Multilingual and Multimedia Information Retrieval from Web Documents

Marta Gatius Manuel Bertran Horacio Rodríguez
TALP Research Center, Technical University of Catalunya, Barcelona
gatius@lsi.upc.es mbertran@lsi.upc.es horacio@lsi.upc.es

Abstract

Web documents present new challenges to conventional Information Retrieval (IR) technologies. This paper describes how these challenges are faced in FameIR, a multilingual multimedia IR shell. In this shell Cross-Language IR (CLIR) and query expansion are performed using EuroWordNet (EWN), the best developed and most widely used lexical resource for several languages. Techniques to extract information from Web documents, Wrapper Generation (WG) techniques, are used to access a finer information granularity than the whole Web page. By combining IR and WG techniques with the use of EWN, FameIR provides a powerful facility to perform CLIR from multimedia Web documents.

1. Introduction

The initial form of Information Retrieval (IR) reduces the search space to closed collections of textual documents in one language. However, there is an increasing interest in dealing with questions and documents in different languages and media, and in going beyond closed collections and performing IR over open and changing information sources as the Internet.

In response to these evolving needs, systems combining technologies from different fields have been developed. This paper describes how these challenges are faced in FameIR, a multi-language and multimedia shell by coupling techniques from several disciplines: IR, Wrapper Generation (WG) and Natural Language Processing (NLP).

Conventional IR technologies cannot be applied directly to Web sources (considering Web sites simply as collections of documents) due to the singular characteristics Web documents present in content, organisation and access. Obtaining information from the Web is usually performed by means of wrappers. In the Web environment, a wrapper can be defined as a processor that converts information implicitly stored in a document in HTML (or other code schema) into information explicitly stored as a data structure for further processing. The system described in this paper proposes combining IR and WG techniques for accessing a finer information granularity than the whole Web page.

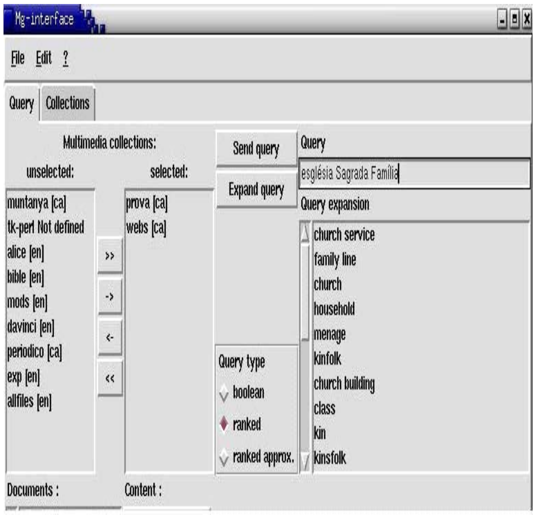
The large amount of documents in different languages available in Internet has increased the need for retrieving documents written in a language that differs from that of the query. Several lexical resources have been used to perform Cross-Language IR (CLIR). FameIR uses EuroWordNet (EWN), the best developed lexical resource for English, Spanish, and other languages (described in [9]). FameIR also uses the set of synonyms included in each EWN word-entry to expand the terms in the query and thus improve IR accuracy.

An overview of the FameIR shell is given in the following section. Section 3 describes the manner in which the FameIR shell handles the challenges that CLIR from Web documents presents. Section 4 describes the wrapper system incorporated to access Web contents. Section 5 draws some conclusions.

2. Overview of the FameIR Shell

The FameIR environment (see [5] for a detailed description of the system) was built within the framework of the European project FAME (Facilitating Agent for Multicultural Exchange, [10]). Basic capabilities of the FameIR shell include indexing, retrieving and presenting multimedia collections (including text, image, voice and video) and Web documents. FameIR also supports CLIR covering queries and documents in Spanish, Catalan and English and query expansion, using EWN.

The FameIR shell has been built over the *Managing Gigabytes* (MG), described in [16], which provides the basic IR capabilities. Term weighting is done using the conventional factors *term frequency* and *inverse document frequency*.



unselected:
muntanya [ca]
tk-perl Not defined
alice [en]
bible [en]
mods [en]
davinci [en]
periodico [ca]
exp [en]
allfiles [en]

selected:
prova [ca]
webs [ca]

Query
església Sagrada Família

Query expansion
church service
family line
church
household
menage
kinfolk
church building
class
kin
kinsfolk

Query type
○ boolean
● ranked
○ ranked approx.

Figure 1. The query page in FameIR

In the FameIR shell, CLIR is performed using EWN, a widely used lexical database for several languages. The main advantage of using EWN is that it provides relations between the word senses in different languages. Unfortunately, EWN, as most available thesaurus-like resources, has some limitations when used in CLIR. This is due to the limitations of its vocabulary coverage.

