Ranking in a Domain Specific Search Engine

CS6998-03 - NLP for the Web
Spring 2008, Final Report
Sara Stolbach, ss3067 [at] columbia.edu

Abstract

A search engine that runs over all domains must give equal weight to all words. However, if a search engine is domain specific it is intuitive that some words are more important than others. These are words specific to the domain. They should carry different weight which will affect the ranking. In this paper I have explored search in the clothing domain. I have extracted important features and will discuss how I gathered them and why they are important. In addition, I will discuss how they are used to improve the rankings and evaluate its performance.

1 Introduction

It is well known that the popular search engine Google uses PageRank (1) as its core algorithm for ranking documents. PageRank is a very successful algorithm, because it associates document as being important based on who links to it and the importance of those links. However, sometimes PageRank isn't enough. In clothing, links can imply relation (such as recommended items), but it is not always the case. In addition, it is difficult to define the importance of a specific article of clothing. What makes one better than another? It is very much a matter of taste. Therefore, it is important to use context. Context is approached in this system using natural language processing techniques such as giving weight to domain related features and using part of speech to find the focus of the query.

Section 2 lists the components of the project, Section 3 discusses websites that provide clothing search, Section 4 describes the data and resources used, Section 5 explains the ranking methods, Section 6 evaluates the system, section 7 discusses the feasibility of the project, and section 8 discusses future work.

2 Components

1. Web Crawler including:
    - Crawling site for links
    - Extracting parts of page
    - Removing html tags
2. Clothing documents
    - In XML format using Xerces (2)
    - 4998 documents from 3 websites
3. IR (indexing and querying) using Lucene (3)
4. Part of speech tagging using Stanford Part-of-Speech Tagger (4)
5. Clothing feature list and scores

    - Format: “term \t score”
6. Ranking System built on top of Lucene (3)
7. Web Interface
8. Detailed evaluation of 55 clothing-related queries

3 Related Work

A number of clothing domain search engines were encountered upon researching the area. Some notable ones are Nexttag.com (5), Like.com, which uses vision to improve their results (6), shopping.com (7), and Become.com which uses what they define as *AIR technology* which is essentially page-rank but within a specific area based on context (8). Nexttag.com and Shopping.com do not mention how they produce their results.

They all work fairly well and Like.com is especially interesting. All but Like.com run on multiple domains which could be detrimental due to word ambiguity. Even though these sites exist it is still a young market and there is much more work that could be done. The sites that run on multiple domains must use some method of clustering either automatically or manually. They also have a lot more data which will automatically produce relevant, though not necessarily wanted, results. The downside to most of these websites is that they tend to use e-bay and other personal selling sites heavily as opposed to manufactures. In addition they all use catalog format instead of just a simple search engine design which would be to contain one text box and a list of results.

4 Data and Resources

Data was extracted from three clothing websites were. It contains men, women, and children clothing as well as accessories. The dataset consists of 4998 documents, divided into 2600, 549, and 1839 documents each in the three websites (The names are being excluded for anonymity) and 28,515 unique terms with stemming. The data was cleaned and processed using a crawler and page extractor that was implemented in Java. Each document was stored in XML format.

The domain specific search engine was built in conjunction with the Java-based search engine Lucene (3), and Stanford’s Part of Speech Tagger (4). The additions used to improve the results are explained in section 5.

5 Ranking Methods

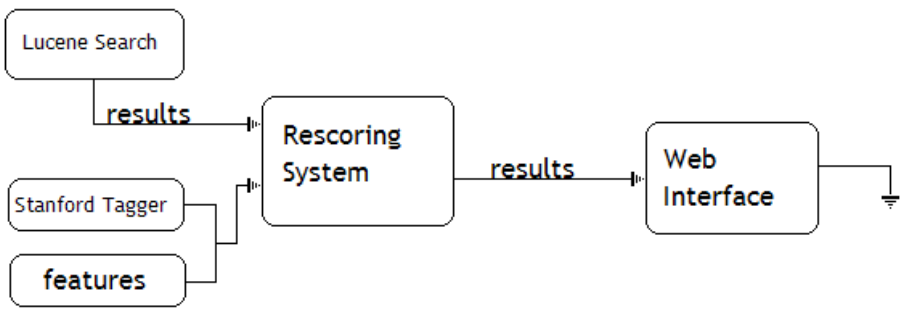

Figure 1 - Data Flow Diagram of System

The standard Lucene ranking was expanded upon to improve results. First, the core analyzer included additions that the standard analyzer does not include. The additions are ignoring common stop words such as “the”, ignoring case, and stemming words. The documents and primary scores were obtained following these additions and then the documents were re-ranked by attempting to find the focus of the query. Once the final results are received they are displayed to the user via a Web Interface (see the DFD in Figure 1).

