite graphs as results in[13], we considered the blog articles 6 months before the U.S. presidential election 2008. This had high intensity of debates and discussions. During the period, the blogs had clearly declared positions. The list of blogs is show in the results section of the thesis. This contained 22 blogs with around 23,800 articles. This data set had a better quality, both in the polarizations of topics as well as better NER and frequency analysis phases.

INFLUENTIAL CONSERVATIVE/LIBERAL PEOPLE LIST

The blog data list that we collected wanted to represent highly politically influential people. Since these were liberal/conservative blogs, we cross checked if the persons mentioned in the blogs were present in the top most conservative and liberal people (using frequency analysis). The list of most influential conservative/liberal people is got from:

- Most influential liberal/conservative people for the year 2007 http://www.telegraph.co.uk/news/uknews/1435447/The-top-US-conservatives-and-liberals.html

- Most influential liberal/conservative people for the year 2010 http://www.telegraph.co.uk/news/worldnews/northamerica/usa/6951961/Top-100-US-liberals-and-conservatives.html

Chapter 3

DATA EXTRACTION

Once the data is decided, the next step is to extract the blog articles. This is a very important process as it involves getting only the crux of the blog – the articles. It should not contain the headers, footers, menus, advertisements, comments, timestamp etc. The role of collecting only the content involves reasons such as (a) the menus, headers, advertisements etc. are not relevant to the actual content. These may be totally irrelevant to the political text, making the data to be noisy. (b)The superfluous content can cause complications for the NLP techniques used in NER tagging, POS tagging and sentiment analysis etc. which are used in later stages. This may cause the accuracy of the NLP techniques to go down.

Blog sites, in general display the content (web pages) in templates. They use standard templates to display the advertisements, the headers, the menus and the actual content. Most of the blogsites uses blog hosting websites such as wordpress, blogspot, typepad sites etc. and uses the standard templates/themes provided through the sites. There are custom blogs which do not follow standard templates, they use their custom design to display the blog posts. The good thing with blogs, using a standard template or a custom template, is that they are highly structured, and we know where the main blog post is going to be inserted on the webpage. And most of the blogs provide few/all articles through RSS feeds. In this work, we consider 3 methods, to collect the blog articles. Whenever blogs provide all the articles through RSS, we use RSS reading mechanisms to get the articles; when they do not, we use site specific crawlers, which uses Blog

extraction techniques based on the structure of the web page. The mechanisms are discussed as follows.

USING RSS – FOR BLOGGERS TEMPLATE

RSS is a family of web feed formats used to publish frequently updated works – such as blog entries, news headlines etc. Mostly, all the blogs, because of the frequent updates of new posts, generally publish through RSS feeds. They generally, provide the most recent few articles through their RSS feeds. And generally the RSS feeds are either in XML format / ATOM format through, which we can read the articles through RSS readers. We can see the sites provide the RSS updates through the above formats in URLS which are generally mentioned in their HTML source as shown in Figure 7.

```
<link rel="alternate" type="application/atom+xml" title="The Abercrombie View - Atom" href="http:
<link rel="alternate" type="application/rss+xml" title="The Abercrombie View - RSS" href="http://
<link rel="service.post" type="application/atom+xml" title="The Abercrombie View - Atom" href="ht
```

Figure 7. HTML code snippet with URLs mentioning the RSS source links.

The websites which uses blogger templates (blogspot.com templates), exceptionally contains all the blog posts posted in the RSS feed. These sites generally have www.sitename.blogspot domain name and their feed posts are available through the link www.sitename.blogspot.com/feeds/posts/default. There is a tool called Blogger backup[5] (shown in Figure 8), an RSS reader, which reads the RSS feed and saves the entire article in XML format.

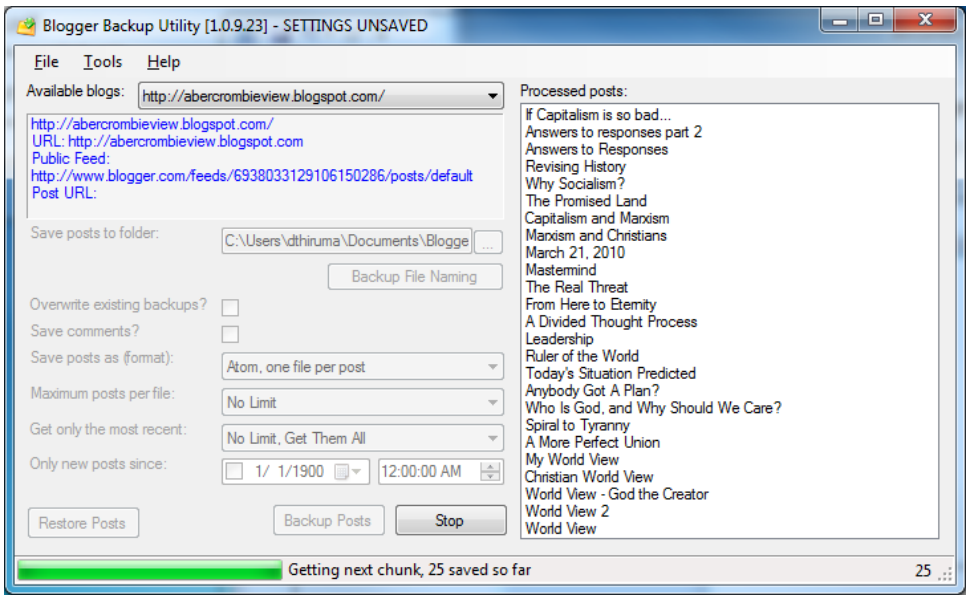

Figure 8. Screenshot of Blogger Backup utility, saving the posts.

Once the feed url is given and start collecting the RSS feeds, the blogger backup utility can store each blog post into a separate XML file as shown in Figure 9.

```
<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom">
    <id>tag:blogger.com,1999:blog-5406449.post-107656063547326985</id>
    <link type="application/atom+xml" rel="edit"
        href="http://www.blogger.com/feeds/5406449/posts/default/107656063547326985"/>
    <link type="application/atom+xml" rel="self"
        href="http://www.blogger.com/feeds/5406449/posts/default/107656063547326985"/>
    <link type="text/html" rel="alternate"
        href="http://contrapositive.blogspot.com/2004/02/padilla-watch-in-calculated-pr-
        gambit.html"/>
  - <author>
        <gd:extendedProperty value="04604614962689284447" name="OpenSocialUserId"
            xmlns:gd="http://schemas.google.com/g/2005"/>
        <name>Contrapositive</name>
        <email>noreply@blogger.com</email>
        <uri>http://www.blogger.com/profile/08645775885446988732</uri>
    </author>
    <content type="html"><b><font size=4>Padilla Watch</font></b> In a calculated PR
        gambit, the Pentagon will now <a href="http://www.reuters.com/newsArticle.jhtml?
        type=domesticNews&storyID=4339055" target=_blank>allow</a> Padilla to see a
        lawyer.<div class="blogger-post-footer"><img width='1' height='1'
        src='https://blogger.googleusercontent.com/tracker/5406449-107656063547326985?
        l=contrapositive.blogspot.com' alt='' /></div></content>
    <updated>2006-11-14T09:06:54-07:00</updated>
    <published>2004-02-11T21:37:00-07:00</published>
</entry>
```

Figure 9. Snapshot of XML file of each blog post collected using Blogger Backup

The xml file contains a blog entry with all the properties such as author, the date it got published etc. The actual content of the blog is in the <content> tag, which has the content in html (with type attribute html). Few blogger users, provide only the summary and the URL to the actual post instead of the content. In that case, the summary is in the <summary> tag. Once all the posts have been collected, we need to get the actual content from the XML, which is done by writing an xml parser, written using C#.NET. What the parser does is, goes through all the xml files sequentially, and gets the content within the <content> tag, which matches the xpath(//entry/content). Then it saves only the content (in html format) into separate files. We use this strategy for blogspot.com templates since they publish all the blog posts through RSS.

