

Anchor Text Analysis

An anchor text analysis of links to five state government websites – California, New York, New Jersey, Texas, Massachusetts for the years 2004 and 2005.
Spring 2008 (05/16/2008)

Jasim Mohammed (jm487)
Prashant Baktha Kumara Dhas (pb327)

Advisor
Prof. William Y. Arms, Dept. of Computer Science
Cornell University

Contents

Abstract	3
Introduction	4
Description	6
Results	10
Conclusion	24
Acknowledgements	24
References	24
Appendix A – Tool Usage	25
Appendix B – Pig Latin	26

Abstract

We have built a tool for analyzing anchor text of links to various websites. The input to the tool is the test collection which is a file containing rows of data. Each row of data is comprised of the “from” link, the “to” link and the anchor text, all tab separated. While running the anchor text analysis tool we provide an option to the user for stemming words that are contained in the output. The stemming option can be used by the user as required based on the analysis that needs to be carried out. The tool produces a collection of 3 output files:

linkCount.tsv – contains the total number of links present in the input test collection.

termCount.tsv – contains the total number of anchor text words present in the input test collection

wordRank.tsv – contains a list of each anchor text word and its frequency. The list is sorted on frequency.

While parsing the test collection, the tool also performs the following processing before producing the output:

- 1) It cleans the input data by removing any special characters like “;”, “&”, “%”, “#” etc.
- 2) It filters out words in anchor text that are present in a list of words called the stop word list. This list is composed of words that are uninteresting with respect to the test collection being analyzed.

We used the anchor text analysis tool to study the test collections from 5 US state governments: California, Massachusetts, New York, New Jersey and Texas. We analyzed the test collections of each of these states for the years 2004 and 2005. The purpose of our study was to use the output of the tool to get an insight into the activities of these various state governments and to compare how their focus has changed from 2004 to 2005. We also made a comparison of the state governments with regard to their attention in the areas of education, health and business. This analysis done by us is intended to show by example how the tool can be used to analyze a random test collection.

Introduction

Web archives are very large collections of data with lots of interesting information. A great amount of useful information can be retrieved from such collections provided we have good tools to aid us in analyzing the collections. Our aim is to build a set of tools that can help us analyze various aspects of web archive data.

One set of data from web archives that is particularly useful is anchor text. Anchor text is the visible, clickable text in a hyperlink that is found on web pages. Anchor text gives useful information about the page being pointed to. It gives us some information about how the referred entities are perceived by others. Anchor text can be analyzed in basically 2 ways:

- 1) Term analysis – Here we try to get some statistical data regarding individual words or terms in the anchor text.
- 2) Phrase analysis – Here we try to analyze phrases as they appear in anchor text to get some idea of how the referred entities are perceived by different organizations.

Currently our objective is to build a tool to conduct term analysis on a random collection of data. Presently, our tool can be used to get statistical data such as number of anchor text links, number of terms present in the anchor text and the ranking of different words present in the anchor text of the provided input collection. With this kind of statistical data it is possible to get some information about the interest of referred entities and to perform a comparison of different entities based on some parameter.

Since the input collections to our tool are typically very large collections of data, we need to employ a programming model that performs some kind of parallel processing on the input data. Sequential processing of such large quantities of data can take an unacceptable amount of time such as a few days or even longer. The programming model that we employ for this problem is the map-reduce model. We use the Hadoop distributed system for this purpose and build the tool using a series of map-reduce programs.

Previous work on anchor text analysis has been carried out at The Web Lab by Ashish Virmani and Neha Arora. Their project involved a set of map reduce programs to analyze the anchor text of two companies: Microsoft and Google. Their work provided us with some valuable ideas on how to carry out an effective analysis of data generated by map-reduce programs.

To test our tool consisting of map-reduce programs we used a Hadoop cluster consisting of 6 machines. Our input dataset schema consisted of the following columns:

- From-host: This is the URL of the webpage containing the anchor text.
- To-host: This is the URL of the webpage to which the anchor text points to.
- Anchor text: This is the anchor text of the link

The input test collections to our tool were files of the above schema for five different states – New York, New Jersey, Massachusetts, California and Texas. We studied the collections

of the above states for 2 different years – 2004 and 2005. This way we have a total of 10 input files with the above schema.

The only constraint faced by us was in getting the entire input test collection in one shot since it takes considerable time in running SQL queries on the web crawls for 2004 and 2005. But to alleviate this problem we went ahead with testing our tool with a sample test collection provided to us that contained a subset of the actual data. Once the tool was completed successfully we were able to perform our analysis on the entire test collection which took a couple of weeks to generate.

Description

To perform the analysis the application has a few map-reduce jobs which are explained in detail below. The analysis is performed on the output of the map-reduce jobs. The input data set as mentioned in the introduction section needs to be pre-processed before it can be used for the map-reduce tasks. The following are the various pre-processing steps which are done:

- 1) Cleaning input data
- 2) Removing stop words
- 3) Stemming (this is optional)

Cleaning input data

This involves removing the special characters like ?, %, \$ etc. from the input data.

Removing stop words

This involves removing frequently occurring words from the input list. Some of the general stop words are and, about, or, the etc. Some of the specific stop words related to this analysis are California, Texas, NY, NJ etc. The stop word is implemented as a hash map so that the look up time for a stop word search will be a minimal constant.

Stemming

Stemming is the process for reducing inflected (or sometimes derived) words to their stem, base or root form. We use Porter's stemming algorithm for performing the stemming. The implementation is made flexible by making the stemming operation optional.

