Multimedia Systems (1995) 3:15–24

Multimedia Systems
© Springer-Verlag 1995

Automatic authoring and construction of hypermedia for information retrieval

Maristella Agosti*, **Massimo Melucci** [1], **Fabio Crestani**[2]

[1] Department of Electronics and Informatics, University of Padua, Via Gradenigo 6a, 35131 Padua, Italy
[2] Department of Computing Science, University of Glasgow, Glasgow, Scotland

Abstract. This paper describes the complete process and a tool for the automatic construction of a multimedia hypertext starting from a large collection of multimedia documents. Through the use of an authoring methodology, the document collection is automatically authored, and the result is a multimedia hypertext, also called a hypermedia, written in hypertext markup language (HTML), almost a standard among hypermedia mark-up languages. The resulting hypermedia can be browsed and queried with Mosaic, an interface developed in the framework of the World Wide Web Project. In particular, the set of methods and techniques used for the automatic construction of hypermedia is described in this paper, and their relevance in the context of multimedia information retrieval is highlighted.

Key words: Information storage and retrieval – Content analysis and indexing – Content-based retrieval – Hypertext/hypermedia – Automatic authoring of hypermedia

1 Introduction

In information retrieval (IR) systems, the user starts the search for documents pertinent to his requirements by entering a query. The system replies by retrieving the documents matching the user's query from a large collection. This querying strategy might be considered a batch process, since it seems that the user cannot adequately interact with the system. On the contrary; in hypermedia systems, browsing is the main feature of user-system interaction.

IR systems, as well as hypermedia,can accommodate browsing. IR systems have the ability to move between related topics or documents and support relevance feedback (van Rijsbergen 1979). Unlike hypermedia, which generally has static links fixed by an expert user, relevance feedback allows the user to create links dynamically at run time by searching for documents similar to others marked as relevant. However, users will only browse if it is easy to do. Browsing by means of relevance feedback is a very complex process, and most of the existing IR systems supporting relevance feedback do not have a good user interface for browsing. Moreover, though early work on browsing text collections in IR dates back to the 1970s (Oddy 1975), very few experimental IR systems allow browsing (Frisse 1988; Thompson 1989). Only fairly recently has there been a new impulse in this research direction (Agosti et al. 1989; Dunlop 1991).

Systems providing either browsing or querying search strategies allow users access to a hypermedia by browsing after a query has been issued. Thus, users have access to documents that do not match the query. In particular, given a retrieved document, the user can access its neighbouring documents even though they do not match the query. This mixed access is useful, especially if the collection is made up of multimedia documents. Indeed, the indexing of multimedia documents is rather difficult because of the number of kinds of media and their different nature and representation. For these reasons, multimedia document indexing needs more methodological and experimental work; textual document indexing, however, has been thoroughly studied in several contexts. Several approaches have been proposed for indexing multimedia document collections; we adopt the approach proposed by Dunlop (1991) because he makes nontextual document indexing possible via neighbouring documents. In fact, cluster-based techniques are used to relate indexed documents neighbouring the multimedia document. In the same way, multimedia portions of a complete document can be indexed and interconnected to construct a hypermedia document. For example, if a figure in a document is indexed with the descriptors of its caption, these descriptors are related to the descriptors in neighbouring, clustered text portions of the complete document.

The approach presented in this paper aims at enabling users of large collections of multimedia documents collections to browse the document base in a natural way, navigating through connections representing statistical or semantic relationships between multimedia IR (MIR) objects. A MIR object can be a text, figure, term, picture, concept, etc. The approach is content based because it uses various IR techniques for content representation to link MIR objects in a coherent way. It should be noted that these techniques have been developed separately

* e-mail: agosti@ipdunivx.unipd.it.
Correspondence to: M. Agosti

and now join in a complete approach for the automatic authoring of a multimedia collection to construct and make available an IR hypermedia. The model presented in Agosti and Crestani (1993) provides a conceptual reference for the network structure of the IR hypermedia to be built. An IR hypermedia is a multimedia document base that allows access to multimedia documents mainly by browsing, but has been authored with IR techniques of content representation and linking. An IR hypermedia is composed of nodes that are stores of information and links that connect the nodes. While browsing, the user navigates from node to node by using links. The series of navigational choices leads, hopefully, through the document base to the desired information. The automatic authoring of multimedia documents is made easier by the indexing approach presented in Dunlop (1991). In this approach, authoring a multimedia document marks up the neighbouring textual documents. This means that a multimedia node is inserted into the IR hypermedia if one of more neighbours are nodes as well. In other words, the descriptors representing a textual document are used to represent content-based links between nontextual documents. From this, it appears that the main core of the work is the automatic authoring and contruction of the IR hypermedia, starting from the collection of textual documents. Therefore, this paper concentrates on the presentation of the research results that permit the building up of a IR hypermedia.

Manual authoring is only feasible if the collection to be authored is not as large as the ones typically managed by an IR system. This is because manual authoring is very time consuming. Moreover, manual authoring depends on the individual and his subjective criteria.

In contrast, automatic authoring represents a way of constructing a hypermedia from a large collection of multimedia documents, with neither time limitations nor the expert user subjectiveness. In fact, the methodology we propose is based on well-known and sound IR techniques, and it allows the user to construct a hypermedia that is the result of a unbiased process, because links are fixed according to statistical measures. The presence of dictionaries and thesauri helps the user with query formulation and browsing while he is looking for relevant documents.

Automatic authoring is becoming more and more important as a task within electronic publishing, information dissemination, and retrieval, because much information is indeed found in journals and, in general, in written form. If novel hypermedia techniques are used, such as automatic authoring, the user can overcome the traditional linear reading of documents previously available only in textual format on paper. For example, the ACM document collection could be automatically authored. Querying and browsing would be easier for the user if the ACM classification scheme were available as a content-based representation tool. We feel that, while maintaining the same scientific value, such a document collection could be accessed and browsed from remote sites with nontraditional tools that make retrieval, reading, and understanding the content easier.

