CS595 Intro to Web Science, Assignment #2

Valentina Neblitt-Jones

September 26, 2013

1 Link Extraction from Twitter

Write a Python program that extracts 1000 unique links from Twitter. You might want to take a look at: `http://thomassileo.com/blog/2013/01/25/using-twitter-rest-api-v1-dot-1-with-python/`. **But there are many other similar resources available on the web. Note that only Twitter API 1.1 is currently available; version 1 code will no longer work. Also note that you need to verify that the final target URI (i.e., the one that responds with a 200) is unique. You could have different shortened URIs for www.cnn.com. You might want to use the search feature (Figure 1) to find URIs, or you can pull them from the feed of someone famous (e.g., Tim O'Reilly). Hold on to this collection. We'll use it later through the semester.**

The Files

Files Used to Complete Q1

1. TwitterLink.py - Gathering tweets from 27 users
2. tweetlink.txt - File created by TwitterLink.py - contains URIs retrieved from tweet
3. UnpackURIs.py - Unshorten links from tweets
4. unpackedURLs.txt - File created by UnpackURIs.py - contains full URIs
5. DedupeURIs.py - Remove duplicate URIs
6. uniqueURIs.txt - File created by DedupeURIs - contains unique URIs

Tweeters

I used 27 tweeters. I could have used less by continuing to loop through some of the more prolific tweeters' timelines, but I thought more tweeters would provide a better variety of links.

- Michael Moore
- Rachel Maddow
- New York Times
- Sesame Street
- The Daily Show
- Washington Post
- Barack Obama
- Cory Booker

- Joel Spolsky
- Governor Christie
- Planned Parenthood
- National Zoo
- United Nations
- Washington City Paper
- Entertainment Weekly
- TMZ
- NPR
- Virginia Tech News
- New York Public Library
- Library of Congress
- Chicago Sun Times
- Chicago Tribune
- USA.gov
- Harvard University
- NFL
- NPR News
- Neil deGrasse Tyson

The Execution

Figure 1:

2 TimeMaps Exercise

Download the TimeMaps for each of the target URIs. We'll use the ODU Memento Aggregator, so for example:
URI-R = `http://www.cs.odu.edu`
URI-T = `http://mementoproxy.cs.odu.edu/aggr/timemap/link/http://www.cs.odu.edu/`

Create a histogram of URIs vs. number of Mementos (as computed from the TimeMaps). For example, 100 URIs with 0 Mementos, 300 URIs with 1 Memento, 400 URIs with 2 Mementos, etc.

The Files

Files Used to Complete Q2

1. uniqueURIs2.txt - Copy of file created by DedupeURIs - contains unique URIs
2. GetTimemaps.py - Loops through unique URIs to aggregate and count the number of mementos for each
3. timeMapResults.txt - Output of the URIs and number of mementos found for each
4. TimeMapAnalysis.xslx - Excel workbook that has the raw data, processed data and the histogram

3 Carbon Date Exercise

Estimate the age of each of the 1000 URIs using the "Carbon Date" tool: `http://ws-dl.blogspot.com/2013/04/2013-04-19-carbon-dating-web.html`. Note you'll have to download and install; don't try to use the web service. For URIs that have >0 Mementos and an estimated date, create a graph with age (in days) on one axis and number of Mementos on the other.