Ontology-based Information Selection

by

Latifur R. Khan

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Computer Science)
August 2000

Dedication

This Dissertation is dedicated to my father and mother--A. Baqui Khan, and Lutfe Ara Begum, who helped me grow into the person I am today and who have been "with me" in every sense in all phases of this Ph.D. journey.

Acknowledgements

All praises to almighty God, most gracious and most merciful, who allowed me to write this dissertation. He made me fortunate enough to be associated with so many competent and intellectually stimulating individuals who have guided me at every stage of my Ph.D.

First, and foremost, I would like to thank Dennis McLeod, my research advisor and "mentor," for his excellent guidance and encouragement throughout my research, and for his friendship during my years as a graduate student. His high standards of scholarship and intellectual integrity, as well as his openness and flexibility, contributed substantially to the understanding of the complex intersection of information retrieval, databases, and artificial intelligence which is presented in this dissertation. I was extremely fortunate to work under his supervision for my Ph.D.

I am also grateful to the other members of my qualifying examination and dissertation committees--Eduard Hovy, Craig Knoblock, Daniel O'Leary, and Larry Pryor. Eduard Hovy, in particular, provided me valuable guidance as an ontology expert. He first introduced me to the subject of ontologies during my first year as a graduate student at USC. While the work on this dissertation was in progress he gave useful feedback that accelerated the research. Daniel O'Leary's suggestions and comments also helped me to steer this research in the right direction.

I want also to sincerely thank all the members, old and new, of the Integrated Media Systems Laboratory at USC, which provided a home with a friendly and helpful research environment, greatly strengthening the outcome and quality of this work.

Last but not least, my friends and colleagues: namely Goksel Aslan, Greg Barish, Ahmed Helmy, Chad Jenkins, Jonghyun Khang, Chawa Lin, Kevin Lio, David Lyon-Buchanan, Graham Philips, Maytham Safar, Hyun-Woong Shin, Yan Wang, and Roger Zimmerman deserve my special thanks for their endurance and helpful input during my various forays down different paths of research, helping me prepare for and to overcome many obstacles along the way.

Furthermore, I am grateful to my sister, Amina Begum, and to my younger brother, Khan Tarik, who specifically offered strong moral support during the last year of my Ph.D. studies.

Finally, my years as an undergraduate in Computer Science and Engineering at the Bangladesh University of Engineering and Technology were fruitful laying the foundation for what I could do further as a graduate student.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Abstract

Technology in the field of digital media generates huge amounts of non-textual information, audio, video, and images, along with more familiar textual information. The potential for exchange and retrieval of information is vast and daunting. The key problem in achieving efficient and user-friendly retrieval is the development of a search mechanism to guarantee delivery of minimal irrelevant information (high precision) while insuring relevant information is not overlooked (high recall). The traditional solution employs keyword-based search. The only documents retrieved are those containing user specified keywords. But many documents convey desired semantic information without containing these keywords. This limitation is frequently addressed through query expansion mechanisms based on the statistical co-occurrence of terms. Recall is increased, but at the expense of deteriorating precision.

One can overcome this problem by indexing documents according to meanings rather than words, although this will entail a way of converting words to meanings and the creation of an index structure. We have solved the problem of an index structure through the design and implementation of a concept-based model using domain-dependent ontologies. An ontology is a collection of concepts and their interrelationships, which provide an abstract view of an application domain. With regard to the converting words to meaning the key issue is to identify appropriate concepts that both describes and identifies documents, as well as language employed in user requests. This dissertation describes an automatic mechanism for selecting these concepts. An important novelty is a scalable disambiguation algorithm which prunes irrelevant concepts and allows relevant ones to associate with documents and participate in query generation. We also propose an automatic query expansion mechanism that deals with user requests expressed in natural language. This mechanism generates database queries with appropriate and relevant expansion through knowledge encoded in ontology form.

Focusing on audio data, we have constructed a demonstration prototype. We have experimentally and analytically shown that our model, compared to keyword search, achieves a significantly higher degree of precision and recall. The techniques employed can be applied to the problem of information selection in all media types.

Chapter 1 Introduction

The development of technology in the field of digital media generates huge amounts of non-textual information, such as audio, video, and images, as well as more familiar textual information [34]. The potential for the exchange and retrieval of information is vast, and at times daunting. In general, users can be easily overwhelmed by the amount of information available via electronic means. The need for user-customized information selection is clear. The transfer of irrelevant information in the form of documents (e.g. text, audio, video) retrieved by an information retrieval system and which are of no use to the user wastes network bandwidth and frustrates users. This condition is a result of inaccuracies in the representation of the documents in the database, as well as confusion and imprecision in user queries, since users are frequently unable to express their needs efficiently and accurately. These factors contribute to the loss of information and to the provision of irrelevant information. Therefore, the key problem to be addressed in information selection is the development of a search mechanism which will guarantee the delivery of a minimum of irrelevant information (high precision), as well as insuring that relevant information is not overlooked (high recall).

1.1 The Traditional Solution

The traditional solution to the problem of recall and precision in information retrieval employs keyword-based search techniques [7]. Documents are only retrieved if they contain keywords specified by the user. However, many documents contain the desired semantic information, even though they do not contain user specified keywords. This limitation can be addressed through the use of query expansion mechanism [87].

Additional search terms are added to the original query based on the statistical co-occurrence of terms. Recall will be expanded, but at the expense of deteriorating precision [12, 70, 72, 103].

1.2 Our Approach

In order to overcome the shortcomings of keyword-based technique in responding to information selection requests we have designed and implemented a concept-based model using ontologies [50, 51]. This model, which employs a domain dependent ontology, is presented in this dissertation. An ontology is a collection of concepts and their interrelationships which can collectively provide an abstract view of an application domain [14, 30, 31].

There are two distinct questions here: one is the extraction of the semantic concepts from the keywords and the other is the indexing. With regard to the first problem, the key issue is to identify appropriate concepts that describe and identify documents on the one hand, and on the other, the language employed in user requests. In this it is important to make sure that irrelevant concepts will not be associated and matched, and that relevant concepts will not be discarded. In other words, it is important to insure that high precision and high recall will be preserved during concept selection for documents or user requests. To the best of our knowledge, in conventional keyword search the connection through the use of ontologies between keywords and concepts selected from documents to be accessed for retrieval is carried out manually [28, 87, 96], a process which is both subjective and labor intensive. In this dissertation, we propose an automatic mechanism for the selection of these concepts. Furthermore, this concept selection mechanism includes a novel, scalable disambiguation algorithm using domain specific ontology. This algorithm will prune irrelevant concepts while allowing relevant concepts to become associated with documents and participate in query generation.

With regard to the second problem, one can use vector space model of concepts or more precise structure by choosing ontology. We adopt the latter approach. This is because vector space model does not work well for short queries. Furthermore, one recent survey about web search engines suggests that average length of user request is 2.2 keywords [7]. For this, we have developed a concept-based model, which uses domain dependent ontologies for responding to information selection requests. To improve retrieval, we also propose an automatic query expansion mechanism which deals with user requests expressed in natural language. This automatic expansion mechanism generates database queries by allowing only appropriate and relevant expansion. We will demonstrate analytically and empirically that our ontology-based model allows us to achieve a significant higher degree of precision and recall than is possible using traditional keyword-based search techniques. Intuitively, to improve recall during the phase of query expansion, only controlled and correct expansion is employed, guaranteeing that precision will not be degraded as a result of this process. Furthermore, for the disambiguation of concepts only the most appropriate concepts are selected with reference to documents or to user requests by taking into account the encoded knowledge in the ontology.

1.3 Experimental Context

In order to demonstrate the effectiveness of our model we have explored and provided a specific solution to the problem of retrieving audio information. We chose audio as the medium employed for the prototype model. Audio is one of the most powerful and expressive of all media. It is of special note that audio information can be of particular benefit to a person who is visually impaired, and for the general population, audio, as a streaming medium (i.e., temporally extended), has become an increasingly popular medium for capturing and presenting information. At the same time, audio's very properties as a medium, along with its opaque relationship to computers, presents distinct technical problems from the perspective of data management [27, 85]. Thus, the effective selection/retrieval of audio information entails several tasks, such as metadata generation (description of audio), and the consequent selection of audio information in response to a query.

Relevant to our purpose, ontologies can be fruitfully employed to facilitate metadata generation. For metadata generation, we need to do content extraction. Content extraction here is carried out by following the current state-of-the-art procedures in speech recognition technology. This will involve the use of a fully automated content extraction [35] technique (speech to text conversion), and selected content extraction using word-spotting [43, 99], which determines the occurrence of keywords in audio where these keywords are derived from ontologies. In this dissertation we argue that ontology, by reducing the chance of speech recognition error, can provide a means for selected content extraction which will determine which keywords should be identified in audio documents. After generating transcripts we can deploy our ontology-based model to identify appropriate concepts that describe and identify audio documents and serve as metadata for the documents.

At present, an experimental prototype of the model has been developed and implemented. As of today, our working ontology has around 7,000 concepts for the sports news domain, with 2,481 audio clips/objects of metadata in the database. For sample audio content we use CNN broadcast sports [18] and Fox Sports audio [22], along with closed captions. Using our disambiguation algorithm, these associated closed captions are connected with the ontology. The performance of our disambiguation algorithm has been studied by considering what percentage of the audio objects selected are associated with relevant concepts. We have observed that through the use of our disambiguation algorithm 90.5% of the objects extrapolated from closed captions successfully associate with concepts of ontologies, while only 9.5% of the objects fail to associate. Among these 90.5%, up to 76.9% of the objects are associated with relevant concepts (pure); in other cases, objects are associated with relevant and irrelevant concepts (mixed).

To illustrate the power of ontology-based over keyword-based search techniques we have taken the most widely used vector space model as representative of keyword search. For comparison metrics we have used measures of precision and recall, and an F score that is the harmonic mean of precision and recall. Fifteen sample queries were run based on the categories of broader query (generic), narrow query (specific), and context query formulation. We have observed that on average our ontology outperforms keyword-based technique. For broader and context queries, the result is more pronounced than in cases of narrow query.

1.4 Contributions

The main contributions of this dissertation are as follows:

- We propose an automatic concept selection mechanism for documents/user requests including a scalable disambiguation algorithm.
    - This mechanism employs a domain-specific ontology that facilitates high precision and high recall in response to information selection requests.
- We demonstrate analytically, and empirically, the superiority of the retrieval effectiveness of our ontology-based model over traditional keyword-based search techniques. The reasons for the superiority of this model can be summarized by noting several points:
    - First, the heuristic based algorithm in the mechanism retains only those concepts which are relevant and associated with documents/user requests, while irrelevant concepts are pruned.
    - Next, during the phase of query expansion, only controlled and correct expansion is employed, guaranteeing that the use of additional terms in the query will not hurt precision.
- We devise a framework for allowing user requests expressed in natural language to be automatically mapped into SQL database queries, with no user knowledge of the database or SQL queries.

- We also demonstrate that some novel optimization techniques which rewrite the SQL query with the help of knowledge that comes from the ontology can be employed, without a loss of precision and recall.

1.5 Outline of the Dissertation

The remainder of this dissertation is organized as follows. In Chapter 2 we review related work. In Chapter 3, we introduce the research context in terms of the information media used (i.e., audio) and some related issues that arise in this context. In Chapter 4, we introduce our domain dependent ontology. In Chapter 5, we present our heuristic-based concept selection mechanism, including the disambiguation algorithm that allows us to choose appropriate concepts for audio information unit. We also discuss metadata management issues. In Chapter 6, we present a framework through which user requests expressed in natural language can be mapped into database queries in order to support index structure. In Chapter 7 we give a detailed description of the prototype of our system, and provide data showing how our ontology-based model compares with traditional keyword-based search technique. Finally, in Chapter 8 we present our conclusions and plans for future work.

Chapter 2 Related Works

Several attempts have been made to improve the effectiveness of information retrieval in various areas. In this chapter we will begin by summarizing key related efforts. First, we will discuss the matter from an information retrieval perspective. Second, we will present some modeling techniques for information selection from a database management perspective. Third, we will present some findings from research conducted in the specific area of audio information retrieval. Fourth, we will present related features from the perspective of a commercial search engine, and finally, from a natural language perspective.

2.1 The Information Retrieval (IR) Perspective

The classic models in the area of information retrieval, Boolean [97], vector space [77, 80], and probabilistic [23], all begin by identifying each document through a set of representative keywords called index terms. An index term is simply a word whose semantic reference serves as a mnemonic device for recalling the main themes of the document. Thus index terms are used to index and summarize the content of a document. Given a set of index terms for a document, we notice that not all terms are equally useful. In fact, with regard to describing content there are index terms which are simply more vague than others. Deciding upon the importance of a term for summarizing the contents of a document is not a trivial issue.

The Boolean model is a simple retrieval model based on set theory and Boolean algebra. This model provides a framework which is easy to grasp for ordinary users of an IR system. Furthermore, queries are specified as Boolean expressions which have precise semantics, since the Boolean model retrieval strategy is based on a binary decision criterion without any notion of a grading scale of the kind which prevents good retrieval performance.

In the vector model, documents and queries are represented as vectors in a multi dimensional space. The vector model proposes a framework in which partial matching is possible. This is accomplished through the assignment of non-binary weights to index terms in both queries and documents. These weights are ultimately used to compute the degree of similarity between a user query and each document stored in the system.

In the probabilistic model, the framework for modeling documents and query representations is based on probability theory. Given a user query, this model presumes, there is a set of documents which contain exactly the relevant documents and none other, and which is the ideal answer set. Given a complete description of this ideal answer set, we would have no problem retrieving its documents. Thus, we can think of the querying process as a process of specifying the properties of this ideal answer set. Clearly, there are certain index terms whose semantic references will characterize the properties of this ideal answer set. However, since these properties are not known at the time of the query, an effort will be made initially to guess, or estimate, what these terms might be. This initial estimate allows us to generate a preliminary probabilistic description set approximating the ideal answer. This probabilistic description set is then used to retrieve an initial approximate set of documents. An interaction with the user is then initiated with the purpose of improving the probabilistic description of the ideal answer set through user feedback.

Beside these, alternative modeling paradigms such as fuzzy [106], extended Boolean retrieval models [79], latent semantic indexing [24], and neural networks [94, 100] have been proposed. Among these, the vector space model is one of the most popular and widely used in IR [7].

In response to a request that information of a specific nature be selected, a search technique is generally employed, implemented by a Boolean or vector model. Documents are only retrieved in response to keywords specified by the user, a limitation which can be addressed through the use of query expansion mechanisms. In this process, additional search terms are added to the original query, based on the statistical co-occurrence of these terms [20]. However, attempts at query expansion of this nature have not been very successful, since the use of a statistical method does not result in good control over which terms should be added and which terms pruned out. Although recall is expanded through the addition of new terms, this occurs at the expense of deteriorating precision [12, 70, 72]. The inferiority of the keyword search technique relative to a concept-based technique can be seen through an example. Take the case in which a query is specified in terms of motor vehicle, and new terms like bus, truck, and car, are added to the original query. Given the fact that the intent of the original query is to retrieve information about automobiles, the addition of the terms bus and truck is not helpful. In this situation, we require the use of a conceptual hierarchy in which one concept subsumes other concepts [103]. In the example given, the concept "automobile" would rest on the top of a hierarchy in which a variety of sub-concepts would be enumerated. In our model, this type of hierarchy constitutes an ontology within which concepts related to queries and documents can be mapped into conceptual space in which measures of similarity are applied.