The focus of the query is the important part of the query, usually the noun. For example, if a person is looking for ruffle skirt, returning a ruffle shirt will usually not be helpful. In other words, skirt is the “focus” of the search and while a ruffle skirt is wanted a ruffle shirt is not desired at all. The focus of the query was found by using domain specific features, discussed in Section 5.1 and part of speech, discussed in Section 5.2.

The main point of the ranking method is to reorder documents to in order to bring the better ones to the top. As will be clear in the evaluation section (Section 6), the baseline engine produced scores that were too close and did not differentiate heavily between different documents. The method described in this section successfully improved scores.

It is important to note that while the system was created with the clothing domain in mind, it can be used in any domain by simply providing a list of features for the desired context.

5.1 Features

As shown in the Data Flow Diagram in Figure 1, features are inputted in to the scoring system via a text file. The text file was simply “term \t score” per line. The list of domain specific features was created manually by examining high frequency features in the documents. Terms were considered to have high frequency by document frequency. The list had to be sifted through manually because there are some terms that appear frequently that should not be considered. For example, there are terms that occur in every page of a specific dataset – they are clearly not relevant. In addition, certain category words are not significant since they appear on virtually every page. For example, color. On the other hand, specific colors are relevant such as black and white. There are certain words that are obvious such as pant, skirt, women, and men. However there are others that are not as obvious. One word examined in the top document frequency list that is not immediately thought of is button. Some other examples of useful features are gown, and knot.

The general scoring rule for scores was major categories such as Men, Women, and children received a score of 3, general categories such as skirt, pant, and shoe received a score of 2, and attributes such as black, silver and bead received a score of 1.5.

5.2 Part of Speech

In addition to using domain features, The Stanford Part of Speech Tagger (4) was also used to improve scores. The Stanford POS Tagger was inputted in to the ranking system (See Figure 1). If a term was not in the list of features it was analyzed for part of speech. If it was a noun and the document contained the word, it increased the score by 1.75 points and if it was an adjective and the document contained the word, it increased the score by 1.25 points.

Using Part of Speech was very helpful when a term was not recognized. If a term is not recognized it does not imply that it is not important, it merely implies that we do not know it. Such an example is *clog*. If the term *black clog* is queried, documents with the word *black* would receive high score because it is in the feature list

but documents with the word *clog* would not, because it is not in the feature list. However, using POS will cause documents with the word *clog* to also have an increased score because it is a noun.

6 Evaluation

Evaluation Method	Top 1	Top 10 Exact	Top 10 Exact + Close	Score Min	Score Max	Num Results
Best	**69.1%**	42.4%	**81.8%**	13.35	16.07	678.25
Base	63.6%	34.5%	65.6%	0.00	0.30	406.29
No POS	65.5%	40.5%	**79.0%**	12.30	14.71	678.22
Only POS	67.3%	41.9%	74.3%	0.12	0.12	678.25

Table 1 - Evaluation Statistics

Evaluation was performed on four different versions of the search engine. The baseline, BASE, was the demo search engine included with Lucene (3). All the other versions had stemming, same case, and common stop words ignored. The first, POS only, used just part of speech to improve ranking, the second, No POS, used just clothing domain specific features, and the final search engine, BEST, is the search engine described in the paper that includes part of speech and domain specific features.

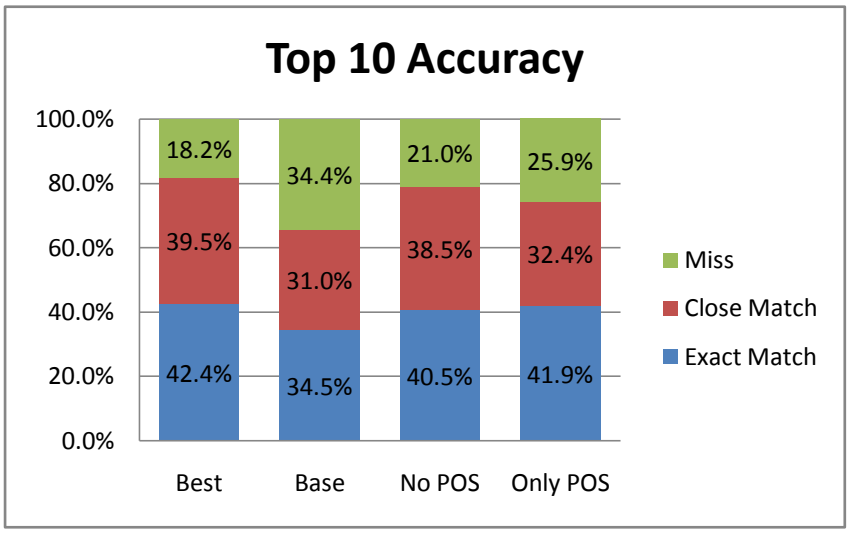

Figure 2 - Accuracy of Top 10 results. Blue implies exact match, Pink close match, and green implies miss.