The application has map-reduce jobs for

- 1) Finding the frequency of anchor text terms in the collection.
- 2) Finding the number of links in the collection.
- 3) Finding the number of anchor text terms in the collection

1) Finding the frequency of terms in the collection

There are two map-reduce jobs for doing this, wordcount and wordrank jobs

Wordcount map-reduce job

Map

- a) The special characters are removed from the anchor text input line.
- b) The anchor text input line is split into terms.
- c) The stop words from the list of terms are filtered out.
- d) Emit (term,1)

Reduce

- a) Initialize the counter to 0.
- b) For all occurrences of a term in the input list increment the count.
- c) Emit (term, count)

Code snippet from the map-reduce job

```
public void map( WritableComparable key, Writable value, OutputCollector
output, Reporter reporter ) throws IOException
{
    String token;
    char[] tokenCharArr;
    String[] line = ( (Text)value ).toString().split( "\\t" );
    String anchorText = StopWordsList.cleanData( line[2] );
    StringTokenizer itr = new StringTokenizer( anchorText );

    while ( itr.hasMoreTokens() )
    {
        token = itr.nextToken().toLowerCase().trim();

        if ( !StopWordsList.isStopWord( token ) )
        {
            word.set( token );
            output.collect( word, one );
        }
    }
}
```

```
public void reduce( WritableComparable key, Iterator values, OutputCollector
output, Reporter reporter ) throws IOException
{
    int sum = 0;

    while ( values.hasNext() )
    {
        sum += ( (IntWritable)values.next() ).get();
    }

    output.collect( key, new IntWritable( sum ) );
}
```

Wordrank map-reduce job

The output of wordcount map-reduce is used as input to wordrank map-reduce job.

Map:

- a) Filter the terms on the number of occurrences. If they occur more than 3 times then they are accepted else rejected.
- b) Emit (frequency, term)

Reduce:

- a) For all occurrences of a frequency in the input list increment emit(frequency, term).

Code snippet

```

public void map( LongWritable key, Text value, OutputCollector<IntWritable,
Text> output, Reporter reporter ) throws IOException
{
    String line = value.toString();
    String[] lineArr = line.split( "\\t" );
    word.set( lineArr[0] );
    freq = new IntWritable( Integer.parseInt( lineArr[1] ) );

    if ( freq.get() > 3 )
    {
        output.collect( freq, word );
    }
}

public void reduce( IntWritable key, Iterator<Text> values,
OutputCollector<IntWritable, Text> output, Reporter reporter ) throws
IOException
{
    while ( values.hasNext() )
    {
        output.collect( key, values.next() );
    }
}

```

2) Finding the number of links in the collection

Linkcount map-reduce job

Map

- a) Get the URL from input line.
- b) Emit (URL, 1)

Reduce

- a) Initialize count to 0.
- b) For all occurrences of a URL in the input list increment the count
- c) Emit (URL, count)

Code snippet

```

public void map( WritableComparable key, Writable value, OutputCollector
output, Reporter reporter ) throws IOException
{
    link.set( "link" );
    output.collect( link, one );
}

public void reduce( WritableComparable key, Iterator values, OutputCollector
output, Reporter reporter ) throws IOException
{
    int sum = 0;
    while ( values.hasNext() )
    {
        sum += ( (IntWritable)values.next() ).get();
    }
}

```

```

    }
    output.collect( key, new IntWritable( sum ) );
}

```

3) Finding the number of anchor text terms in the collection

Termcount map-reduce job

Map

- a) The special characters are removed from the anchor text input line.
- b) The anchor text input line split into terms.
- c) The stop words from the list of terms are filtered out.
- d) Emit (“word”, 1)

Reduce

- a) Initialize the counter to 0.
- b) For all occurrences of the constant “word” in the key increment the counter
- c) Emit (“word”, count).

Code snippet from the map-reduce job

```

public void map( WritableComparable key, Writable value, OutputCollector
output, Reporter reporter ) throws IOException
{
    String token;
    char[] tokenCharArray;
    String[] line = ( (Text)value ).toString().split( "\\t" );
    String anchorText = StopWordsList.cleanData( line[2] );
    StringTokenizer itr = new StringTokenizer( anchorText );

    while ( itr.hasMoreTokens() )
    {
        token = itr.nextToken().toLowerCase().trim();

        if ( !StopWordsList.isStopWord( token ) )
        {
            word.set( "word" );
            output.collect( word, one );
        }
    }
}

```

```

public void reduce( WritableComparable key, Iterator values, OutputCollector
output, Reporter reporter ) throws IOException
{
    int sum = 0;
    while ( values.hasNext() )
    {
        sum += ( (IntWritable)values.next() ).get();
    }
    output.collect( key, new IntWritable( sum ) );
}

```

Results

From the output of the map-reduce jobs the following lists are generated for analysis

- 1) Top 20 list
- 2) Change over time list
- 3) Term comparison among the states

1)Top 20 list

Below are tables which show the list of the frequently occurring words for the years 2004 and 2005 for the five states.

California

Terms from 2005	Frequency	Terms from 2004	Frequency
tax	8171	caltrans	5462
ab	8439	services	6004
code	8859	programs	6372
resources	9044	dmv	6624
sb	9267	cdhs	6794
conditions	9550	legislative	7094
leginfo	9630	conditions	7600
bill	10201	cde	7894
amp	10285	resources	8081
cde	11032	corrections	8781
corporations	12347	page	9483
office	12416	commission	9965
energy	12419	energy	10344
page	13137	office	12594
commission	13444	education	12883
education	13675	county	12993
county	17523	information	13096
information	18304	dept	14029
board	24478	board	15083
department	49112	department	25210

If we look at this list “Education” is one term which occurs frequently so we can assume that California has high focus on education.