2 The approach for automatic authoring and construction of hypermedia

The approach starts with the usual set of IR raw data: a flat, large, document collection. Documents are available as individual unrelated objects.

The steps of the approach consist of setting:

- a homogeneous collection of terms, namely the index terms collection
- the concept collection
- the network of links within each collection: documents (D-D links), terms (T-T links), and concepts (C-C links)
- the network of links between a pair of collections: documents – terms (D-T links), terms – concepts (T-C links).

Each step comprises one or more actions. With the exception of the first step which must be completed before any others, there is no strict or unique order for the remaining steps. Some steps can also be taken in parallel for a faster construction of the hypertext. The order we use in the presentation is based on a simple idea: first determine the objects, then build up links between homogeneous objects, and finally create links between objects of different collections.

For the presentation of specific methodological details related to this approach, the reader is referred to Agosti and Crestani (1993).

2.1 Construction of the collection of index terms

In this step, the index terms are created and connected to documents (D-T links). The collection of index terms is created by extracting terms from documents. This process known as *automatic indexing*. Automatic indexing ensures that individual terms or groups of terms found in documents become index terms, assuming a representational power that places them on a higher level of abstraction than the documents.

Indexing, a very complex process, has long been studied in IR. It constitutes the core of IR research because it is the technique by means of which document content is represented, and it makes content-based retrieval possible. In fact, the description of the document content can then be used to find an answer for the user by matching with the user query, which is represented by the same type of indexing.

There are many ways of indexing a set of documents. The steps of indexing are: term extraction, stop-term removal, conflation, and weighting. We adopt all these techniques for indexing. Please note that these techniques are described well in classical IR textbooks (Salton and McGill 1983; van Rijsbergen 1979).

2.2 Links between concepts using semantic relationships

We assume that we can identify a set of concepts in the application domain. From an IR point of view, there is no operative advantage in having a set of application domain concepts if they are not connected to each other according to their semantics. It is by looking at the relationships a concept has with other concepts that we can understand the "meaning" of the concept in the context of the application domain. When this meaning has been fully understood, it is also possible to understand the "usage" of the index terms connected to the concept. In fact, they just represent the way the concept has been addressed in the documents belonging to the collection. The way a concept has been addressed by the authors of the documents could differ from the way the user of the IR system addresses it. Using a very precise term in addressing a concept increases the precision of the retrieval. However, the user could be interested in considering a concept loosely. This can be done with index terms expressing concepts semantically related to the concept that represents the user's requirements. In this way it is also possible to increase the recall of the retrieval.

The utility of having a tool that provides a set of semantically related concepts for each concept has long been recognised in IR. A *thesaurus* is a tool that provides a set of terms for each term in a specific application domain; the semantic relationships are well defined. Because of its nature, the structure of associations represented in a thesaurus can be mapped directly into a network structure: concepts are mapped to nodes (C nodes) and concept relationships to links (C-C links).

If a thesaurus for a the specific application domain is not available, it is necessary to build up a network of concepts manually. The first essential step is to identify a set of concepts and their relationships. The fundamental types of semantic relationships commonly expressed in a thesaurus are: scope, equivalence, and hierarchical and associative relationships (Srinivasdan 1992). They provide a useful frame of reference on the kind of relationships to be taken into consideration for a manual construction of a network of concepts.

2.3 Association between index terms and concepts

The semantic association between index terms and concepts can be built with various formal approaches. The approach described by Agosti and Marchetti (1992), called semantic association, permits the automatic construction of links between index terms and concepts (T-C links). Agosti and Marchetti give a complete description of this technique.

2.4 Statistically determined relationships between index terms

There are many techniques for identifying relationships between index terms. For example, using the concept network, it is possible to relate index terms by means of objects on a higher level of abstraction. In this work we find relationships between index terms with a technique using only the information present on the same level of abstraction. This technique does not involve the semantics of index terms, but only the information provided by the statistical analysis of index term occurrence in documents. See Agosti and Crestani (1993) for a detailed description of the technique used for the construction of T-T links.

2.5 Automatic determination of relationships between documents

For an automatic set-up of links between documents (D-D links), it is possible to use statistical techniques very similar to those employed for the construction of links between index terms. Other techniques for setting up a network of related documents make use of bibliographic citations. Bibliographic citations can be used to build up a network implicitly, assuming that the documents cited by another document must somehow be related to it.

Most operational IR systems use only D-T links in the retrieval process. These are represented in the inverted file structure, which is the most common storage structure in IR. Only very few operational IR systems enable the user to take advantage of relationships like those established by C-C and T-C links, and they are used only as an aid to query formulation. Relationships like those represented by T-T and D-D links are used only in a few experimental IR systems. A scheme for an IR hypermedia produced by the authoring process is depicted in Fig. 1.

3 The automatic authoring of an IR hypermedia

A *library of MIR object classes* has been developed with C++. The library implements the basic IR structures, and its abstract interfaces allow the user to use IR functionalities. It is important to note that the class library has been developed to be as independent of any specific application as possible, so that a designer using it can find and reuse basic IR structures and functionalities without having to reimplement them. Because of this independence requirement, the class library includes classes that are independent of a specific application so that it can be used as a generic IR framework.

Using this class library, we automatically author an IR hypermedia from a collection of documents. Automatic authoring produces a hypertext in which each MIR object is connected to others by means of links. Links connecting MIR objects are set up on the basis of various criteria, such as similarity among documents (D-D links), synonymity and contiguity among index terms (T-T links), pertinence between documents and index terms (D-T links), and semantics between index terms and concepts (T-C links).

An instance of an object of the `document` class refers to the set of index terms extracted from it and describing its informative content. D-D links are placed among documents on the basis of measures of similarity. The reference of one document to another is an attribute encapsulated by the first document. We can represent a collection by defining an ad hoc subclass of the `document` class.