USING GOOGLE READER

The problem with the above strategy is that only few blog hosting websites provide the entire blog articles through RSS. Most of them provide only recent articles through RSS, which is insufficient for our needs. In those cases, we can use Google Reader[3] – a web-based aggregator, capable of reading Atom and RSS feeds online. Google reader provides a good interface where you can subscribe to several RSS feeds and get the news/articles. Google reader, not only acts as a feed reader, it also acts a platform for archiving the feeds. This means that it stores all the posts from the subscribed feeds , If you visit a blog or a news site, the feed will only contain the latest 10-20 posts, Google reader can show that history of feeds that were subscribed.

We can get all the blog posts, stored in the google reader through an URL,

http://www.google.com/reader/atom/feed/FEED_URL?r=n&n=NUMBER_OF_ITEMS

Where, the FEED_URL is the url of the feed/RSS

NUMBER_OF_ITEMS is the number of blog posts that you would like to see.

One such example url is

http://www.google.com/reader/atom/feed/http://feeds.feedburner.com/redstate/?r=n&n=2147483647,

Where, http://feeds.feedburner.com/redstate is feed url for www.redstate.com and,

n = 214748 is the maximum number of blogs google reader can extract.

This URL returns all the blog posts in a single XML file as shown in Figure 10. Each blog entry is in the <entry> tag. And each <entry> tag has all the information such as published date, updated date, author, title etc. The actual content of the blog post is contained in the <summary> tag (with type="html" attribute). In order to get only the content of all the posts, again an XML parser using C#.NET is written, which reads the entire XML document and gets all the inner text of XPath – //entry/summary and iterates through all the entries and stores it into different files.

Figure 10. Snapshot of XML file of all the posts collected using Google Reader

SITE SPECIFIC CRAWLER

Using Google Reader to get all the blog posts faces few problems: (a) Google Reader stores only the RSS feeds which were subscribed the users. If there were not subscribed they would not have been stored. (b) The blog posts are quite random and Google reader does not take pain to check if it contains all the blog posts. And few feeds just give a description rather than the entire content.

In the cases where the RSS (using the previous two methods) doesn't work, we need to extract the articles through some other means. We can extract all the content of the website through crawlers, which finds the URLS linked in page. But these traditional crawlers are very time consuming. Most of the blog

sites maintain some kind of a structure (DOM structure – Document object model) and follow certain standard templates. Blog hosting sites such as wordpress, typepad, blogspot etc. provide various themes/templates based on which users can post the articles. We can make use of this structure, because the structure tells us, where the header is going to be, where the advertisements are going to be, where the articles are going to be etc., as part of the page. There are many sites which follow non-standard templates, but even those sites have a proper structure.

For example, let us consider www.stoptheaclu.com, the website looks like as in Figure 11. We can see the site follows some structure throughout all the pages. The section 1 in the image is the header, the section 2 contains menus to navigate within the website, section 3 contains advertisements and section 4 contains important links for the website and section 5 contains the actual article. We can make use of this structure and extract only the main content in section 5. Any web page would be in HTML(includes CSS, javascript), and we can find the structure in HTML which corresponds to different section. We need to the find the HTML code which corresponds to section 5 and extract the data. A program which extracts the data from a webpage using its structure is called a wrapper. This extraction process can be done in two distinct phases:

(a) Analysis phase:

(i) Conceptual modeling: the process of identifying the part of the document which is of interest. In our case its section 5.

(ii) Extraction plan: In this step, we decide how to extract the pattern. This can be done by regular expressions, which matches string by patterns.

(b) An automated implementation phase: The wrapper (program) which automatically fetches the required part/parts of the document.

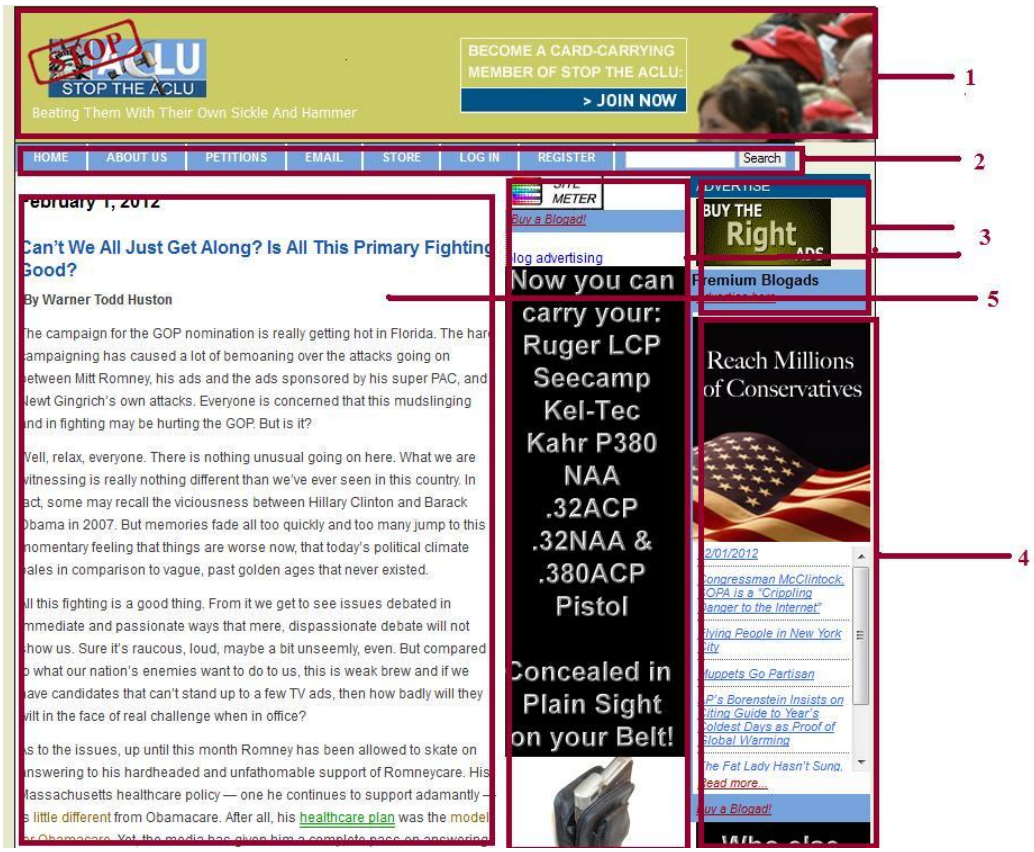

Figure 11. Snapshot of the website www.stoptheaclu.com with numbers indicating section numbers.

Analysis phase. In analysis phase, to find the exact section 5, we can use a developer tool called Firebug which is an Add-on in Firefox Browser. You can actually view the HTML code in a window, when you select the text in HTML as

shown in Figure 12. One way to exactly find the section 5, is to get the entire page as HTML and perform a pattern matching/regular expression of the HTML code corresponding to section 5.

Automated implementation phase. Using the HTML code which just contains the article, we need to form a generic regular expression which matches the section 5 text without, the date, author name, and separator from another article, comments etc. A separator has to be used to make sure that regular expression does not match more than the core article text that we want. Guidelines on using the separator and how important it is were referred from [10].

Figure 12. Usage of Firebug tool to extract only the blog article content.

The regular expression used in this case was: ***</h1> (.*?)<a rel=\"nofollow\"*** where, (.*?) is the actual text group that we are interested in.

The group was preceded by the sub-expression </h1> which refers to end of h1 tag and start of p tag – which refers to the text after ***By Warner Todd Huston***. The author name was in h1 tag and the actual text started with p tag. The group was followed by <a rel=\"nofollow\"> which referred to a link just after the blog article.

The overview of the HTML code was: <h1> By Warner Todd Huston</h1> The text <a rel="nofollow">.

The code for the site-specific crawler (a wrapper for each website) was written using Java which performs the regular expression matching. Apart from the structure of the article inside the page, the site specific crawlers made use of the organization of the blog articles (or pages itself). Generally all the blog articles are archived / organized in different ways. The articles are sorted by time, or organized by the authors who have written it, or organized by the topics they represent.

In our stoptheaclu.com case, the blogs were sorted by the decreasing order of the time. For example,

http://www.stoptheaclu.com/page/1/ contained the most recent blog posts. http://www.stoptheaclu.com/page/2/ contained little older blog posts.