Massachusetts

Terms from 2005	Frequency	Term from 2004	Frequency
osd	204	index	203
index	215	portal	252
consumer	225	romney	256
mitt	236	mgl	277
management	260	governor	342
website	298	commonwealth	596
policy	327	emergency	658
official	377	management	663
governor	429	ma	693
romney	453	ozone	1588
web	539	level	1588
office	577	service	1820
site	857	osd	1828
commonwealth	1075	map	1830
ozone	1215	site	2053
level	1221	privacy	2356
ma	1295	policy	2383
online	2157	online	7926
page	2461	page	8195
services	10032	services	15875

If we look at the above list terms like “ozone”, “level” occur frequently. From this we can conclude that Massachusetts is working towards pollution control or it could be that they hosted a talk on pollution etc.

New Jersey

Terms from 2005	Frequency	Term from 2004	Frequency
analysis	408	statistics	624
statistics	414	analysis	643
safety	459	dot	833
governmen	546	capital	882
division	546	improvements	882
commission	561	commuter	890
site	654	community	916
department	718	srp	1049
info	749	contact	1183
online	959	online	1398
units	1032	njdep	1498
njdep	1073	units	1675
programs	1269	my	1920
my	1932	njdot	2254
dep	2479	programs	2591
services	2555	dep	3098
search	3153	search	3546
departments	7843	people	6321
people	8443	departments	7545
njhome	8630	njhome	7900
business	10691	business	9520

In the above list “business” is one term which occurs frequently. Based on this we can infer that New Jersey is attracting a lot of business. May be they provide an environment for business to thrive by having tax waivers etc. These are some of the various things we can infer.

New York

Terms from 2005	Frequency	Term from 2004	Frequency
finance	2130	division	734
amp	2257	official	739
service	2260	pataki	752
taxation	2283	commission	755
system	2528	aging	765
attorney	2758	information	790
general	2956	assistance	845
dept	2963	empire	891
board	3202	dec	1013
conservation	3204	tax	1027
division	3305	court	1075
website	3333	attorney	1168
site	3548	dept	1211
tax	3553	finance	1219
senate	3585	service	1220
environmental	3638	system	1234
insurance	3707	general	1280
labor	3921	taxation	1374
court	4246	insurance	1389
information	4405	governor	1440
services	4701	services	1687
governor	4728	environmental	1759
page	5334	conservation	1807
dec	5458	labor	2034
research	7415	senate	2313
office	9296	nys	3743
nys	15846	office	4127
health	17746	page	5599
department	19240	department	6568

As expected some of the words which occur frequently in the above list are tax, insurance, finance etc. This is expected considering the fact that New York City is the financial capital of the world. But “health” is another term which occurs frequently.

Texas

Terms from 2005	Frequency	Term from 2004	Frequency
web	5677	governor	3641
registration	5862	attorney	3742
offender	5906	hqqs	4015
general	6315	tcleose	4035
office	6511	office	4148
records	6626	park	4334
page	6985	tea	4344
online	7229	health	4407
library	7374	education	4487
amp	7856	online	4783
park	7894	general	4785
agency	8308	tsl	4842
public	8387	amp	5225
site	8425	public	5311
wildlife	9023	library	5470
tea	9048	information	5574
report	9960	legislature	5661
health	9976	link	5867
parks	10379	wildlife	5883
board	10499	board	6005
trail	10626	tdh	6072
education	11738	parks	6231
statewide	13249	trail	11755
information	18673	statewide	11817
search	22902	commission	12916
department	23595	department	13146
commission	24121	search	24174

Some of the expected words like “trail”, “parks”, “wildlife” occur frequently. “Education”, “library” are other words which occur frequently, which reflects that there is focus on literacy etc.

2)Change over time Analysis

In this analysis we calculate the relative frequencies of the terms and see how the term frequencies have changed from 2004 to 2005. The various columns in the tables are explained below:

Column A: Term

Column B: Frequency of the term in year 2005

Column C: Frequency of the term in year 2004

Column D: Frequency of the term for year 2005 / Number of terms in the collection for year 2005

Column E: Frequency of the term for year 2004 / Number of terms in the collection for year 2004

Column F: Column D – E

Column G: $(D + E) / 2$

Column H: F / G

There are three cases for the value in column H

1) Less than zero.

In this case the frequency of the term in 2004 is more than 2005.

2) Greater than zero

In this case the frequency of the term in 2005 is more than 2004.

3) Equal to zero

In this case the frequencies for both years are the same

For this analysis we consider the top words which increased and decreased in frequencies over the years.

California

Term	Frequency		T05/ N05(2041 340)	T04/ N04(1142 819)	D - E	(D + E)/ 2	F/G
	'05	'04					
cdhs	226	6794	0.000110712	0.005944948	-0.00583424	0.00302783	-1.926870669
comments	128	2299	6.27E-05	0.002011692	-0.00194899	0.001037198	-1.8790898
compliance	172	1603	8.43E-05	0.001402672	-0.00131841	0.000743465	-1.773336019
course	193	1551	9.45E-05	0.00135717	-0.00126263	0.000725858	-1.73949247
corrections	1156	8781	0.000566295	0.007683631	-0.00711734	0.004124963	-1.72543042
accountability	250	1693	0.000122469	0.001481424	-0.00135896	0.000801947	-1.694571712
students	260	1688	0.000127367	0.001477049	-0.00134968	0.000802208	-1.682458243
query	129	795	6.32E-05	0.000695648	-0.00063245	0.000379421	-1.666893586
programs	1458	6372	0.000714237	0.005575686	-0.00486145	0.003144961	-1.545789831
corr	100	417	4.90E-05	0.000364887	-0.0003159	0.000206937	-1.5265481

In the above list corrections has gone down significantly from 2004 to 2005. This can reflect the fact that crime rate was less.

