A Methodology for Analyzing Web Search Results

Gloria Bordogna[1] and Giuseppe Psaila[2]

[1] Consiglio Nazionale delle Ricerche, Dalmine (Bg), Italy
[2] Università degli Studi di Bergamo, Facoltà di Ingegneria, Dalmine (BG), Italy

Abstract. A methodology based on the use of soft aggregation operators for filtering shared contents between the results of distinct Web searches, organized into granules of distinct resolution, is described.

Keywords. clustered results, soft aggregation operators, Web exploration.

1 Introduction

This work aims at improving the potential exploitation and comprehension of the contents retrieved by multiple Web searches to search engines [8]. In previous works, we approached this objective in several ways, by first proposing the use of operators to combine clustered results [1], then by the automatic generation of disambiguated queries from clusters [3], and finally by personalized facilities for re-ranking the clusters [2]. All these approaches were defined within the *Matrioshka* project, and implemented in the homonymous prototypal system.

In this paper, we describe a methodology for exploring the results of several web searches to filter out documents containing shared and correlated contents. Highlighting hidden content relationships between documents retrieved by distinct queries can help understanding the topics dealt with in the documents text, and, thus, give new hints of their relevance [8, 10]. In order to make this task feasible, without accessing the full text of a retrieved, our solution extracts the necessary information from within the contents reported in the result lists provided by the search engines [6,8,9]. Then, to analyse the content relationships between the retrieved documents we have defined soft operators based on fuzzy set theory [11].

2 Soft Operators for combining granules of search results

The finest information granule we consider is the *item i*, representing a document in a ranked list retrieved by a search engine as a result of a query evaluation. *i* is defined by an Uri_i , i.e., the Uniform Resource Identifier of the web document; its $Title_i$, $Snippet_i$ and Bag_i that is a bag of strings (single terms), each one weighted with a score in [0,1], expressing the significance of the string in representing the contents of the item. The strings in Bag_i are obtained by performing lexicographic analysis of Uri_i, $Titles_i$ and $Snippets_i$ of item *i* by applying *Lucene* functions, removing stop-words, conflating terms having the same stem, expanding single terms with associated terms by using *Wordnet* [7]; then, all the selected single terms in Uri_i, $Titles_i$ and $Snippets_i$ are included in the bag of strings. Each string s in Bag_i is then associated with a weight $w_s \in [0,1]$: an occurrence in the title is considered as twice occurrences in the snippet and Uri, and the total number of occurrences of a string is then normalized with respect to the maximum weight of the strings in Bag_i. An item i has also an $Irank_i \in [0,1]$ that expresses the estimated relevance of the retrieved Web document with respect to the query, and is computed as a function of the position of the item in the query result list normalized by the list's length. Thus, $Irank_i$ is independent of the actual relevance score computed by the search engine.

The intermediate information granule is the *cluster c*, that is a fuzzy set of items. It has a $Label_c$ that is the title of the item which is the most relevant in the cluster [1], and a $crank_c \in [0,1]$, that, by default, is defined as the average of the *Iranks* of its items, or can be computed based on personal preferences evaluating some cluster properties, such as the cluster cardinality, novelty, heterogeneoity [4]. A cluster can be generated by applying an operator combining two other clusters, or by a clustering operation. In this context, we do not focus of the clustering algorithm. For extracting the features necessary to cluster the items we parse the result list provided by the search engine, containing the first *N* results, and extract all the information which constitutes the representation of an item. In the *Matrioshka* system [2], *Lingo* clustering is applied [9]. We are aware that the effectiveness of the proposed approach strongly depends on the clustering. Nevertheless, the combination of clusters can aid to better understand the clusters' contents, and thus complements the information provided by a clustering algorithm.

The coarsest information granule is the *group g*, composed of ranked clusters. *g* has a $Label_g$ that semantically synthesizes its main contents. A direct way to generate a group is submitting a query to a search engine and cluster the *N* top ranked items in the results' list. Alternatively, a group can be generated by an operator working on groups [1]. When a group is generated by a query to a search engine, its label is the text of the query, otherwise it is the title of the most representative item of the group [1].

Notice that, the same web page retrieved by different search engines (or by different queries) may be represented by distinct items in distinct result lists. In this case, the document is uniquely identified by the same *Uri*, while it may have distinct *Snippet*, *Bag* and *Irank*. On the other side, distinct web pages with distinct *Uris* may share the same or very similar *Title* and *snippet*, because they are indeed duplicated documents at distinct web sites retrieved by the same query.

To filter documents retrieved by distinct searches that have different *snippet* and *bag* but same *uri*, we first introduced in [1] the ranked intersection, *RIntersection,* and the ranked union, *RUnion*, operations as the usual intersection and union of fuzzy set, since clusters are regarded as fuzzy sets of ranked items. They are crisp operations uniquely identifying the items by their *Uri*, which are compared based on an exact matching. The membership degree of the resulting item is obtained as the minimum and maximum of the *Iranks* of the items in *RIntersection*, and *RUnion*, respectively. To obtain the *Title*, the *Snippet* and the *Bag* of the resulting items, we select those belonging to the document having the minimum (in the case of *RIntersection*) or the maximum *Irank* (in the case of *RUnion*). By this choice we represent the cluster by its worst (best) representative in case of intersection (union), in accordance with fuzzy set theory [11].