2.1.1 Ontology-based Retrieval

Historically ontologies have been employed to achieve better precision and recall in the text retrieval system [32]. Here, attempts have taken two directions, query expansion through the use of semantically related-terms, and the use of conceptual distance measures, as in our model. Among attempts using semantically related terms, query expansion with a generic ontology, WordNet [63], has been shown to be potentially relevant to enhanced recall, as it permits matching a query to relevant documents that do not contain any of the original query terms. Voorhees [96] manually expands 50 queries over a TREC-1 collection using WordNet, and observes that expansion was useful for short, incomplete queries, but not promising for complete topic statements. Further, for short queries, automatic expansion is not trivial; it may degrade rather than enhance retrieval performance. This is because WordNet is too incomplete to model a domain sufficiently. Furthermore, for short queries less context is available, which makes the query vague. Therefore, it is hard to choose appropriate concepts automatically.

In [65] the query expansion mechanism is manual, and is used for simultaneously multiple domain ontologies, along with obtaining a measure of the imprecision of the retrieval process.

The notion of conceptual distance between query and document provides an alternative approach to modeling relevance. Smeaton et al. [86] and Gonzalo et al. [28] focus on managing short and long documents, respectively. Note here that in these approaches queries and document terms are manually disambiguated using WordNet. In our case, query expansion and the selection of concepts, along with the use of the disambiguation algorithm, is fully automatic.

We introduce the subject of ontology in detail in Chapter 4.

2.2 The Database Perspective

Database management systems are concerned with the storage, maintenance, and retrieval of the factual data which is available in the system in explicit form. It is important to note that the information does not appear as natural language text but is only in the form of specific data elements which are stored in tables. In a database environment each item or record is thus separated into several fields with each field containing the value for a specific characteristic or attribute identifying the corresponding record with which it is linked. The information retrieved in a query will consists of all those records or items which are an exact match for the stated search request. The retrieved information will consists of all records which match the stated search request exactly. In information retrieval, as opposed to database management system, it is often difficult to formulate precise information requests, and the retrieved information may include items that may or may not match the information requests exactly [104].

Although we use audio as the medium with which to demonstrate our model, we also show the related work in the video domain which is closest to and which complements our approach in the context of data modeling. Table 1 summarizes the key efforts in the area of data modeling techniques from a multimedia information selection perspective.

Key related work in the video domain for the selection of video segments includes [1, 38, 68]. Of these, Omoto et al. uses a knowledge hierarchy to facilitate annotation, while others use simple keyword based techniques without a hierarchy (see Table 1). The model of Omoto et al. fails to provide a mechanism that automatically converts generalized descriptions into specialized ones. Further, this annotation is manual, and does not deal with the disambiguation issues related to concepts.

Table 1 Comparisons of Different Data Modeling for Information Selection

	Basis	Existing Query Language	Use of Hierarchy	Optimizations
Omoto et al.	Schemaless	No	Yes (limited use)	No
Adali et al.	Spatial data structure and segment tree	No	No	No
Hjelsvold et al.	General data model using ER	No	No	No
Our model	Relies on ontology	Yes (SQL)	Yes (ontology)	Yes

Omoto et al. [68] also proposes a schemaless video object data model. In this model, a video frame sequence is modeled as an object, with contents described in terms of attributes and attribute values. Each object is described by a set of starting and ending frames. In addition, an interval is described in the same manner, by a starting and an ending frame. New objects are composed from existing objects, and some attributes/values are inherited from these existing objects based on the principle of interval inclusion. Omoto et al. also proposes a query language, VideoSQL, to retrieve video objects through the specification of specific attributes and values.

Adali et al. [1] develop a video data model that exploits spatial data structures (e.g., characters in a movie), rather than temporal objects as is the case in our data model. In Adali et al. objects, activities, and roles are identified in the video frames. Each object and event is associated with a set of frame sequences, and a simple SQL-like video query language is developed. This query language provides the user more flexibility for controlling presentations.

Hjelsvold et al. [38] propose a "generic" video data model and query language for the support, structuring, sharing, and reuse of video. Their scheme builds on an enhanced-ER data model. Their simple SQL-like query language supports video browsing. For annotation, they use a stratification approach [85]. Note that their framework assumes fixed categories for annotations such as persons, locations, and events. Hence, their data model follows conventional schema design with a fixed, rather than an arbitrary, attribute structure.

We propose an ontology-based model for customized selection and delivery, and demonstrate that for annotation an ontology can provide representational terms for objects in a particular domain. We also propose a mechanism which will permit automatic concept selection from documents and user requests using a domain dependent ontology along with a disambiguation algorithm. For the query, we have not built anything from scratch. We use the most widely used query language, SQL. We also demonstrate some novel optimization techniques that rewrite the SQL with the help of knowledge derived from the ontology, without jeopardizing precision and recall.

Other than ours, the only data model to employed a hierarchy is the one proposed by Omoto et al. While we use an ontology to build a hierarchy Omoto et al. use IS-A. An advantage to using an ontology is that the ontology automatically provides different levels of abstraction for querying the system. This is because in regard to objects only descriptions which have been reduced to specificity are annotated. User requests which are originally described in terms of generalized descriptions become recast in the form of specific descriptions by traversing the ontology, and are then expressed in SQL. This top-down approach enables the user to query the system in a more expressive or abstract way.

By contrast, the data model of Omoto et al. fails to offer any mechanism which automatically converts generalized descriptions into specific descriptions. Instead, on the contrary, IS-A begins with existing objects and then facilitates the generation of generalized descriptions based on the specific descriptions of these objects. In this case, the user creates a new object by explicitly merging two existing objects. The description of the new object then takes the form of a generalization based upon the original specific descriptions. However, there is a possibility that certain objects annotated with specific descriptions will not be retrieved in the form of a subsequent generalized description because the user has not yet chosen to merge these objects. For example, in the Omoto et al. model one video object is annotated as "President: Bill Clinton" and another video object annotated as "President: George Bush."   When the user defines a new object by merging these two video objects the new object is automatically annotated as "President: American Statesman." This is because in the IS-A hierarchy "American Statesman" is the generalization of "Bill Clinton" and "George Bush." If the two video objects are merged, the query, specified by video object "American Statesman," can retrieve this new video object. However, if these video objects are not merged by the user, the query specified by "American Statesman" will not succeed in retrieving them at all.

2.3 The Audio Retrieval Perspective

Existing research into issues involved in the management of content-based audio data is very limited. Two general directions can be identified. The first direction found in this research involves the study of mechanisms for processing spoken queries [8, 19, 102], as in the case of query-by-humming [26]. A natural way of querying an audio database of songs is to hum the tune of a song. The techniques for doing this address the issue of how to specify a hummed query and how to report on an efficient query execution implementation by using approximate pattern matching. The approach hinges upon the observation that melodic contour, defined as the sequence of relative differences in pitch between successive notes, can be used to discriminate between melodies.

Within this sphere, in [13], researchers are working on a video mail retrieval system that currently accepts 35 spoken query words. The second direction, audio analysis, involves the study of the retrieval of both video information and spoken documents. Video information retrieval [36, 69], such as Informedia indexing, relies not only on closed captions but also upon audio transcription, in which the Sphinx speech recognition is used for the conversion of speech to text [35]. In spoken document retrieval systems documents are processed by a speech recognition engine system, while transcripts generated for these documents are fed into a classical (textual) IR system. Through the use of such transcripts, a number of spoken documents can be retrieved in response to a textual query.

2.4 The Search Engine Perspective

Our ontology-based audio information retrieval system is in essence an ontology-based search engine. Therefore, in this connection, we would like briefly to present a picture of the underlying technology for commercial web search engines. The present underlying technology for commercial search engines requires that all queries be answered without accessing the text, since only indices are available in the web search engine [33, 59]. This is similar to the situation obtaining in our ontology-based model. However, and this is different from what we propose, matching in commercial search engines relies on keyword-based techniques. For this purpose, most search engines use a centralized crawler-indexer architecture. Crawlers are programs (software agents) that traverse the web, sending new or updated pages to a main server, where they are indexed. In spite of the name, a crawler does not actually move to and run on remote machines. The crawler runs on a local system and sends requests to a remote web server. These requests are indexed and the index then used in a centralized fashion to answer queries submitted from different places in the web. Most search engines carry out ranking through the use of some variation of a Boolean or vector model [105]. As with searching, ranking has to be carried out on the basis of the index alone, without accessing the text. Most indices use variants of an inverted file, which is a list of sorted words (vocabulary), each one having a set of pointers referring to the pages where its elements occur. By using compression techniques the size of the index can be reduced. Thus in commercial search engines a query is answered by doing a binary search on the sorted list of words of the inverted file. If the search involves multiple words, the results have to be combined through aggregation to generate the final answer [9, 10, 11].

Key function in search mechanisms is the categorization of documents; ontologies also have a role in the categorization of documents. For example, Yahoo represents an attempt to organize web pages into a hierarchical index of more than 1,50,000 categories. Here, classification of submitted web pages to categories is done manually. However, Labrou et al. [55] have conducted some experiments which make the classification automatic using n-grams.

2.5 The Natural Language Perspective

A large part of the information stored in bibliographic retrieval systems, and the web in general, consists of data in the form of a natural language, since many users prefer to approach a retrieval system by using natural language formulations of their information requests [42, 49]. For example, automatic question-answering systems are being designed in which the system is expected to give explicit answers to incoming search requests such as "what is boiling point of water?" with answer, "100 degrees Celsius," as opposed to merely furnishing bibliographic references/web pages expected to contain the answer. Cymfony [84], TARGET, FREESTYLE [93], and WIN [81] are the few systems that provide functional “natural language search.” Currently, usable language processing techniques appear inadequate for full utilization under operational retrieval conditions. This is because the task of language analysis is difficult and raises complicated issues [91].

Chapter 3 Research Context: Audio

In this study the mechanism for information selection has been developed in the domain of audio. In this chapter we will initially present techniques for breaking down audio data into segments. Next, we will mention different techniques for content extraction of audio. Finally, we will give a formal definition of audio objects.

Audio is one of the most powerful and expressive of the non-textual media. Moreover, audio information can be of significant benefit to the visually impaired. Audio is a streaming medium (temporally extended), and its properties make it a popular medium for capturing and presenting information. At the same time, these very properties, along with audio's opaque relationship to computers, present several technical challenges from the perspective of data management [27].

The type of audio considered here is broadcast audio. In general, within a broadcast audio stream, some items are of interest to the user and some are not. Therefore, we need to identify the boundaries of news items of interest so that these segments can be directly and efficiently retrieved in response to a user query. After segmentation, in order to retrieve a set of segments that match with a user request, we need to specify the content of segments. This can be achieved using content extraction through speech recognition. Therefore, we present segmentation and content extraction technique one by one.

3.1 Segmentation of Audio

Since audio is by nature totally serial, random access to audio information may be of limited use. To facilitate access to useful segments of audio information within an audio recording deemed relevant by a user, we need to identify entry points/jump locations. Further, multiple contiguous segments may form a relevant and useful news item.

As a starting point both a change of speaker and long pauses can serve to identify entry points [3]. For long pause detection, we use short-time energy (E_n), which provides a measurement for distinguishing speech from silence for a frame (consisting of a fixed number of samples) which can be calculated by the following equation [73]:

$$\begin{aligned} E_n &= \sum_{m=-\infty}^{m=\infty} [x(m)w(n-m)]^2 \\ &= \sum_{m=n-N+1}^{m=n} x(m)^2 \end{aligned} \qquad \textbf{(3.1)}$$

Where $x(m)$ is discrete audio signals, n is the index of the short-time energy, and $w(m)$ is a rectangle window of length N. When the E_n falls below a certain threshold we treat this frame as pause. After such a pause has been detected we can combine several adjacent pauses and identify what can be called a *long pause*. Therefore, the presence of speeches with starting and ending points defined in terms of long pauses allows us to detect the boundaries of audio segments.

3.2 Content Extraction

To specify the content of media objects two main approaches have been employed to this end: fully automated content extraction [35], and selected content extraction [99]. Due to the weakness of the fully automated content extraction in state-of-art audio/speech recognition we have chosen the latter approach, i.e., selected content extraction. This is because, as Hauptman has shown in the Informedia project, automatic transcription (indexing) of speech is a difficult task [35]. This follows from the fact that the current speech recognition systems support limited vocabulary (64,000 word forms), at least an order of magnitude smaller than that of a text retrieval system [98]. Additionally, environmental noise generates inevitable speech recognition error. Hence, in the Informedia project it was shown that, after the conversion of speech into text, information retrieval results in terms of precision and recall may suffer. In other words, the user may receive too much irrelevant data or miss some relevant data. Therefore, to insure the appropriate selection and presentation of audio information, we advocate selected content extraction. For this process, our goal is to identify a particular set of keywords in the audio segment. For this, the techniques developed in word-spotting can be employed [43, 99].

3.2.1 Word-spotting

Word-spotting techniques can provide selected content extraction in a manner that will make the content extraction process automatic. Word-spotting is a particular application of automatic speech recognition techniques in which the vocabulary of interest is relatively small. In our case, vocabularies of concepts from the ontology can be used. It is the job of the recognizer to recognize and pick out in the speech (in our case an audio information unit/ an audio object/a number of contiguous segments) only occurrences of keywords from this vocabulary. Thus, the input for word-spotting is the audio object and the keywords from ontologies. The output of a wordspotter is typically a list of keyword "hits," or matches in the audio object. For example, if the occurrence of the keyword "NFL" is determined in a particular audio object, the output of the word-spotting technique for this object should contain keyword, “NFL.” Through employing a restricted number of keywords we argue that we will get a better rate of speech recognition. For example, Brown et al. [13] have investigated using 35 pre selected keywords for video mail retrieval system and reported retrieval accuracy near 98% for spoken queries.

3.2.2 Manual Annotation

Human intervention may be required to reduce speech recognition error. Furthermore, content description can be provided in plain text, such as closed captions. However, this manual annotation is labor intensive. For content extraction we rely on closed captions that came with audio clip itself from fox sports and CNN web site in our case (see Section 7.1).

3.3 Definition of an Audio Object

An audio object, by definition and in practice, is composed of a sequence of contiguous segments. Thus, in our model the start time of the first segment and the end time of the last segment of these contiguous segments are used respectively to denote start time and end time of the audio object. Further, in our model, pauses between interior segments are kept intact in order to insure that speech will be intelligible. The formal definition of an audio object indicates that an audio object's description is provided by a set of self-explanatory tags or labels using ontologies (see Section 5.1 for more details).

- An audio-object O_i is defined by five tuple (id_i, S_i, E_i, V_i, A_i) where
  - Identifier: Id_i is an object identifier which is unique
  - Start time: S_i is the start time

- End time: E_i is the end time. Start time and end time satisfy E_i-S_i>0
- Description: V_i is a finite set of tag or label, i.e., $V_i=\{v_{1i}, v_{2i}, \dots, v_{ji}, \dots, :v_{ni}\}$ for a particular j where v_{ji} is a tag or label name.
- Audio data: A_i is simply audio recording for that time period.

For example, an audio object is defined as {10, 1145.59, 1356.00, {Gretzky Wayne}, *}. Here, the identifier of the object is 10, start time and end time are 1145.59, and 1356.00 unit respectively, and the description is "Gretzky Wayne", while * denotes audio data. Of the information in the five tuple, the first four items (identifier, start time, end time, and description) are called *metadata*.

