OLD DOMINION UNIVERSITY

CS 495: Introduction to Web Science
Instructor: Micheal L. Nelson, Ph.D
Fall 2014 Thursdays 4:20pm – 7:10pm ECSB 2120

Assignment # 3
Joseph Elder UIN: 00844802

Honor Pledge
I pledge to support the Honor System of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community it is my responsibility to turn in all suspected violations of the Honor Code. I will report to a hearing if summoned

October 2, 2014

Contents

1.1 Question 1

1.1.1 The Problem

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
```

Keep both files for each URI (i.e., raw HTML and processed).

If you're feeling ambitious, "boilerpipe" typically does a good job for removing templates:

https://code.google.com/p/boilerpipe/

1.1.2 The Solution

The majority of the solution is handled by the python program dl.py, named dl for download. The version of python used is 2.7.6. The program wget is used to download the raw HTML from each of the 1000 unique URI's supplied in the file links.txt from assignment 2. In addition to URIs, the file links.txt also contains metadata pertaining to each URI which must be stripped leaving only URIs to be processed. The simple python program, stripInfo.py, takes links.txt and strips the metadata for each line returning a file containing a list of only URIs with one URI per line called 1000URIs.txt.

The os python module is imported for this solution to allow certain operating system functionalities in python. The first is the ability to call shell commands inside of python. The function os.system(“somecommand”) works the same as calling “$ somecommand” from a Linux command prompt. This method is how wget is called for each unique URI from python. Each line is processed individually to download the raw HTML source for each URI. Since URIs can potentially contain characters which are not allowed in file names, to save the raw HTML a filename which represents each URI is generated by removing all potentially dangerous characters. The downloaded HTML is stored in that representative file in the directory named src.

All the raw HTML files are stored in the src directory. Using another function of the os python module “listdir(src)” a list is generated of all the filenames contained in src. The majority of all HTML markup is removed using “lnyx -dump -force_html” leaving a file which should be easier to process later. Each processed file is stored in the directory named prc.

1.2 Question 2

1.2.1 The Problem

Choose a query term (e.g., "shadow") that is not a stop word (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/
0.085	0.008	10.680	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
```

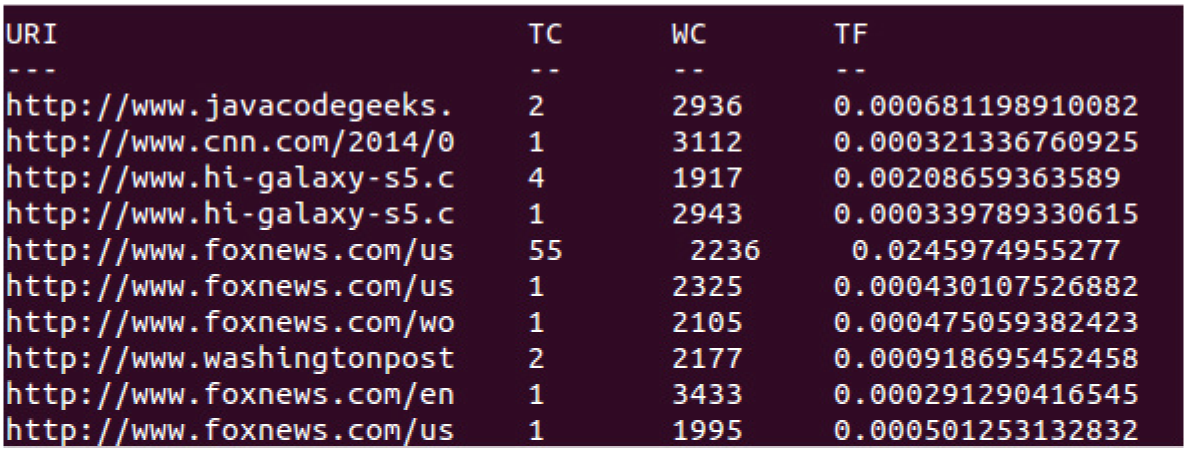
It won't be completely accurate, but it will be probably be consistently inaccurate across all files. You can use more accurate methods if you'd like.

Don't forget the log base 2 for IDF, and mind your significant digits!

1.2.2 The Solution

Using python(dl.py) files will be chosen to be searched for the word “shooting” from the prc directory. Out of the 1000 URI's only around 15 contain the term. The first 10 files which contain the search term are added to a list to be evaluated for TFIDF, TF, and IDF values. The URI of each file must also be supplied. Since the URI and filename are not identical but similar, the list of URIs is opened again. Each URI in the list undergoes the same string manipulations it did earlier in order to become the filename. If the modified URI matches the filename then the URI is returned for each of the 10 files processed by lynx.