Term	Frequency		T05/ N05(2041 340)	T04/ N04(1142 819)	D - E	(D + E)/ 2	F/G
	'05	'04					
people	1284	203	0.000628999	0.000177631	0.000451368	0.00040332	1.119144924
motor	3178	493	0.001556821	0.000431389	0.001125431	0.00099411	1.132104927
station	763	112	0.000373774	9.80E-05	0.000275771	0.00023589	1.169071862
franchise	4112	569	0.002014363	0.000497892	0.001516471	0.00125613	1.207259342
sb	9267	1276	0.004539665	0.001116537	0.003423128	0.0028281	1.21039794
loan	1054	143	0.000516328	0.000125129	0.000391198	0.00032073	1.219718626
opinions	3354	439	0.001643038	0.000384138	0.001258901	0.00101359	1.242023821
virus	1351	173	0.00066182	0.00015138	0.00051044	0.0004066	1.255386081
mail	1385	172	0.000678476	0.000150505	0.000527971	0.00041449	1.273782938
nile	1389	170	0.000680435	0.000148755	0.00053168	0.0004146	1.282408619
elementary	2015	232	0.000987097	0.000203007	0.00078409	0.00059505	1.317683621
card	1691	194	0.000828377	0.000169756	0.000658622	0.00049907	1.319707302
campaign	1358	113	0.000665249	9.89E-05	0.000566371	0.00038206	1.482399011
cad	3753	262	0.001838498	0.000229258	0.001609241	0.00103388	1.556509253
pub	2899	198	0.001420146	0.000173256	0.00124689	0.0007967	1.565066818
corporations	12347	800	0.006048478	0.000700023	0.005348455	0.00337425	1.585079217

If we observe the above list terms like “virus” “mail” have appeared more in 2005 than 2004. This can reflect the fact that there were more e-mail viruses etc in year 2005.

Massachusetts

Term	Frequen cy '05	Frequen cy '04	T05/ N05(932 19)	T04/ N04(8030 6)	D - E	(D + E)/ 2	F/ G
service	115	1820	0.00123365	0.022663313	-0.02142966	0.011948483	-1.793504492
map	157	1830	0.00168421	0.022787837	-0.02110363	0.012236021	-1.724713455
privacy	203	2356	0.00217767	0.029337783	-0.02716012	0.015757725	-1.723606345
osd	204	1828	0.0021884	0.022762932	-0.02057454	0.012475663	-1.649173756
policy	327	2383	0.00350787	0.029673997	-0.02616613	0.016590933	-1.577134256
emergency	164	658	0.0017593	0.008193659	-0.00643436	0.004976479	-1.292954665
online	2157	7926	0.02313906	0.098697482	-0.07555842	0.060918271	-1.240324477
page	2461	8195	0.0264002	0.10204717	-0.07564697	0.064223683	-1.17786723
management	260	663	0.00278913	0.008255921	-0.00546679	0.005522526	-0.989907534
portal	102	252	0.0010942	0.003137997	-0.0020438	0.002116097	-0.965834406
site	857	2053	0.00919341	0.025564715	-0.01637131	0.01737906	-0.942013567
services	10032	15875	0.10761755	0.197681369	-0.09006382	0.152649457	-0.590004214
mgl	191	277	0.00204894	0.003449306	-0.00140037	0.002749122	-0.509387232
ozone	1215	1588	0.01303382	0.019774363	-0.00674054	0.016404093	-0.410905946

In the terms related to pollution like “ozone” have gone down from 2004 to 2005. This can reflect change in emphasis for pollution.

Term	Frequen cy '05	Frequen cy '04	T05/ N05(9321 9)	T04/ N04(8030 6)	D - E	(D + E)/ 2	F/ G
level	1221	1588	0.013098188	0.019774363	-0.006676175	0.016436276	-0.406185384
dem	130	145	0.001394565	0.001805594	-0.000411028	0.00160008	-0.256879803
index	215	203	0.002306397	0.002527831	-0.000221434	0.002417114	-0.091611025
consumer	225	194	0.002413671	0.00241576	-2.09E-06	0.002414715	-0.000864982
governor	429	342	0.004602066	0.00425871	0.000343356	0.004430388	0.077500131
mitt	236	179	0.002531673	0.002228974	0.000302699	0.002380323	0.127166984
romney	453	256	0.004859524	0.003187807	0.001671718	0.004023665	0.415471343
commonwealt	1075	596	0.011531984	0.007421612	0.004110371	0.009476798	0.433729985
official	377	183	0.00404424	0.002278784	0.001765456	0.003161512	0.558421533
website	298	127	0.003196773	0.001581451	0.001615322	0.002389112	0.676118237
web	539	202	0.005782083	0.002515379	0.003266704	0.004148731	0.78739848
office	577	112	0.006189725	0.001394665	0.00479506	0.003792195	1.264454871

The above list does not have significant information for inference.