An instance of the `auxiliary_data` class represents auxiliary data, and it is used to represent the semantic content of a set of documents. The abstract interface of the `auxiliary_data` class enables the application designer to access generic auxiliary data without considering the specific criterion by which the auxiliary data have been constructed and associated to the pertinent documents. This means that the

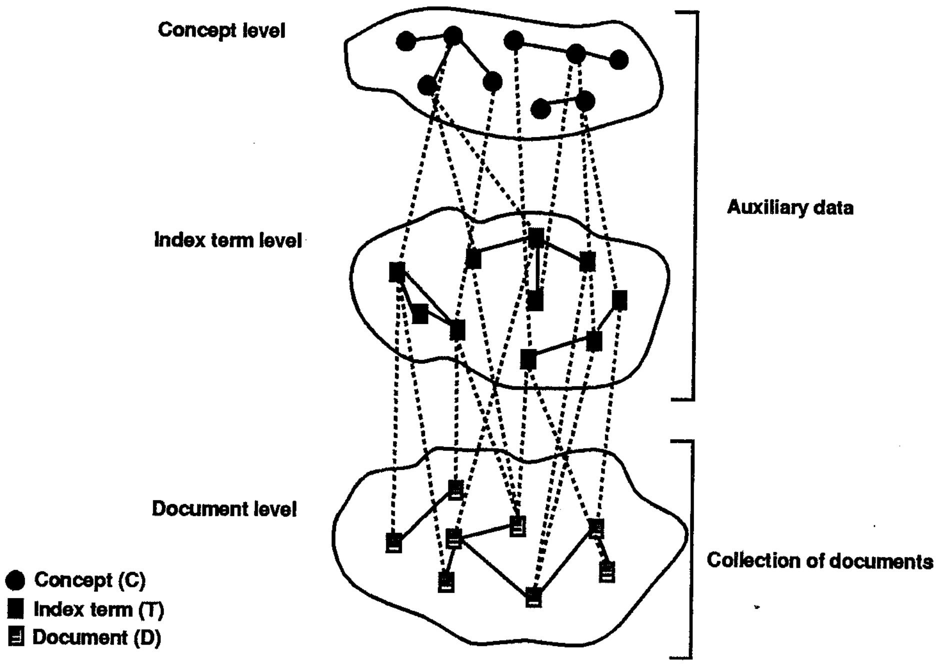


Fig. 1. A conceptual schema of an IR hypermedia

class `auxiliary_data` provides an "umbrella" to manage a generic auxiliary data subclass specified in it. This provides the designer with ad hoc tools to manage specific types of auxiliary data. Our approach provides two types of auxiliary data: index terms and concepts. For their distinctive characteristics, we consider it useful to distinguish the subclasses `concept` and `index_term` in the `auxiliary_data` class to emphasise the specific feature of concepts and index terms with respect to generic auxiliary data.

Index terms are auxiliary data that have been extracted automatically from documents by indexing. An index term is associated with its frequency of occurrence within the collection; it is also associated with the set of documents from which it has been extracted. T-T and D-T links are placed on the basis of information provided by the statistical analysis of index terms and document occurrence. The references of a document to its extracted index terms, the references of an index term to another index term and to the pertinent documents are attributes encapsulated by the objects representing documents and index terms.

Concepts are represented by instances of the class `concept` that implements the third level of the conceptual architecture. C-C links are set up on the basis of the semantic relationships among concepts. A relationship between two concepts is an entity holding the semantics to be represented. A semantic relationship between two concepts is represented by an instance of the `relationship` class. Since different types of relationship can exist between concepts, it is useful to break the class `relationship` up into various subclasses. Thus, the `relationship` class is subdivided into more subclasses describing the fundamental types of semantic relationships commonly expressed in a semantic structure. These subclasses are: `scope`, `hierarchy`, `synonymity` and `association`. The subclass `hierarchy` has a further subclass `specialisation` to represent the relationship between a given concept and other concepts on a more specific level.

We stressed the fact that it is almost always necessary to model the semantic relationships between concepts manually. It is the user himself or a team of domain experts who must build up the semantic structure that represents important and useful application-domain knowledge. However, if this semantic structure is represented and stored in a machine-readable form, the prototype can build the network of concepts automatically. This means that the tool can recognise concepts and relationships among concepts that are coded in a machine-readable form, and it is no longer necessary to build up the network of concepts manually . It is also possible to set up the T-C links automatically. We have already described the semantics of these links: concepts can be linked to several index terms, and an index term can be linked to various concepts. For example, the concept "information retrieval" is linked to the index terms "information" and "retrieval," but the index term "information" is also linked to the concept "information processing." In general, an automatic indexing algorithm considers index terms made by one word, such as "information" or "retrieval". The indexing algorithm we developed treats index terms in that way too. However, our class library provides functions to split concepts into two or more terms. If a split term is an index term, the connection with the concept is set up. In an analogous way, index terms can be concatenated to construct a multiword term. If the multiword term is a concept, the T-C link is completed. Therefore, the passage from the index-term level to the concept level, and vice versa, that is, the T-C linking mechanism, is possible due to the splitting of a concept into terms and the concatenation of terms to build a concept.

4 The automatic authoring process

The automatic authoring process makes use of the class library presented in the previous section. The process is depicted in Fig. 2. The input is a flat document collection and the output is an IR hypermedia that is written in hypertext mark-up language (HTML). IR hypermedia can be browsed and queried with Mosaic (for machines with a graphical interface) or Lynx (for machines able to deal only with text). It is important to bear in mind that our approach is general and applicable to several types of collections as long as they appear in standard, machine-readable forms. At present we can handle plain ASCII, LaTEX, BibTEX, the standard of the Bath Institute of Scientific Information Data Service (BIDS), and INSPEC. We are currently adding capabilities for translating document formats written with other standards into HTML. In addition, we have automatically authored the ACM classification scheme to provide an IR hypermedia with a widespread and wide-ranging concept collection for computing and computer science. This means that the ACM document collections could be automatically authored and queried with the ACM classification scheme itself. Automatic authoring becomes more important if one takes into account the way documents in machine-readable form and on-line bibliographies broaden over the Internet. Documents are indeed physically stored at various Internet sites, but they are interrelated by citation links. If this approach were used, it would make available all those documents connected by means of content-based links.

