REINA at WebCLEF 2007. Selecting Usefull Snippets

Carlos G. Figuerola, José L. Alonso Berrocal, Ángel F. Zazo Rodríguez, Emilio Rodríguez
REINA Research Group, University of Salamanca
reina@usal.es

Abstract

The task for this year consist in retrieve snippets or pieces of text from web documents about several topics. The extraction of such snippets can be approached in several ways, as well as the selection of most usefull of them. We describe the segementation process adopted, and the selection of snippets carried out.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [**Database Managment**]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Web retrieval, Text segmentation

1 Introduction

This year, the WebCLEF track is focused in retrieving text snippets or fragments of web pages which bring up information about a topic; additionally, snippets must be in a language from a set of accepted ones. As a point of start, we have a set of topics, each with a title and a short description, as well as several documents or *known sources* about the topic. Additionally, for each topic, we have one or several searches in Google, with the first 1000 documents retrieved.

The general approach, by our part, consists of considering, for each topic, the documents retrieved by Google as the document collection with which to work. Since the task demands to obtain snippets, these documents must be divided in fragments or snippets, each one of which will be considered like an independent document.

As the query for each topic, we can use the description that we have for each one of them. Additionally, this query can be enriched with more terms from the *known sources.* Also we can use the available anchors which point at documents recovered by Google.

Finally, it is possible to apply filters or restrictions that eliminate those found documents that they are not in some of the accepted languages for each topic.

In this way, the task can be approached like a classic problem of retrieval, and apply, consequently, conventional techniques.

2 Building the database of documents

As we said, the collection or database of documents will be formed by snippets from the documents retrieved by Google for each topic. For each one of these topics, has taken place one or more searches in Google, and the (mor or less) 1000 first recovered documents have been taken for each one from those searches. This implies a variable number of documents by topic.

We have valued equal all the searches in Google for same topic. So, for each one of the documents retrieved by Google we would have to obtain the original document, to convert it to text, to disturb it in fragments, to obtain the terms of each fragment and to calculate its weights.

The organizers of the task have already solved the first part of such operations, since they have provided us the original documents as well as their conversions to plain text. In general, the conversion to flat text is good (this is a non-trivial aspect). Nevertheless, the use of codifications of characters is disparate, although it's affirmed that the plain text is codified in UTF-8. For languages that use characters noncontained in the standard ASCII, the codification and decoding of those characters are a source of headaches; the simple detection of the used system of codification is problematic in many cases. As an example, we have used the Universal Encoding Library Detector (chardet) [2], a module for Python based on the libraries for detection of Mozilla; which surprisingly indicates that most of the plain text versions is codified in Latin-2.

2.1 Segmentation of the text

To segment documents and to obtain fragments or short text passages can be applied diverse techniques. Basically, ones are based on the size in bytes, or words; and others are oriented in the separation in phrases or paragraphs [4]. The former techniques produce, of course, pieces more homogenous in size, but often devoid of sense, as the partition point is blind. The other techoques tend to produce fragments of very different size. In addition, its application not always is simple; in many cases the conversion to plain text of a web document loses the separations between paragraphs, nondifference between soft and hard line feeds, or blurs structural elements, like the tables.

A simplist approach, like the election of a orthographic character, as the period (.) like reference to fragment the text, tends to produce passages too short and, therefore, little useful for the objectives of this task. In our case, we adopted a mixed approach. After several tests, we decided that the suitable size for each fragment was around the 1500 bytes, but as we wanted fragments that had informative sense, our fragmenter looks for the period closest the 1500 bytes, and part by that point.

2.2 Other processes of lexical analysis

Some other transformations were carried out: conversion to small letters, removing accents, removing stopwords, (with a long list of stop words for all the accepted languages), application of a simple s-stemmer.

Each fragment thus obtained and transformed was considered an independent document. Terms were extracted and they were weighed according to scheme ATU (slope=0.2) [3], applying to the good well-known vectorial model.

