CSCI 572 – Information Retrieval and Web Search Engines
Spring 2014

Spatial Search Using Apache Solr and Google Maps

Section – Professor Horowitz

Yang, Po-hsuan	Almaslukh, Abdulaziz	Alejo, Matthew	Liu, Yuexi
pohsuany@usc.edu	almasluk@usc.edu	matthema@usc.edu	yuexiliu@usc.edu

I. Geo-Tagging the Documents

As described in the assignment spec, the first challenge was to geo-tag the corpus of documents in the vault. Accomplishing this was a matter of extracting the most frequently occurring term in each of the pdf documents and using that term to receive a latitude and longitude for the document from the geonames.org dataset. Of course, we had to do our best to avoid stop words and also have our algorithm search geonames.org with the next highest-occurring term if the first one didn't produce any matches.

Retrieving the list of terms in each pdf was done through repurposing our first Tika assignment to extract the text of the pdf then trimming out any non-alphanumeric characters, as they were quite prevalent in the content that Tika would return. The words of the document were then sorted from highest to lowest occurring through the use of a hash map to track the frequency of each word. A simple sorting of this hash map based on frequency gave us the list of terms we needed to check with the geonames.org dataset to assign each pdf a set of coordinates.

We downloaded the US.txt dataset from geonames.org, which contained the names and coordinates of nearly every point of interest in the United States. For every term on the list, starting from the highest occurring, we would check if that term was in the dataset. If it was in the dataset, regular expression was used to extract the latitude and longitude; otherwise the next term would be searched. If no terms were found, a default value of zero would be set for both Latitude and Longitude.

II. Indexing into Solr

As a result of this procedure, we would have the name of the pdf we were processing, a complete list of terms that appeared in the pdf, and a latitude and longitude for the document. It was this information that we would be indexing into Solr rather than the actual pdfs, as it saved space and contained complete summaries of the pdf content thanks to the word list. All this information was formatted into an XML structure and output into a file. This XML was then inserted into the Solr index.

III. Google Maps Integration

The last step was to visualize our results using the Google Maps API and using markers to show the geo-location of each pdf document depending on the search query. Everything we needed to implement Google Maps and markers into our website was readily available in Google's documentation of the API, so the map was simple to integrate. Most of our challenge lay in querying the Solr index using

javascript. Once we ironed out the errors in getting our javascript to connect with the index, we were able to receive a JSON string containing the results of our query. From there, the name of the pdf and its coordinates were extracted from the JSON and given to our code that integrated Google Maps, which would in turn display the markers according to their location. Clicking on the marker would also show the user the name of the pdf that is tagged there.

IV. Results of the Project

Conspiracies and their search terms:

JFK : JFK, Dallas, shooter

UFOs : UFO, alien, extraterrestrial

Watergate: Watergate, Nixon, tapes

Link to video:
http://youtu.be/s8Y-M0owH2g