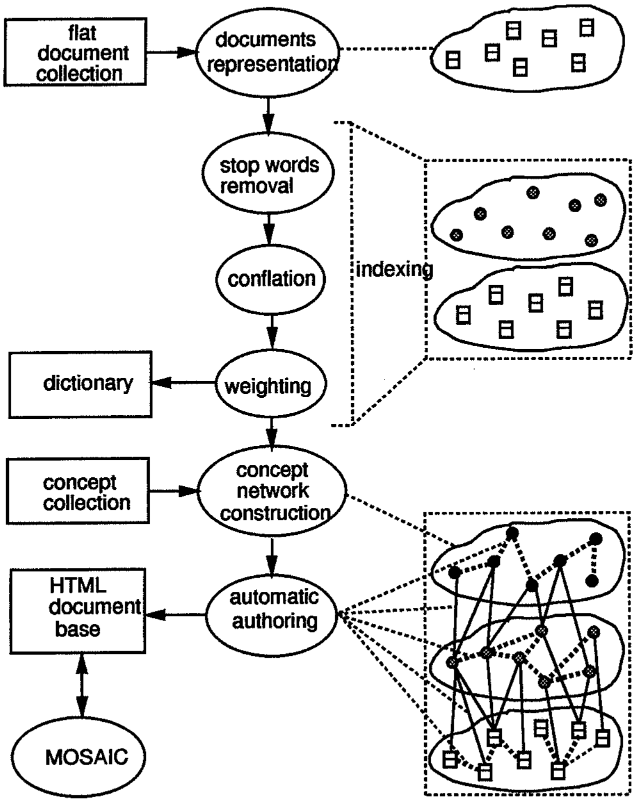


Fig. 2. The automatic authoring process

The authoring process is divided into the following sequence of steps:

1. *Collection loading and document representation.* The collection is analysed to produce document representations in terms of objects of the class `document`.
2. *Indexing.* This task aims at populating the class `index_term` that makes up the dictionary of the collection. As words are extracted from documents, stop words are detected or, otherwise, they are conflated. Porter's stemming algorithm (Porter 1980) is used to conflate words to index terms. Index terms extracted from each document item are merged into a unique list and associated with the document.
3. *Semantic structure loading and concept representation* make up the second phase of the design process. Like the first task, this one, too, depends on the particular semantic structure. We have already outlined how the tool can read automatically and represent and manage a semantic structure if it is stored in a standard format and includes the information for setting up the relationships among the concepts. Thus, the C-C links are set up during this task.
4. *Automatic authoring.* This step can be considered as the core of the entire process because it makes an automatically constructed IR hypermedia available to the user. This task implements the last three phases of the design process. The similarity measures are computed and the D-D, D-T, T-T, and T-C links are automatically set during this step. During this task items of the three levels of the conceptual scheme are stored in a collection of HTML documents. The HTML documents are linked by the mechanisms which the HTML provides. A HTML document is linked to another, or to a part of itself, by means of a pair of tags, say, "link" and "anchor" tags; Mosaic is provided with the functionality of retrieving and displaying the anchored document after the user has clicked the link tag. A document node is authored with link tags on the index terms extracted from its text. Special tags give access to the dictionary and to the classification system. The mode of a given index term is authored by linking all the documents from which it is extracted, all the index terms similar to it, and all the related concepts of the classification system.

5 Browsing and querying an IR hypermedia

Using Mosaic, the user can access any MIR object of the MIR hypertext by means of two procedures: browsing and querying. In Agosti and Crestani (1993) we have stressed and justified the importance of using browsing and querying together to access IR documents. On the network structure of the IR hypermedia, it is possible to browse among concepts, index terms, and documents, exploring the large document and auxiliary data space.

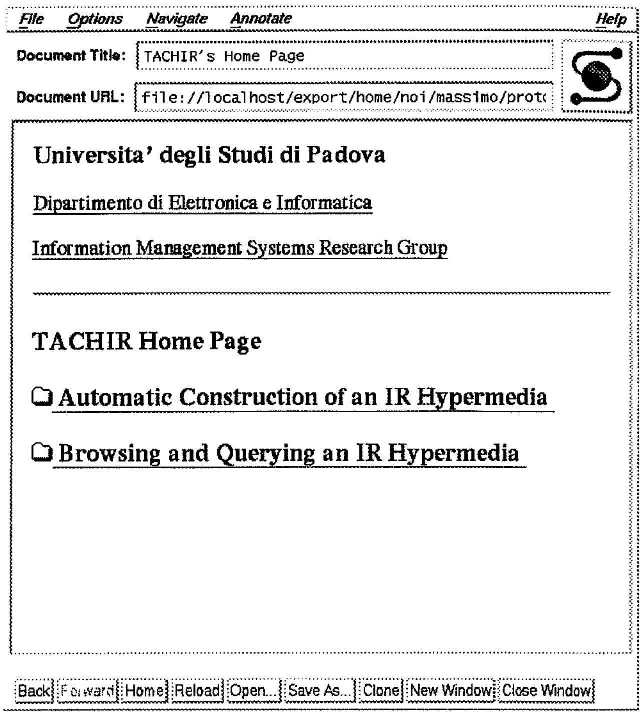
It is also possible to query the IR hypermedia with the keyword search available in Mosaic. A more complex technique for querying is being developed. In fact, using an IR hypermedia, the process of querying can be enhanced by spreading activation techniques (Salton and Buckley 1988). Once the user has entered the network structure of the IR hypermedia using a concept, an index term, or a document, he can go on building up a query by browsing through other concepts, index terms, or documents and including in the query those that he thinks are relevant. After the user has built up a query by browsing, an automatic procedure can be activated. This makes use of the semantics associated to links and node types, and can spread activation over the network and use concepts, index terms, or documents that are closely related to those indicated in the query. The user can provide feedback to the system by marking the nodes that he considers relevant in the retrieved list. In this way, the user assesses the success of the spreading in including new MIR objects to his query. This process is similar to the relevance-feedback technique used in advanced IR systems in which relevance feedback is used to modify the query terms according to the suggestions the user gives to the system after he has marked the relevant documents. New query terms are determined by the system on the basis of the weights of the previous query terms. Our approach provides query modification based not only on statistical analysis (D-T and T-T links), but also on semantic relationships (T-C and C-C links). After a new query has been formulated, the user can activate spreading again, and the search continues in an iterative and interactive process controlled (constrained) by the system.