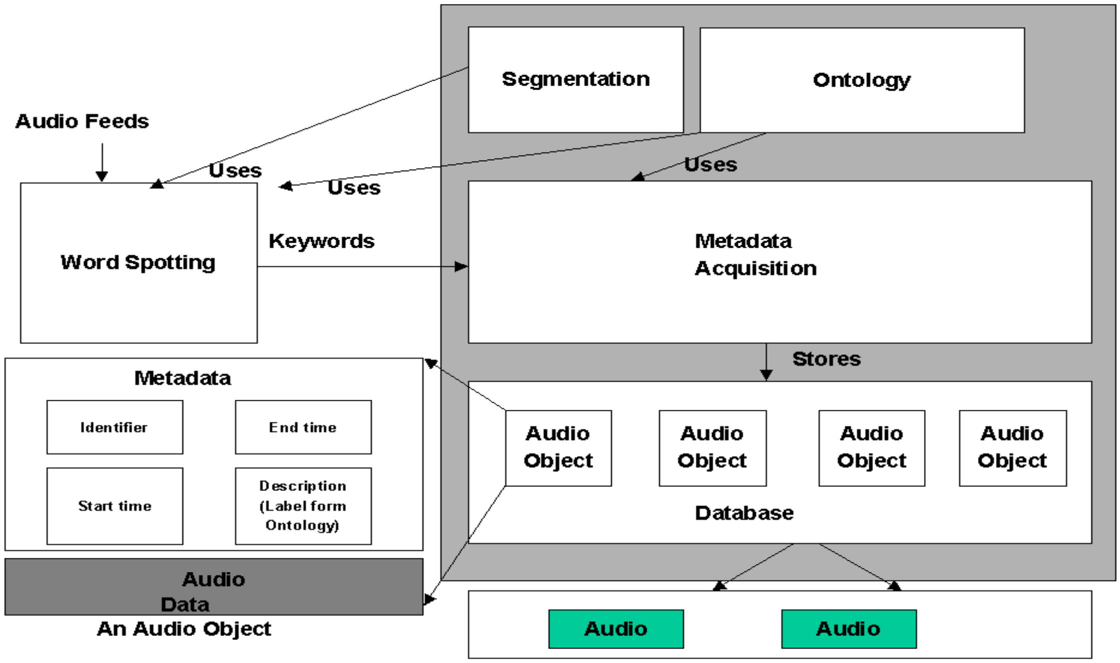


Figure 3.1 Architecture of our Experimental Metadata Generation Context

Figure 3.1 discusses the architecture of our metadata generation system. After obtaining keywords in an audio object in the phase of content extraction, we need to select the right concepts from ontologies for this audio object. The mechanism for the selection of concepts is discussed in Section 5.1, where we demonstrate that ontologies help to disambiguate concepts for audio objects. It is important to note that these jobs are done in a phase of preparation or off-line.

Chapter 4 Ontologies

The purpose of this chapter is to introduce ontologies.

An ontology is an explicit specification of a conceptualization [14, 30, 31]. The term is borrowed from philosophy, where ontology is a systematic account of being or existence, from the Greek ontos (being). For AI systems, what "exists" is that which can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. This set of objects, and the describable relationships among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge. Thus, in the context of AI, we can describe the ontology of a program by defining a set of representational terms. In such an ontology, definitions are used to associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names mean, and formal axioms that constrain the interpretation and focus the well-formed use of these terms. Formally, an ontology is the statement of a logical theory.

The word "ontology" seems to generate a lot of controversy in discussions about AI. While everybody agrees that ontologies are important, there is debate about how to draw a dividing line between ontologies and a number of other approaches (e.g object models) to representing concepts and conceptualization. The quibbling arises when we dive deeper and ask: "how formal or rich does this specification need to be before one can call it an ontology." The AI community views ontologies as formal logical theories whereby we are not only defining terms and relationships, but formally defining the context in which the term (relationship) applies, and facts and relationships are implied. These ontological theories are formal enough to be testable for soundness and completeness by theorem provers. In contrast, databases and other communities view ontologies more as object models, taxonomies and schemas, and do not explicitly express important constraints. Linguistic ontologies (e.g., WordNet) and thesauri express various relationships between concepts (e.g synonymy, antonymy, is_a, contains_a), but do not explicitly and formally describe what a concept means.

Therefore, an ontology defines a set of representational terms, which are called *concepts*. Interrelationships among the concepts describe a target world. An ontology can be constructed in two ways, domain dependent and generic. CYC [57, 58], WordNet [63], and Sensus [92] are examples of generic ontologies; their purpose is to make a general framework for all (or most) categories encountered by human existence. Generic ontologies are generally very large but not very detailed--it is difficult to build them.

For our purposes, we are interested in creating domain dependent ontologies which are generally much smaller. First, this is because a domain dependent ontology provides concepts in a fine grain, while generic ontologies provide concepts in coarser grain. Second, domain ontologies do not contribute the large number of concepts which result in speech recognition errors characteristic of generic ontologies. Finally, encoded knowledge in domain dependent ontologies helps us to disambiguate concepts and to choose those which are most relevant for audio objects.

Ontologies are usually constructed by a domain expert, someone who has mastery over the specific content of a domain [66]. During the construction of ontologies the following points are kept in mind [40]. Ontologies should be:

- Open and dynamic: Ontologies should have fluid boundaries and be readily capable of growth and modification.
- Scalable and inter-operable: An ontology should be easily scaled to a wider domain and adapt itself to new requirements.
- Easily maintained: It should be easy to keep ontologies up-to-date. Ontologies should have a simple, clear structure, as well as be modular. They should also be easy for humans to inspect.

4.1 Ontology Concepts

Figure 4.1 shows an example of an ontology for sports news. An ontology of this nature is usually obtained through the use of generic sports terminology, as well as information provided by domain experts [21], and is described by a directed acyclic graph (*DAG*) in which each node in the DAG represents a concept. Concept is a class of items that together share essential properties that define that class. In general, each concept in the ontology contains a label name which is unique to the ontology, and a list of synonyms. Further, this label name is used as a basis for associating concepts with audio objects. The list of synonyms of a concept contains a vocabulary (a set of keywords) through which the concept can be matched with user requests and associated with audio objects. Formally, each concept has a list of synonyms $(l_{1,}\ l_{2,}\ l_{3,} \ldots,\ l_{i}\ ,\ldots,l_{n})$ through which user requests are matched with a given set l_i, which constitutes an *element*

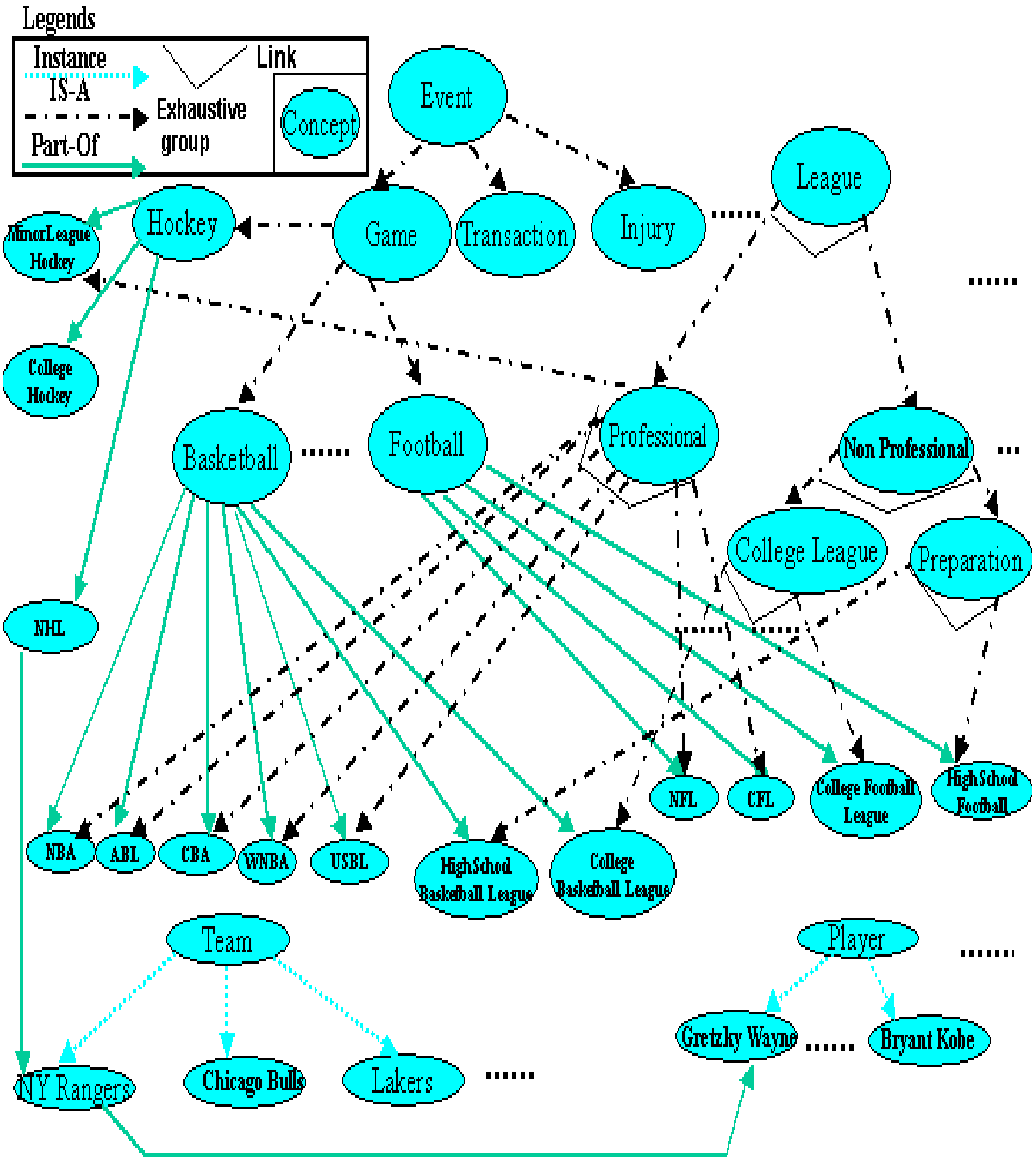


Figure 4.1 A Small Portion of an Ontology for Sports Domain

in the list. Note that a keyword may be shared by lists of synonyms referring to multiple concepts. For example, players "Bryant Kobe," "Bryant Mark," "Reeves Bryant" share the common word "Bryant" which may create problems of ambiguity. Moreover, each belongs to the league NBA. Hence, each of these concepts' labels has as a prefix the label for the concept NBA that permits the efficient generation of queries for upper level concepts (see Section 6.2 for more details).

4.2 Interrelationships

In the ontology, concepts are interconnected by means of interrelationships. If there is an interrelationship, *R*, between concepts C_i and C_j, then there is also an interrelationship, *R'*, between concepts C_j and C_i. In Figure 4.1, interrelationships are represented by labeled arcs/links. Three kinds of interrelationships are used in the creation of ontologies of the type we employ: IS-A, Instance-Of, and Part-Of. These correspond to key abstraction primitives in object-based and semantic data models [6].

4.2.1 IS-A

This "IS-A" interrelationship is used to represent concept inclusion. A concept represented by C_j is said to be a specialization of the concept represented by C_i if C_j is a kind of C_i, or an example of a C_i. For example, "NFL" is a kind of "Professional" league. In other words, "Professional" league is the generalization of "NFL." In Figure 4.1, the IS-A interrelationship between C_i and C_j goes from generic concept C_i to specific concept C_j, represented by a broken line. Sets of IS-A interrelationships can be further categorized into two types: *exhaustive group and non-exhaustive group*. An exhaustive group consists of a number of IS-A interrelationships between a generalized concept and a set of specialized concepts, and places the generalized concept into a categorical relation with a set of specialized concepts in such a way so that the union of these specialized concepts is equal to the generalized concept. In other words, the category exhausts or encompasses all its possible members, and vice versa, the sum of the members constitute the category. For example, "Professional" relates to a set of concepts, "NBA", "ABL", "CBA", ..., by exhaustive group (denoted by caps in Figure 4.1). Further, when a generalized concept is associated with a set of specific concepts by only IS-A interrelationships that fall into the exhaustive group, this generalized concept will not participate explicitly the metadata acquisition and SQL query generation. This is because this generalized concept is entirely partitioned into its specialized concepts through an exhaustive group. We call this generalized concept a *non-participant concept (NPC)*. For example, in Figure 4.1 the concept "Professional" is an NPC. On the other hand, a non-exhaustive group consisting of a set of IS-A does not exhaustively categorize a generalized concept into a set of specialized concepts. In other words, a union of specialized concepts is not equal to the generalized concept.

Specialized concepts inherit all the properties of the more generic concept and add at least one property that distinguishes them from their generalizations. For example, "NBA" inherits the properties of its generalization "Professional," but is distinguished from other leagues by the type of game, skill of the participants, and so on.

4.2.2 Instance-Of

An instance denotes a single named existing entity but not a class. This is used to show membership. If a C_j is a member of concept C_i then the interrelationship between them corresponds to an Instance-Of denoted by a dotted line. Player "Wayne Gretzky" is an instance of the concept "Player." In general, all players and teams are instances of the concepts, "Player" and "Team" respectively.

4.2.3 Part-Of

A concept is represented by C_j is Part-Of a concept represented by C_i if C_i has a C_j (as a part), or C_j is a part of C_i. For example, the concept "NFL" is Part-Of the concept "Football," and player, "Gretzky Wayne" is Part-Of the concept team, "NY Rangers."

4.3 Disjunctness

When a number of concepts are associated with a parent concept through an IS-A interrelationship, it is important to note that these concepts are disjoint, and are referred to as concepts of a disjoint type. When, for example, the concepts "NBA", "CBA", or "NFL" are associated with the parent concept "Professional," through IS-A, they become disjoint concepts. Moreover, any given object's metadata cannot possess more than one such concept of the disjoint type. For example, when an object's metadata is the concept "NBA," it cannot be associated with another disjoint concept, such as "NFL." It is of note that the property of being disjoint helps to disambiguate concepts for keywords during the phase of concept selection and disambiguation (see Section 5.1 and 6.1). Similarly, the concepts "College Football League" and "College Basketball League" are disjoint concepts due to their associations with the parent concept "College League" through an IS-A interrelationship. Furthermore, "Professional" and "Non Professional" are disjoint. Thus, we can say that "NBA," "CBA," "ABL," "College Basketball," and "College Football," are all disjoint.

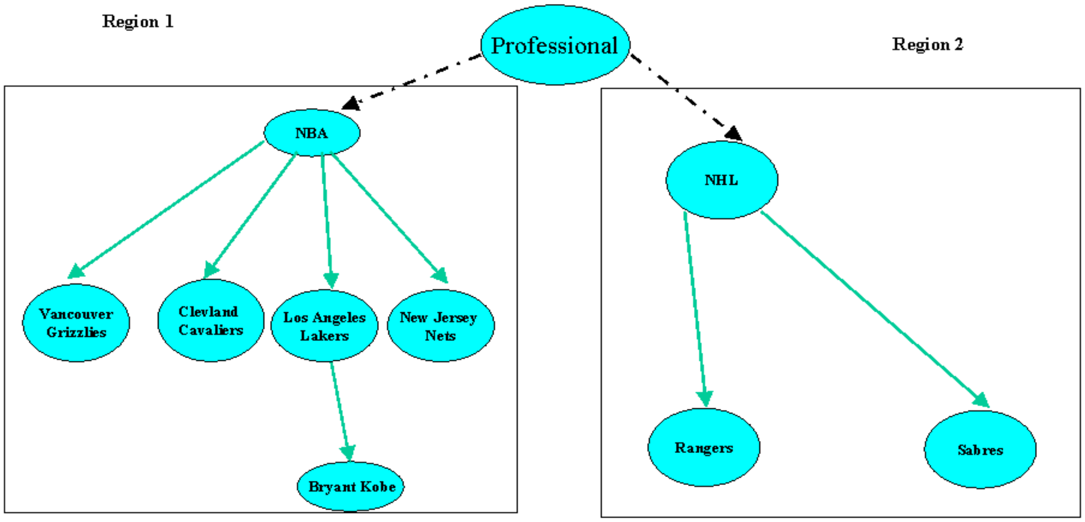


Figure 4.2 Different Regions of Ontologies

Each of these leagues, or associations, along with its teams and players, forms a boundary around what we call a *region* (see Figure 4.2). During the disambiguation of concepts for an audio object our goal is to arrive at a particular region. This is because an audio object can be associated with only one concept of the disjoint type. However, it may be possible that a particular player may play in several leagues. For this we consider two alternatives. First, we will generate multiple instances of the player in the ontology. In other words, for each league in which the player plays he will be represented by a separate concept. In this manner we are able to preserve the property of disjunction. In this case, each region is simply a sub-tree. Second, we will keep just one node for the player that have two parents (say), two teams. In this case, each region is DAG. With the former approach, maintenance or update will be an issue; inconsistency may arise. With the latter approach, maintenance will be easier; however, precision will be hurt. This is because if the query is requested in terms of a team where this player plays, some of retrieved objects will be related to other team and vice versa. This is because both teams have common child concept and query expansion phase allows to retrieve all associated audio objects to this player regardless of his teams. For example, player "Deion Sanders" plays two teams "Dallas Cowboys" (under NFL region) and Cincinnati Reds (under MLB region). If user request is specified by "Dallas Cowboys" some objects will be retrieved that contain information Cincinnati Reds along with Deion Sanders. For this, we adopt former approach.

