Addendum to Project Assignment 2

The following discussion provides further details on the Query Suggestion project, i.e., Project 2, and offers other guidelines in the design and the implementation of the project.

1. Stopword removal

   While processing a user query *Q*, any leading stopwords of *Q* are excluded from consideration, whereas non-leading stopwords should be retained and considered for query suggestions. For example, “A” in the query “A workshop” is ignored and only “workshop” will be considered in making suggestions. (See the Stopword List of Project 1.)

2. Making possible suggestions for a given query *Q*

   a. Given a query *Q* with *m* (≥ 1) words, a suggested query must be of length at least *m* + 1. For example, given the query *Q*, “fish”, a suggested query can be “fish tank”. However, if *Q* is ”tropical fish aquarium”, then “tropical fish” is not a valid suggested query.

   b. For this project assignment, you can assume that no queries will be suggested until the user has entered a correctly-spelled word, i.e., you are not required to make any suggestions till a completed word is entered followed by either a space, tab, or new line, which serves as the delimiter of words.

   c. A query *Q* from the AOL log file is a suggested query *SQ* for *Q*, if *Q* has been modified to *SQ* within the 10-minute interval by the same user in a session. Consider the following data in an AOL user session, where XX:YY:ZZ is a time stamp:

      information 12:05:15

      information retrieval 12:05:25

      information retrieval system 12:06:01

   in this particular example, “information” is treated as *Q*, and "information retrieval" is a valid suggestion for Q. However, "information retrieval system" should not be

considered as a modified query for *Q*, since it is not directly modified from *Q*. However, if the given query Q is "information retrieval", then "information retrieval system" is a valid suggestion for *Q*, since it is directly modified from *Q* in the log file.

3. freq(*SQ*) and mod(*Q*, *SQ*) in SuggRank(*Q*, *SQ*)

   Given a query *Q*, the frequency of a suggested query *SQ* of *Q*, denoted freq(*SQ*), is the *normalized* frequency of occurrence of *SQ* in the AOL query logs. You may choose to use any approach to normalize *SQ*. One of the normalization approaches is to count the number of times *SQ* appears in the query log logs and then divide it by any suggestion that appears the most in the log files. The same idea can be applied to compute the normalized frequency of occurrence of *Q* that has been modified to *QS*, i.e.., mod(*Q*, *SQ*).

4. WCF(*Q*, *SQ*) in SuggRank(*Q*, *SQ*)

   Given a query *Q* with *m* (≥ 1) words and a suggestion *SQ* with *m* + *i* words, consider the last word in *Q* as $word_1$, and the first suggested word in *SQ*, i.e., the $m+1^{th}$ word in *SQ* as $word_2$, $WCF(word_1, word_2)$ is the WCF(Q, SQ). For example, if *Q* is “tropical fish” and *SQ* is ”tropical fish pond”, *WCF(“fish”, “pond”)* is the computed value of WCF(*Q*, *SQ*).

   You can access the website http://peacock.cs.byu.edu/CS453Proj2/ to obtain the WCF value of two stemmed words. You are required to stem the words to extract the WCF values (see the Porter Stemmer in Project 1 for stemming). The Project homepage includes a sample java program that demonstrates how to access the link and retrieve a WCF value. For Example, http://peacock.cs.byu.edu/CS453Proj2/?word1=fish&word2=pond returns the WCF value of “fish” and “pond”, which is 1.5967200397426E-6.

   A return value of -1 for two stemmed words indicates that there is no WCF value for the two words, and the WCF value of the two words should be treated *zero*. (The Java program uses Jsoup library. If you are using another programming language, you should find similar library to access the webpage.)