6 Initial experimental results of constructing and browsing of an IR hypermedia

We have developed a tool for the automatic construction of an IR hypermedia that makes use of the class library presented in the previous section: a tool for the automatic construction of hypermedia for information retrieval (TACHIR). TACHIR can be activated from inside a Mosaic session by clicking the button assigned for the purpose on the home page. In Fig. 3 we can see the automatic construction of an IR hypermedia button that activates such a function. The user is asked to give the location of a collection of documents and a collection of concepts such as a thesaurus or the ACM classification scheme. TACHIR automatically builds up the corresponding IR hypermedia that can be browsed and queried with Mosaic.

If one or more IR hypermedia are already available, the user can begin with "browsing and querying of an IR hypermedia" to browse or query effectively the chosen IR hypermedia, which has available the functionalities illustrated in Sect. 5.

At present, only document collections of the the BIDS, LaTEX, BibTEX and plain ASCII types have been used to generate an IR hypermedia automatically. At the current stage, we tested the prototype by adopting a large BibTEX bibliographic reference collection. Our approach is general and applicable to several collections as long as they appear in machine-readable forms. We are implementing new TACHIR functionalities to translate other types of collections into IR hypermedia. Some collections are made up of documents that include references to figures and bibliographies other than in the usual structured full text – for example, LaTEX documents. Nowadays, tools for translating LaTEX documents into HTML documents are available, but they lack automatic authoring.



Document Title: TACHIR's Home Page

Document URL: file://localhost/export/home/noi/massimo/prot(

Universita' degli Studi di Padova

Dipartimento di Elettronica e Informatica

Information Management Systems Research Group

TACHIR Home Page

Automatic Construction of an IR Hypermedia

Browsing and Querying an IR Hypermedia

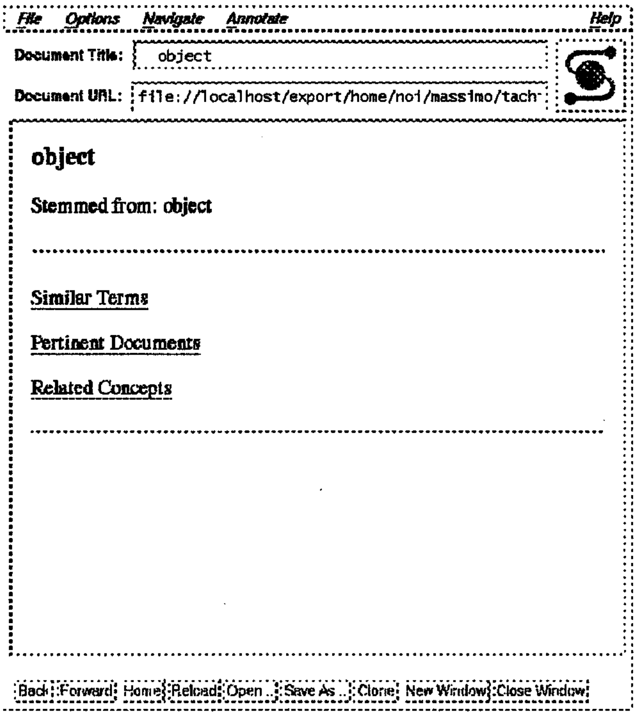
Fig. 3. TACHIR home page

In the following, a guided tour of the construction of an IR hypermedia for a BibTEX collection is presented to explain the complete approach and construction of an IR hypermedia with a real case study. Of course, this tour is not exhaustive, but it is representative of the possibilities available to TACHIR users. We have used a BibTEX collection as the raw input data, since a BibTEX collection can include abstracts that are full text documents. A BibTEX item is a bibliographic record including various kinds of entries: keywords, abstracts other than the usual data fields (such as titles, authors, affiliations, and so forth). In the following, a BibTEX collection of 18 000 entries on object orientation is employed.

As we have pointed out, we have also chosen the ACM classification scheme to be the semantic structure placed at the third level of the architecture, since it is one of the most widespread semantic structures in computing and computer science. The entries of the ACM classification scheme are hierarchically organised, and each entry is a concept that can hold one or more narrower concepts and also has a broader concept. Each entry of the ACM classification that can be picked up by the user is underlined by the prototype.

The user can select a whole entry or a part of it; for example, given a document title, it is possible to retrieve the entire document or the information associated with a title term, no matter whether the user clicks the whole title or a title term. However, it must be noted that these characteristics are typical of other collections as well.

Once the user has selected a document collection (the BibTEX collection in this guided tour), he chooses a starting point for browsing among the three levels of the architecture depicted in Fig. 1:



Document Title: object

Document URL: file://localhost/export/home/noi/massimo/tach

object

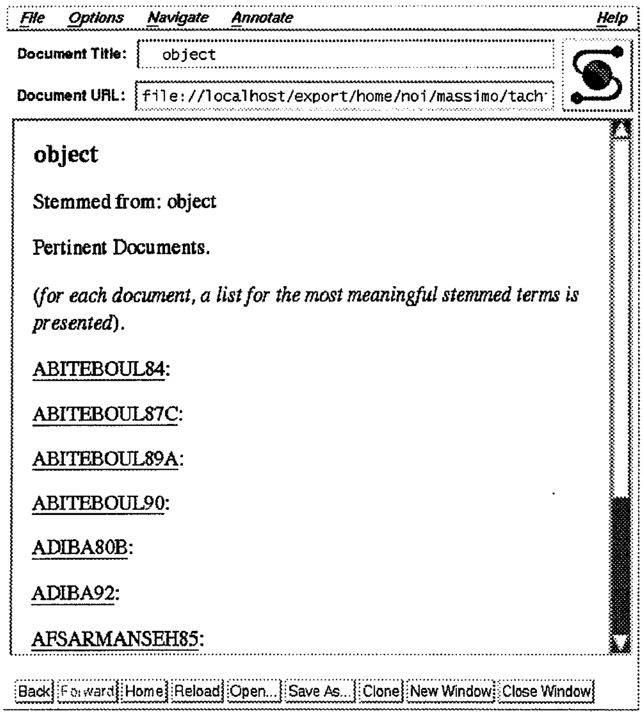
Stemmed from: object

Similar Terms

Pertinent Documents

Related Concepts

4



Document Title: object

Document URL: file://localhost/export/home/noi/massimo/tach

object

Stemmed from: object

Pertinent Documents.