The results described in this section were created by analyzing the 4 versions on a list of 55 queries. The queries were created manually by asking people to name searches as well as looking at some of the words found in the documents, to see if those results would come up when queried. Queries were only examined if results were produced for at least one version of the search engine.

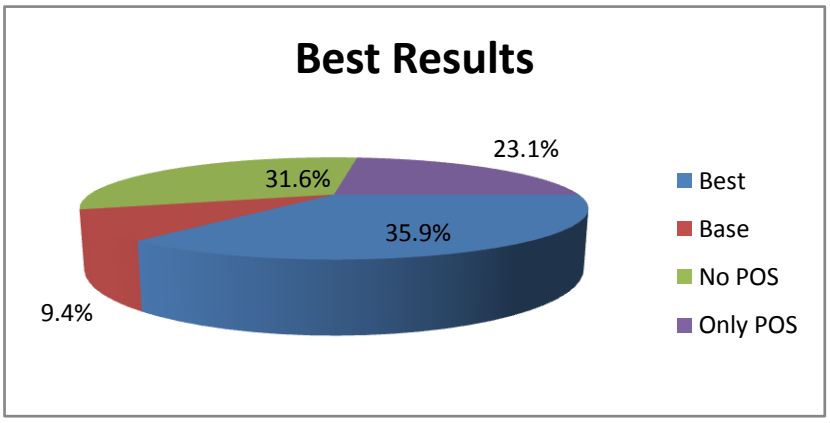

Figure 3 - Percentage of time each search engine was considered to produce the best results.

Figure 2 displays the accuracy of the top 10 results produced by the queries. The results were considered to be an exact match if the item matched

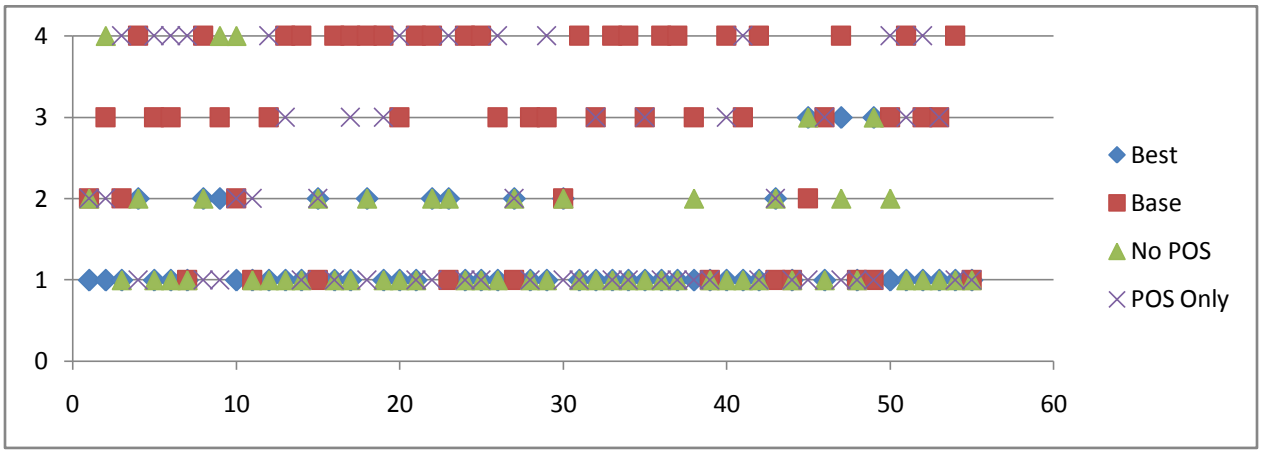

Figure 4 - Comparison of order of placement (1st-4th) for each version depicting which was considered to be the best for that query.