In order to find TF, first the occurrences of the term “shooting” must be counted for each of the 10 files. The word count of each of the 10 files must be obtained as well. Python is used to generate the term count and word count. Terms are simply counted as they occur, words involve a more complicated process to count. They are counted by using the python method .split() which when called with no parameters splits up the text every time a space occurs in between two other substrings. The substrings could be considered words and are put into a list. The length of that list would be the number of words in the file. TF is generated by dividing term count by word count for each file.



```
URI                             TC       WC        TF
---                             --       --        --
http://www.javacodegeeks.       2        2936      0.000681198910082
http://www.cnn.com/2014/0       1        3112      0.000321336760925
http://www.hi-galaxy-s5.c       4        1917      0.00208659363589
http://www.hi-galaxy-s5.c       1        2943      0.000339789330615
http://www.foxnews.com/us       55        2236      0.0245974955277
http://www.foxnews.com/us       1        2325      0.000430107526882
http://www.foxnews.com/wo       1        2105      0.000475059382423
http://www.washingtonpost       2        2177      0.000918695452458
http://www.foxnews.com/en       1        3433      0.000291290416545
http://www.foxnews.com/us       1        1995      0.000501253132832
```

Figure 1: Python output showing abbreviated URI, term count, word count, and TF

To calculate IDF for the term “shooting” it must entered into a search engine. Using Bing we can assume 20 Billion total documents in corpus. “Shooting” returns 202,000,000 total documents with the term. IDF is calculated by taking the base 2 log of the quotient of the total documents in corpus by the total documents returns 6.63.

TDIDF	TF	IDF	URI
.0046	.0007	6.630	http://www.javacodegeeks.com/2013/10/hello-world-what-every-cs-student-should-know-about-the-first-job.html?utm_content=buffer41ae7&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer
.0020	.0003	6.630	http://www.cnn.com/2014/09/24/living/censoring-history-schools-denver-protest/index.html
.0139	.0021	6.630	http://www.hi-galaxy-s5.com/2014/03/samsung-galaxy-s5-camera-and-gallery.html?utm_content=buffer99adb&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer
.0020	.0003	6.630	http://www.hi-galaxy-s5.com/2014/02/samsung-galaxy-s5-hands-on-and-initial.html
.1631	.0246	6.630	http://www.foxnews.com/us/2014/09/24/3-dead-in-shooting-at-alabama-ups-facility-report-says/
.0027	.0004	6.630	http://www.foxnews.com/us/2014/09/24/65-ton-armored-vehicle-to-roll-through-pa-woods-in-hunt-for-suspected-cop/?cmpid=cmty_twitter_fn
.0033	.0005	6.630	http://www.foxnews.com/world/2014/09/24/australia-police-kill-terror-suspect-who-stabbed-2-police-officers/
.0060	.0009	6.630	http://www.washingtonpost.com/posteverything/wp/2014/08/05/these-people-i-interviewed-in-iran-clearly-loved-the-country-so-why-did-it-put-them-in-jail/
.0020	.0003	6.630	http://www.foxnews.com/entertainment/2014/09/24/orange-is-new-black-star-michael-harney-on-playing-bad-guys-but-keeping-god/?cmpid=cmty_twitter_fn
.0033	.0005	6.630	http://www.foxnews.com/us/2014/09/24/residents-criticize-roadblocks-in-search-for-pennsylvania-ambush-suspect/

1.3 Question 3

1.3.1 The Problem

```
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.
```

1.3.2 The Solution

PR Checker (http://www.prchecker.info/check_page_rank.php) was used to compare page rank of the 10 URI's containing the search terms however all 10 URI's were not available for page rank using the tool. Since page rank could not be calculated for the URIs themselves the page rank has been found for their top-level domains. By using the page rank of the top-level domains a great deal of accuracy is lost, however based on my results I would venture to believe that URIs with higher TF and TFIDF values would be more likely to have a larger page rank.

Page Rank	Top-level Domain	Original URI
.4	http://www.javacodegeeks.com/	http://www.javacodegeeks.com/2013/10/hello-world-what-every-cs-student-should-know-about-the-first-job.html?utm_content=buffer41ae7&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer
.9	http://www.cnn.com/	http://www.cnn.com/2014/09/24/living/censoring-history-schools-denver-protest/index.html
0	http://www.hi-galaxy-s5.com/	http://www.hi-galaxy-s5.com/2014/03/samsung-galaxy-s5-camera-and-gallery.html?utm_content=buffer99adb&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer
0	http://www.hi-galaxy-s5.com/	http://www.hi-galaxy-s5.com/2014/02/samsung-galaxy-s5-hands-on-and-initial.html
.8	http://www.foxnews.com/	http://www.foxnews.com/us/2014/09/24/3-dead-in-shooting-at-alabama-ups-facility-report-says/
.8	http://www.foxnews.com/	http://www.foxnews.com/us/2014/09/24/65-ton-armored-vehicle-to-roll-through-pa-woods-in-hunt-for-suspected-cop/?cmpid=cmty_twitter_fn
.8	http://www.foxnews.com/	http://www.foxnews.com/world/2014/09/24/australia-police-kill-terror-suspect-who-stabbed-2-police-officers/
.8	http://www.washingtonpost.com/	http://www.washingtonpost.com/posteverything/wp/2014/08/05/these-people-i-interviewed-in-iran-clearly-loved-the-country-so-why-did-it-put-them-in-jail/
.8	http://www.foxnews.com/	http://www.foxnews.com/entertainment/2014/09/24/orange-is-new-black-star-michael-harney-on-playing-bad-guys-but-keeping-god/?cmpid=cmty_twitter_fn
.8	http://www.foxnews.com/	http://www.foxnews.com/us/2014/09/24/residents-criticize-roadblocks-in-search-for-pennsylvania-ambush-suspect/

References

1. https://docs.python.org/2/
2. https://www.bing.com/
3. http://www.prchecker.info/