USING INFORMATION RETRIEVAL TECHNIQUES FOR KEYWORD EVALUATION AND EXTRACTION

Christopher Burrows
Ricardo Malheiro
Miguel Torga University College
Information Systems and Technology Laboratory
Largo da Cruz de Celas Nº 1, 3000-132 Coimbra - Portugal
{cburrows,rsmal}@ismt.pt

ABSTRACT

With the growth of online businesses, it is necessary for consumers to have easy access to the desired product. This access is usually achieved through search features which associate lists of keywords to the available products or by browsing through the different categories. Using Information Retrieval techniques like indexing and searching, this paper shows how to create wordlists from the collections of documents sold by an online publisher and compare the lists of associated keywords with the indexes so as to evaluate their completion, and if new keywords are obtained, a proposition will be made to be added to the existing lists. This will be particularly useful for the consumers whose access to the documents will be simplified, and to the business itself who will obtain customer satisfaction.

KEYWORDS

Document Indexing, Keyword Extraction, Taxonomy, Information Retrieval Tools.

1. INTRODUCTION

The problem that led to this study was proposed by a company whose business model is strictly Internet based. This company supports the authors' creative process and both their and their works' visibility, as well as streamlining communication between researchers, entities responsible for R&D and civil society.

The company in question publishes articles, theses, literary and technical papers, skills and other sorts of information, such as opinion papers (e.g., summaries and reviews) in a range of knowledge areas (e.g., environment, biotechnology, health, Portuguese literature). To have their works published, authors must create an account in the system and submit their work, selecting the area of knowledge with which it is associated from a given list. When submitting they must also specify up to five keywords from a list suggested by the system for the current domain, which they consider identifies their work. They must also suggest other keywords which they think are more representative of their document. These words will be processed by the system and may be included in the list of words associated with the domain. In this process of disclosing their accomplishments, authors must pay a fee, and in return, not only do they obtain the recognition of having their work published, but they also receive royalties from its sale.

For this reason, it is obvious that the company must have its contents organized in a way such that the people interested in accessing them may do so in a simple and intuitive manner, whether by using the search option, or through browsing each area of knowledge.

For this, the documents are categorized in a taxonomic tree with several levels, depending on the domain in which they are classified (as mentioned above), and according to the domain and sub-domain selected by the author at the time of submission. The suggested keywords may be added to a list already existing in the system, used as a basis for customer searches for the article they want to find.

As the keywords are suggested by the author, they are very likely to be the ones which best describe the paper from his/her point of view, but they may neither be the most logical from the future customers' point of view, nor which have greater representativeness in the existing documents. Furthermore, no linguistic processing has been applied to the words in the list, which leads to there being many repeated words in different forms (e.g., several forms of the same verb or words with suffixes).

The aim of this work is to verify that the words in the keyword list do, in fact, represent their associated documents, and to identify the words which have greatest representativeness in the documents within each area of knowledge, so that in future, they may be added to the list, with the ultimate aim of making it easier for the company's customers to access the publications in which they are interested, which will also lead to authors becoming more interested in and continuing to use the system.

Initially this was to be done by analyzing the groups of documents within each area of knowledge and creating an index of the words they contain, excluding most frequent words, converting verb forms to their infinitive and reducing indexed words to their root morpheme (e.g., the word "happily" is composed of two morphemes, "happy" - which is the root morpheme – and the affix "-ly"), so that the index would only contain the words which are relevant to the study.

However, as the process developed, the conclusion reached was that part of this analysis was not necessary, and another analysis, which had not been planned for, was. This extra analysis consists of indexing expressions which contain multiple words, and it has to be done as some of the keywords are composed of multiple words.

Once this index had been created it was compared with the list of keywords to study their representativeness, and the index itself was analyzed so that other words with high representativeness could be identified and eventually be added to the suggested keywords.

Subsequently, the possibility of automating this process for the publication of new documents has been considered, which will maintain a level of consistency between the documents, the taxonomy and the list of suggested keywords.

This process was done using modern Information Retrieval (IR) [1] techniques, specifically using Apache Lucene (http://lucene.apache.org).