2.3 Formation of queries

Somehow, the objective is to solve the task using conventional or already known retrieval techniques. From the document collection formed with snippets, we must select those that are more usefull for each topic. The key is in composing suitable queries that can produce this selection. As sources of information to compose those queries, we have topics with a short title and a brief description. Additionally, we also have, for each topic, a few documents denominated *known sources*, in full text. We also have the queries formulated to Google, but, since the document collection

	run 0	run 0.25	run 1
Precision	0.1415	0.1599	0.1624
Recall	0.1796	0.2030	0.2061

Table 1: Official Runs and Results

comes exclusively from the answers to such queries, the information contained in them already is taken advantage of.

So we can use topics (title and description) like nucleus of each query, and enrich this one with terms coming from the *known sources*. The *known sources* are complete documents, some of very long, which can contain many terms. It is possible to ask oneself if this will introduce perhaps too much noise in the query; a possibility is to weight the terms coming from those *known sources* in a different way that terms coming from title and the description from topics.

Additionally, it is also possible to consider the different structural fields from the *known sources* (title, body, headings, meta tags, etc.). Previous experiments, in previous editions CLEF [1], show importance of some of such fields, as well as the little interest of others. The most interesting field is anchors of backlinks. In this sense, since we very have a reduced document set, we do not have many backlinks with which to work; nevertheless, they seem specially important those that, from the *known sources* point at some of documents retrieved by Google.

Thus, we have used in the queries the terms of topics (title and description), plus the terms of the above mentioned anchors. To this, we have added the terms of the *known sources*, but weighted in different ways. In previous editions of WebCLEF we work on the use of different sources of information in retrieval and how to mix or to fuse these sources. In this time, we have chosen to modify the weights of the terms operating on the frequency of these in each document. The scheme of chosen weight also for the queries is ATU (slope=0.2), reason why the weight is directly proportional to the frequency of the term in the document; thus we have fixed a coefficient by which to multiply this frequency.

Runs carried out varies based on that coefficient: one of them maintains the same frequency, reason why the terms of the *known sources* weight just like those of topics. Another one run weights the terms of the *known sources* in a quarter (frec. x 0,25), and third it does not use these terms. The idea is to value to what extent such terms are useful or, on the contrary, introduce noise.

3 Results

Results of the trhee runs submitted show litle diference between. It seems that using terms of the *known sources* is more usefull than not. But we must note that several topics (about half of them) produce none usefull results. We not applied any restriction nor filter based on accepted languages; but restrictions, maybe, based on the type of information contained in the snippets would be desirable. By example, several of such snippets were references to another source of information (bibliographic references, academic courses on the topic, etc.). It seems that this type of information is not very usefull for this task.

4 Conclusions

We based our work on building queries with terms from the *known sources*, as well as with terms from the description of the topics. Using the terms from *known sources* produces better results. Nevertheless, it seems more interesting the process of obtaining of the text segments and the selection of these being based on its content type and its language

References

[1] Carlos G. Figuerola, José Luis Alonso Berrocal, Ángel F. Zazo Rodríguez, and Emilio Rodríguez. REINA at WebCLEF 2006: Mixing fields to improve retrieval. In A. Nardi, C. Peters, and J.L. Vicedo, editors, *ABSTRACTS CLEF 2006 Workshop, 20-22 September, Alicante, Spain. Results of the CLEF 2006 Cross-Language System Evaluation Campaign*, 2006.

[2] Mark Pilgrim. Universal Encoding Detector. http://chardet.freeparser.org.

[3] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 18–22, 1996, Zurich, Switzerland (Special Issue of the SIGIR Forum)*, pages 21–29. ACM, 1996.

[4] Ángel F. Zazo, Carlos G. Figuerola, José Luis Alonso Berrocal, and Emilio Rodríguez. Reformulation of queries using similarity thesauri. *Information Processing & Management*, 41(5):1163–1173, 2005.