Assignment 3

CS595

Introduction to Web Science

Old Dominion University

Computer Science

Due: 11:59 pm Oct 5

Lulwah Alkwai

Question One-

Download the 1000 URIs from assignment # 2. "curl", "wget", or "lynx" are all good candidate programs to use. We want just the raw HTML, not the images, stylesheets, etc. from the command line:

```
% curl http://www.cnn.com/ > www.cnn.com
% wget -O www.cnn.com http://www.cnn.com/
% lynx -source http://www.cnn.com/ > www.cnn.com
```

```
"www.cnn.com" is just an example output file name, keep in mind
that the shell will not like some of the characters that can occur
in URIs (e.g., "?", "&").You might want to hash the URIs,like:
```

```
% echo -n "http://www.cs.odu.edu/show_features.shtml\?72" | md5
41d5f125d13b4bb554e6e31b6b591eeb
```

("md5sum" on some machines; note the " n" in echo this removes the trailing newline.) Now use a tool to remove (most) of the HTML markup. "lynx" will do a fair job:

```
% lynx -dump - force_html www.cnn.com > www.cnn.com.processed
```

Use another (better) tool if you know of one. Keep both files for each URI (i.e., raw HTML and processed).

Answer One-

To Answer this question I have written a python code which reads the file containing all the links and create two text files. The first is created using curl,where it extract the links raw data. And the second uses lynx to get the data without tags. Both files are saved with a hash name. As I run the code I display the line count, the links name and the hashed name of the file.

Code-

Attached are:
(a3q1.py): the python code.
(finallinks.txt): the list of 1000 links.
(All folder): all the files containing the links data and the processed data.

Question Two-

Choose a query term (e.g., "shadow") that is not a stop words (see week 4 slides) and not HTML markup from step 1 (e.g., "http") that matches at least 10 documents (hint: use "grep" on the processed files). If the term is present in more than 10 documents, choose any 10 from your list. (If you do not end up with a list of 10 URIs, you've done something wrong).

As per the example in the week 4 slides, compute TFIDF values for the term in each of the 10 documents and create a table with the TF, IDF, and TFIDF values, as well as the corresponding URIs. The URIs will be ranked in decreasing order by TFIDF values. For example:

```
Table 1. 10 Hits for the term "shadow", ranked by TFIDF.

TFIDF TF IDF URI
-------------
0.150 0.014 10.680 http://foo.com/
0.044 0.008  5.510 http://bar.com/
```

You can use Google or Bing for the DF estimation. To count the number of words in the processed document (i.e., the deonminator for TF), you can use "wc":

```
% wc -w www.cnn.com.processed
    2370 www.cnn.com.processed
```

It won't be completely accurate, but it will be probably be consistently inaccurate across all files. You can use more accurate methods if you'd like. Don't forget the log base 2 for IDF, and mind your significant digits!

Answer Two-

Term searched: iPhone, I chose this term since most of the links were gathered based on people who are interested in technology. I needed to select only 10 documents with this term, but it turned out that the term was found in over 70+ documents so I randomly selected 10.

Searching criteria:
grep o i iphone *.processed — uniq c

- (-i)ignore case
- (-o)only matching
- (uniq -c) report or omit repeated lines prefix lines by the number of occurrence ...

The list of the selected uri:

- http://gigaom.com/2013/09/19/beware-command-center-in-ios-7-is-nice-but-could-make-your-lost-iphone-hard-to-find/
- http://9to5mac.com/2013/09/17/iphone-5s-to-be-constrained-at-launch-as-apple-preps-app-to-check-availability/
- http://store.apple.com/us/product/HA815ZM/A/tech21-impact-mesh-case-for-iphone-5?fnode=47
- http://bgr.com/2013/09/12/iphone-5s-fingerprint-scanner-motorola-mocking/
- http://blog.rememberthemilk.com/2013/05/manage-your-evernote-reminders-with-remember-the-milk/
- http://mashable.com/2013/09/10/iphone-5s-hands-on/
- http://shop.oreilly.com/product/9781933820170.do?cmp=tw-na-books-videos-product-dod_daily_tweet&code=DEAL
- http://sciencehouse.wordpress.com/2013/06/17/patent-perspiration-not-inspiration/
- http://science.time.com/2013/09/11/your-tiny-roommates-meet-the-microbes-living-in-your-home/

- `http://rootbridges.blogspot.com/2009/08/blog-post.html?m=1` ...