exactly. For example, *black skirt* produced *black skirt*. The results were considered a close match if the focus of the query was in the title of document (as that part of speech), or clearly visible in the image. For example, *black skirt* produced *brown skirt*. The results were considered to be a miss if they did not match at all. For example, *black skirt* produced *black linen*. The most interesting results to look at are the combination of Exact and Close matches. This is especially useful because exact matches may not be very high due to lack of data. If there are only 5 black skirts, it is impossible to receive 10 in the results. As can be seen in Table 1, BEST has the best results of 81.8% an improvement of over 15% of the baseline. While No POS and POS only do well on their own, together they do even better. Using part of speech gives just a 3% increase to using domain specific features only. In fact, quite often part of speech was not used since the term was in the domain. However, part of speech is important for uncommon domain features as mention in Section 5.2. The variation of the search engines were also compared by ordering them as to which appeared to have the best overall results each time. Figure 3 shows the percentage of time each search engine came in first place. Again, BEST, came in first place the most at 35.9% with No POS, or only domain specific features in second at 31.6%. There were in fact many overlaps which cause the percentages to appear less. The best search engine was in 1st place 76.4% of the time, 2nd 18.2% of the time, 3rd 5.5% of the time and never in 4th place. A comparison of order of placement is visible in Figure 4; it is clear that Best tends to be in 1st place very often, the baseline in 4th place and Best and No POS tend to overlap in first place (and in fact were often identical results).

A few other interesting points worth mentioning is that on average there were 406 results for the baseline and 678 results for the best and other variations. The baseline has fewer results because it does not use stemming. In addition, the scores had an increase in over 13 points in the Best search engine as shown in Table 1.

Therefore, the search engine described in this paper was the best in all evaluation methods, whether looking at just the first result (69.1% accuracy), ordering of the search engines, or top 10 results.

7 Feasibility

This system can easily be ported to other domains. In addition it can easily be used on many other sites as long as the data can be obtained and cleaned. Obtaining and cleaning the data can be difficult (though doable). Each website must be treated individually as described in the next paragraph.

A lot of time was spent on obtaining and cleaning up the data. Extracting data can be difficult depending on the construction of the website. In particular, dynamic data generated via JavaScript can cause data to not be available through an extraction script. In addition, in an ideal situation the document would only contain domain relevant data, and not information about the website. Preprocessing is very time consuming and was not the focus of this project. Therefore most of the irrelevant data was simply ignored. This is acceptable because a person should only be searching for domain related terms. If they do search they should know that they won't get reasonable results. For example, *e-mail a friend*, is not an item of clothing. In addition, such terms will tend to not affect the top features because they are too frequent, appearing in most of the documents for the specific website. Some other difficulties related to data were due to the little useful unique text in each document. Each document will have one small paragraph describing its item. In addition, different sites have different weights when they shouldn't because of repetitive words. For example, if one website repeats the title of the document 3 times it will get higher results even though it is not more important than the document in the other website that contains the title only once.

8 Future Work

The system described is an excellent foundation for domain specific search. One interesting addition to improve the important features would be to use synonyms in WordNet (9). For example, to obtain more colors that may not have been picked up in the dataset.

It also may be worthwhile to explore creating clusters of related terms. For example, all clothing articles, or genders. This could be beneficial because if a person searches for women clothing, all other genders can be excluded. However, it may not give a large boost in scoring and can also be detrimental when multiple fields in a cluster are desired or unimportant. For example, getting a different color is sometimes acceptable.

It would also be useful to use the title of the webpage to improve a documents score. In other words, if the document contains the query word in the title it should carry more weight than in just the text.

In addition, bigrams were not used because of the way Lucene is implemented. It would also greatly increase the index size. It may be worthwhile to examine in the future.

Finally, numbers propose a difficult problem. There are useful ones, such as size, and un-useful ones, such as quantity. In addition, they are usually stored in dropdown boxes which cause difficulty in associating them properly. It would be interesting to analyze what sort of techniques can be used to gather this useful information successfully.

9 Conclusion

The results of this paper show that associating special ranking rules with natural language processing techniques by using domain specific features and part of speech significantly improves search results. In addition to being successful in the clothing domain the methods can easily be adapted to any domain by simply creating a new list of features which make it a very useful and well-rounded method.

10 Bibliography

1. **Brin, Larry Page and Sergey.** PageRank. *Google.* [Online] http://www.google.com/technology/.
2. *Xerces Java Parser 1.4.4.* s.l. : Apache Software, 2001.
3. **Bialecki, Andrzej, et al.** *Lucene.* s.l. : Apache Software Foundation, 2008.
4. **Toutanova, Kristina, et al.** *Stanford Log-linear Part-Of-Speech Tagger.* Stanford, CA : s.n., 2006.
5. www.nextag.com. *Nextag.* [Online] www.nextag.com.
6. www.like.com/aboutus.py. *Like.com.* [Online] www.like.com/aboutus.py.
7. www.shopping.com. *Shopping.com.* [Online] http://www.shopping.com/.
8. http://www.become.com. *Become.* [Online] http://www.become.com/technology.html.
9. **Fellbaum, Christaine.** *WordNet: An Electronic Lexical Database.* s.l. : MIT Press, 1998.