Concepts are not disjoint, on the other hand, when they are associated with a parent concept through either the relationship of Instance-Of or Part-Of. In this case, some of these concepts may serve simultaneously as metadata for an audio object. An example would be the case in which the metadata of an audio object are team "NY Rangers" and player "Wayne Gretzky," where "Wayne Gretzky" is Part-Of "NY Rangers."

The sample content of concepts is as follows:

- NY Rangers:

  Label: NHLTeam11

  Instance: Team

  Part-Of: NHL

  Synonyms list: NY Rangers, New York Rangers, . . .

- NHL

Label: NHL

IS-A: Professional

Part-Of: Hockey

Synonyms list: NHL, National Hockey League, . . .

Thus, the labels for the concepts NY Rangers and NHL are NHLTeam11, and NHL respectively. The concept NY Rangers is associated with the concepts "Team" and "NHL" through Instance-Of and Part-Of interrelationships.

4.4 Creating an Ontology

For the construction of sports domain dependent ontologies, first, we list all possible objects necessary to cover a given sports domain. This possible object list should include different sports, such as basketball, football, baseball, hockey etc, and different leagues within a given sport, as in basketball, NBA, ABL, CBA and so forth. Furthermore, different sports can be qualified by characteristics such as injuries, player transactions, strikes, etc.

Now, the question becomes what level of granularity of knowledge do we need to take into account in the ontology? Since our goal is to build a search mechanism which is more powerful than keyword-based search technique, without relying upon the understanding of natural language, we do not, in our ontology, represent concepts at the level of granularity necessary, for example, for the extraction of highly specific factual information from documents (e.g., how many times a batter attempted to execute a hit and run play in a particular game). Yet, the term transaction has been further qualified through the use of such designations, within a given sports domain, as "sign a player," "retire a player," "release a player," "trade a player," and "draft a player," and so on. Furthermore, all instances of teams, players, and managers are grouped under "team", "player," and "manager" respectively through Instance-Of interrelationship. Note that, specifically, a set of players who play in a team are grouped into that team through Part-Of interrelationships. All teams, players, and managers for different leagues are taken from Yahoo. Note that Yahoo has a hierarchy of 1,50,000 categories. Only a part of this hierarchy has been used for the construction of our domain dependent ontology. Therefore, upper level concepts are chosen manually; however, lower level concepts are borrowed from the Yahoo hierarchy. Furthermore, there, all the last names of the players come first, with the first names coming later. Therefore, e.g., in Figures 4.1, 4.2, 5.1, 5.2, and so forth the players' last names are shown first. The maximum depth of our ontology's DAG is six. The maximum number of children concepts from a concept is 28 (branching factor).

Now the question is: how much does this ontology cover? What is the criteria for the test model? Since lower level concepts are taken from the Yahoo hierarchy, and more upper level concepts are added in the ontology, we believe that the ontology covers the domain reasonably. For this, we have done some experiments in order to estimate coverage. In these experiments our goal is to select concepts from ontologies for the annotated text of audio clips. These clips are taken from the CNN sports and the Fox sports web site along with their closed captions, not from the Yahoo web site. We have observed that 90.5% of the clips are associated with concepts of ontologies, while 9.5%

of the clips failed to associate with any concept of ontologies due to incompleteness of ontologies (see Section 7.3.1).

It is important to note that players may change teams frequently and that this may raise problems of maintenance or consistency. In this dissertation, we have not addressed this issue, doing so will be part of our future work (see Section 8.1). Furthermore, one important observation about our ontology is that it is constructed from the perspective of a database. Therefore, in the ontology the mechanism of inference sometimes may not be supported among concepts for different links or arcs.

Chapter 5 Metadata Acquisition

Metadata acquisition is the name for the process through which descriptions are provided for audio objects. In this chapter we first, present the model of the mechanism though which concepts are selected from ontologies to facilitate words to meaning mapping, along with the automatic disambiguation algorithm. Next, we will present metadata management issues that are raised in association with these operations.

5.1 Concept Selection and Disambiguation Mechanisms

Our model features an automatic disambiguation algorithm [52] for choosing appropriate concepts for a group of keywords, and we propose further refinements along these lines.

For each audio object we need to find the most appropriate concept(s). Recall that using word-spotting or closed-captions we get a set of keywords which appear in a given audio object. Now we need to map these keywords in conceptual space. In other words, we need to extract concepts from keywords. This is because matching between user requests and documents is done in conceptual space rather than through keyword matching. For this, concepts from ontologies will be selected based on matching terms taken from their lists of synonyms with those based on specified keywords. Furthermore, each of these selected concepts will have a score based on a partial or a full match. It is important to note that keywords in the list of synonyms might only be a variant of keywords present in a relevant document. Plural, gerund forms, and past-tense suffixes are examples of syntactic variations which prevent a perfect match between keywords from the list of synonyms and keywords in matching documents. This problem can be partially overcome through replacing these keywords with their respective stems. This is called *stemming* [71]. A stem is the portion of the keyword which is left after the removal of its affixes (i.e., prefixes and suffixes). For example, connect is the stem for the variants: connected, connecting, connection, and connections. For stemming we used the same algorithm employed in WordNet [64].

It is possible that a particular keyword may be associated with more than one concept in the ontology. In other words, association between keyword and concept is one:many, rather than one:one. Therefore, the disambiguation of concepts is required. The basic notion of disambiguation is that a set of keywords occurring together determine a context for one another, according to which the appropriate senses of the word (its appropriate concept) can be determined. Note, for example, that base, bat, glove may have several interpretations as individual terms, but when taken together, the intent is obviously a reference to baseball. The reference follows from the ability to determine a context for all the terms.

5.1.1 Disambiguation Methods

Thus, extending and formalizing the idea of context in order to achieve the disambiguation of concepts, we propose an efficient pruning algorithm based on two principles: co-occurrence and semantic closeness. This disambiguation algorithm first strives to disambiguate across several regions using first principle, and then disambiguates within a particular region using the second. The basic procedure is as follows. For automatic disambiguation within an ontology a set of regions representing different concepts can be defined. The concepts, as they appear in a given region, will be mutually disjoint from the concepts of other regions. This becomes the basis for determining a group of appropriate concepts for a given keyword or collection of keywords. In short, after keywords are matched to the concepts of a given ontology, the region within the ontology in which the greatest number of selected concepts occurs is determined. This region, the one containing the largest number of selected concepts, will at the time be used to associate with documents and user requests. The selected concepts of other, different, regions will be pruned automatically.

A simple example will make this clear. The keyword "Charlotte" for a particular document is associated with two concepts of the ontology "Charlotte Hornets" and "UNC Charlotte." One is in the region encompassing a professional league, the National Basketball Association, (NBA), the other in the region encompassing college basketball. Thus, at various levels of complexity beyond this simplified example, the disambiguation technique used to distinguish between concepts is based on the general idea that any set of keywords occurring together in context will together determine appropriate concepts for one another, i.e., fall into the same region, in spite of the fact that each individual keyword is multiply ambiguous.

However, since any keyword alone will determine a group of concepts which are both relevant and irrelevant, and which can occur in different regions, we will need to have a way of dealing with the possibility that even within a region selected for annotation a given keyword will match more than one concept. In other words, within a given region multiple ambiguous concepts will have been selected for a particular keyword, necessitating further disambiguation. In order to further prune irrelevant concepts we will need to determine the correlation between concepts selected in a given region. For this, we use the second principle; semantic distance (in the ontology). When concepts are correlated, concepts closely associated will be given greater weight. This association will be based on minimal distance in the ontology and the matching scores of concepts based on the number of keywords they match. Thus, selected concepts which correlate with each other will have a higher score, and a greater probability of being retained than non-correlated concepts. If scores of particular ambiguous concepts fall below a certain *threshold-score*, which will be a minimum score chosen for selected concepts for that particular object, these concepts will be pruned.

For example, the annotated text for a particular audio object might be:

> Lakers keep grooving with 8th straight win. Kobe Bryant scores 21 points as the Lakers remain perfect on their eastern road trip with a 97-89 triumph over the Nets. Bryant discussed the eight game win streak and his performance in the All Star game.

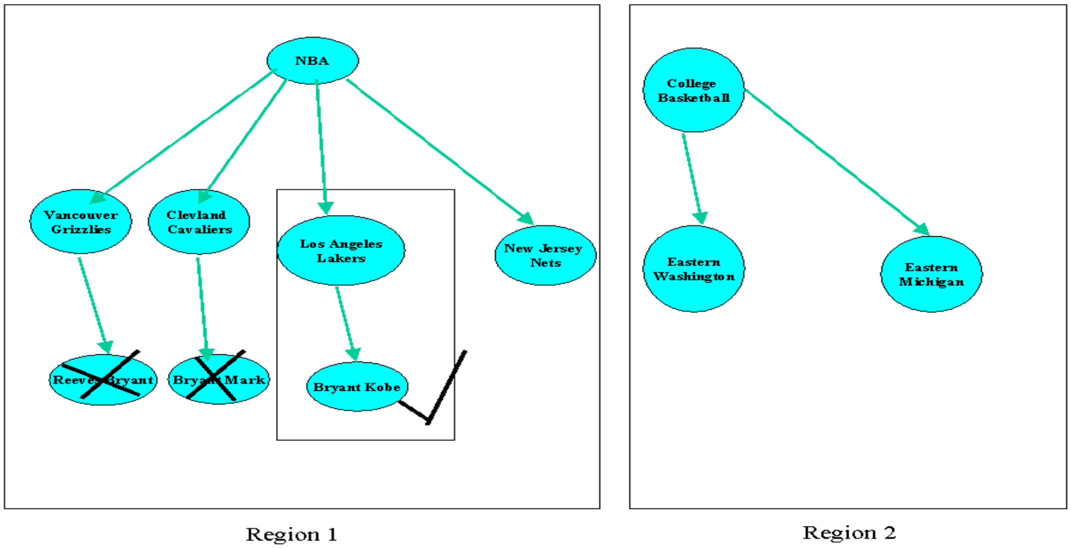


Figure 5.1 Different Regions of Ontology and Disambiguation of Concepts in a Region

The words in italics are the keywords which are associated with the concepts of our ontology. The keywords "Lakers," and "Nets" are associated with the concepts "Los Angeles Lakers" and "New Jersey Nets" respectively. The keyword "Bryant" is associated with the concepts, "Reeves Bryant," "Bryant Mark," and "Bryant Kobe."

It is important to note that all the concepts selected above are found in the region "NBA." However, the keyword "Eastern" is associated with the concepts "Eastern Washington," and "Eastern Michigan" which are not associated with "NBA" but with the region "College Basketball" (see Figure 5.1). If we now choose only the concepts which appear in the region NBA, which is the region in which the greatest number of concepts occur, the concepts "Eastern Washington" and "Eastern Michigan" will be eliminated, since they are not found in that region. Thus, we keep from among the concepts selected those which appear in the region NBA in which the greatest number of concepts occur, and prune other selected concepts.

In the selected region, in this case NBA, a keyword such as "Bryant" may be associated with more than one selected concept. This necessitates further disambiguation. We will want to know what other concept qualifies the concepts selected by keyword "Bryant" through correlation. As noted above in the case of the keyword "Bryant" the concepts "Bryant Kobe," "Bryant Mark," and "Reeves Bryant" are all selected. Among these ambiguous concepts, however, only "Bryant Kobe" is correlated with another selected concept, in this case "Los Angeles Lakers." Therefore, "Bryant Kobe" is kept, and the concepts "Bryant Mark," and "Reeves Bryant" are thrown away (see Figure 5.1).

Thus, we determine the correlation of selected concepts in the region in which the greatest number of keywords have been matched to the audio annotated text, and within that region non-correlated ambiguous concepts are pruned. Finally, the selected concepts for this audio object are "New Jersey Nets," "Los Angeles Lakers," and "Bryant Kobe."

Further, the following example illustrates how the disambiguation algorithm discards irrelevant concepts where a particular audio object semantically carries little information about these concept(s). The annotated text for a particular audio object might be:

> After only two games back, *NBA* bad boy *Dennis Rodman* of the *Dallas Mavericks* has been ejected, fined and suspended. *Rodman* has been suspended without pay for one game and fined $10,000 by the *NBA* for his actions during Tuesday night's home loss to the *Milwaukee Bucks*. *Rodman* expressed his dissatisfaction with the suspension Wednesday by challenging *NBA* commissioner *David Stern* to a boxing match.

The concepts chosen for this audio object by our disambiguation algorithm are player "Dennis Rodman," team "Dallas Mavericks," and team "Milwaukee Bucks," all of which belong to the region "NBA." It is important to note that our disambiguation algorithm chooses relevant concepts correctly, while irrelevant concepts are automatically pruned (e.g., the concept “Boxing”). If, as in this case, the user request embodies the term boxing, our ontology-based model will not retrieve this object. By contrast, this object will be retrieved when keyword-based technique is employed, with a consequent loss of precision, even though the concept of boxing is not part of what is required to provide conceptual closure in this query.

We have implemented the above idea using score-based techniques. To illustrate this technique we first define some terms, and then present our score-based algorithm.

5.1.2 Formal Definitions

Each selected concept contains a score based on the number of keywords from the list of synonyms which have been matched with the annotated audio text. Recall that in an ontology each concept (C_i) has a complementary list of synonyms ($l_{1,}$ $l_{2,}$ $l_{3,}$ ..., l_i ,...,l_n). Keywords in the annotated text are sought which match each keyword on the element l_j of a concept. The calculation of the score for l_j, which we designate an *Escore*, is based on the number of matched keywords of l_j. The largest of these scores is chosen as the score for this concept, and is designated *Score*. Furthermore, when two concepts are correlated, their scores, called the *Propagated-score*, are inversely related to their position (distance) in the ontology. Let us formally define each of these scores.

Definition 5.1: Element-score (Escore): The Element-score of an element l_j for a particular concept C_i is the number of keywords of l_j matched with keywords in the annotated text divided by total number of keywords in l_j.

$$Escore_{ij} \equiv \frac{\#of\ keywords\ of\ l_j\ matched}{\|\#\ of\ keywords\ in\ l_j\ \|} \qquad \textbf{(5.1)}$$

The denominator is used to nullify the effect of the length of l_j on $Escore_{ij}$ and ensures that the final weight is between 0 and 1.

Definition 5.2: Concept-score (Score): The Concept-score for a concept, C_i is the largest score of all its element-scores. Thus,

$$Score_i = max\ Escore_{ij}\ \ where\ 1 \leq\ j\ \leq n \qquad (5.2)$$

Definition 5.3: Region-score (Cscore$_R$): The Region-score ($Cscore_R$) for a region R is the summation of Concept-score of selected concepts that are belonged to this region. Note that for ambiguous concepts for a particular keyword, their average concept-score is calculated and added to the sum rather than taken as the mere sum of the individual scores.