Now based on the word count on each link I calculated the TF for each link based on (word count matching/wc total), and I calculated the IDF by selecting Bing which result in (total docs in corpus: 20B) and (docs with term: 170M) After calculating the log based 2 we have IDF of (6.878) for all documents since we have searched the same term.

The TF calculations is as following:

- gigaom.com
  (160/2062=0.0775)
- 9to5mac.com
  (53/1469=0.0360)
- store.apple.com
  (78/2684=0.0290)
- brg.com
  (31/1094=0.0283)
- blog.rememberthemilk.com
  (8/899=0.0089)
- mashable.com
  (6/793=0.0076)
- sciencehouse.wordpress.com
  (4/966=0.0041)
- shop.oreilly.com
  (4/1105=0.0036)
- science.time.com
  (4/2342=0.0017)
- rootbridges.blogspot.com
  (7/9510=0.0007)

  ...

Links TFIDF:

Rank	TFIDF	TF	IDF	URI
1	0.5330	0.0775	6.8783	gigaom.com
2	0.2476	0.0360	6.8783	9to5mac.com
3	0.1994	0.0290	6.8783	store.apple.com
4	0.1946	0.0283	6.8783	brg.com
5	0.0612	0.0089	6.8783	blog.rememberthemilk.com
6	0.0522	0.0076	6.8783	mashable.com
7	0.0282	0.0041	6.8783	sciencehouse.wordpress.com
8	0.0248	0.0036	6.8783	shop.oreilly.com
9	0.0117	0.0017	6.8783	science.time.com
10	0.0048	0.0007	6.8783	rootbridges.blogspot.com

Question Three-

Now rank the same 10 URIs from question # 2, but this time by their PageRank. Use any of the free PR estimaters on the web, such as:

`http://www.prchecker.info/check_page_rank.php`
`http://www.seocentro.com/tools/search-engines/pagerank.html`
`http://www.checkpagerank.net/`

If you use these tools, you'll have to do so by hand (they have anti-bot captchas), but there is only 10. Normalize the values they give you to be from 0 to 1.0. Use the same tool on all 10 (again, consistency is more important than accuracy).
Create a table similar to Table 1:
Table 2. 10 hits for the term "shadow", ranked by PageRank.

```
PageRank URI
-----------
0.9 http://bar.com/
0.5 http://foo.com/
```

Briefly compare and contrast the rankings produced in questions 2 and 3.

Answer Three-

To get the page rank of the links I used the website: http://www.checkpagerank.net/ and the result is as following:

Page Rank:

Order	PageRank	URI
1	0.09	store.apple.com
2	0.08	gigaom.com
3	0.08	mashable.com
4	0.07	brg.com
5	0.07	science.time.com
6	0.07	shop.oreilly.com
7	0.06	9to5mac.com
8	0.05	blog.rememberthemilk.com
9	0.05	sciencehouse.wordpress.com
10	0.05	rootbridges.blogspot.com

TFIDF and Page Rank Comparison:

TFIDF	PR	URI
1	2	gigaom.com
2	7	9to5mac.com
3	1	store.apple.com
4	4	brg.com
5	8	blog.rememberthemilk.com
6	3	mashable.com
7	9	sciencehouse.wordpress.com
8	6	shop.oreilly.com
9	5	science.time.com
10	10	rootbridges.blogspot.com

In this list I have notice some ties in second place,third place and fifth place, so I sorted alphabetically. By comparing the two results I have noticed that gigaom.com and store.apple.com are still one of the top three on both lists, and that brg.com is the fourth in both lists, and rootbridges.blogspot.com is the last on both lists. So the two lists are somewhat similar.

Question 4- (for 3 points extra credit)

Compute the Kendall Tau_ b score for both lists (use "b" because there will likely be tie values in the rankings). Report both the Tau value and the "p" value.

Answer Four-

The Tau-b statistic measures the strength of association of two lists, with the adjustments of ties. Range from 1 (negative association)to +1 (positive association).

Based on the online calculator on the link:
`http://calculator-fx.com/post/calculator-result/kendall-tau-correlation`
The result was:
Kendall tau score= 0.422222
This indicate a 42 percent of association between the two lists.

On the other hand, based on another tool from the link:
`http://www.wessa.net/rwasp_kendall.wasp#output`
The results were:
Kendall tau score= 0.422222226858139
2-sided p-value=0.10740464925766
Score=19
Var(Score)=125
Denominator=45

So both tools gave same Kendall tau score, which is 42 percent of association between the two lists.