New Jersey

Term	Frequen cy '05	Freque ncy '04	T05/ N05(105 473)	T04/ N04(8646 4)	D - E	(D + E)/ 2	F/ G
capital	113	882	0.001071364	0.010200777	-0.009129413	0.005636071	-1.619818739
improvements	113	882	0.001071364	0.010200777	-0.009129413	0.005636071	-1.619818739
commuter	122	890	0.001156694	0.010293301	-0.009136607	0.005724998	-1.595914552
srp	282	1049	0.00267367	0.012132217	-0.009458547	0.007402943	-1.277673792
community	251	916	0.002379756	0.010594004	-0.008214248	0.00648688	-1.266286448
contact	392	1183	0.003716591	0.013681995	-0.009965404	0.008699293	-1.145541823
homeowner	201	495	0.001905701	0.005724926	-0.003819225	0.003815313	-1.001025214
help	238	502	0.002256502	0.005805885	-0.003549383	0.004031193	-0.880479496
programs	1269	2591	0.012031515	0.029966229	-0.017934714	0.020998872	-0.854079857
library	121	247	0.001147213	0.00285668	-0.001709467	0.002001947	-0.853902485
dept	103	206	0.000976553	0.002382494	-0.001405941	0.001679524	-0.837107024
military	150	284	0.001422165	0.003284604	-0.001862439	0.002353384	-0.791387518
employment	134	249	0.001270467	0.002879811	-0.001609344	0.002075139	-0.775535376
news	183	322	0.001735041	0.003724093	-0.001989052	0.002729567	-0.728706019
veterans	158	273	0.001498014	0.003157383	-0.00165937	0.002327699	-0.712879982

Term	Frequen cy '05	Frequen cy '04	T05/ N05(1054 73)	T04/ N04(8646 4)	D - E	(D + E)/ 2	F/ G
commerce	311	242	0.002948622	0.002798853	0.000149769	0.002873737	0.052116531
economic	311	241	0.002948622	0.002787287	0.000161335	0.002867955	0.056254283
safety	459	355	0.004351825	0.004105755	0.00024607	0.00422879	0.05818914
growth	361	275	0.003422677	0.003180514	0.000242162	0.003301596	0.073347098
people	8443	6321	0.080048922	0.07310557	0.006943353	0.076577246	0.090671222
child	166	108	0.001573863	0.001249075	0.000324788	0.001411469	0.230106244
commission	561	345	0.005318897	0.0039901	0.001328797	0.004654498	0.285486588
labor	247	150	0.002341832	0.001734826	0.000607006	0.002038329	0.297795676
information	282	135	0.00267367	0.001561343	0.001112327	0.002117507	0.52530014
department	718	333	0.006807429	0.003851314	0.002956116	0.005329372	0.554683697
insurance	323	147	0.003062395	0.00170013	0.001362266	0.002381262	0.572077073
education	339	136	0.003214093	0.001572909	0.001641184	0.002393501	0.685683384
division	546	215	0.00517668	0.002486584	0.002690096	0.003831632	0.702075818
office	302	108	0.002863292	0.001249075	0.001614217	0.002056183	0.785055104

From the above list we can clearly conclude that New Jersey focus on education, child care, labor etc are more in 2005 than 2004.

New York

Term	Frequen cy '05	Frequen cy '04	T05/ N05(654 208)	T04/ N04(1033 52)	D - E	(D + E)/2	F/G
academic	143	199	0.000218585	0.001925459	-0.001706874	0.001072022	-1.592200605
hiicap	292	387	0.000446341	0.003744485	-0.003298144	0.002095413	-1.573982585
nysgis	402	440	0.000614483	0.004257295	-0.003642812	0.002435889	-1.495475081
liberty	116	120	0.000177314	0.001161081	-0.000983767	0.000669197	-1.470070522
science	227	217	0.000346984	0.002099621	-0.001752636	0.001223303	-1.432708716
mentally	113	105	0.000172728	0.001015946	-0.000843218	0.000594337	-1.418753939
advocate	267	237	0.000408127	0.002293134	-0.001885007	0.001350631	-1.395649594
assistance	955	845	0.00145978	0.008175942	-0.006716162	0.004817861	-1.394013129
holocaust	221	188	0.000337813	0.001819026	-0.001481213	0.00107842	-1.37350353
oasas	438	371	0.000669512	0.003589674	-0.002920162	0.002129593	-1.371230237
social	177	147	0.000270556	0.001422324	-0.001151768	0.00084644	-1.360719778
worker	182	151	0.000278199	0.001461026	-0.001182827	0.000869613	-1.360177269
persons	260	214	0.000397427	0.002070594	-0.001673167	0.00123401	-1.355877203
processing	214	176	0.000327113	0.001702918	-0.001375805	0.001015016	-1.355452118
counseling	509	414	0.00077804	0.004005728	-0.003227688	0.002391884	-1.349433325
alcoholism	343	277	0.000524298	0.002680161	-0.002155863	0.00160223	-1.345539364