2. APACHE LUCENE

In the selection process, a superficial comparison of five Information Retrieval tools/libraries was made, based on the following characteristics:

- Type of text supported, because the documents to be analyzed consist of unstructured text, and some indexing tools only support Structured/Semi-structured text (e.g., XML documents).
- Support for custom stopword lists. 'Stopwords' or 'stop words' are "words and symbols which constantly appear in texts, and therefore do not add any value to the determination of their contents", [2] (e.g., "the", "or", "and"). As the analyzed texts are mainly in Portuguese, it is important that the tool can accept a custom stopword list as input so that these words are not indexed. This also reduces the index's size and makes searches more efficient.
- Support for Portuguese stemming. A stemming algorithm is "a procedure to reduce all words with the same stem to a common form" [3]. This "common form" is the part of the word which gives a general idea of the concept it describes, and it is necessary to avoid the consideration of words which are essentially repeated.
- The words should be stored in a vector, ranked by their relevance. This means that they will be stored in such a way that a word which occurs more often in a document or group of documents is considered more relevant than a word with fewer occurrences.
- And, in case the process is automated some time in the future, it is important that it is, in fact, a development library.

Of the tools, Apache Lucene came up as being the most complete and user-friendly, and was used throughout the rest of the project. This selection was partly influenced by the extensive documentation and support available. Apache Lucene is a high performance IR Application Programming Interface (API) which allows the inclusion of indexing and search capabilities in applications developed in any programming language to which Lucene has been ported. Initially developed in Java, it has currently been ported to other programming languages and is used in a variety of web pages and applications.

Even though the use of stemming algorithms has been dismissed thus far, it cannot be considered useless for future work or improvement.

3. KEYWORD EVALUATION

Once the tool was selected, an Indexer was created to handle the document formats to be analyzed. This process uses the Microsoft Word Document parser from Apache POI Project (http://poi.apache.org), PDFBox (http://www.pdfbox.org), for handling Adobe Acrobat documents, and Java's own handler for Rich Text Format (javax.swing.text.rtf). Also, this Indexer makes use of Lucene's StandardAnalyzer, and the NgramAnalyzerWrapper by Sebastian Kirsch (http://www.sebastian-kirsch.org), to create an index containing the necessary N-Grams.

Once the index was created, it was necessary to build a class which would load the keywords to be evaluated. These are stored in a raw text file, and when the Keyword Evaluator is run, it reads the text file and stores the lines in memory, to compare the words with equivalent terms in the index.

The result of this comparison is a report containing the terms, the documents in which they occur, each term's average term frequency, inverted document frequency, tf-idf (Term Frequency-Inverted Document Frequency), and a list of the documents associated with the keyword.

Analysis of this report showed that terms with a tf-idf over 0.8 were, in general, irrelevant, and not particularly specific to the document subject. However, those with a tf-idf value lower than 0.8 were clearly related to the documents they were found to be associated with.

4. CONCLUSION

From the study concluded so far, we have been able to identify the keywords in the list which are more representative of document contents, and those which are not. We have also been able to specify relations between documents which had not been considered previously.

Work so far suggests that it will be possible to create an automatic keyword suggestion system on submission of a document, based not only on term frequency, but also on terms associated with similar documents. This can be done by extracting the terms from the document while submitting it, and rating them by usage in the keyword list and the index itself, while maintaining the 0.8 tf-idf limit to filter out those considered irrelevant.

REFERENCES

[1] Baeza-Yates, R., Ribeiro-Neto, B. 1999. *Modern Information Retrieval 1st Edition.* Addison Wesley, New York, USA.

[2] Malheiro, R. 2007. *Anotação Automática de Páginas Web em Português utilizando Tecnologias da Web Semântica.* PhD Submitted for Approval, Coimbra, Portugal.

[3] Lovins, J., 1968. *Development of a Stemming Algorithm.* Mechanical Translation and Computational Linguistics 11, pp. 22-31.

Book

Baeza-Yates, R., Ribeiro-Neto, B. 1999. *Modern Information Retrieval 1st Edition.* Addison Wesley, New York, USA.

Gospodnetic, O. and Hatcher, E., 2005. *Lucene in Action.* Manning Publications Co., Greenwich, USA.

Gülsen, G., 2005. *Information Retrieval Services for Conceptual Content Management: Evaluation and Systems Integration*. MSc Thesis, Hamburg, Germany.

Journal

Singhal, A., 2001. *Modern Information Retrieval: A Brief Overview*. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering 24, pp 35-43

Lovins, J., 1968. *Development of a Stemming Algorithm*. Mechanical Translation and Computational Linguistics 11, pp. 22-31.