(for each document, a list for the most meaningful stemmed terms is presented).

ABITEBOUL84:

ABITEBOUL87C:

ABITEBOUL89A:

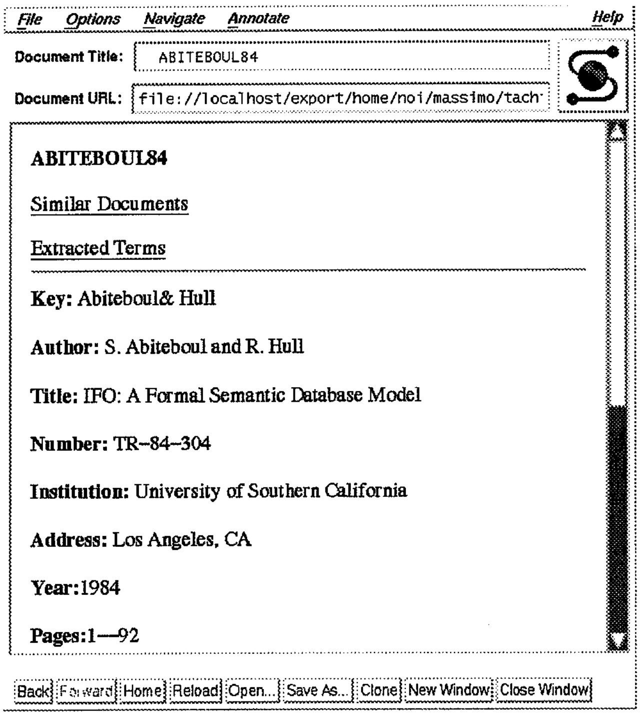
ABITEBOUL90:

ADIBA80B:

ADIBA92:

AFSARMANSEH85:

5



Document Title: ABITEBOUL84

Document URL: file://localhost/export/home/noi/massimo/tach

ABITEBOUL84

Similar Documents

Extracted Terms

Key: Abiteboul& Hull

Author: S. Abiteboul and R. Hull

Title: IFO: A Formal Semantic Database Model

Number: TR-84-304

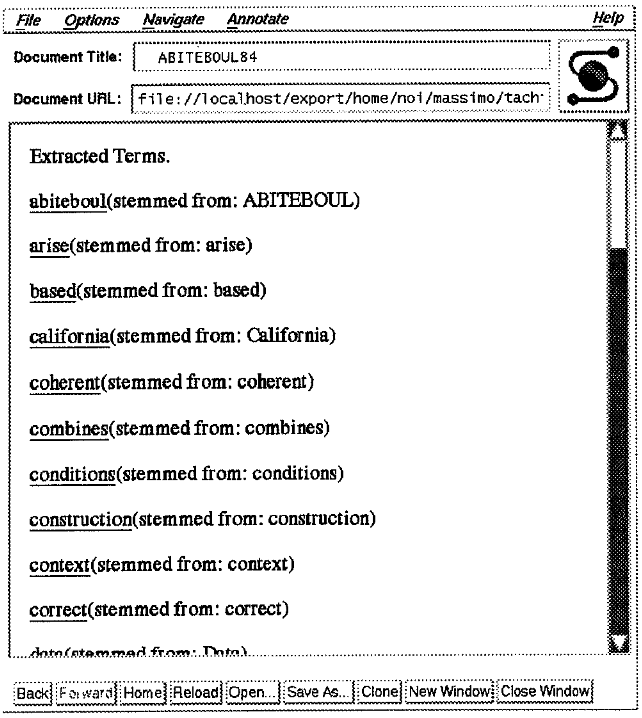
Institution: University of Southern California

Address: Los Angeles, CA

Year:1984

Pages:1—92

6



Document Title: ABITEBOUL84

Document URL: file://localhost/export/home/noi/massimo/tach

Extracted Terms.

abiteboul(stemmed from: ABITEBOUL)

arise(stemmed from: arise)

based(stemmed from: based)

california(stemmed from: California)

coherent(stemmed from: coherent)

combines(stemmed from: combines)

conditions(stemmed from: conditions)

construction(stemmed from: construction)

context(stemmed from: context)

correct(stemmed from: correct)

data(stemmed from: Data)

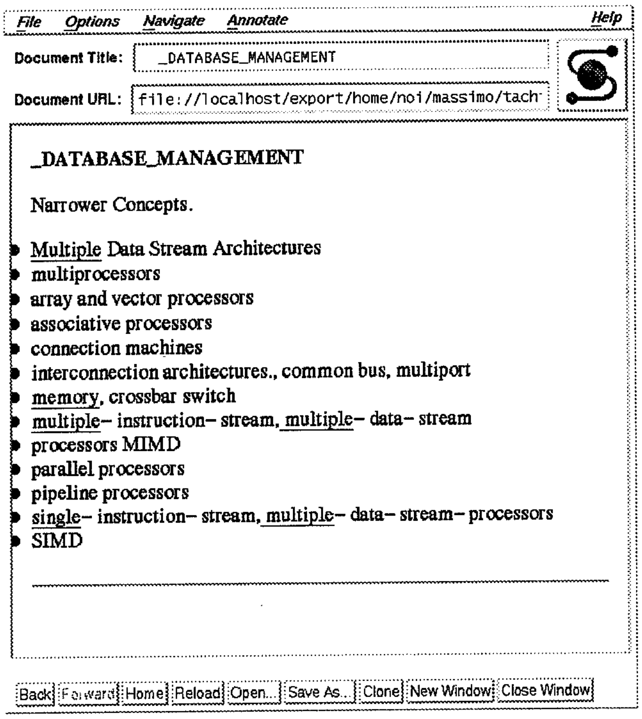
7

Fig. 4. An index term and its related information
Fig. 6. A document pertinent to `object`

Fig. 5. The list of documents pertinent to `object`
Fig. 7. Terms extracted from a document

1. The collection of documents
2. The set of index terms that are automatically extracted from the documents
3. The set of concepts that take part in the ACM classification scheme.