EWN is also used to improve IR by expanding the terms that comprise the query vector. This is accomplished by adding terms chosen from the list of terms correlated with the user's original terms. The correlated terms are obtained from the domain-independent lexical relation included in EWN. Expanding the terms in the query reduces the *Term Mismatch* problem (a relevant document is not retrieved if a term in the query does not occur in the document). Several authors ([8], [12], and [14]) have proposed different approaches to query improvement: document clustering, dimensionality reduction, document and query expansion, use of semantic and syntactic relations between words, etc. In [13], [15] and [17], we can find excellent reviews of the techniques applied to improve IR from multimedia documents.

A graphic interface has been incorporated into the FameIR shell to facilitate the operations of indexing and querying. In the example shown in Figure 1, two collections have been selected: *webs* (representing a set of Web sources that have been previously indexed) and *prova*. The query *església Sagrada Família* has been introduced. This query has been translated and extended by choosing several words from those provided by EWN. The kind of query can be also chosen (by default ranked).

3. Combining IR, WG and NLP Techniques

3.1. The challenges of accessing the Web

Conventional information retrieving technologies cannot be applied directly to access information on the Web. The main reasons are the following:

- The information is placed in heterogeneous sources with different ways of accessing it.
- Web pages change rapidly.
- Most Web sites are designed to facilitate browsing, not querying.
- Many Web pages are generated on the fly upon user's request.

There are many Web crawlers and Web directories (Yahoo!-like) designed to solve the problem of locating Web sites containing specific information. Nevertheless, many Web sources are not indexed yet by most well-known search engines. Most of these sources are Web services where pages are generated on the fly from some database. [1] claims that the content of these databases is 10 to 500 times the content of the visible Web. Moreover, many crawlers also avoid complex pages containing files in different formats and pages whose content changes very often. Although there are already several intelligent crawlers accessing hidden Web sources (such as BrightPlanet^{TM}) the problem of retrieving Web documents efficiently is not completely solved.

Furthermore, there is an increasing need for tools not only to retrieve collections of Web documents but also to extract the relevant information from these documents. In order to face the new challenges Web documents provide for the information extraction (IE) field (large volume, semi-structured organisation, and hyperlinked information), a new discipline within this field, the WG, has emerged in the last few years. A wrapper obtains specific information from a particular source. Although there are several commercial wrapper systems performing comparisons of specific information (e.g. the prices of a particular product) in different Web pages (such as the comparison-shopping systems Jungle, Jango and mySimon, described in [2]) their scope is usually restricted. Besides, the cost of generating a wrapper is still a problem.

3.2. Using WG techniques in FameIR

FameIR was primarily designed as an aid to communication in different languages. The main functionality of FameIR consists of providing online information (in different media) related to the topics appearing in a conversation. Retrieving information from the Web below document level is useful to achieve this goal because there are large documents covering different topics and containing information in different formats. Besides, retrieving the relevant parts of relevant documents from a set of previously selected Web sources seems more appropriate than retrieving all documents that can be found in the Web, as most crawlers do. For this reason, a system of wrappers was integrated into FameIR. The use of wrappers in the FameIR shell proposes an intermediate point between the IR model, in which a collection of Web documents are retrieved, and the WG model, in which very specific information from several documents is extracted. Additionally, the wrapper system provides a unified access to the contents of heterogeneous Web sources, even those in the hidden Web. The system of wrappers facilitates the adaptation of the retrieval shell to the user's needs and interests. By writing a wrapper for a specific Web document, users select the relevant information to be extracted from it (e.g., prices, specific images). The main goal when adapting the wrapper system has been reducing the effort required for writing a wrapper.

3.3. Indexing and querying Web documents

In the FameIR shell, the set of Web documents to be accessed during communication must be selected and indexed in advance. A Web document consists of a file containing the keyword describing the Web source and the wrapper extracting the relevant data in that source. The different types of sources that can be accessed are indexed, like the other kinds of documents considered in FameIR, by textual features (words or stems). These textual features are obtained from the keywords and should refer to the specific data (an image, a paragraph) to be retrieved from the Web document. The keywords can be introduced manually by the user or, alternatively, they can be extracted automatically from the document (e.g. from the title, the keywords, the image legends and other words in the page). The keywords attached to the Web services requiring parameters from the user may include all possible values of those parameters. The services requiring parameters usually include them in their web addresses. For instance, a service with two parameters will probably have an address similar to *http://serviceaddress/par1=par1value&par2=par2value,* where *par1value* and *par2value* are the values introduced by the user. At run-time those values are obtained from the terms contained in the user's query (by comparing each term in the query to the set of values allowed for each parameter).