…

http://www.stoptheaclu.com/page/1200/ contained the oldest blogs(last page)

The site-specific crawler got the HTML content through all the pages and does a regular expression matching and gets the content. In case of temporal blog data, blog articles within a time range, we need to consider pages within that time range. We performed some strategies like binary search - O(log n) time - to find the blogs which were posted between the upper time limit and the lower time limit. In case of the blogsites which did not sort according to the time, we performed another regular expression matching for the date posted and extract the posts only within the time-range. Few websites did not have the entire content on the main page of the blog itself. Instead, the links to the article and short description of the articles are given in the main page. In those cases, we initially crawled using regular expression matching to get the links alone and then perform the regular expression matching to get the entire content on those links.

Vs. Traditional crawlers. There are advantages of using site-specific crawlers as:

(a) The site-specific crawlers were very faster compared to traditional crawlers. It took approximately 2 or 3 days just to collect the urls specified within the blogs using tradition crawler. Whereas, with site specific crawlers it took around 3 hours to crawl and extract the entire blog post urls along with the content. The time taken by traditional crawler and a site-specific crawler can be shown in a graph as in Figure 13.

(b) Apart from the time taken, the traditional crawlers have another drawback of crawling the same blog post several times. Each post can be contained

in different categories and can have different URLs (more than one) even though the content is same. Traditional crawlers do not find that if several URLs point to the same blog posts. With site-specific crawlers we can consider only the unique links, without un-necessarily crawling the same page.

Chapter 4

DATA CLEANING AND ENTITY TAGGING

DATA CLEANING

The next step involved is cleaning the HTML data collected in the previous step. It involves removing the HTML (css, JavaScript) code and get the required text in English sentences/paragraphs only. English paragraphs usually have a high density of sentences with auxiliary verbs[4].

This role of content filtering involves reasons such as (a) the html content is only used for displaying it for the browser. It is not relevant to the actual content itself (b)The superfluous content can cause complications for the NLP techniques used in NER tagging, POS tagging and sentiment analysis etc. which are used in later stages. This may cause the accuracy of the NLP techniques to go down. The Data cleaning is done using pattern matching and regular expression techniques.

The regular expression which matches html tags and code are stripped off. We used regular expression similar to this /<(.|\n)*?>/g, which matches all HTML tags pairs including attributes in the tags. The data cleaning program was also written using Java, which reads all the files with the blog content in HTML, performs regular expression matching, and replaces the html tags with an empty space, and saves the articles into new files.

ENTITY TAGGING

Once we have collected all the blog data with, only English sentences, the next step is to analyze the text for finding Named Entities. Named Entity Recognition (NER) is a subtask of information extraction that seeks to locate and classify atomic elements in text into predefined categories such as names of the persons, organizations, locations, quantities etc. It uses grammar based techniques, and other NLP (natural language processing) techniques to tag the entities. We have used NER to tag the entities, and we consider only person names. We have used two strategies for tagging the entities, they are as follows:

Using Stanford NER. Stanford NER is an implementation of NER using Conditional Random Field(CRF) sequence models, coupled with very good feature extractors for NER. The Stanford NER contains 3 class (PERSON, ORGANIZATION, LOCATION) named entity recognizers for English. We used the PHP implementation of the Stanford NER to find the entities of the articles (with a open Calais key). And we discarded the entities with class type location or organization. We considered only the class type Person.

Our php program opens the articles one by one, finds the entities with class type Person and saves it into another file along with index of the first character of the entity in the text file itself. The first character of the text of article starts with 0. By using the index positions, we were able to extract 5 words before and 5 words after the named entity using space separators(white space or blank space characters). We extracted this to analyze the sentiment expressed

towards the entity, which is explained in the next chapter. An example output of tagging the text using Stanford NER looks as follows:

<pos> FG </pos> <pos> asks, </pos> <pos> should </pos> <pos> President </pos> <pos> </pos> <person> Bush </person> <pos> expand </pos> <pos> the </pos> <pos> government </pos> <pos> spending </pos> <pos> in </pos>

The <pos> tags contain the words which occur before and after the named entity(Bush). The <person> tag contains the name of the person. Each person is preceded by 5 words(5 <pos> tags) and followed by 5 words. The output is strucutured in the above format using tags for two reasons:

1. To differentiate between the words which occur around the entity and the named entity itself. This structure can be useful for sentiment analysis, used in the later phase.
2. Easier for parsing the person names in frequency analysis.

Using AlchemyAPI. AlchemyAPI[4] – is an online api – that utilizes statistical natural language processing technology and machine learning algorithms to analyze the content, extracting information about people, places, companies etc. They provide REST based API service, through different languages and the extracted meta-data may be returned in XML, JSON, RDF formats etc. They have their libraries in different languagues like c++, java, C#.net etc. We have used the Java version of the API library.

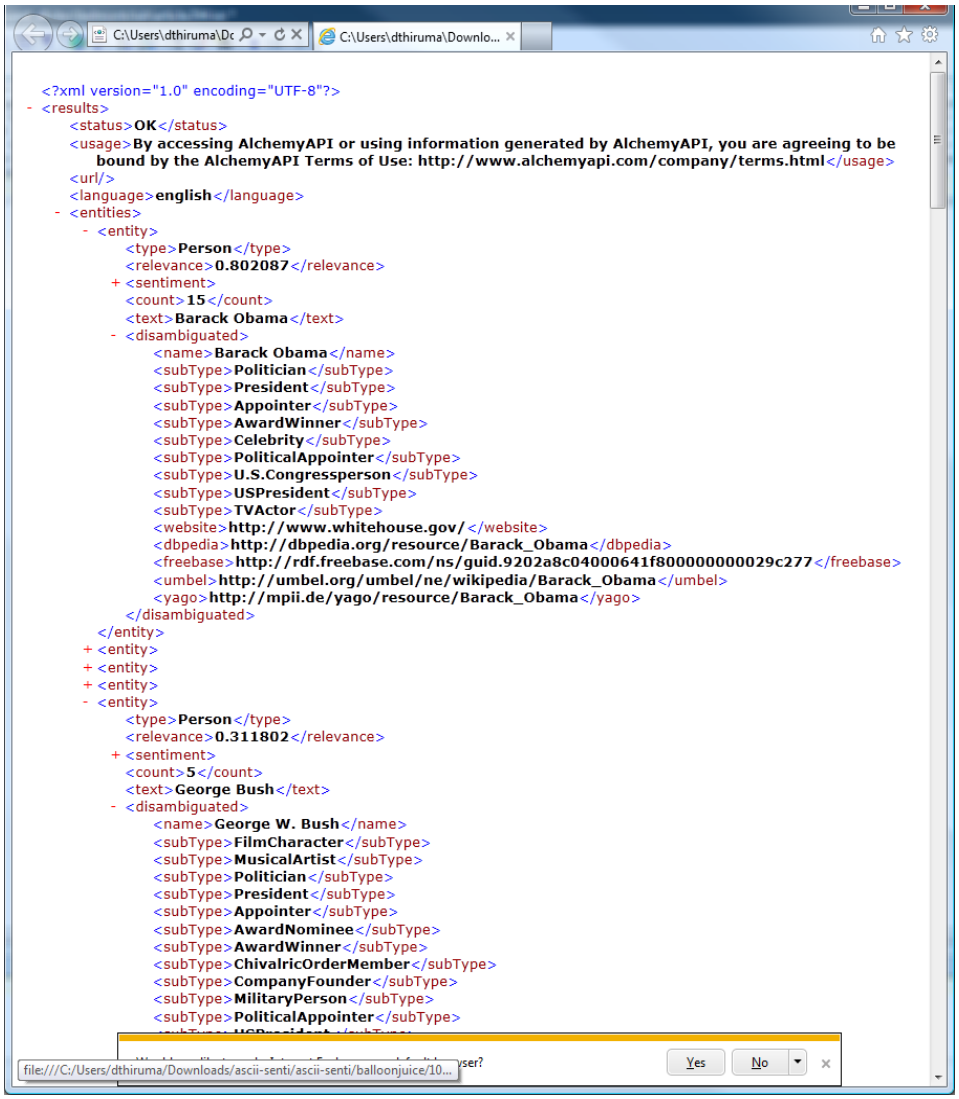

Figure 14. Snapshot of the XML file of NER of an article using Alcyhemy API.