Term	Freque ncy '05	Freque ncy '04	T05/ N05(6542 08)	T04/ N04(1033 52)	D - E	(D + E)/2	F/G
disabled	366	113	0.000559455	0.001093351	-0.000533896	0.000826403	-0.646047739
reform	370	114	0.000565569	0.001103027	-0.000537457	0.000834298	-0.644202933
courts	1608	481	0.002457934	0.004653998	-0.002196064	0.003555966	-0.617571755
tax	3553	1027	0.005430994	0.009936915	-0.00450592	0.007683955	-0.586406412
resource	507	139	0.000774983	0.001344918	-0.000569935	0.001059951	-0.537700014
court	4246	1075	0.006490291	0.010401347	-0.003911056	0.008445819	-0.463076043
division	3305	734	0.00505191	0.007101943	-0.002050033	0.006076926	-0.337346977
children	926	181	0.001415452	0.001751297	-0.000335845	0.001583374	-0.212106881
mental	912	170	0.001394052	0.001644864	-0.000250812	0.001519458	-0.165066815
banking	1890	112	0.002888989	0.001083675	0.001805314	0.001986332	0.908868173
health	17746	623	0.027125929	0.006027943	0.021097986	0.016576936	1.272731316
research	7415	237	0.011334316	0.002293134	0.009041182	0.006813725	1.326907332

We can infer from the above table that the focus on banking, health and research are more in 2005 than in 2004.

Texas

Term	Freque ncy '05	Frequen cy '04	T05/ N05(1329 080)	T04/ N04(7352 70)	D - E	(D + E)/ 2	F/ G
contracts	122	654	9.18E-05	0.000889469	-0.000797676	0.000490631	-1.625817286
privacy	873	1237	0.000656845	0.001682375	-0.00102553	0.00116961	-0.876813312
arts	2032	2872	0.001528877	0.003906048	-0.002377171	0.002717463	-0.874775971
food	335	473	0.000252054	0.000643301	-0.000391247	0.000447678	-0.87394828
división	109	153	8.20E-05	0.000208087	-0.000126075	0.000145049	-0.869189143
inmunizacione	109	153	8.20E-05	0.000208087	-0.000126075	0.000145049	-0.869189143
colleges	319	424	0.000240016	0.000576659	-0.000336643	0.000408337	-0.824424512
pesticide	101	134	7.60E-05	0.000182246	-0.000106254	0.000129119	-0.822910648
legislature	4271	5661	0.003213501	0.007699213	-0.004485711	0.005456357	-0.82210743
basketball	100	130	7.52E-05	0.000176806	-0.000101566	0.000126023	-0.805931125
banking	648	839	0.000487555	0.001141077	-0.000653522	0.000814316	-0.802540824

Term	Frequen cy '05	Frequen cy '04	T05/ N05(132 9080)	T04/ N04(735 270)	D - E	(D + E)/ 2	F/ G
industry	938	190	0.000705751	0.000258408	0.000447343	0.00048208	0.927943398
motor	631	120	0.000474764	0.000163205	0.000311559	0.000318985	0.976720592
utility	896	168	0.000674151	0.000228487	0.000445663	0.000451319	0.987467908
ethics	981	150	0.000738105	0.000204007	0.000534098	0.000471056	1.133831838
texasonline	1664	244	0.001251994	0.000331851	0.000920143	0.000791922	1.161910571
renewal	1497	216	0.001126343	0.00029377	0.000832573	0.000710056	1.172545554
construction	1964	283	0.001477714	0.000384893	0.001092821	0.000931303	1.173432246
gulf	697	100	0.000524423	0.000136004	0.000388418	0.000330214	1.176263934
affairs	1189	167	0.000894604	0.000227127	0.000667476	0.000560866	1.190082464
housing	1131	131	0.000850965	0.000178166	0.000672799	0.000514565	1.307509174
ozone	1119	129	0.000841936	0.000175446	0.00066649	0.000508691	1.310206645
losures	1855	188	0.001395702	0.000255688	0.001140014	0.000825695	1.380671357

The above list does not have any interesting details for analysis.

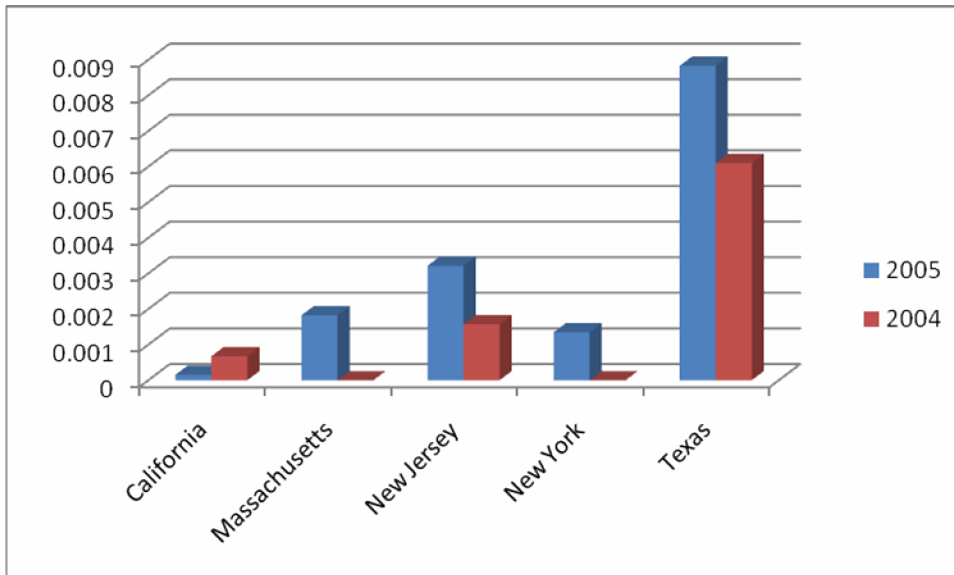
3)Term comparison among the states

In this analysis we take three words and compare their relative frequencies obtained from the above tables in section 2. The three words are:

- 1) Education
- 2) Health
- 3) Business

Education

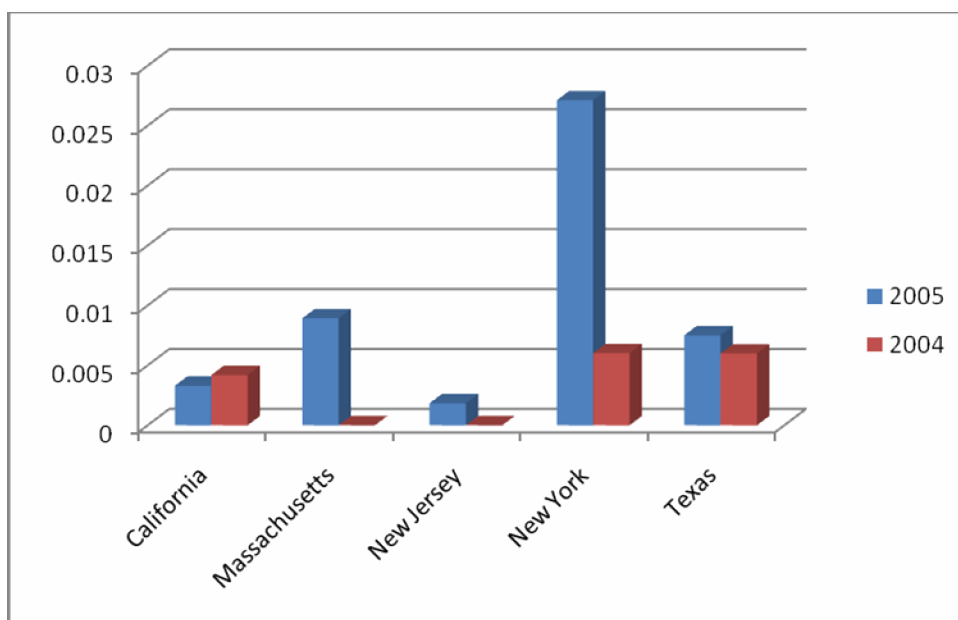
State	Relative frequency for year 2005	Relative frequency for year 2004
California	0.000152841	0.000673772
Massachusetts	0.001823663	0
New Jersey	0.003214093	0.001572909
New York	0.001345138	0
Texas	0.008831673	0.00610252



The above graph shows that among the five states Texas places more emphasis on education. Apart from California all other states have increased focus on education from the previous year.

Health

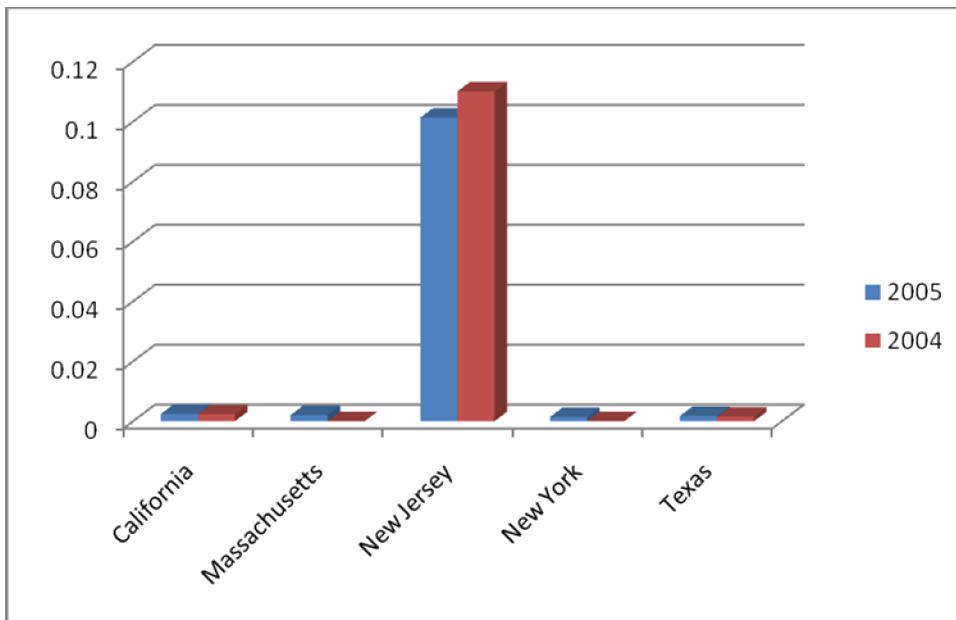
State	Relative frequency for year 2005	Relative frequency for year 2004
California	0.003306162	0.00419314
Massachusetts	0.008925219	0
New Jersey	0.001839333	0
New York	0.027125929	0.006027943
Texas	0.007505944	0.005993717



The above graph shows that among the five states New York places more emphasis on Health. Apart from California all other states have increased focus on education from the previous year. Another point to be noted is though Texas and New York were placed at the same level in 2004, in 2005 New York's frequency went up significantly. This clearly reflects the emphasis on health.

Business

State	Relative frequency for year 2005	Relative frequency for year 2004
California	0.002443003	0.002365204
Massachusetts	0.002070393	0
New Jersey	0.101362434	0.110103627
New York	0.001444495	0
Texas	0.00177416	0.00151101



The above graph shows that among the five states New Jersey places a significant emphasis on business though it reduced from 2004. All other states are about the same level.

