Ranking in a Domain Specific Search Engine

CS6998-03 - NLP for the Web, Spring 2008, Semester Project

Sara Stolbach, ss3067 [at] columbia.edu

Overview

The ultimate goal of my project is to create a search engine for clothing. However, the main focus is on improving the ranking of results by finding important domain specific features. This analysis can be used to build the search engine, but the techniques used can hopefully be used for other domains.

I plan on creating a domain specific ranking system that will be used in the clothing search engine and comparing it to a general ranking system. If time permits, I will create a web interface for the clothing search engine, but continuously improving the ranking system is the top priority.

Achievements

I have gathered data from three clothing websites (excluding names for privacy issues). The data contains clothing and accessories for Men, Women, and Children. There are a total of 4,988 documents (divided into 2600, 549, and 1839 documents each in the three websites). The data was cleaned and processed using a crawler and page extractor that I implemented in Java.

I've created a general index on the data using Lucene [1]. The index is used to analyze the data to obtain important domain specific features and also can be used as a baseline for the ranking. I've done some analysis based on document frequency. There are 29263 terms in the corpus in total. This includes numbers. I imagine that the list of important features will be around 250-500. Some potential features include blue, button, pants, white, men, and pink.

All the data collected currently will be used for training. After the data has been thoroughly analyzed and important features have been extracted, data from a new website will be used to test whether it is successful. I have two possible sites for training, but am excluding the names for privacy issues.

Finally, I've been spending time to get more familiar with Lucene as my future goals will be to use it on a deeper level.

Main Future Goals

The current index is a standard one included with Lucene (Using Lucene's demo). It does not stem the terms and only indexes unigrams. My first immediate goal which is currently being worked on is to create my own indexing tool using Lucene to get a better baseline and create a good list of important features. Features are found partially using methods, such as document frequency, and also manual analysis.

After an initial list of features has been created I will create a ranking system on top of the standard Lucene system which will alter the standard ranking to place focus on the important features in the query. Each feature will have a level associated with it, because some are more important than others.

It is obvious that all important features cannot be examined. Therefore, I will attempt to create additional general rules to improve the rankings of the results. For example, (as mentioned by Kathy in the proposal feedback), doing NP analysis on the query to focus on the noun phrase. Other rules that will be attempted are ones that deal with numbers such as price ranges.

If time permits I would like to use Wordnet [2] to obtain more features by examining synonyms. For example, Wordnet can be used to get more colors by using the ones that exist in the list of features.

Questions

I spoke to Kathy about methods for finding features other than document frequency. She didn't have any suggestions but mentioned that I should ask you as well. Do you have any ideas?

I also spoke to Kathy about numbers in text, such as size. Size is difficult to capture in the documents because of the html. Size tends to be listed in a dropdown and it is unclear that it is a size (For example, it could be quantity). The setup is also different for each website. At the moment I will be ignoring size, but I was wondering if you have any suggestions for labeling size automatically?

References

[1] Lucene: http://lucene.apache.org/

[2] Wordnet: http://wordnet.princeton.edu/