Our program, sends the text from all the cleaned articles to process through the Alchemy API(using a key) and obtains the processed results with named entities. One such example is as shown in Figure 14. We can see that person name Barack Obama and Gerge W. Bush have been identified, along with the count(the number of occurences in the document). The count is not exactly the term frequency, the API looks for words which also describes that person using

NLP techniques such as he/she, him/her etc. When there is a conflict of the name, it disambiguates, understands the context and provides the reference and the website of the name it has identified through dpedia and other references. There are also several peroperties such as the type of the entity etc.

We can set the options such as, whether we need sentiment for the entities, the number of maximum entities to be found in a given article. With our initial data using politicalhub sites, there were around 350,000 articles. And technorati rank based websites had around 530,000 articles. Processing all the articles took a long time, and there was a limit on the number of requests(30,000) that can be made to the API. In order to efficiently use API, we merged documents together. We merged 10 documents together, and made it as a single document and sent it for processing to get named entities. Even though, each document was a single request, it took a long time for process it. And the accuracy of the NER was going down, which was bad. So we analyzed each blog article once, but for a lesser number of documents – articles within 6 months before the presidential elections(May 2008 – Oct 2008), which had very high intensity of debates. (included around 23,800 articles).

Chapter 5

SENTIMENT ANALYSIS AND FREQUENCY ANALYSIS

SENTIMENT ANALYSIS

Once the entities have been extracted, the next step is to find the sentiment expressed towards the entities by the blogs. Since, we have considered liberal/conservative blogs; we can see the opinion towards the entities. If a text mentions, “Bob is bad”, it is a negative sentiment. If a text mentions, “bob is good”, it is positive sentiment. “Bob is very good”, expresses more positive sentiment than bob is good. The sentiment analysis needs to capture whether the blog expresses positive or negative sentiment towards the entity and try to quantize the sentiment. And we employed two different strategies to perform sentiment analysis. They are discussed as follows:

Using snippets around the entity. Generally the words around a person name in a text, tries to describe about the person or what the person does. They generally tend to express their opinion towards the person using adjectives. This may be called as ascriptions. Hence, while we performed the NER tagging, we extracted 5 words before and 5 words after the entity. And then we can extract the sentiment from them. Few examples are:

<pos> about </pos> <pos> the </pos> <pos> goodness </pos> <pos> of </pos> <pos> </pos> <person> Rudy Giuliani </person> <pos> </pos> <pos> about </pos> <pos> how </pos> <pos> the </pos> <pos> attacks </pos>

<person> Bush </person> <pos> was </pos> <pos> Right </pos>

The surrounding words are in <pos> tags and the person name is in <person> tag. We can see that in the few words within the surrounding range expresses their opinion towards the person. In fact, we should be looking for describing words (such as" goodness" and "right") in the above example about the persons – adjectives/adverbs. In order to find the describing words, we need to find if the word is noun or an adjective or adverb. To figure it out, we used POS tagger. We used Stanford POS tagger to tag the part-of-speech, and it is based on [21]. But the results were not good. When we tried to perform frequency analysis (counting the number of times the entity occurs and the word that occurs around it), there were lots of problems:

(a) There were lots of stop words. [words such as the, was, is, a, an etc.]

(b) There were lots of neutral words or nouns such as "President" Bush

(c) It was difficult task to combine person names such as "Bush" and "President Bush" and "George W. Bush". Even though they referred to the same name, it was difficult to combine them and see how many times the name Bush has been used and what is the global sentiment. It is very complicated task and we tried to consider only names of length 2 words (first name and last name), but we lost a lots of names with a single word, which was bad. To overcome these problems, we used another approach, which is described in the next section.

Using Alchemy API. As explained in the previous chapter, Alchemy API is to analyze the content, extracting information about people, places, companies etc. They provide REST based API service, through different languages and the extracted meta-data may be returned in XML, JSON, RDF formats etc. It not only extracts the named entities but also finds the sentiment towards that person with a score. The sentiment can be positive, negative or a neutral along with a score. Unlike, the previous method which considers local descriptions of the entity, Alchemy API uses a global sentiment towards the entity. Consider the sentence, "Ugly Bob attacked beautiful Susan."

AlchemyAPI decodes three different sentiment values for the statement above. It marks "Bob" as negative (because he was indicated as being "Ugly"), and Susan as positive (because she is "beautiful").Additionally, using directional-sentiment, AlchemyAPI decodes the fact that Bob is emitting negative sentiment towards Susan (he is attacking her). Named entity extraction is also incorporated, so AlchemyAPI knows that Bob and Susan are both Persons.Using Alchemy have a lots of advantages over the previous method:

(a) It also looks for pronouns which refer to that person. Like he/she, him/her etc.

(b) It eliminates the stop words and finds the sentiment based on the global text.

(c) It tries to find a reference to person or the entity it has identified. In case of names like Bush, it tries to disambiguate the person named based on the context. Here Bush, George W. Bush, President – all refer to George W.

Bush, avoiding the disambiguation. It also gives puts the entity into subtype based on the profession and other characteristics such as, politician, artist etc.


```
<?xml version="1.0" encoding="UTF-8"?>
<results>
    <status>OK</status>
    <usage>By accessing AlchemyAPI or using information generated by AlchemyAPI, you are agreeing to be
        bound by the AlchemyAPI Terms of Use: http://www.alchemyapi.com/company/terms.html</usage>
    <url/>
    <language>english</language>
  - <entities>
      - <entity>
            <type>Person</type>
            <relevance>0.802087</relevance>
          - <sentiment>
                <type>neutral</type>
            </sentiment>
            <count>15</count>
            <text>Barack Obama</text>
          - <disambiguated>
                <name>Barack Obama</name>
                <subType>Politician</subType>
                <subType>President</subType>
                <subType>Appointer</subType>
                <subType>AwardWinner</subType>
                <subType>Celebrity</subType>
                <subType>PoliticalAppointer</subType>
                <subType>U.S.Congressperson</subType>
                <subType>USPresident</subType>
                <subType>TVActor</subType>
                <website>http://www.whitehouse.gov/</website>
                <dbpedia>http://dbpedia.org/resource/Barack_Obama</dbpedia>
                <freebase>http://rdf.freebase.com/ns/guid.9202a8c04000641f800000000029c277</freebase>
                <umbel>http://umbel.org/umbel/ne/wikipedia/Barack_Obama</umbel>
                <yago>http://mpii.de/yago/resource/Barack_Obama</yago>
            </disambiguated>
        </entity>
      + <entity>
      + <entity>
      + <entity>
      - <entity>
            <type>Person</type>
            <relevance>0.311802</relevance>
          - <sentiment>
                <type>negative</type>
                <score>-0.104415</score>
            </sentiment>
            <count>5</count>
            <text>George Bush</text>
          - <disambiguated>
                <name>George W. Bush</name>
                <subType>FilmCharacter</subType>
                <subType>MusicalArtist</subType>
```


Figure 15. Snapshot of the XML file of Sentiment Analysis of an article using Alcyhemy API.

In Figure 15, we can see that George Bush(seen in the <text> tag) is disambiguated as George W. Bush(seen in the <name> tag)and we can see that

the text expresses a negative sentiment towards George bush and neutral sentiment towards Barack Obama. The count also includes all the pronouns such as he/she, her/him etc. It also provides the reference for the entity through different database sources, such as dbpedia,freebase etc. to support it. In fact the NER tagging using Alchemy and the frequency analysis can be both combined into a single step, by providing setSentiments parameter to be true, while extracting the entities. Again we used the java version of Alchemy API. While doing the sentiment analysis for the articles one by one, because of the huge amount of articles, we combined 10 documents and made it a single document and sent it for processing. Due to the global sentiment analysis, their results were not accurate because of the different sentiments expressed in different articles and also because the difference in context between different articles. Hence, we considered to send single article at a time for processing, but for a short duration of time, 6 months before the presidential elections [May 2008 – Oct 2008], which has high intensity of debates.