Let us suppose that the user has chosen the second level, namely, the dictionary-term level. After the user has chosen the index term `object`, a page containing links to the related information is displayed. The information related to an index term are represented by three buttons linking similar terms,



Document Title: _DATABASE_MANAGEMENT

Document URL: file://localhost/export/home/noi/massimo/tach

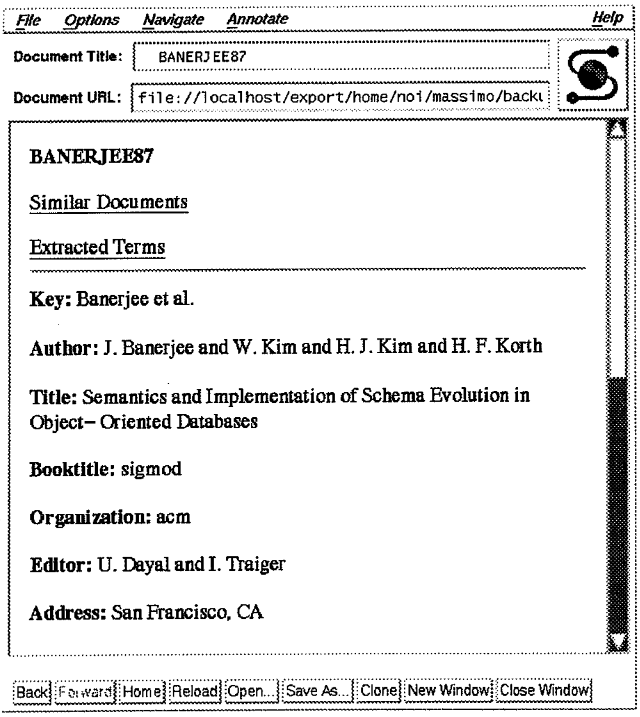
_DATABASE_MANAGEMENT

Narrower Concepts.

- Multiple Data Stream Architectures
- multiprocessors
- array and vector processors
- associative processors
- connection machines
- interconnection architectures., common bus, multiport
- memory, crossbar switch
- multiple– instruction– stream, multiple– data– stream
- processors MIMD
- parallel processors
- pipeline processors
- single– instruction– stream, multiple– data– stream– processors
- SIMD

8

Fig. 8. A concept related to `database`



Document Title: BANERJEE87

Document URL: file://localhost/export/home/noi/massimo/backu

BANERJEE87

Similar Documents

Extracted Terms

Key: Banerjee et al.

Author: J. Banerjee and W. Kim and H. J. Kim and H. F. Korth

Title: Semantics and Implementation of Schema Evolution in Object– Oriented Databases

Booktitle: sigmod

Organization: acm

Editor: U. Dayal and I. Traiger

Address: San Francisco, CA

9

Fig. 9. A document relevant to `database`

pertinent documents, and related concepts. Each of these sets of information represents a "direction" along which an index term spreads its semantics: two directions are vertical ones, towards the higher and lower levels, and the other is horizontal, the level where the term is placed. Clicking one of these "directions" gets the concepts explaining the index term semantics or the documents with the semantics explained by the index term. At this point it should be noted that this is but one of the possible ways of getting started. Another strategy is based on querying, but we are now interested mainly in browsing, since we are addressing content-based hypermedia functionalities.

When the user is looking at an index term, he can pick up one of several entries. For example, the user might be interested in documents that are pertinent to the term object. Index terms (Fig. 4) have been associated to a triplet of sets: the sets of similar terms, pertinent documents, and related concepts.

Let us suppose that the user clicks the anchor of the pertinent documents. In Fig. 5, the list of documents pertinent to `object` is presented. He then selects the document identified as Abiteboul84, which is then presented (Fig. 6).

Only after the user has read the selected document can he decide if it is effectively relevant to his query. If a user cannot find the relevant documents after the first selection, he must reformulate the query by clicking the button of terms extracted from the selected document, by going back to the term level, or by reading through the list of the similar documents. After having chosen the list of extracted terms, the user selects the term `database` by trying a query reformulation. Selecting the term *database* from the extracted terms appearing in Fig. 7, the user could collect the pertinent documents. From these documents it is possible to choose another document in the list and to place it on a page. However, the term `database` is rather general, and it is not semantically meaningful to the user, so that he may wish to access the third level by looking for related concepts. The concept `database management` is useful just to clarify to the user the possible contexts in which the term `database` is used. Fig. 8 displays such a concept and the underlined terms are those output by the indexing process. The ACM classification scheme entries are alphanumeric strings representing concepts. These concepts are organised in a hierarchical manner according to a narrower-broader relationship. Such an entry consists of one or more terms, and some of these terms can be index terms and belong to the second level of the architecture as well.

Accessing concepts through an index term allows the user to see the related concepts. The index term-concept (T-C) association rule is used in automatic authoring: each index term is connected to the concepts containing it, and each concept is connected to the component index terms. This rule is based on a quite straightforward mechanism. When the user is browsing the IR hypermedia at the second level, he can retrieve the concepts with component words that equal the pointed index term, together with the available narrower and broader concepts. Symmetrically, the user can ask the system to retrieve the index terms forming the concept he is considering while browsing the third level. We are going to address the difficulties in relating terms to concepts in a more "intelligent" way, since, for example, more concepts could be related to a term. However, we are aware of the complexity of this task and of similar ones. such as automatic construction of thesauri and passage retrieval.