Our approach to improve IR from Web documents exploits the organisation of the Web sources and focuses on improving user queries. Frequently, Web sites contain several pages sharing a similar structure and content. In this case, the Web document can be the entire Web site (or a set of its pages). The main advantage of indexing pages in a Web site by the words appearing in the site is that, as in document clustering, the queries are compared against the site, thus saving the cost of comparing those of each page.

In order to improve user queries, the terms of the user's query are expanded to include the synonyms provided by EWN. Given that usually not all these synonyms are highly correlated with the user's original terms in the set of Web documents consulted, the FameIR shell allows the user to choose the terms to be included. Expanding user's query terms seems appropriate in FameIR because the search space is reduced to a set of Web sites selected in advance (this approach would not be very appropriate to access the whole Web).

In FameIR, Web documents can be indexed and queried in English, Spanish and Catalan. The FameIR facilities include translation of the terms appearing in the indexes and the user's queries to the document language using EWN.

4. The System of Wrappers in FameIR

The system of wrappers used in FameIR was designed to provide Web access for different types of applications in the area of the NL communication. It had been previously integrated into a dialogue system on Web contents (GIWEB, described in [11]). The wrapper system was developed to extract information from different types of Web sources, including those not accessible for most crawlers. When adapting the system to the FameIR needs we have enhanced its capabilities with advanced features. New functionalities of the wrapper system include dealing with new types of Web sources, such as non-structured documents and complex documents with data in different formats and media (text, graphics and voice). The system also performs selectively hyperlink searching to obtain data from connected sources. Additionally, the wrapper system has been provided with a tool for building wrappers semi-automatically.

4.1. The language for writing wrappers

Wrappers exploit the document structure to extract the information required. In Web pages, the traditional NLP techniques for IE are not well suited. Since information on the Web is frequently presented in a structured or semi-structured form, the document organisation can be exploited instead of using linguistic knowledge. Moreover, even in not-structured Web documents HTML delimiters give relevant information. Most systems of wrappers use delimiter-based methods (such as HTML tags) to locate data.

In the system used in the FameIR shell, as in many wrapper systems, the HTML page code is represented as an HTML syntax tree where nodes are labelled by tag names. The information in the page is obtained by traversing the HTML parsed tree following the description of its location, provided by the wrapper program. The location of the data to be extracted is usually described by the tag names corresponding to the parse tree path to reach it.

Writing a wrapper for a Web source requires a considerable human effort. Frequently, the same wrapper can be used for many pages in a Web site because they share a common organisation. For example, pages generated on the fly from a database share the same structure because they are generated by filling the same template. However, when the page organisation changes (and it often does) a new wrapper program should be created. Several approaches are being proposed to reduce the cost of implementing a different wrapper for every page: special languages for writing wrappers (such as that described in [6]), helping tools ([3]), and wrapper-learning systems ([7]).

The wrapper system used in FameIR provides a special language for building wrappers. This language is simple and easy to use. A wrapper is generated by describing the data to be extracted from a page using this language. This description must be given following the general scheme shown in Figure 2. First, the Web source address and the Web source home address must be given (the home address is used to complete partial addresses appearing on the page). Then, the Web source type must be indicated. Two different types are considered: pages and services. In the case of services where parameters are required, these must be given. Next, the wrapper type must be stated. According to the organisation of the data to be extracted several types of wrappers are considered: the type *multimedia*, for pages containing data in different media, the type *table* for semi-structured pages where information is represented as tables or lists of values, etc. The internal properties of the data to be extracted must be included in the wrapper description. This information consists in the kind of data (e.g. for the type of wrappers *multimedia*, text, image, voice and addresses; for the wrapper type *table*, the type of table or list). If only part of the data must be extracted, specific information must be included about its organisation on the page (e.g., its relative location, from a specific point). Finally, the last line of the Web source description gives the absolute location (from the page top) of the data to be extracted.

1) Web source home address
2) Web source address
3) Web source type
4) Wrapper type
5) Internal properties of the data to be extracted
6) Data location

Figure 2: The definition of a wrapper

The absolute and relative location of data is given by the parsed tree path to be followed to reach it, that is, the sequence of the number and the HTML tag corresponding to the next daughter to access from a specific node in the tree (e.g. the path *body 2 table 1 tr* indicates that from the node with tag *body* its second daughter node, with tag *table*, must be accessed and from it, its first daughter, tag *tr*) . However, in many documents the data location cannot be easily described by the user (e.g. the parsed tree path to be followed may be too large). To solve this problem, our system allows easier ways to describe data location (similar to those described in [6] and [4]). The system accepts incomplete specified tree paths, disjunctive tree paths as well as other types of information describing the data location, such as strings delimiting it (the text appearing above, below and/or within) and spatial information (e. g. the second and fourth column in a table, the second image). For example, no data location is required in the definition of a wrapper to extract all files contained in a page in a specific media. In case of a wrapper in charge of extracting a specific image file its location can be given in different forms: the parsed tree path to reach it from the root node, the text appearing below it, the number of image files placed above it.