FREQUENCY ANALYSIS

To check the entities which occur the most number of times in the entire blog (consisting of all blog articles), we need to analyze the NER results from all the articles. We can analyze the 5 words before and 5 words after format(with pos tags) to count the number of occurrences of the different names occurring in the blog along with the most occurring word which is +/- 5 words surrounding the entity. We wrote this frequency analysis program in Java. Figure 16 shows the

entities of word length more than 1, along with the most occurring neighboring words, with the elimination of stop words.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
al qaeda	224	attack	9	Abu	7	Al	7	Iraq	7	terrorist	7	groups	6
saddam hussein	113	Iraq	7	former	5	power	5	regime	5	capture	4	dictator	3
osama bin laden	62	Afghanistan	4	Alinsky	3	West	3	between	3	terrorist	3	Al	2
kofi annan	54	UN	15	General	11	Secretary(	11	Secretary	10	Nations	4	very	4
george bush	45	President	10	one	4	11	3	Septembe	3	whether	3	American	2
john kerry	42	Bush	3	America	2	President	2	antiwar	2	candidate	2	defeat	2
glenn reynolds	40	links	12	tip	11	Hat	6	hat	5	article	3	out	3
abu ghraib	30	abuses	3	images	3	Baghdad	2	displaced	2	pictures	2	prison	2
george w. bush	30	President	11	Truman	2	United	2	1945	1	1990	1	Bill	1
donald rumsfeld	26	Secretary	10	Defense	6	Hubris?	2	broadest	2	defence	2	descriptio	2
al jazeera	25	Reuters	5	Moreover	3	Philippine	3	broadcast	3	deputy	3	footage	3
andrew sullivan	25	Paul	3	links	3	Bremer	2	Kagan	2	describes	2	quotes	2
ward churchill	23	professor	4	Leftist	2	himself	2	plagiarize	2	those	2	Actually	1
abu sayyaf	22	Philippines	3	Abdel	2	began	2	claimed	2	demand	2	fact	2
juan cole	22	joke	5	professor	4	Goldberg	2	Iraq	2	Jonah	2	modern	2
paul bremer	20	Administrator	4	CPA	4	Andrew	2	Bremer	2	Former	2	George	2
saddam husseins	19	regime	7	forces	2	killed	2	sons	2	two	2	Baathist	1
roger simon	18	hat	3	tip	3	describes	2	one	2	time	2	Bais	1
tommy franks	18	Franks	3	General	3	CENTCOM	2	Tenet	2	Tommy	2	judgment	2
yasser arafat	18	Palestinian	3	President	3	Israeli	2	Marxist	2	According	1	AlSadr	1
ronald reagan	17	John	2	left	2	nation	2	Although	1	Bill	1	Bush	1
al qaedas	16	Arabia	1	Darling	1	Force	1	Middle	1	Saddam	1	Saudi	1
dan darling	16	Crucis	9	Regnum	8	Madrid	2	Winds	2	bombings	2	thinks	2
ralph peters	16	argued	3	article	3	writing	2	1	1	15ths	1	April	1
phil carter	15	Iraq	2	military	2	out	2	place	2	pointed	2	quotes	2
abdel rahman	14	STEWART	5	told	4	YOUSRY	3	Sayyafâ€™	2	Stewart	2	demand	2
angelo de la cruz	14	hostage	6	driver	4	Filipino	3	release	3	truck	3	Philippine	2
jacques chirac	14	President	8	French	7	one	2	Ariel	1	Bush	1	Gourdault	1
john kerrys	14	George	2	President	2	policy	2	Bin	1	Both	1	Bushs	1
steven den beste	14	Tranzism	2	den	2	long	2	refers	2	thanks	2	2003	1
bin laden	13	Basically	1	Bernard	1	Besides	1	Bush	1	Laden	1	Ladens	1

Figure 16. Snapshot of the result of Frequency Analysis of NER output.

When we performed the frequency analysis on NER tagged results from Stanford NER, as mentioned before, the results had a lot of noise, such as high stop words, and many non-descriptive neutral words such as president. We can check the quality of the data by checking if the frequency analysis of the blogs contains names from the most liberal/conservative people list mentioned in section 2.5. The result of Alchemy API can be analyzed by frequency analysis by taking into consideration the only entities with politician, for example, as one of their subtypes (because people like politicians, journalist, senator etc. would be the important issue, on which these blogs express sentiments on).

Chapter 6

SYNTHESIS OF BI-PARTITE GRAPH AND KEYWORD EXTRACTION

SYNTHESIS OF BI-PARTITE GRAPH

Once the sentiment analysis is performed and the sentiment scores of the persons by all the articles is obtained, the next step is to synthesize the overall bi-partite graph of the blogs and the people. We consider the sentiment towards people who are in the top most influential 100 liberal and 100 conservative people list (total 200). The sentiment scores of all those people, expressed by all the articles in a single blog are combined to form an overall sentimental score expressed by the blog towards the people. Similarly the overall sentimental scores for all the blogs towards the people are calculated. We wrote a matlab program, to read the sentimental analyzed articles in XML, and for the person names only in the top 200 most influential liberal/conservative lists, we combined the sentimental scores as cumulative sum of the product of relevance score and sentiment score for each blog. For example, if blog1 had articles article1, article2, article3 and article4. For each person, if the person is found in the 200 list, we found the overall sentiment expressed by blog1 as follows. We calculate the product of relevance and sentiment expressed by each every article – article1, article2, article3 and article4. And sum it up. The summed up value is the overall sentiment score expressed by the blog1 towards people. We considered entities which had relevance greater than 20% and sentiment greater than magnitude of sentiment score 0.1. When we considered the U.S. Presidential data, we obtained 135 people from the most

influential list (200). Then we performed the above procedure and recorded the cumulative sentimental score expressed towards the people as a two dimensional adjacency matrix, with rows representing the blogs and the columns representing the people.

Persons ->

Adjacency matrix or bi-partite graph: Blogs $\begin{bmatrix} & \\ & \end{bmatrix}$

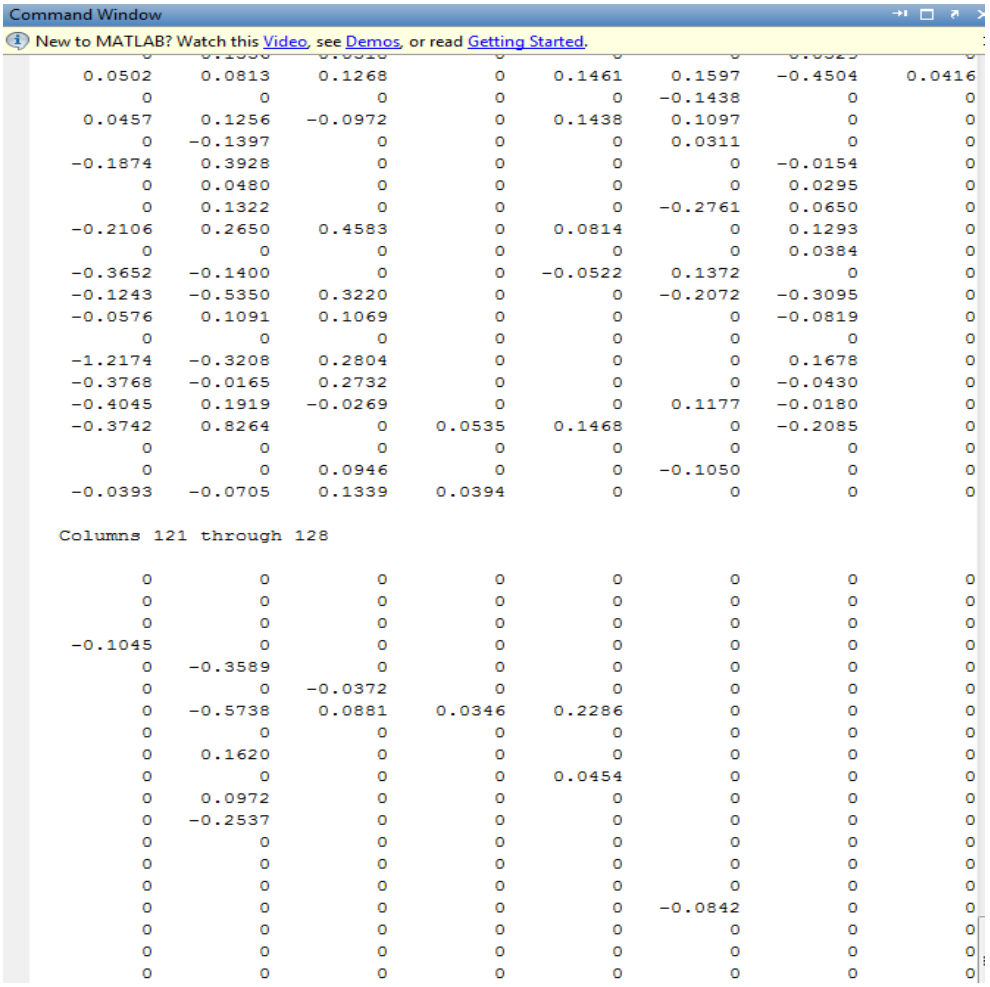

Figure 18. Snapshot of the result of adjacency matrix.

