**Homework 4: Adding Spell Checking and AutoComplete to Solr Search Engine**

**Name: Vrushali Peshave**

**USC ID: 4171565740**

**School Name: USC Gould School of law**

**Steps followed to complete homework:**

1. Configured solr so as to enable in built SuggestComponent of solr

* SuggestComponent in Solr provides users with automatic suggestions for query terms. I have used this to implement a powerful suggest feature in my search application.
* I made all the required changes as described in pdf:

1. Added a search component in solrconfig.xml

2. Added a request handler in solrconfig.xml

3. The default values for the number of suggestions is set to 5.

4.Reloaded the core so as to reflect the changes made.

1. Implemented autocomplete feature using Jquery:

* Used autocomplete feature of JQuery UI (https://jqueryui.com/autocomplete/)
* As user types the character, characters are passed to get json result from suggest component of the solr
* For multiple queries, each time passed last query term to get the json result assuming previous query terms are correct according to user.
* I was getting **Cross Origin Request Blocked Error** when I made ajax call to solr server from my main php file(index.php) to get suggestions. Thus I created one more php file (suggestData.php) which acts as proxy server.
* Result obtained in json is parsed in 2nd php file itself and I returned all the suggested terms in descending order of the weights of term.
* While showing results for multiple words, I appended results of all previous suggestions along with suggestions of last query term.

1. Used Norvig’s program for spell correction.

* Norvig’s algorithm does spell correction uses big.txt as a dictionary so I appended all the content of the crawled data using python script (extractText.py).
* Algorithm uses dynamic programming technique to calculate edit distance between strings along with Bays theorem and gives correct result based on frequency of words in big.txt
* There was a bug in the PHP file (SpellCorrector.php) which was corrected by replacing line number 69 by the following code:
* if(array\_key\_exists($f, $model))

$model[$f] += 1;

else $model[$f] = 1;

* I used Beautiful soup the popular html parser to extract text from html files.
* Correct(String word) function in Spellchecker program gives the correct spelling for misspelled word
* I splitted all query terms and called this function on each query term for the spell correction.
* Like Google, when there are spelling errors, it corrects it and asks “Did you mean: Corrected word”. So I implemented same way. Corrections are provided and link to corrected query is presented as a link So that user can click and will get results for correct query.

1. Stop Words removal

* Solr has in built feature for stop words removal if JSON results contains any stop words. For each core, in conf folder there will be stopwords.txt file creted by solr when we create core. File contains no stop words.(Basically it is an empty file) I added stopword list into this file.
* It can be seen from schema.xml file which has followinh line:

<filter class="solr.StopFilterFactory" words="stopwords.txt" ignoreCase="true"/>

This tells to use stopwords.txt and ignoreCase is set to true,hence “An” or “an” are treated as same word “an”.

Thus for autosuggestion when I type “an” I don’t get suggestions for stop words like “an” and “and”. Also when I type “is” I don’t get results for “is“ as a suggestion.

1. Used in built solr stemmer for Stemming

* Solr has in built feature for stemming. It uses Snowball Porter stemmer to stem all the words. So “run”, “running” gets mapped to “run” which is a stem root for all
* To enable built in feature of stemming, I made following changes in schema.xml file: I added following line: <filter class="solr.SnowballPorterFilterFactory"/> in schema.xml. And it looks like:

<fieldType name="text\_general" class="solr.TextField" positionIncrementGap="100" multiValued="true">

<analyzer type="index">

<tokenizer class="solr.StandardTokenizerFactory"/>

<filter class="solr.StopFilterFactory" words="stopwords.txt" ignoreCase="true"/>

<filter class="solr.LowerCaseFilterFactory"/>

**<filter class="solr.SnowballPorterFilterFactory"/>**

</analyzer>

<analyzer type="query">

<tokenizer class="solr.StandardTokenizerFactory"/>

<filter class="solr.StopFilterFactory" words="stopwords.txt" ignoreCase="true"/>

<filter class="solr.SynonymFilterFactory" expand="true" ignoreCase="true" synonyms="synonyms.txt"/>

<filter class="solr.LowerCaseFilterFactory"/>

**<filter class="solr.SnowballPorterFilterFactory"/>**

</analyzer>

</fieldType>

**Analysis of Results:**

1. Output for some misspelled words by spelling correction project:

* goud gould
* profsoor professor
* calfornia california

Observations:

For query **goud** spelling correction gives result **gould** and not **gold** because frequency of gould is more in big.txt than gold.

1. Output of autosuggestion:

* gogould ,gouldlaw,gould.usc,government
* ne news,newsbid.com
* facult faculti
* us usc,usc.edu

Observations:

1. gould has higher weight than government so it is suggested in the top position.

2. As stemming is done,it stems faculty word and gives faculti as a suggestion.

3. For query “ us”, “us” does not come as a suggestion as it is a stopword.