If the user cannot find any useful information in the concept `database management`, he can go down to the second level to try another strategy, such as the retrieval of the documents relevant to `database`, as in the Fig. 5, from which a relevant document is retrieved (Fig. 9).

7 Conclusions

We have presented a complete content based approach for the automatic construction of an IR hypermedia and an effective tool based on it. We have developed such a tool to enable the automatic creation of a hypertext structure written in HTML from a collection of documents. This hypertext, which we call an IR hypermedia because it can be enriched by multimedia documents as well, can be browsed and queried with any of the World Wide Web graphical interfaces supporting HTML, for example Mosaic, running on various platforms, and ranging from Unix to Macintosh. The availability of such a tool can make large collections of documents available for browsing and querying to a large number of Internet users.

At present we are addressing the problems connected with the enhancement of the querying capabilities. In particular, we are developing a querying tool that activates a form of constrained spreading over the IR hypermedia to produce and present to the user a ranking of the documents. This is because a form of spreading would provide the user with iterative and interactive browsing/querying possibilities.

Acknowledgements. This work was partially funded by a 1993 MURST grant from the Italian Ministry of Education. The work of Massimo Melucci was supported in part by a grant of IDOMENEUS, the ESPRIT Network of Excellence No. 6606 on Information and Data on Open Media for Networks of Users, for a period in which he worked at the Department of Computing Science of the University of Glasgow, Glasgow, Scotland.

References

Agosti M, Crestani F (1993) A methodology for the automatic construction of a hypertext for information retrieval. Proceedings of the ACM Symposium on Applied Computing, Indianapolis, pp 745–753

Agosti M, Marchetti PG (1992) User navigation in the IRS conceptual structure through a semantic association function. Comput J 35:194–199

Agosti M, Gradenigo G, Mattiello P (1989) The hypertext as an effective information retrieval tool for the final user. In: Martino AA, (ed) Preproceedings of the 3rd International Conference on Logics, Informatics and Law, Florence, Italy, pp 1–19

Dunlop M (1991) Multimedia information retrieval. PhD Thesis, Department of Computing Science, University of Glasgow, Glasgow, UK

Frisse ME (1988) Searching for information in a medical handbook. Commun ACM 31:880–886

Oddy RN (1975) Reference retrieval based on user inducted dynamic clustering. Phd Thesis, Computing Science Department, University of Newcastle upon Tyne, UK

Porter MF (1980) An algorithm for suffix stripping. Program 14:130–137

Salton G, Buckley C (1988) On the use of spreading activation methods in automatic information retrieval. In: Chiaramella Y (ed) Proceedings of ACM SIGIR, Grenoble, France, pp 147–160

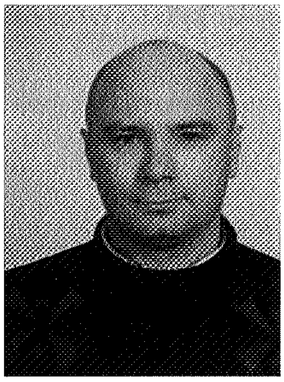
Salton G, McGill MJ (1983) Introduction to modern information retrieval. McGraw-Hill, New York

Srinivasdan P (1992) Thesaurus construction. In: Frakes WB, Baeza-Yates R (eds) Information retrieval: data structures and algorithms. Prentice Hall, Englewood Cliffs, N. J., pp 161–218

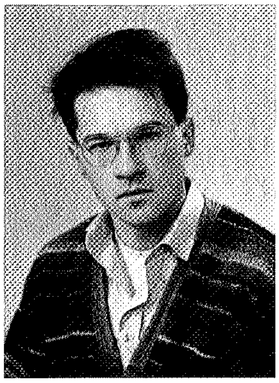
Thompson RH (1989) The design and implementation of an intelligent interface for information retrieval. Technical report, Computer and Information Science Department, University of Massachusetts

Rijsbergen CJ van (1979) Information retrieval. Butterworths, London

Maristella Agosti has been an Associate Professor of Computing Science since 1987 at the Department of Electronics and Computer Science at the University of Padua, Italy. She is Scientific Secretary of the European Information Retrieval Specialist Group of CEPIS (CEPIS-EIRSG), Domain Leader for information retrieval and multimedia of the IDOMENEUS ESPRIT No. 6606 Network of Excellence, and a member of the Research Panel of the EEC Information Engineering programme, Luxembourg. She has participated in the development of research prototypes of advanced information-retrieval systems(e.g. HYPERLINE of ESA and HyperLaw). Present research interests are: advanced IR models, hypertext information retrieval, automatic authoring of hypermedia for information retrieval, hypermedia and information retrieval systems interaction evaluation.



FABIO CRESTANI received a degree in statistics from the University of Padua in 1987. He then worked in the area of Information Retrieval collaborating with the research group of Prof. Agosti at the Unversity of Padua. Since 1990 he has been working with Prof. van Rijsbergen at the University of Glasgow. He participated in various research projects, including the Esprit Projects "SHAPE" and "MIRO". Since 1992 he has been on the staff of the Department of Electronics and Computer Science of the University of Padua. He is currently on leave at the University of Glasgow where he is working in the Esprit Project "FERMI". His current research interests include information retrieval, knowledge representation, knowledge acquisition, cognitive science, and connectionism.



MASSIMO MELUCI is a PhD student in computing science at the Department of Electronics and Computer Science of the University of Padua. He graduated in Statistics and Economics at the University of Padua with a thesis in computing science addressing the object-oriented data base modelling. In 1994 he was a visiting research student at the Computing Science Department of the Glasgow University for studies on the modelling of hypertexts for IR. His main areas of interest are: conceptual and logical modelling in IR, hypertext and database; object orientation; multimedia.