The output of the representation of the adjacency matrix for the U.S. Presidential looks as in Figure 18. The matrix is represented as a bi-partite graph with green and red edges after scaling with ANCO-HITS is shown as in Figure 20.


```
<?xml version="1.0" encoding="UTF-8"?>
<results>
  <status>OK</status>
  <usage>By accessing AlchemyAPI or using information generated by Alchemy
    bound by the AlchemyAPI Terms of Use: http://www.alchemyapi.com/co
  <url/>
  <language>english</language>
  <keywords>
    <keyword>
      <text>the oath</text>
      <relevance>0.979541</relevance>
      <sentiment>
        <type>positive</type>
        <score>0.238891</score>
      </sentiment>
    </keyword>
    <keyword>
      <text>mental reservation</text>
      <relevance>0.818756</relevance>
      <sentiment>
        <type>positive</type>
        <score>0.0165806</score>
      </sentiment>
    </keyword>
    <keyword>
      <text>Law School</text>
      <relevance>0.63474</relevance>
      <sentiment>
        <type>positive</type>
        <score>0.0202934</score>
      </sentiment>
    </keyword>
    <keyword>
      <text>Senator Feingold</text>
      <relevance>0.618841</relevance>
      <sentiment>
        <type>positive</type>
        <score>0.358647</score>
      </sentiment>
    </keyword>
    <keyword>
```


Figure 19. Snapshot of keyword extraction & their sentiments using AlchemyAPI

KEYWORD EXTRACTION

All this while, we had considered the leaders/people as the issue. We also did some experiments with topics as an issue. We performed a similar task, instead of

finding persons (entities) and finding the sentiment expressed towards them, here we consider extracting important topics and finding the sentiment expressed towards them. We used Alchemy API, using keyword extraction tool to find the keywords and their sentiments. The output of one such snapshot is shown as in Figure 19. When we obtained a similar bi-partite graph with blogs vs. Keywords, and performed partitioning [13], we got the most extreme topics as "Iraq War", "Polygamy", "Same sex-marriage", which was very intuitive. These were the topics most of the liberal/conservative blogs were debating on.

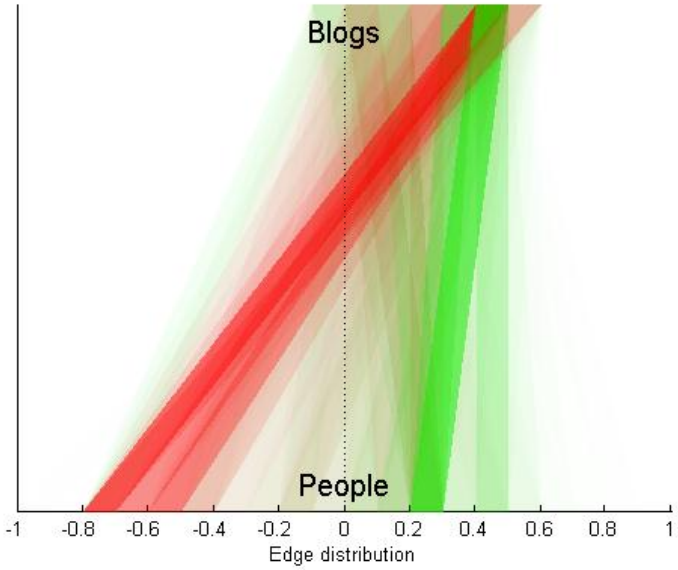

Figure 20. Bi-partite graph after scaling with ANCO-HITS

Chapter 7

RESULTS

The results of each and every step of the process are shown with a screen shot in their respective sections. The blogs decided upon different strategies are shown in the table in the data section and one in the appendix. The screen shot of the sample of the results of data extraction, NER tagging, frequency analysis, are shown in examples and screen shots. Table 3, shows the 3 strategies used to collect the blog sites, along with the time-span covered and the number of posts.

Table 3

Blog sites collected using different strategies.

Strategy Used	**Time Span**	**Number of websites**	**Number of Posts**
Using political hubs	2002 - 2011	147	350,000
Using high technorati ranking	2008 - 2011	26	530,000
Presidential election Data	May 2008 – Oct 2008	22	23,800

Table 2 is the list of blogs from presidential election data (May 2008 – Oct 2008) which was considered to build the sentimental bi-partite graph. There were 22 blogs out of which 13 were liberal and 9 were conservative.

Table2

List of political blogs during Presidential elections (May2008 – Oct 2008)

Blog Name	URL	Political camp	Posts
Huffington Post	http://www.huffingtonpost.com/	Liberal	3959
Daily Kos	http://www.dailykos.com/	Liberal	1957
Boing Boing	http://www.boingboing.net/	Liberal	1576
Crooks and Liars	http://www.crooksandliars.com/	Liberal	1497
Firedoglake	http://www.firedoglake.com/	Liberal	1354
AMERICABlog	http://americablog.com/	Liberal	1297
Think Progress	http://thinkprogress.org/	Liberal	1197
Talking Points Memo	http://www.talkingpointsmemo.com/	Liberal	1081
Wonkette	http://wonkette.com/	Liberal	1064
Balloon Juice	http://www.balloon-juice.com/	Liberal	923
Digby's Hullabaloo	http://digbysblog.blogspot.com/	Liberal	553
Informed Comment	http://www.juancole.com/	Liberal	179
Truthdig	http://www.truthdig.com/	Liberal	159
Hot Air	http://hotair.com/	Conservative	1579
Reason - Hit and Run	http://reason.com/blog	Conservative	1563
Little green footballs	http://littlegreenfootballs.com	Conservative	787
Atlas shrugs	http://atlasshrugs2000.typepad.com/	Conservative	773
Stop the ACLU	http://www.stoptheaclu.com/	Conservative	741
Wizbangblog	http://wizbangblog.com/	Conservative	621
Michelle Malkin	http://michellemalkin.com/	Conservative	532
Red State	http://www.redstate.com/	Conservative	311
Pajamas media	http://pajamasmedia.com/	Conservative	97

In chapter 3, we discussed the extraction of blog content using site specific crawlers instead of traditional crawlers. Here is the graph which shows how site specific crawlers outperform the traditional crawlers in time. We used a tool called GSite Crawler[24] to crawl the webpages using Traditional method.

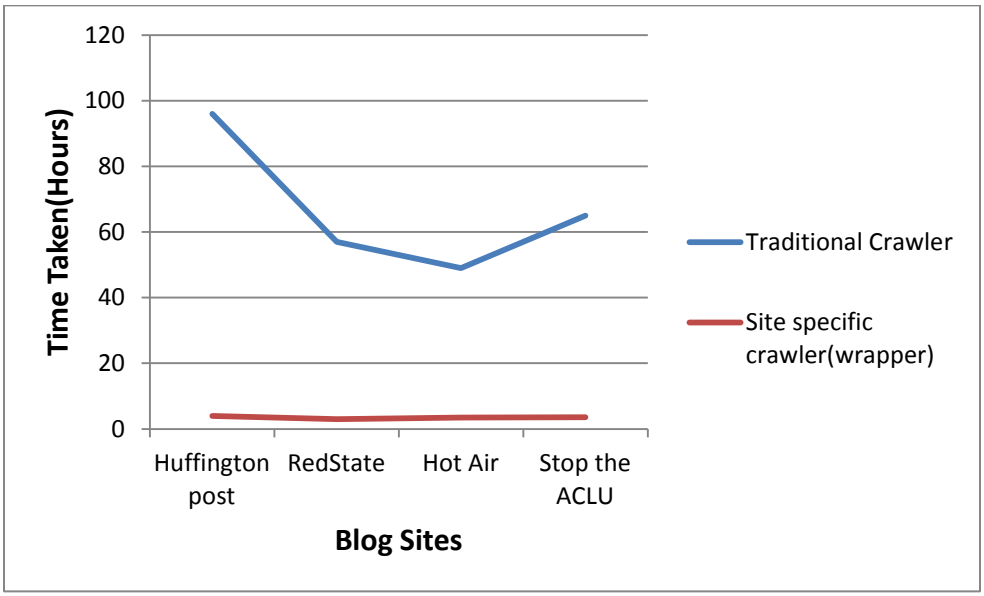

Figure 13. Comparison of time taken by site-specific crawler vs. traditional crawler

In chapter 4, we used Stanford NER tagger to extract the named entities. Figure 17 shows a snapshot of the result of an article using the Stanford NER.