Definition 5.4: Semantic distance (SD (C_i , C_j)): *SD (C_i , C_j)* between concepts C_i and C_j is defined as the shortest path between two concepts, C_i and C_j in the ontology. Note that if concepts are in the same level and no path exists, the semantic distance is infinite. For example, the semantic distance between concepts “NBA” and team “Lakers” is 1 (see Figure 5.2). This is because the two concepts are directly connected via a Part-Of inter-relationship. Similarly, the semantic distance between “NBA,” and “Bryant Kobe” is 2. The semantic distance between “Los Angeles Lakers,” and “New Jersey Nets” is infinite. This distance measure has been studied extensively by Aggire et al. [3] who attempt using WordNet to resolve the lexical ambiguity of nouns. Their use of measures of conceptual distance between concepts is not only sensitive to the shortest path that connects the concepts involved, but also to the depth and density of the hierarchy in which the concepts appear.

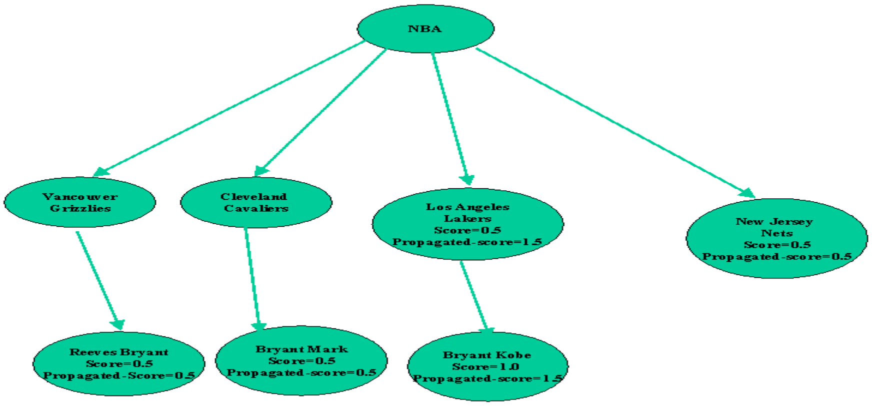

NBA
Vancouver Grizzlies
Cleveland Cavaliers
Los Angeles Lakers Score=0.5 Propagated-score=1.5
New Jersey Nets Score=0.5 Propagated-score=0.5
Reeves Bryant Score=0.5 Propagated-Score=0.5
Bryant Mark Score=0.5 Propagated-score=0.5
Bryant Kobe Score=1.0 Propagated-score=1.5

Figure 5.2 Illustration of Scores and Propagated-scores of Selected Concepts

Definition 5.5: Propagated-score (S_i): If a concept, C_i, is correlated with a set of concepts *(C_j, C_{j+1},...,C_n)*, the propagated-score of C_i is its own Score, $Score_i$ plus the scores of each of the correlated concepts' *(C_k $k=j, j+1, ..., n$) $Score_k$* divided by *$SD(C_i, C_k)$*. Thus,

$$\begin{aligned} S_i &= Score_i + \sum_{k=j}^{k=n} \frac{Score_k}{SD(C_i,C_k)} \\ &= Score_i + \frac{Score_j}{SD(C_i,C_j)} + \frac{Score_{j+1}}{SD(C_i,C_{j+1})} + \ldots + \frac{Score_n}{SD(C_i,C_n)} \end{aligned} \qquad \textbf{(5.3)}$$

Thus, when two concepts are correlated with each other and semantic distance is greater than one, these concepts will have a lower S_i and S_j compared to concepts with the same concept-scores and a semantic distance which is one. This is because for higher semantic distances concepts are correlated in a broader sense. Thus, correlated concepts have a higher S_i than non-correlated concepts. For example, in Figure 5.3 the values of $Score_i$ for "Los Angeles Lakers" and "Bryant Kobe" are 0.5 and 1.0 respectively. Furthermore, these concepts are correlated with a semantic distance of 1, and their Propagated-score (sum of concept scores) is 1.5 (0.5 + 1.0).

Definition 5.6: S_{max}: For an object, S_{max} is the largest score of all its selected concepts' propagated-score, S_i.

Definition 5.7: Threshold-score (γ_{Score}): The Threshold-score for an object is a certain fraction of its S_{max}. It is simply determined by the product of S_{max} and a threshold-constant. This threshold-constant can be between 0 and 1.

It is important to note that the same definitions will be used to identify appropriate concepts that describe user requests (see Section 6.1).

The pseudo-code for the disambiguation algorithm is as follows:

```
For each audio object
    Find concepts ($C_1$, $C_2$, $C_3$, …,$C_i$, …, $C_m$) that are associated with keywords of annotated text of this audio object
    For each region $R$
        $Cscore_R = 0$ //initially
        //Sum of all selected concepts concept-score for a region, $R$
        For each keyword
            If a non ambiguous concept $C_1$ is selected in this region, $R$
                    //add $C_1$ score to Region-score
                    $Cscore_R \equiv Cscore_{RA} + Score_{c1}$
            Else If
                    //ambiguous concepts are selected
                    Selected ambiguous concepts ($C_{k+1}$ ,$C_{k+2}$, …$C_{k+r-2}$, $C_{k+r-1}$, $C_{k+r}$) are in this region, $R$
                    //Calculate their average concept-score, $\mathrm{Cscore}_{RA}$
                    $Cscore_{RA} = \frac{Scorec_{k+1} + Scorec_{k+2} + ... + Scorec_{k+r}}{r}$
                    $Cscore_R = Cscore_R + Cscore_{RA}$
         //End of For Loop for each keyword
    //End of For Loop for each region
    Choose a region with maximum score, $Cscore_R$
            and prune selected concepts in different regions
    For this selected region, determine correlation of concepts ($C_i$, $C_j$,$C_{j+1}$, .., $C_n$) and update their propagated-scores by
    $S_i = Score_i + \sum_{k=j}^{k=n} \frac{Score_k}{SD(C_i, C_k)}$
    $= \mathrm{Score}_i + \frac{Score_j}{SD(C_i, C_j)} + \frac{Score_{j+1}}{SD(C_i, C_{j+1})} + ... + \frac{Score_n}{SD(C_i, C_n)}$
    //Prune non-correlated ambiguous concepts
    Determine maximum score $S_{max}$ among all selected concepts' propagated score $S_i$ for this object
    For each ambiguous concept's propagated score $S_i$
            If ($S_i < \gamma_{Score}$ ($S_{\max}$ * threshold-constant))
                    Simply discard this concept which has $S_i$
            Else
                    Keep this concept
    //End of For Loop each ambiguous concept
//End of For Loop for each audio object
```

Figure 5.3 Pseudo code for Disambiguation Algorithm

There is a trade-off associated with the selection of value of threshold-constant (γ); γ can be 0, 0.1, 0.2,… For high values of γ, we may lose some relevant concepts and at the same time discard many irrelevant concepts for audio objects. On the other hand, for a lower value of γ, we may keep many irrelevant concepts along with those which are correct. Our goal, for a given audio object, is to keep as many relevant concepts as possible and to throw away the maximum number of irrelevant concepts. By increasing γ, we may discard many ambiguous concepts. In this case, some of those discarded are indeed irrelevant for the object, and by throwing out these concepts better precision can be achieved. This is because in the latter case a given irrelevant object will not be retrieved when the user query is related to one of these discarded concepts.

For the example (given in Figure 5.1), the concepts "Los Angeles Lakers," "New Jersey Nets," "Bryant Kobe," "Bryant Mark," and "Reeves Bryant," are selected in the selection of region “NBA.” S_i, propagated-scores for these concepts are 1.5, 0.5, 1.5, 0.5, 0.5 respectively (see Figure 5.2). Note that "Los Angeles Lakers," and "Bryant Kobe" are correlated with semantic distance 1 and "Los," and "New" are removed due to the fact that they belong to a stop list of common words. S_{max} is 1.5 here and ambiguous concepts are "Bryant Kobe," "Bryant Mark," and "Reeves Bryant." If we set $\gamma = 0.6$, then the ambiguous concepts "Bryant Mark," and "Reeves Bryant" are discarded since their S_i scores fall below 0.9 ($S_{max} * \gamma = 1.5 * 0.6$). Although S_i for "New Jersey Nets" is 0.5, which falls below the threshold-score, we keep it because it is not an ambiguous concept.

5.1.3 Characteristics of Disambiguation Algorithm

At this point we present the features of our disambiguation algorithm.

First, through the use of the algorithm it might be possible that a relevant concept may be discarded along with irrelevant ones. This is because a relevant concept may not correlate with other concepts, hence its S_i is low. When relevant concepts are discarded recall will be hurt, because objects with these concepts will not be retrieved if the user request is framed in terms of these concepts. For example, the annotated text for an audio object is: "*Flyers* fall to *Leafs*. *Eric* scored two goals and the *Leafs* staved off *Flyers'* third-period rally to hang on for a 4-2 victory Wednesday night over the *Philadelphia Flyers*." The concepts "Desjardins Eric," "Lindros Eric", "Philadelphia Flyers", and "Toronto Maple Leafs" are selected. The propagated-scores S_i for these concepts are 1.5, 0.8333, 1.5, and 0.833 respectively. The interrelationships between player "Desjardins Eric," and team "Philadelphia Flyers" and player "Lindros Eric" and team "Toronto Maple Leafs" are Part-Of. If γ=0.6 is chosen as a threshold-constant, among two ambiguous concepts "Lindros Eric" (0.8333 < 1.5*0.6) will be thrown away, and "Desjardins Eric," will be kept. In other words, the relevant concept, "Lindros Eric" will be discarded.

Second, note that if there is no correlation, the algorithm fails to resolve ambiguity. In that case, we keep all the selected concepts. For example, the annotated text for an audio object is: "*Young Tiger* hurlers hoping balance offense." Major league baseball's team "Detroit Tigers" and players "Tiger Dmitri" and "Tiger Eric" are selected. The S_i

scores for these concepts are 0.5. Due to a lack of correlations, we cannot throw away irrelevant concepts "Tiger Dmitri" and "Tiger Eric."

Furthermore, due to the incompleteness of the ontology, some irrelevant concepts may be associated with audio objects. For example, the annotated text for an audio object is:

> Team Up exciting part of *NBA* All-Star weekend for commissioner. *NBA* commissioner *David Stern* believes that Team Up, a program that encourages young people to volunteer their time to the community, is the most exciting part of the All-Star weekend. Former players Bob Lanier and *Michael* Cooper agree, and say the program is about making a difference in people's lives.

Among the concepts selected, *NBA* players "Cage Michael," "Curry Michael," "David Kornel," "Dickerson Michael," "Robinson David," "Wingate David," are wrongly selected because our ontology does not contain knowledge about the NBA commissioner.

Third, one important observation is that when a keyword selects one concept we assume that it is unambiguous, although this unambiguous concept may have a low score as a result of not being correlated with other concepts. In the Figure 5.1, as a case in point, the concept "New Jersey Nets" has S_i=0.5. Further, some of these concepts may not be relevant to audio objects. If the annotated text for an audio object is:

> *Titans* coaches bring game plan to Atlanta. The *Tennessee Titans* fight through the cold of Atlanta and the absence of a bye week to prepare for the SuperBowl against the Rams Sunday. *Titans* quarterback *Steve McNair* believes that the cold *weather* might actually help his turf toe.

Besides, concepts "Mcnair Steve" and "Tennessee Titans," player "Weathers Andre," a concept which is not relevant, is also selected.

Finally, it may be possible that among ambiguous concepts one will simply subsume the other. For example, the annotated text for an audio object is: "*Caps Oates* scores

300th goal; beat Islanders. *Adam Oates* scores his 300th career goal with 5:01 left Monday night, giving the *Washington Capitals* a 3-2 victory over the *New York Islanders*." Concepts "Oates Adam," "Washington Capitals," "York Mike," and "York Rangers" are selected. Note that "York Mike," and "York Rangers" are ambiguous concepts, and the interrelationship between "York Mike" and "York Rangers" is Part-Of. In that case we discard concept, "York Mike." This is because for this audio object, one team "Washington Capitals" has already selected. Most probably the object conveys information about one team's performance over the other. It is important to note that if concept "York Mike" is selected with higher S_i, we keep this concept.

Therefore, disambiguation fails to disambiguate concepts when there is little or no context among the concepts selected. This is an extremely rare occurrence (see Section 7.3.1). In a case in which it does occur, we keep all selected concepts; where some of them are relevant and some are irrelevant. However, whenever some context is available in almost cases disambiguation discards the irrelevant concepts which have been associated with audio objects. In only a few cases are relevant concepts also discarded.

This way we guarantee that precision will not be hurt in the course of the extraction of concepts from keywords in the phase of document representation. This will contribute a gain over keyword-based search on one side of the coin (i.e., document representation) with the other side of the coin being the querying mechanism (see Chapter 6).

5.1.4 Further Refinements

Besides regions, several upper level concepts of the ontology can be selected. If no region is selected, or a tie for the greatest number of concepts selected occurs between regions, we rely on these upper level concepts in metadata acquisition. Otherwise, we ignore them. This is because if a query comes in terms of an upper level concept that concept will be expanded in terms of more specific concepts by traversing the ontology (see Section 6.2 for more details). In a case in which no region is selected, an audio object will simply be associated with these concepts. In case of a tie, we will strive to associate each of these concepts in the selected regions with other upper level selected concepts. Further, if selected upper level concepts display disjoint properties, these can be used to disambiguate. For example, "Basketball," "Football," "Soccer," and "Hockey," are disjoint concepts. If an upper-level concept is selected which is associated with a selected concept in a particular region, we keep this region and throw out the others. However, it may happen that several regions which are in a tie, with the same number of selected concepts, may both be associated with one selected upper level concept. In that case we cannot resolve the tie.

To illustrate further, if a tie occurs between the regions "NBA", and "NFL", and the upper level concept, "Basketball" is selected, we keep "NBA" and throw out "NFL." This is because the interrelationship between "NBA" and "Basketball" is Part-Of. On the other hand, "NFL" is associated with "Football" which is a concept of a disjoint type in relation to the concept, "Basketball."

5.2 Management of Metadata

Effective management of metadata facilitates efficient storing and retrieval of audio information. To this end, in our model most specific concepts are considered to be metadata.

Several concepts of an ontology, for example, can become candidates for becoming the metadata of an audio object, regardless of the metadata acquisition technique employed. However, some of these may be the children of others. Two alternative approaches can be used to address this problem.

First, we can simply store the most general concepts. But we may get many irrelevant objects, and precision will be hurt for queries related to specific concepts. For example, an audio object becomes the candidate for the concepts "NHL," "Hockey," and "Professional." We might simply store the general concept, "Professional" for this object. When a user request comes in terms of a specific concept, "NHL," this object will be retrieved along with other irrelevant objects that do not belong to NHL (say, NFL, CFL, and so on). Therefore, precision will be hurt.

Second, the most specific concepts can be stored in the database. Corresponding generalized concepts can then be discarded. In this case, recall will be hurt. Suppose, as before, an audio object becomes the candidate for the concepts "NHL", "Hockey", and "Professional." During the concept selection and disambiguation process the object might be annotated with the most specific concept, "NHL." In this case, the metadata of the audio objects stored in the database will be comprised of the most specific concepts. If a query contains the terms "hockey" or "professional," this object will not be retrieved.

We follow the latter approach. By storing specific concepts as metadata, rather than generalized concepts of the ontology, we can expect to achieve the effective management of metadata. In order to avoid problems of recall, user requests are first passed through ontology on the fly and expressed in terms of the most specific concepts. In other words, ontology virtually resides on top of the database (see Figure 5.4). Even so, the audio object containing the concepts "NHL," "Hockey," and "Professional," in the above example, can still be retrieved through querying the system by any of the three terms, "NHL", "Hockey", or "Professional."