4.2. Generating wrappers semi-automatically

Our wrapper system includes a tool for building wrappers semi-automatically. This tool allows the user to generate a wrapper in an interactive way without knowing neither the specific details of the page organisation nor the language to describe it. Furthermore, the wrappers generated are less error-prone than those built manually by the user.

When using this tool, the user should only give one example of the data to be retrieved from a Web source. Examples can be provided easily by selecting them from the Web page. From the example, the generator tool obtains the information about its representation in the page (its situation in the HTML syntax tree, the data type, the tags delimiting it, etc.) and represents it in the wrapper language. The wrapper generated can be refined interactively if the data it extracts is not exactly what the user expected. The strategy applied is based on pruning the more specific descriptions (the inner nodes of the HTML tree path), thus having a more general description. For example, if a Web page contains a table and the user selects the data contained in the first column in the first row, the wrapper generated will obtain all the data in the first column of the table. If this result does not satisfy the user, a new wrapper will be created for extracting the data in all columns.

5. Conclusions

Web documents present new challenges to conventional IR technologies. We have described how these challenges are faced in FameIR, a multilingual and multimedia IR shell designed as an aid to improve communication. By coupling techniques from the IR field, those from the WG discipline and the use of lexical resources, the FameIR shell was provided with a powerful facility to perform multilingual and multimedia IR from Web documents. WG techniques provide a unified access to the Web sources (including those in the hidden Web). They also allow access to a finer granularity of information than the whole Web document.

Our approach to improve Web IR exploits the organisation of the Web sources (sites containing pages sharing a similar structure and content) and focuses on improving user queries. Indexes can be generated for a particular page as well as for a Web site, saving the cost of comparing each page for every user query.

Despite presenting vocabulary coverage limitations, EWN has proven useful for enhancing IR capabilities, both for performing CLIR and query expansion. The results of query expansion using EWN have been improved by allowing the user the possibility to choose the synonyms to expand the query from those provided by EWN, thus reducing semantic ambiguity.

Future work includes experiments to compare our approach to a baseline of a basic vector space model without query expansion for different types of queries (short and complex), users (casual and experts) and documents.

Acknowledgments. This research is partly supported by the European Union under the grant IST-2001-28323 (FAME project).

We would like to thank Victoria Arranz for her helpful comments.

6. References

[1] About Brightplanet^{TM}. http://www.brightplanet.com/
[2] About MySimon. http://www.mysimon.com/
[3] N. Ashish and C. A. Knoblock, "Wrapper generation for semistructured Internet sources", *Proceedings of the ACM SIGMOD Workshop on Management of Semi-structured Data,* 1997.
[4] R.Baumgartner, S.Flesca and G. Gottlob, "Visual Web information extraction with lixto", Proceedings of *VLDB*, 2001.
[5] M. Bertran, M. Gatius and H.Rodríguez, "FameIR, a Multimedia Information Retrieval Shell", Proceedings of JOTRI, Madrid, 2003.
[6] W. Cohen and L.S. Jensen, "A structured wrapper induction system for extracting information from semi-structured documents", Proceedings of *IJCAI Workshop on Adaptive Text Extraction and Mining*, 2001.
[7] W. Cohen, "Whirl: A word-based information representation language", *Artificial Intelligence*, 2000, 118.
[8] F. Crestani. "Exploiting the Similarity of Non-matching Terms at Retrieval Time", Journal of Information Retrieval, 2000, 2.
[9] Eurowordnet. http://www.let.uva.nl/ewn
[10] Fame. http://isl.ira.uka.de/fame/
[11] M. Gatius and H. Rodríguez, "Natural Language Guided Dialogues for Accessing the Web". Springer-Verlag in Lecture Notes in Artificial Intelligence, 2001, Vol. 2448.
[12] H. Joho, C. Coverson, M. Sanderson and M. Beaulieu, "Hierarchical presentation of expansion terms", Proceedings of *ACM SAC*, 2002.
[13] M.T. Maybury, ed. *Intelligent Multimedia Information Retrieval,* MIT Press, 1997.
[14] M. Mittendorfer and W. Winiwarter, "A simple way of improving traditional IR methods by structuring queries". Proceeding of the IEEE Systems, Man and Cybernetics Conference, 2001.
[15] P. Schäuble, *Multimedia Information Retrieval*, Kluwer Academic Publishers, 1997.
[16] I.H. Witten, A. Moffat and T.C. Bell, *Managing Gigabytes:Compressing and Indexing Documents and Images,* Morgan Kaufmann Publishing, San Francisco, 1999.
[17] J.K. Wu, M.S. Kankanhalli, J-H Lim, D. Hong, *Perspectives on Content-Based Multimedia Systems*, Kluwer Academic Publishers, 2000.