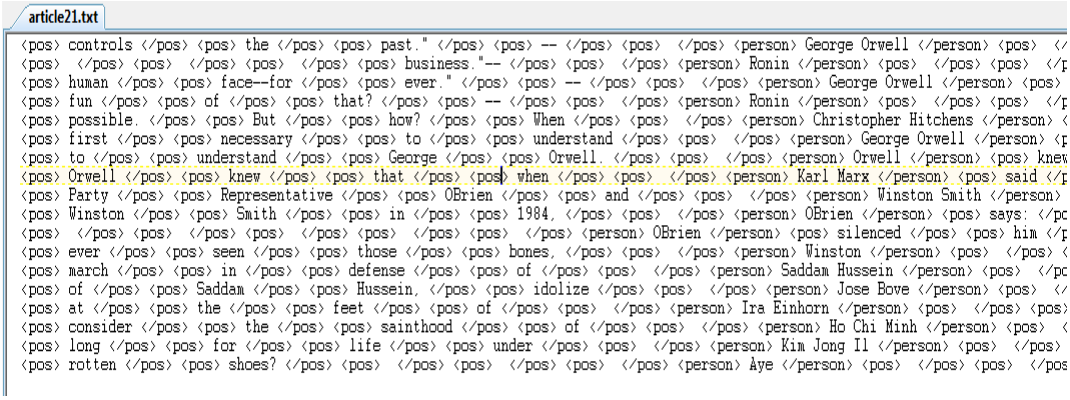

Figure 17. Snapshot of result of entity extraction using Stanford NER .

Chapter 8

CONCLUSIONS AND FUTURE WORK

CONCLUSION

In this thesis, we understood the impact of blogosphere on politics, and modeled the interesting interactions occurring in political blogs. We considered American conservative/liberal blogs, because of high intensity of debates and difference in opinions towards political issues. We considered to model the interactions(sentiment) between the blogs and the leaders (among different issues) into a structure, based on which several interesting data mining techniques can be employed to find further interactions/analysis. We modeled it as a bi-partite graph between blogs and leaders and the links between them having green edges or red edges, expressing positive or negative sentiments towards the leaders. Our thesis concentrated on making a bi-partite graph from the American blogs through sequentially steps.

First we considered the different strategies to find the most important blogs – through political hubs, sites with high Technorati rankings, 6 months before U.S. Presidential elections(May 2008 – Oct 2008). Then we employed different mechanisms to extract the data (only the crux of the blog content) from the blog websites – through RSS reading mechanisms for blogspot templates, using Google reader and using site specific crawlers, which uses extraction techniques based on structure of the webpages. Then the extracted HTML content was then cleaned using regular expression techniques and person names involved

in the articles were extracted using different approaches – using Stanford NER and Alchemy API. The sentiment expressed towards the entities by blogs were evaluated by sentiment analysis using several approaches – using snippets for find the describing words(5 words before and after) around the entity then, and using Alchemy API. We also extracted the important keywords around the articles to understand the context of the blog. We used the results from Sentiment analysis and synthesized the bi-partite graph by cumulative values of the sentiments of the blog articles. We synthesized the final bi-partite graph using the U.S. Presidential election data with 22 blog sites consisting of 22,800 blog articles.

FUTURE WORK

Using template independent wrapper for blog extraction. Instead of using the site specific crawlers which extracts the blog articles based on the structure of the template, using regular expressions, we can make use of template independent wrappers. The template independent wrappers can try to learn where the blog articles are present based on few training blog sites, and try to extract on the new blog sites.

Partitioning and scaling. [13] .We can try to use the bi-partite structure to perform various data mining techniques and find interesting patterns/interactions. The problem of partitioning the signed bi-partite graphs is interesting. While we partition the blogs trying to increase the number of green edges within each partition and increase the number of red edges between the partitions, we will actually partition the graph into two opposing groups. There can be several

algorithms try to perform this. Lots of research work is going in [13] which uses algorithm related to HITS and spectral clustering.

Scaling tries to scale both the blogs and the underlying issues on a uni-variate scale. Using this scale, researchers can identify moderate and extreme blogs within each camp, polarizing vs unifying issues. One can also develop techniques based on this structure, for detecting and presenting both friendly and unfriendly neighborhoods of a blog or an issue, and their agreements and disagreements. One can also incorporate longitudinal analysis to detect trends and trajectories over time.

REFERENCES

[1] Blogosphere. http://en.wikipedia.org/wiki/Blogosphere

[2] Political Blogs. http://en.wikipedia.org/wiki/Political_blog

[3] Google Reader. http://en.wikipedia.org/wiki/Google_Reader

[4] AlchemyAPI http://www.alchemyapi.com/

[5] Blogger Backup Utility. http://bloggerbackup.codeplex.com/

[6] Graph. http://en.wikipedia.org/wiki/Graph_%28mathematics%29

[7] The internet and campaign 2010. http://www.pewinternet.org/~/media/Files/Reports/2011/Internet%20and%20Campaign%202010.pdf

[8] Lada A. Adamic, N. Glance**.** The political blogosphere and the 2004 U.S. election: divided they blog. Proceedings of the 3rd international workshop on Link discovery ACM New York, NY, USA 2005

[9] John N. Brown. Who reads blogs: An Examination of Blog readers. Master's thesis, Appalachian State University, May 2007

[10] D. Cao, X. Liao, H. Xu, Shuo Bai. Blog post and comment extraction using information quantity of web format. Proceeding AIRS'08 Proceedings of the 4th Asia information retrieval conference on Information retrieval technology Springer-Verlag Berlin, Heidelberg - 2008

[11] H. Deng, M. Lyu, and I. King. A generalized co-hits algorithm and its application to bipartite graphs. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 239 - 248. ACM, 2009.

[12] D. Drezner and H. Farrell. The power and politics of blogs. Public Choice, 134:15-30, 2008.

[13] S. Gokalp, H. Temkit, H. Davulcu, and H. Torolu. Partitioning and scaling signed bipartite graphs forpolarized political blogosphere. In Proceedings of the 21st international conference on world wide web (submitted). ACM, 2012.

[14] W. Gryc, K. Moilanean. Leveraging Textual Sentiment Analysis with Social Network Modelling: Sentiment Analysis of Political Blogs in the 2008 U.S.

Presidential Election, T2PP Workshop, 9-10 April 2010, Vrije Universiteit Amsterdam

[15] E. Hargittai, J. Gallo, and M. Kane. Cross-ideological discussions among conservative and liberal bloggers. Public Choice, 134(1):67–86, 2008.

[16] John D. Lafferty, Andrew McCallum, Fernando C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Proceeding ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning

[17] Tony Mullen and Robert Malouf. Preliminary investigation into sentiment analysis of informal political discourse. In Computational Approaches to Analyzing Weblogs: Papers from 2006 AAAI Spring Symposium, pages 159 - 162. Stanford, California, USA, March 27-29 2006.

[18] M. Rege, M. Dong, and F. Fotouhi. Co-clustering documents and words using bipartite isoperimetric graph partitioning. In Data Mining, 2006. ICDM'06. Sixth International Conference on, pages 532-541. IEEE, 2006.

[19] V.K Singh,D. Mahata, R.Adhikari; Mining the Blogosphere from a socio-political perspective. Computer Information Systems and Industrial Management Applications (CISIM), 2010 International Conference , pages 365 – 370.