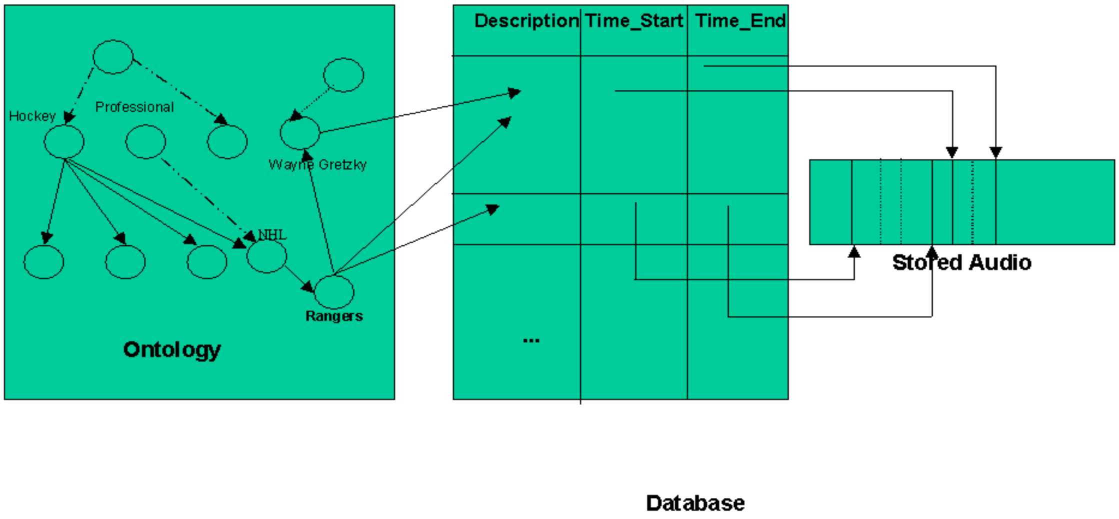


Figure 5.4 Big Picture of Our Ontology-based Model

Here, we consider an efficient way of storing audio objects in the database: We maintain a single copy of all the audio data in the database (see Figure 5.4). Further, each object's metadata are stored in the database. Thus, this start time, and end time of an object point to a fraction of all the audio data. Therefore, when the object is selected, this boundary information provides relevant audio data that are to be fetched from all the audio data and played by a scheduler routine. The following self-explanatory schemas are used to store audio objects in the database: *Audio_News (Id, Time_Start, Time_End, ...),* and *Meta_News (Id, Label)*. Each audio object's start time, end time and description correspond to Time_Start, Time_End, and Label respectively. Thus, each object's description is stored as a set of rows or tuples in the Meta_News table for normalization purpose [104].

Chapter 6 Query Mechanisms

A mechanism for the selection of information filters unwanted information. This chapter begins with a discussion of the initial stage of filtering to resolve ambiguity in concept selection in response to user requests. Second, we present a technique for SQL generation from selected concepts, along with mechanisms to expand query. Third, we present different types of queries. Fourth, we present various techniques for optimizing SQL queries. Finally, we present a technique to narrow down search.

6.1 Concept Selection and Disambiguation from Users Requests

We now focus specifically on our techniques for utilizing an ontology-based model for processing information selection requests [53]. In our model the structure of ontology facilitates indexing. In other words, ontology provides index terms/concepts which can be used to match with user requests. Furthermore, the generation of a database query takes place after the keywords in the user request are matched to concepts in the ontology.

We assume that user requests are expressed in plain English. Tokens are generated from the text of the user's request after stemming and removing stop words. Using a list of synonyms these tokens are associated with concepts in the ontology through Depth First Search (DFS) or Breadth First Search (BFS). Each of these selected concepts is called a *QConcept*. Among QConcepts, some might be ambiguous. However, through the application of a pruning technique that will be discussed in Section 6.1.1 only relevant concepts are retained. These relevant concepts will then be expanded, and will participate in SQL query generation as is discussed in Section 6.2.

6.1.1 Pruning

Disambiguation is needed when a given keyword matches more than one concept. In other words, multiple ambiguous concepts will have been selected for a particular keyword. For disambiguation, it is necessary to determine the correlation between selected concepts, discussed in Section 5.1. When concepts are correlated, the scores of concepts strongly associated with each other will be given greater weight based on their minimal distance from each other in the ontology and their own matching scores based on the number of words they match. Thus, ambiguous concepts which correlate with each other will have a higher score, and a greater probability of being retained, given a particular threshold score, than ambiguous concepts which are not correlated.

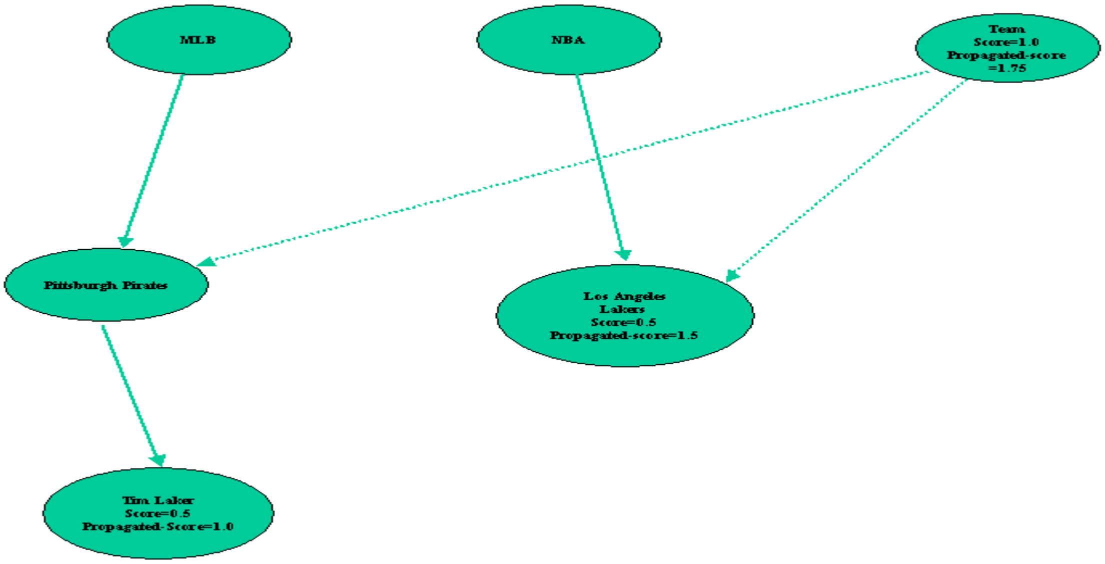


Figure 6.1 Impacts of Semantic Distance on Propagated-scores

For example, if a query is specified by "Please tell me about team Lakers," QConcepts "Team," "Los Angeles Lakers," and major baseball player, "Tim Laker" (of team "Pittsburgh Pirates") are selected (see Figure 6.1). Note that selected concepts, "Los Angeles Lakers," and "Tim Laker" are ambiguous. However, "Los Angeles Lakers" is associated with selected QConcept, "Team" due to Instance-Of interrelationship. Therefore, we prune the non-correlated ambiguous concept, player "Tim Laker." The above idea is implemented using score-based techniques. Now, we would like to present our concept-pruning algorithm for use with user requests. It is important to note that in this algorithm we borrow the definitions from Section 5.1.2 in the context of QConcept (*QC*) rather than concept (*C*).

The pseudo code for the disambiguation algorithm is as follows:

```
$QC_1, QC_2, \ldots, QC_l, \ldots, QC_r$ are selected with concept-score $Score_1, \ldots, Score_l, \ldots Score_r$
Determine correlation of selected concepts $(C_i, C_j, C_{j+1}, .., C_n)$ and update their Propagated-scores using
        $S_i = Score_i + \sum_{k=j}^{k=n} \frac{Score_k}{SD(QC_i, QC_k)}$
        $= Score_i + \frac{Score_j}{SD(QC_i, QC_j)} + \frac{Score_{j+1}}{SD(QC_i, QC_{j+1})} + \ldots + \frac{Score_n}{SD(QC_i, QC_n)}$
Sort all QConcepts ($QC_i$) based on $S_i$ in descending order
//Find Ambiguous QConcepts and prune some of them which have low
//Propagated-score
For a keyword that associated with ambiguous QConcepts, $QC_i, QC_j, QC_{l,} \ldots$
where $S_i > S_j > S_l, \ldots$
    Keep only $QC_i$ and discard $QC_j, QC_l, ..$
//End of For Loop for a keyword.
Keep all specific QConcepts and discard corresponding generalized concepts
For each QConcept that are not pruned
        Query_Expansion_SQL_Generation (QConcept) //see Figure 6.4
//End of For loop each QConcept
```

Figure 6.2 Pseudo Code for Pruning Algorithm

Using pruning algorithm (see Figure. 6.2), for a user request, “Team Lakers,” at the beginning selected QConcepts are “Team”, “Los Angeles Lakers” and “Tim Laker” (see Figure 6.1). Note that ambiguous concepts are “Los Angeles Lakers,” and “Tim Laker.” In Figure 6.1 the SD between concepts, "Team," and "Los Angeles Lakers" is 1 while the SD between concepts, "Team" and "Tim Laker" is 2. Furthermore, the Scores for concepts, "Team," "Los Angeles Lakers," and "Tim Laker" are 1.0, 0.5, 0.5 respectively. It is important to note that when two concepts are correlated with each other where semantic distance is greater than one, they will have a lower Propagated-scores, S_i and S_j compared to concepts with the same concept-scores and a semantic distance of 1. This is because for the higher semantic distance concepts are correlated in a broader sense. Thus, concepts which are correlated have a higher S_i in comparison with non-correlated concepts. Now, the Propagated-score for these concepts becomes 1.75 (1.0+0.5/1+0.5/2), 1.5 (0.5+1.0/1), and 1.0 (0.5+1.0/2) respectively (see Figure 6.1). Therefore, we keep the concept "Los Angeles Lakers" from among these ambiguous concepts and prune the other. Thus, the SD helps us to discriminate between ambiguous concepts.

It is important to note that in this pruning algorithm, unlike the disambiguation algorithm in metadata acquisition, QConcepts may be selected from more than one region. Intuitively, a user request may contain references to more than one league, such as NBA, NHL, and so on.

Among selected concepts, one concept may subsume the other concept. In this case, we use specific concept for SQL generation. For example, if a user request is expressed in terms of "Please tell me about Lakers' Bryant," the QConcepts, team "Los Angeles

Lakers," players, "Bryant Kobe", "Bryant Mark," "Reeves Bryant," are selected. Their concept-scores are 0.5, 0.5, 0.5 respectively. The latter three are ambiguous concepts. However, among these selected concepts, only "Bryant Kobe," and "Los Angeles Lakers" are correlated with a semantic distance of 1. Therefore, their propagated-scores S_i are high as compared to other concepts, in this case, 1.0, 1.0, 0.5, 0.5 respectively (see Figure 6.3). Consequently, we throw away "Bryant Reeves" and "Bryant Mark." Furthermore, "Bryant Kobe" is a sub-concept of "Los Angels Lakers," due to a Part-Of interrelationship. In this case, we keep the more specific concept, "Bryant Kobe," and the SQL generation algorithm will be called (see Figure 6.4).

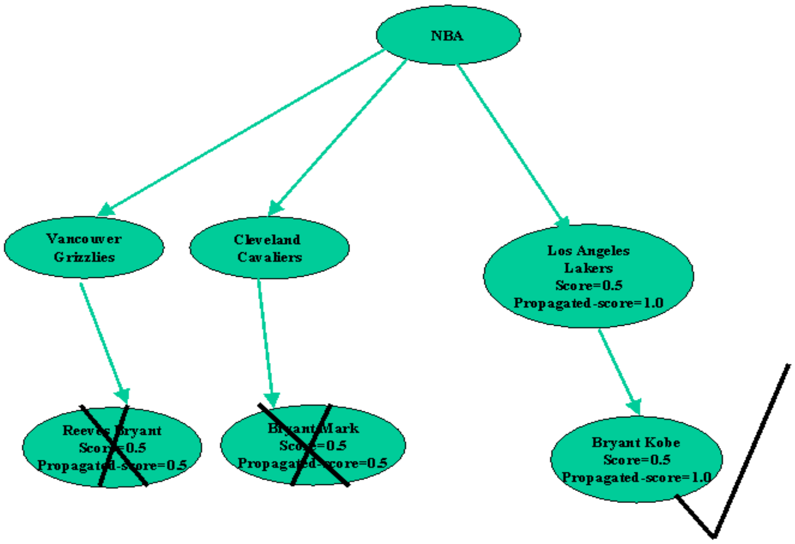

NBA
Vancouwer Grizzlies
Cleveland Cavaliers
Los Angeles Lakers Score=0.5 Propagated-score=1.0
Reeves Bryant Score=0.5 Propagated-score=0.5
Bryant Mark Score=0.5 Propagated-score=0.5
Bryant Kobe Score=0.5 Propagated-score=1.0

Figure 6.3 Illustration of Propagated-scores of QConcepts

Therefore, we guarantee that precision will not be hurt, since only the most appropriate relevant concepts associated with user requests will be identified and allowed to participate in the phase of query generation. This will contribute a gain over keyword-based search in the important area of the query mechanism.

6.2 Query Expansion and SQL Query Generation

We now discuss a technique for query expansion and SQL query [82] generation. In response to a user request for the generation of an SQL query, we follow a Boolean retrieval model. We now consider how each QConcept is mapped into the "where" clause of an SQL query. Note that by setting the QConcept as a Boolean condition in the "where" clause, we are able to retrieve relevant audio objects. First, we check whether or not the QConcept is of the NPC type. Recall that NPC concepts can be expressed exhaustively as a collection of more specific concepts. If the QConcept is a NPC concept, it will not be added in the "where" clause. On the other hand, it will be added into the “where” clause. Likewise, if the concept is leaf node, no further progress will be made for this concept. However it is non-leaf node, its children concepts are generated using DFS/BFS, and this technique is applied for each children concept. One important observation is that all concepts appearing in an SQL query for a particular QConcept are expressed in disjunctive form. Furthermore, during the query expansion phase only correct concepts are added which will guarantee that addition of new terms will not hurt precision. This will contribute further to the gain in precision over keyword search. The complete algorithm is shown in Figure 6.4.

```
Query_Expansion_SQL_Generation (QC_i)
    Mark QC_i is already visited
        If QC_i  is not NPC Type
            Add label of QC_i into where clause of SQL as disjunctive form
    //regardless of NPC type concept
    If QC_i  is not leaf node and not visited yet
            For each children concept, QCh_l  of  QC_i using DFS/BFS
                    Query_Expansion_SQL_Genertaion (QCh_l )
```

Figure 6.4 Pseudo Code for SQL Generation

The following example illustrates the above process. Suppose the user request is "Please give me news about player Kobe Bryant." "Bryant Kobe" turns out to be the QConcept which is itself a leaf concept. Hence, the SQL query (for schema see Section 5.2) generated by using only "Bryant Kobe" (with the label "NBAPlayer9") is:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_News m
WHERE a.Id=m.Id
AND Label="NBAPlayer9"
```

Let us now consider the user request, "Tell me about Los Angeles Lakers." Note that the concept "Los Angeles Lakers" is not of the NPC type, so its label ("NBATeam11") will be added in the "where" clause of the SQL query. Further, this concept has several children concepts ("Bryant Kobe," "Celestand John," "Horry Robert," . . ., i.e., names of players for this team). Note that these player concepts' labels are "NBAPlayer9," "NBAPlayer10," and "NBAPlayer11," respectively. In SQL query:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_news m
WHERE a.Id = m.Id
AND (Label="NBATeam11”
OR Label="NBAPlayer9"
OR Label="NBAPlayer10"
OR Label= . . .)
```

6.2.1 Remedy of Explosion of Boolean Condition in the Where Clause

Since most specific concepts are used as metadata and our ontologies are large in the case of querying upper level concepts, every relevant child concept will be mapped into the "where" clause of the SQL query and expressed as a disjunctive form. To avoid the explosion of Boolean conditions in this clause of the SQL query, the labels for the player and team concepts are chosen in an intelligent way. These labels begin with the label of the league in which the concepts belong. For example, team "Los Angeles Lakers" and player "Bryant Kobe" are under "NBA." Thus, the labels for these two concepts are "NBATeam11" and "NBAPlayer9" respectively, whereas the label for the concept "NBA" is "NBA."