Conclusion

The results we get from the above analysis can be used as a starting point for other studies. As of now we can base conclusions like which state is focusing more on health, business etc. But for more detailed findings like perception of a state government based on the frequency of the word like whether the words show positive or negative feeling etc we cannot use the above results. More detailed analysis need to be done. As of now we only do term analysis the tool needs to be extended to perform the more detailed phrase analysis. By doing phrase analysis the results will be more focused and better.

Acknowledgements

We would like to extend our gratitude to Prof. William Y. Arms for the valuable guidance that he provided in our project. We would also like to thank Lucy Walle for her help in acclimatizing us with the Hadoop system and Manuel Calimlim for providing us with the test collections. We also acknowledge the support from Cornell Web Lab in providing us with the resources to run our tasks and perform the analysis. The Cornell Web Lab is funded in part by National Science Foundation grants CNS-0403340 SES-0537606, IIS-0634677, and IIS-0705774.

References

- Anchor text analysis, Fall 2007 - <http://www.infosci.cornell.edu/SIN/WebLab/papers/Virmani2007.pdf>
- Bursty and Hierarchical Structure in Streams, Jon Kleinberg - <http://www.cs.cornell.edu/home/kleinber/bhs.pdf>
- Hadoop Documentation - <http://hadoop.apache.org/core/>
- Map Reduce Paper from Google - <http://labs.google.com/papers/mapreduce-osdi04.pdf>
- Pig Documentation - <http://wiki.apache.org/pig/> and <http://incubator.apache.org/pig/>
- Yahoo Pig Tutorial - <http://mias.uiuc.edu/files/speakers/pig-tutorial.ppt>
- Porter Stemming Algorithm - <http://tartarus.org/~martin/PorterStemmer/>

Appendix A – Tool Usage

The tool basically consists of 2 components:

- *analysis.jar*: This jar contains all the map-reduce java classes that operate on the input test collection and produce the output files. It also contains a PostProcessor.class that performs some post-processing on the output files.
- *tool.sh*: This is a script that takes the input file as a parameter, runs the map-reduce java classes present in analysis.jar and then does some post-processing where it reorganizes the output files produced into an easily readable structure.

To run the tool, follow these steps:

- Copy *tool.sh*, *analysis.jar* and the *input file* to any directory on the master node of the Hadoop cluster. We'll call this directory as the current working directory. Here the *input file* is a file whose schema has been described previously ie. A file with columns “from-host”, “to-host” and “anchor text” all tab separated.
- Before running the tool ensure that there is no directory named “output” under the current working directory. This is because any directory named “output” will be deleted while running the tool and a new directory named “output” will be created in its place containing the output of the tool.
- Ensure that tool.sh can be executed by executing the following statement at the prompt in the current working directory:
`$chmod +x tool.sh`
- If you want an output containing stemmed words execute the tool by running the following command at the prompt in the current working directory:
`$/tool.sh -stem <input-filename>`

If you want an output without stemming, enter the following command at the prompt:

```
$/tool.sh <input-filename>
```

- After execution of the tool, a directory named “output” will be created under the current working directory. It will contain 3 files: linkCount.tsv, termCount.tsv and wordRank.tsv. These files contain tab separated entries and can be opened using Microsoft Excel to see their contents.

Appendix B – Pig Latin

An alternative to performing data analysis using map-reduce programs is to execute a series of Pig Latin statements. Pig Latin is a language that consists of set of few simple commands that is designed for data analysis. Each command in Pig Latin transforms a set of records to another set of records. The objective of using Pig Latin is to make it easy for the users as it provides a high-level abstraction for data analysis and also to make use of any inherent parallelism and reuse for the data analysis. Internally, Pig Latin statements are transformed to Map-Reduce tasks on Hadoop, but this transformation is transparent to the user.

We experimented with Pig Latin, by trying to find the count of words in a document. Since Pig is still in the incubation period, there is no stable release that can be downloaded. We therefore had to check out the Pig source code online using SVN and then build it. SVN can be downloaded as an Eclipse Plugin. The location of the Pig source code online is <http://svn.apache.org/repos/asf/incubator/pig/trunk>. Once the source code is checked out, we build it from the top-level directory. If the build is successful, we see pig.jar created in that directory.

There are 3 ways to execute Pig Latin commands:

- Using an interactive shell called *grunt*
- Using a script file
- Embed Pig Latin commands in a host language (Eg. Java)

We used the first method for executing our Pig Latin program. To start *grunt* we execute the following command at the command prompt: **java -jar pig.jar -x local** where pig.jar was created earlier. Once *grunt* is started we execute the following statements to perform a word count with the input file “mytext”

```
grunt> A = load 'mytext' using TextLoader();
grunt> B = foreach A generate flatten(TOKENIZE($0));
grunt> C = group B by $0;
grunt> D = foreach C generate flatten(group), COUNT(B.$0);
grunt> store D into 'myoutput';
```

Here the file “myoutput” gets created which contains the word count. Here we use the default tokenizer provided in Pig called TOKENIZE. Users can specify their own tokenizer by writing a java program and creating a jar containing the compiled java class. The user can then use this custom tokenizer by registering the jar as follows :

```
grunt> register ./Tokenize.jar
```

We can then invoke the custom tokenizer using a statement like

```
grunt> B = foreach A generate flatten(Tokenize($0));
```

As we can see from the above code if we want to perform any specific specialized operations like stemming, stop word filtering etc., we need to write java code, create jars and register before running. This is an inflexibility we see with pig latin. But otherwise it is easier

than the regular map-reduce programming as most of the details of map-reduce steps are hidden from the programmer.