[20] Branimir T. Todorovic, Svetozar R. Rancic, Ivica M. Markovic, Edin H. Mulalic, Velimir M. Ilic. Named Entity Recognition and Classification using Context Hidden Markov Model.9th symposium on Neural network applications in electrical engineering, NEUREL – 2008.

[21] K. Toutanova and C.Manning,. *Enriching the Knowledge Sources used in a Maximum Entropy Part-of-Speech tagger.* Joint SIGDAT Conference on Empirical Methods in NLP and Very Large Corpora (EMNLP/VLC-2000) , October 7-8, 2000 , Hong Kong.

[22] H. Zha, X. He, C. Ding, H. Simon, and M. Gu. Bipartite graph partitioning and data clustering. In Proceedings of the tenth international conference on Information and knowledge management, pages 25-32. ACM, 2001.

[23] Z. Zhang, C. Zhang, Z. Lin, B. Xiao. Blog extraction with template independent wrapper. Network Infrastructure and Digital Content, 2010 2nd IEEE International Conference.

[24] GSiteCrawler. http://gsitecrawler.com/

APPENDIX A

LIST OF BLOGS COLLECTED FROM POLITICAL HUBS

BLOG Name	View	Year
The Abercrombie View	Conservative	2008-2010
apologiesdemanded	Conservative	2003-2011
The Absurd Canadian	Conservative	2004-2006
A Good Choice . . . for Ohio	Conservative	2006-2011
Eschaton	Conservative	2002-2011
Belmont Club	Conservative	2003-2005
Junto Boyz	Conservative	2003-2011
Bill's Comments	Conservative	2004-2011
Blog Hogger	Conservative	2003-2007
Blog O' DOB	Conservative	2005-2005
Booker Rising	Conservative	2004-2011
Cavalier Attitude	Conservative	2003-2004
A Pax American and other fun stuff	Conservative	2004-2011
Clayton's Corner	Conservative	2004-2005
Miller's Time	Conservative	2003-2004
CrankyBeach	Conservative	2004-2011
Colby and Beyond!	Conservative	2004-2005
Common Sense and Wonder	Conservative	2008-2011
Confederate Yankee	Conservative	2005-2005
Conservative observer	Conservative	2006-2011
Conservative Politics	Conservative	2005-2005
C-POL	Conservative	2004-2011
Cranky Bastard	Conservative	2004-2008
South Dakota War College	Conservative	2004-2007
Dick McDonald	Conservative	2010-2011
Dissecting Leftism	Conservative	2003-2011
DSS Hubris	Conservative	2004-2005
GotDesign	Conservative	2004-2011
Government Cheese	Conservative	2005-2007
Junto Boys	Conservative	2003-2011

Michael Johns	Conservative	2007-2010
My Echo Chamber	Conservative	2004-2010
84rules	Conservative	2007-2010
New Liberal Democrat	Conservative	2009-2011
Orthogonian	Conservative	2004-2005
Conservative Musings	Conservative	2010-2011
The Political Commentator	Conservative	2008-2011
The Politics Post	Conservative	2005-2006
Poor Justin's Almanac	Conservative	2004-2008
Pragmatic Libertarian	Conservative	
Prevent Truth Decay	Conservative	2003-2007
Rant burger	Conservative	2011-2011
Red Mind in a Blue State	Conservative	2004-2011
A Word From The Right	Conservative	2004-2007
Republican National Convention Blog	Conservative	2004-2011
Section 31	Conservative	2004-2005
Conservative Dubliner	Conservative	2005-2005
Nudnik File, The	Conservative	2004-2006
Ten O'Clock Scholar	Conservative	2004-2011
Thinktown USA Report	Conservative	2010-2011
Truthprobe	Conservative	2011-2011
political blog for the politically incorrect	Conservative	2004-2011
Wince and Nod	Conservative	2003-2011
Boring Made Dull, The	Conservative	2005-2009
Peter Porcupine	Conservative	2005-2010
The Cyber Menace	Conservative	2006-2008
GayPatriot	Conservative	2004-2008
Mockingbird	Conservative	2003-2005
Going to the Mat	Conservative	2004-2011
McCain's Lone Ranger	Conservative	2008-2009
Amy Ridenour's National Center Blog	Conservative	2003-2011
Urban Grind, The	Conservative	2004-2005
Galvin Opinion, The	Conservative	2004-2008
A Better Nation	Liberal	2004-2009
PSoTD	Liberal	2010-2011
altara	Liberal	2002-2011
American Regression	Liberal	2006-2007
Angry Bear	Liberal	2003-2011
Barking Dingo	Liberal	2004-2010
Belly of the Beast	Liberal	2004-2008
Today's World	Liberal	2010-2011
Boileryard	Liberal	2011-2011
Progressive intelligence and opinion	Liberal	2011-2011

24 Hours To Live	Liberal	2004-2011
Conservatives Are America's Real Terrorists	Liberal	2008-2011
Contrapositive	Liberal	2003-2010
Dad in Left Field	Liberal	2010-2011
Daily Texican	Liberal	2004-2010
Demagogue	Liberal	2005-2010
Hullabaloo	Liberal	2003-2006
Digital Dissent	Liberal	2004-2005
Disconnected Rumblings	Liberal	2004-2008
Dissent Channel	Liberal	2004-2008
Freethought	Liberal	2004-2006
End the Nightmare	Liberal	2004-2004
near-far	Liberal	2004-2010
European-American Blog	Liberal	2002-2011
A la Gauche	Liberal	2010-2011
Forewarned is Forearmed	Liberal	2004-2004
Democracy's Daily Posts	Liberal	2003-2008
Fresh Salad	Liberal	2005-2005
Heather Annastasia Siladi's Blogspot	Liberal	2006-2011
Excuse my french	Liberal	2004-2005
Iddybud	Liberal	2003-2010
Independent Observer, The	Liberal	2010-2010
Kick the Leftist	Liberal	2003-2005
Chris Geidner	Liberal	2009-2009
l'enfant terrible	Liberal	2010-2010
LeftIndependent	Liberal	2007-2007
The Limerick Savant	Liberal	2004-2011
Lionboi Blues & News	Liberal	2005-2009
Dispassionate Lib	Liberal	2006-2011
Mixter's Mix	Liberal	2011-2011
Odd Hours	Liberal	2004-2004
PBD - Progressive Blog Digest	Liberal	2004-2011
Political Salon	Liberal	2006-2011
The Preston Caldwell Political Blog	Liberal	2011-2011
The Proponent of Reason	Liberal	2011-2011
Raed in the Middle	Liberal	2004-2011
Rittenhouse Review	Liberal	2002-2007
Seeing the Forest	Liberal	
Shakespeare's Sister	Liberal	2004-2011
The Shameless Antagonist	Liberal	2003-2011
Brown Man Thinking Hard	Liberal	2008-2011
Snues	Liberal	2002-2011

Strange Doctrines	Liberal	2006-2007
sustainablog	Liberal	2003-2011
Americans for reason and truth	Liberal	2010-2011
The Blog Warrior	Liberal	2004-2005
Corey Hawkey	Liberal	2007-2007
tom_thinks	Liberal	2004-2006
Too Serious A Matter	Liberal	2007-2008
Velvel on National Affairs	Liberal	2004-2011
Delivering Hope	Liberal	2004-2009
Watching Washington	Liberal	2004-2009
AMERICAblog	Liberal	2004-2005
Abolish the Death Penalty	Liberal	2004-2011
Estropundit	Liberal	2004-2008
That's Going Too Far!	Liberal	2004-2009
Just to the Left	Liberal	2005-2011
Left is Right	Liberal	2002-2011
Outragedmoderates	Liberal	2006-2011
Rhetoric & Rhythm	Liberal	2003-2011
Richard McNairy	Liberal	
Roger Ailes	Liberal	2002-2011
Someone Take The Wheel	Liberal	2003-2005
StoutDemBlog	Liberal	2002-2011
skippy the bush kangaroo	Liberal	2010-2010
Yoder's Rants	Liberal	2004-2011
An Englishman in New York	Liberal	2004-2005
zenophobia: enlightened fear	Liberal	2009-2011
dldnh	Liberal	2004-2010
Constantly Amazed, Yet Never Surprised	Liberal	2008-2010