Nevertheless, it can happen that the same web page is duplicated at distinct sites, so two web pages may differ just for their *Uris* while they may share similar *Titles*, *snippets* and *bags*. With the *RIntersection* and *RUnion* operations duplicated web pages are filtered out from the results. This could be a limitation, when one would like either to identify documents dealing with shared contents or to eliminate documents dealing with redundant contents. Let us consider, for example, the page of *Expedia* of the same hotel but retrieved in two different searches with two different dates of booking. They refer to the same hotel in the same Web site, but they have different *Uri*s. *RIntersection* considers these documents as distinct, even if their semantics is the same.

This is the reason for introducing the *soft operators* between clusters [4]. The soft intersection, *SIntersection*, and the soft union, *SUnion*, uniquely identify the ranked items by their *bags*, i.e., by fuzzy subsets on strings. A fuzzy relation between any two items can be defined to perform their partial matching as for two fuzzy sets. Thus *SIntersection*, and *SUnion*, are defined as the intersection and union of fuzzy sets of fuzzy sets [4].

In order to filter duplicated documents the *Soft Intersection* between clusters can be applied. The soft intersection relaxes the ranked intersection, so that its resulting cluster includes the results of the ranked intersection, plus other ranked items of the input clusters that share the most specific common contents, as represented by their bags of strings. Let us give a simple example. Given two documents, one dealing with Italian tourist places, and the second with Tourist places in the Mediterranean area, they probably share most of the places listed in the first document, but the vice versa is unlikely to occur, since the second document contains also places of other countries than Italy such as Greece, Spain and so on. So, the soft intersection retains only the shared contents, i.e., the first document on Italian places.

Conversely, the soft union restricts the ranked union, so that the resulting cluster is included in the results of the ranked union. *SUnion* generates a cluster that contains the results of the ranked intersection of the input clusters plus the most general ranked items that share common contents, as represented by their bags. Let us make an example: to have a panoramic overview of the Mediterranean Tourist information; having two documents, one dealing with Italian tourist places, and the second with Tourist places in the Mediterranean area, the second one is most general one and thus it is selected by the soft union. These operations between clusters are the basic bricks on which the operators between Groups of clusters were defined [1].

3. Conclusions

A methodology for exploring the results contents organized into information granules of distinct resolution (Groups, clusters and single documents) and obtained within a Web search process by querying possibly several search engines has been proposed. This method is based on the application of soft operators to combine pairs of granules to filter documents with shared contents. Ongoing research is aimed at improving the understanding of the results yielded by the soft operators, by providing new directions of navigation within the set of retrieved documents.

References

1. Bordogna G., Campi A., Psaila, G., & Ronchi S. A language for manipulating groups of clustered web documents results, In Proc. of the 17th ACM CIKM'08, 23-32. (2008)
2. Bordogna, G., Campi, A., Psaila, G., & Ronchi, S. A Cluster Manipulation Paradigm for Mobile Web Search Interaction. In Proc. of the 1st IIR'10, 53-57(2010).
3. Bordogna, G., Campi, A., Psaila, G., & Ronchi, S. (2009). Query Disambiguation Based on Novelty and Similarity Users Feedback, in Proc. of FQAS09, LNCS, Springer Verlag, 179-190 (2009).
4. Bordogna, G., Psaila, G., Soft operators for exploring Information granules of Web search results, submitted to the World Conference on Soft Computing (San Francisco, May 23-26, (2011).
5. Belew K. Adaptive information retrieval: Using a connectionist representation to retrieve and learn about documents. In Proc. of the 12th ACM SIGIR'89, 11-20 (1989).
6. de Graaf E., Kok J., & Kosters W. Clustering improves the exploration of graph mining results. In Proc. of AII'07, 247 of International Federation for Information Processing, Springer Verlag, 13-20 (2007).
7. Fellbaum, C. (Ed.) WordNet An Electronic Lexical Database. Cambridge, MA; London: The MIT Press. (1998).
8. Jansen, B. J., & Spink, A. How are we searching the World Wide Web? A comparison of nine search engine transaction logs. Information Processing and Management, 42, 248-263. (2006).
9. Osinski, S., &Weiss, D. A concept-driven algorithm for clustering search results. IEEE Intelligent Systems, 20, 48–54. (2005).
10. Roussinov, D. G., & Chen, H. Information navigation on the web by clustering and summarizing query results. Information Processing and Management, 37, 789 – 816. (2001).
11. Zadeh, L.A. Fuzzy sets. Information and control, 8, 338-353. (1965)