Now, when user requests come in terms of an upper level concept (e.g., "Please tell me about NBA.") the SQL query generation mechanism will take advantage of prefixing:

```
SELECT Time_Start, Time_End
FROM Audio_News  a, Meta_News m
WHERE a.Id=m.Id
AND Label Like “%NBA%”
```

On the other hand, if we do not take advantage of prefixing, the concept NBA will be expanded into all its teams (28), and let us assume each team has 14 players. Therefore, we need to maintain 421 (1+ 28 + 28 *14) Boolean conditions in the where clause of SQL query. This explosion will be exemplified by upper level concept like basketball.

6.3 Different Types of SQL Queries Generation

In this section, we discuss different types of queries and the SQL query generation mechanism used in for each type.

6.3.1 Disjunctive Queries

If several QConcepts are selected, they are expressed in disjunctive form unless the user explicitly states that this is a conjunctive or difference query through advanced operators (see Section 7.2.1). Moreover, all sub-concepts of a particular QConcept are expressed in a pure disjunctive form. For example, the user requests "Give me news related to Bryant Kobe and Mike Tyson." Two QConcepts, "Bryant Kobe" (label"NBAPlayer9"), and "Mike Tyson" (label "BoxingPlayer03") are selected from the ontology and added to the where clause connected via "or." In SQL:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_news m
WHERE a.Id = m.Id AND
          (Label="NBAPlayer9" OR Label="BoxingPlayer3")
```

6.3.2 Conjunctive Queries

In a conjunctive query, a user request is associated with a set of QConcepts where all these QConcepts must be satisfied. Thus, selected objects should be relevant to all QConcepts simultaneously. Therefore, all QConcepts can simply be expressed as the Boolean "and" condition in the "where" clause of SQL query. Without loss of generality, we assume that a user request is associated with two QConcepts. The basic idea for SQL query generation is as follows: First, we select an object which is associated with the first QConcept, and then check whether this object fulfills the second QConcept using a self-join of relation, MetaNews.

For example, the user requests "Please tell me about the game between the Los Angeles Lakers and the Portland Trail Blazers." Two QConcepts "Los Angeles Lakers" and "Portland Trail Blazers" (label “NBATeam21”) are selected. MetaNews relation will be joined twice to fulfill these two QConcepts. The SQL query is:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_news m1, Meta_news m2
WHERE a.Id = m1.Id AND a.Id=m2.Id
AND m1.Label="NBATeam11"
AND m2.Label="NBATeam21"
```

6.3.3 Difference Queries

A user may receive a large number of audio objects for a particular QConcept. This is very likely not what the user wants. A difference query, specified by two QConcepts, restricts the universe of retrievable objects by satisfying the first QConcept but not the second. The user will then receive a more limited and desirable number of audio objects. Without loss of generality, one QConcept expresses user interests and the other expresses user disinterests. During SQL generation, initially all objects that bear the first QConcept are retrieved. Then, among these selected objects, those that do not bear the second QConcept are kept. This strategy is written in SQL query using “NOT IN.” For example, the user requests "Please tell me about player Tiger Woods, did not include information related to player Duval. Two QConcepts, player "Tiger Woods" (label PGAPlayer1) and player, "Duval" (label PGAPlayer7) are selected. The SQL query is:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_News m1
WHERE a.Id =m1.Id
    AND m1.Label = “PGAPlayer1”
    AND m1.Id NOT IN
                    (SELECT m2.Id
                    FROM Meta_News m2
                    WHEREm2.Label=“PGAPlayer7”)
```

6.4 Query Optimizations

The basic idea of query optimizations is to rewrite the database SQL query effectively in order to leverage the work of a traditional query optimizer [47].

6.4.1 Qualified Disjunctive Form (QDF)

When one concept qualifies another concept (e.g., professional football), the straightforward approach is to treat this as a conjunctive query. Then, we may generate the query by simply writing two QConcepts as a Boolean "and" in the "where" clause. However, further optimization is possible by taking the intersection of all children concepts of the two QConcepts. Hence, we consider only concepts which intersect for these QConcepts. By discarding the non-intersecting concepts, the number of Boolean conditions in the "where" clause for these

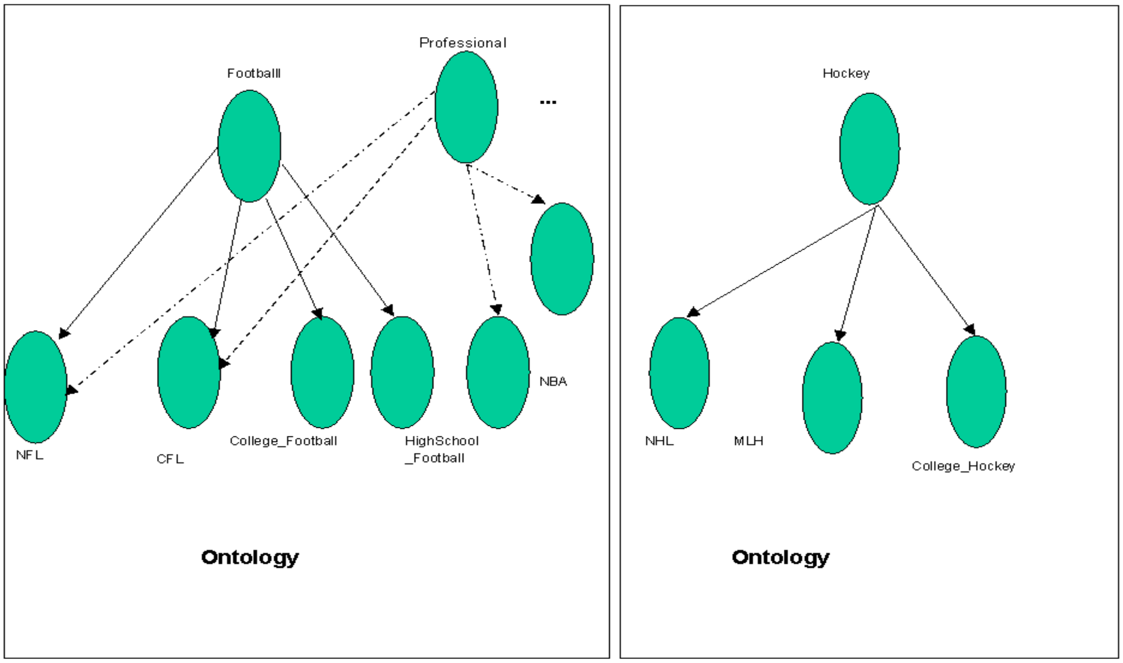


Figure 6.5 Several Concepts of Ontologies to Demonstrate Optimizations

QConcepts is reduced. However, the traditional query optimizer eliminates redundant children concepts by means of transforming a redundant expression into an equivalent one using Boolean algebra [44]. By employing this technique we leverage the work of a traditional query optimizer. Note that here for QConcepts which qualify one another all concepts in the "where" clause are expressed in disjunctive form. The query "give me professional football news" illustrates the conversion into QDF. The intersected concepts of two QConcepts (professional and football) are "NFL," and "CFL (see Figure 6.5). The SQL query for without optimization is

```
SELECT Time_Start, Time_End
FROM Audio_News  a, Meta_News m1, Meta_News m2
WHERE a.Id=m1.Id
AND a.Id = m2.Id
AND ((m1.Label Like “%NFL%”)
    OR (m1.Label Like “%CFL%”)
    OR (m1.Label Like “%College Football%”)...)
AND ((m2.Label Like “%NFL%”)
    OR (m2.Label Like “%CFL%”)
    OR (m2.Label Like “%NBA%”))
```

The SQL for optimized form is:

```
SELECT Time_Start, Time_End
FROM Audio_News  a, Meta_News m
WHERE a.Id=m.Id
AND ((Label Like “%NFL%”)
        OR (Label Like “%CFL%”))
```

6.4.2 Optimization for Conjunctive Queries

In a conjunctive query, the fact that some concepts are disjoint enables us to eliminate certain irrelevant concepts from the "where" cause of the SQL query. These are not necessarily a case in which the presence of these concepts will reduce precision; however, the query response time will be adversely affected. For example, if the user requests "tell me about the game between the Blazers and the Lakers," three QConcepts, "Portland Trail Blazers," "Los Angeles Lakers," and "Tim Laker" (label "MLBPlayer111") are selected. The last two, with regard to each other, are ambiguous concepts, called forth in response to the keyword "Laker." Note that "Los Angeles Lakers," and "Portland Trail Blazers" are two teams in the NBA, whereas "Tim Laker" is a major league baseball player. Since there is no correlation between "Los Angeles Lakers," and "Portland Trail Blazers," we are not led to throw away "Tim Laker" through the use of our disambiguation algorithm which is discussed in Section 6.1.1. The SQL query without optimization is:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_news m1, Meta_news m2
WHERE a.Id = m1.Id AND a.Id=m2.Id
    AND (m1.Label="NBATeam11" OR m1.label="MLBPlayer111")
    AND m2.Label="NBATeam21"
```

Note that the concepts here, some of which are ambiguous, are distributed across several regions. In a conjunctive query, all QConcepts which are selected should be in the same region. Otherwise, they will return an empty set. This is because an audio object can only be associated with one region. Thus, for concepts which are ambiguous for a particular keyword, only those will be retained which appear in a region in which overall the greatest number of concepts are selected in the case of a specific query (conjunctive). For this example, the ambiguous concepts "Los Angeles Lakers," and "Tim Laker" are in the region NBA and the region MLB respectively, while the non-ambiguous concept "Portland Trail Blazers" is in the region NBA. Thus, the concept, "Tim Laker," will be thrown out. Thus ambiguous concepts in different regions are also automatically pruned. The SQL query is:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_news m1, Meta_news m2
WHERE a.Id = m1.Id AND a.Id=m2.Id
AND m1.Label="NBATeam11" AND m2.Label="NBATeam21"
```

6.4.3 Optimization for NPC Concepts

Query generation does not allow an NPC concept to appear as a Boolean condition in the "where" clause; however, it does allow the another non-NPC concept to appear in the "where" clause. The NPC concept is exhaustive, which implies that it can be expressed as a collection of children concepts. Now, for each of these children concepts, NPC types are checked. If these concepts are of the NPC type, they will not be placed in the "where" clause in the SQL query. Thus, due to the avoidance of one/more Boolean conditions, the query response time may be reduced.

6.4.4 Optimization for Difference Queries

The ontology, in particular, helps one to optimize a difference query when the second QConcept is a sub-concept of the first QConcept. Note that this optimization holds for disjoint type concepts. Rather than employing the general approach (using NOT IN), we simply discard the second QConcept from children concepts of the first QConcept before the generation of an SQL query. This is because these children concepts are disjoint. Hence no common object is shared among these concepts. Therefore, as a result of the straightforward discarding of children concepts, this disjoint property guarantees that users do not get any unwanted object. For example, the user requests "give me hockey news except college hockey," which gives the generalized form of SQL query using NOT IN:

```
SELECT Time_Start, Time_End
FROM Audio_News a, Meta_News m1
WHERE a.Id =m1.Id
AND ((Label Like “%NHL%”)
    OR (Label Like “%MLH%”)
    OR (Label Like “%CollegeHockey%”))
AND m1.Id NOT IN
                    (SELECT m2.Id
                    FROM Meta_News m2
                    WHERE m2.Label Like “%CollegeHockey%”)
```

In optimized form:

```
SELECT Time_Start, Time_End
FROM Audio_News  a, Meta_News m
WHERE a.Id=m.Id
    AND ((Label Like “%NHL%”)
    OR (Label Like “%MLH%”))
```

Note that QConcept, “Hockey” has three children concepts, “NHL,” “MinorLeague_Hockey,” and “College_Hockey.” Since QConcept “Hockey” subsumes QConcept “College_Hockey,” children concepts “NHL” and “MinorLeague_Hockey” are only added to the where clause.

6.5 Narrow Down Search

Sometimes a user request may give too many results, as is the case with the broad query "football." To reduce the number of search results, a user might want to do a new search that is restricted to certain objects which have been returned by the first search query. This is called "narrowing a search" or "searching within the current search results." This new query will return a specific subset of the objects which have been returned by the original "too-broad" query. In a "too-broad" query case, a large set of objects is retrieved at a high level of generality. The user will then select a subset of objects that will be used to formulate a "narrowing a search" query. The main idea behind such a query is to identify concepts attached to selected objects that have been identified as relevant by the user, and to then enhance the importance of these concepts in a new query formulation. The expected effect is that the new query will be moved toward relevant objects and away from non-relevant ones. This approach is similar to relevance feedback in IR [76], and has the following advantages over other query reformulation strategies.

- It shields the user from the details of the query reformulation process because all the user has to provide is a relevance judgment about the relevance of objects returned in the original query.
- It breaks down the whole searching task into a sequence of small steps which are easy to grasp.
- It provides a controlled process designed to emphasize some concepts (relevant ones) and to discard/de-emphasize other non-relevant ones.

The basic method for narrowing a search is as follows. The concepts attached to selected objects are identified. From there, emphasized concepts are identified by finding common concepts among these selected objects. The following three rules are applied:

I) We take the intersection of concepts attached to selected objects without any query expansion

II) If (I) results in an empty set then we expand all concepts attached to selected objects to sub-concepts. Then we take the intersection of these sub-concepts.

III) If (II) results in an empty set then we simply take the union of concepts attached to the selected objects.

After taking either an intersection or a union, these concepts will be used for SQL generation. Furthermore, narrowing down a search always works on the most recent result set.

For example, an initial user query is "basketball." Two objects are selected from this "too-broad" query result set which are associated with the concepts "Los Angeles Lakers," for the first object, and "Bryant Kobe" for the second. After taking the intersection of these concepts using step II we notice that the intersected concept is "Bryant Kobe" and the SQL generation mechanism is called. This is because "Bryant Kobe" is associated with "Los Angeles Lakers" through a Part-Of interrelationship. On the other hand, let us assume that user selected objects are associated with the concepts "Los Angeles Lakers" for one object, and "Portland Trail Blazers" for another. For this, I and II will result in an empty set. However, III will be used and the SQL generation will be called using "Los Angeles Lakers" and "Portland Trail Blazers."

Chapter 7 Experiments

In discussing implementation, we first describe our experimental setup and user interface. Next, we present some results connected with the use of our disambiguation algorithm and our inquiry into the power of ontologies over keyword based search techniques.

7.1 Experimental Setup

We have constructed an experimental prototype system which is based upon a client server architecture. The server (a SUN Sparc Ultra 2 model with 188 MBytes of main memory) has an Informix Universal Server (IUS) [41], which is an object relational database system [15].

For the sample audio content we use CNN broadcast sports audio [18] and Fox Sports [22]. We have written a hunter program in Java that goes to these web sites and downloads all audio and video clips with closed captions. The average size of the closed captions for each clip is 25 words, after removing stop words. These associated closed captions are used to hook with the ontology. As of today, our database has 2,481 audio clips. The usual duration of a clip is not more than 5 minutes in length. Wav and ram are used for media format [74, 101].

Currently, our working ontology has around 7,000 concepts for the sports domain (see Table 2). For fast retrieval, we load the upper level concepts of the ontology in main memory, while leaf concepts are retrieved on a demand basis. Hashing is also used to increase the speed of retrieval.

Table 2 Parameters Used for Experimental Results

Media support	Wav and Ram
Total # of clips	2,481
Maximum length of clip	5 min
Average size of closed caption for a clip after removing stop words	25 words
Total # of concepts in ontologies	7,000
Average # of concepts associated with an object	4.47

Our prototype system has five components: database, metadata generator, selector, player, and user interface. The selection of concepts and the use of the disambiguation algorithm takes place in the metadata generator module, where the algorithm chooses appropriate concepts for audio clips from their closed captions (shown by 1 in Figure 7.1; discussed in Section 5.1). Only the concepts, along with their URLs as metadata, are stored in the database. The database does not contain any audio data. It contains URLs that facilitate downloading audio data on demand from the source in which it is stored. The User Interface handles user requests expressed in the form of natural language, and dispatches these to the selector (shown by 2 in Figure 7.1). The selector, using the pruning module, chooses relevant concepts and discards those which are irrelevant (discussed in Section 6.1). From the concepts selected, the selector generates database queries in SQL, using an expansion module with possible optimizations, and then submits these to the database (shown by 3 in Figure 7.1; discussed in Section 6.2). The URLs of relevant clips, with closed captions, are next displayed in the web browser, with the most recent relevant clip shown first. When the user clicks on a closed caption, the browser will invoke the real player/windows media player (shown by 4 in Figure 7.1). Note that concept selection, the disambiguation modules for metadata generation, and the pruning and query expansion modules with possible optimizations for selection are all written in Java. Furthermore, the connectivity between the database and these modules has been achieved through JDBC [45].

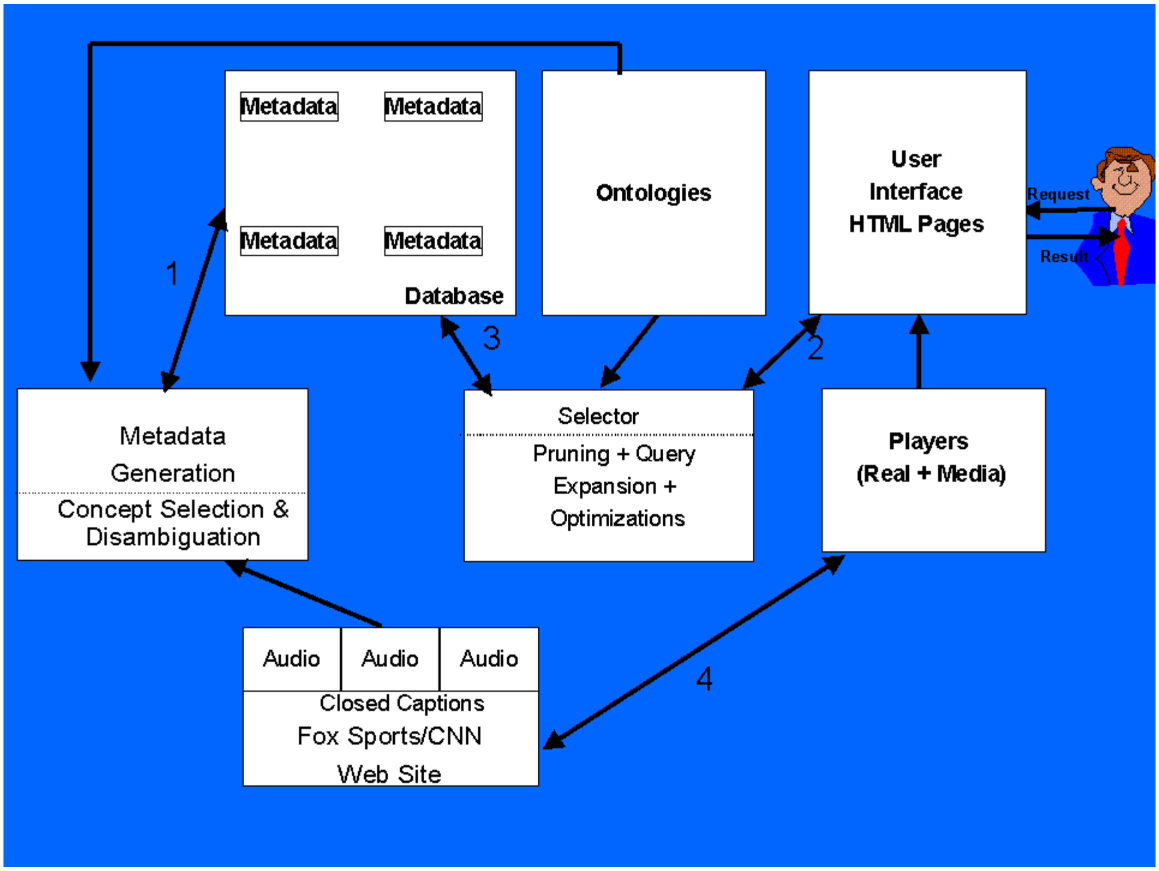


Figure 7.1 Components of our Prototype System

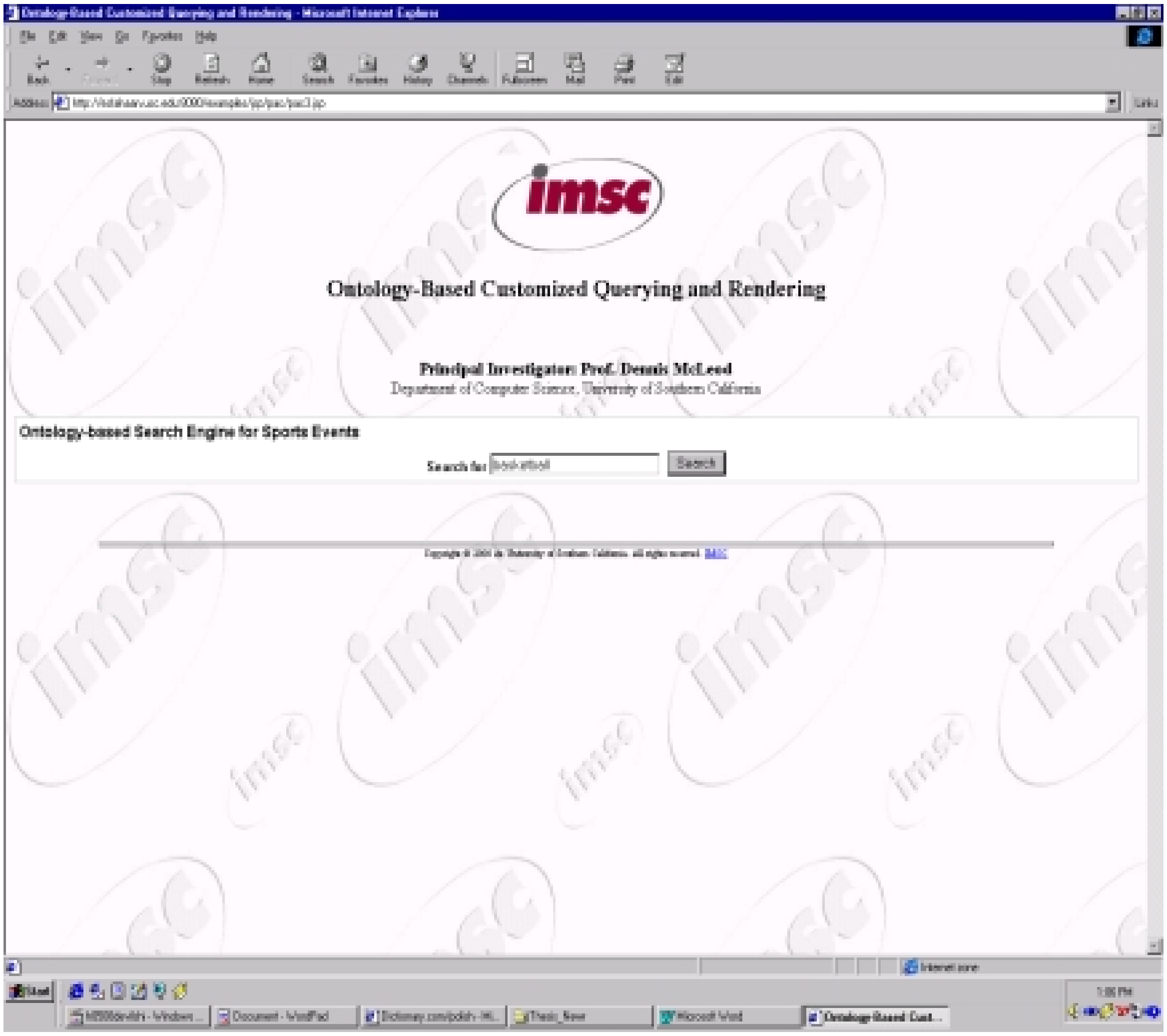
7.2 Interface

Our system is available on the web (URL: http://esfahaan.usc.edu:8080/examples/jsp/pac/pac3.jsp). To enter a query into our system with basic form, users are merely required to type in a few descriptive keywords (see Figure 7.2) and to hit "search." Our interface will then provide a set of html pages that contain the hyperlink (absolute URL) of relevant audio objects, along with closed captions (see Figure 7.3). Each html contains at most 20 audio objects. When a user clicks on a certain closed caption, a real player or windows media player will be invoked, depending on the medium used, and the clip will start to play (see Figure 7.4).

Our interface automatically expands user search using stemming, so when a user types in "boxing" the interface searches for "boxer" as well, and maybe even "box" (discussed in Section 5.1). There is some argument about whether this is good or bad, but it is a necessary reflection of the fact that our matching criterion is from concept to concept rather than keyword to keyword. Furthermore, our disambiguation algorithm always chooses the right concepts from the keywords. Therefore, additional query terms usually do not hurt precision. As discussed in Section 5.1, our Interface removes common words such as "of" and "for" from the query before it starts to search. We ignore these words for two reasons:

- Common words rarely help narrow down a search, and
- Common words slow down searching significantly.

It is important to note that our basic interface will return objects that contain only some of the query terms, not necessarily all of them. For example, a user types "Los Angeles Lakers or Portland Trail Blazers." Our interface will return objects that talk about either "Los Angeles Lakers," or "Portland Trail Blazers," or both. However, if the user types "Los Angeles Lakers and Portland Trail Blazers," the interface will retrieve the previous result set. Thus we do not discriminate between keyword "and" as opposed to keyword "or." Furthermore, when the user types "NBA and NHL," the retrieved object should contain information about either the NBA or the NHL [29, 89]. This is because these concepts are disjoint. Since our goal is not to achieve a level of understanding comparable to that of natural language, these terms "and," and "or" are added to the list of common words. However, the user can construct conjunctive or difference (not) queries using a few advanced search operators.



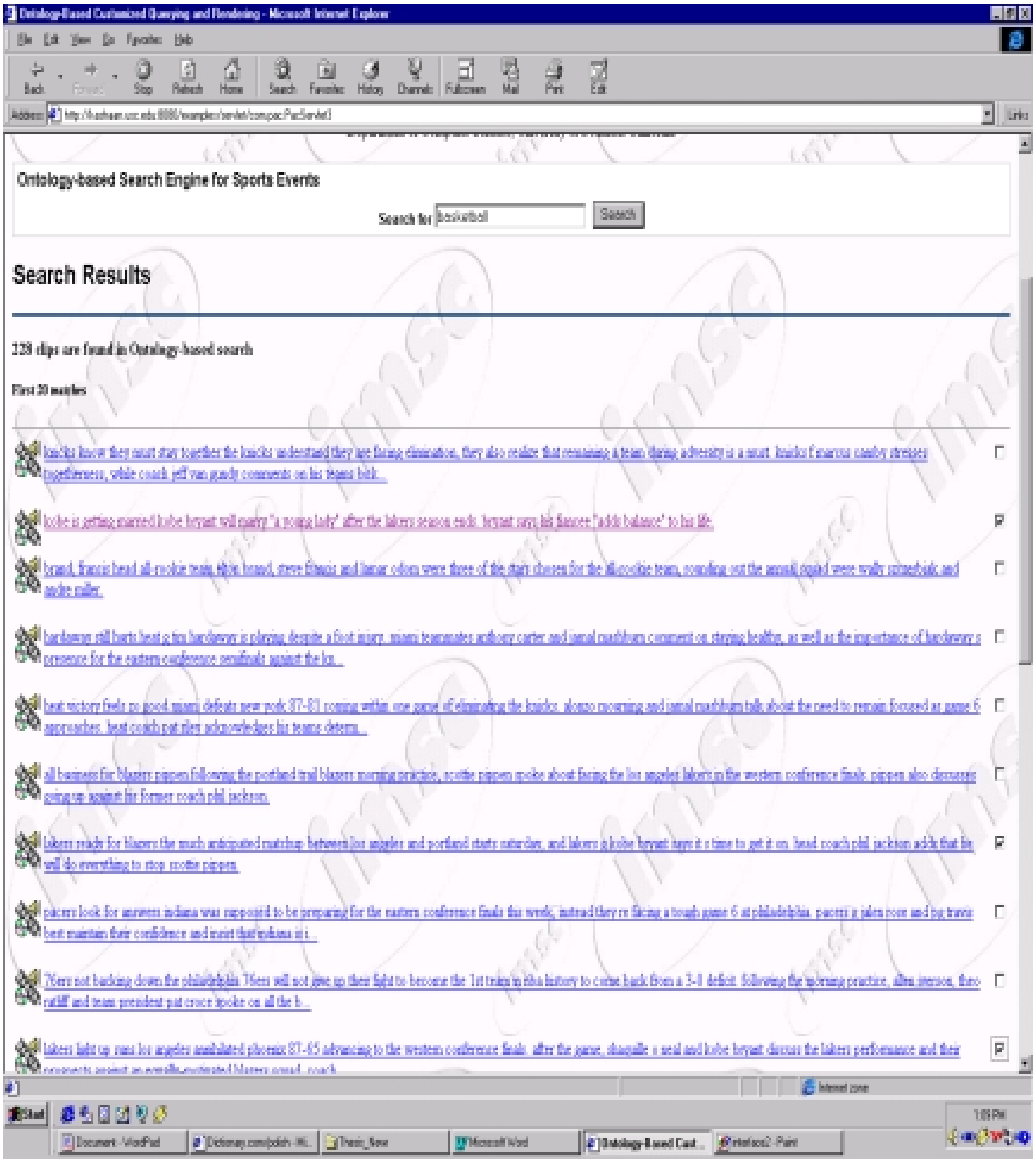
Ontology-Based Customized Querying and Rendering

Principal Investigator: Prof. Dennis McLeod
Department of Computer Science, University of Southern California

Ontology-based Search Engine for Sports Events

Search for: basketball

Figure 7.2 Basic User Interface



Ontology-based Search Engine for Sports Events

Search for basketball Search

Search Results

228 clips are found in Ontology-based search

First 20 matches

- knicks know they must stay together the knicks understand they are facing elimination, they also realize that remaining a team during adversity is a must. knicks f marcus camby stresses togetherness, while coach jeff van gundy comments on his teams bab... □
- kobe is getting married kobe bryant will marry "a young lady" after the lakers season ends. bryant says his fiancee "adds balance" to his life. ☑
- brand, francis head all-rookie team elton brand, steve francis and lamar odom were three of the stars chosen for the all-rookie team, rounding out the annual squad were wally szczerbiak and andre miller. □
- hardaway still hurts heat g tim hardaway is playing despite a foot injury. miami teammates anthony carter and jamal mashburn comment on staying healthy, as well as the importance of hardaway's presence for the eastern conference semifinals against the kn... □
- heat victory feels so good miami defeats new york 87-83 coming within one game of eliminating the knicks. alonzo mourning and jamal mashburn talk about the need to remain focused as game 6 approaches. heat coach pat riley acknowledges his teams determ... □
- all business for blazers pippen following the portland trail blazers morning practice, scottie pippen spoke about facing the los angeles lakers in the western conference finals. pippen also discusses going up against his former coach phil jackson. □
- lakers ready for blazers the much anticipated matchup between los angeles and portland starts saturday, and lakers g kobe bryant says it s time to get it on. head coach phil jackson adds that he will do everything to stop scottie pippen. ☑
- pacers look for answers indiana was supposed to be preparing for the eastern conference finals this week, instead they re facing a tough game 6 at philadelphia. pacers g jalen rose and g travis best maintain their confidence and insist that indiana is i... □
- 76ers not backing down the philadelphia 76ers will not give up their fight to become the 1st team in nba history to come back from a 3-0 deficit. following the morning practice, allen iverson, theo ratliff and team president pat croce spoke on all the b... □
- lakers light up suns los angeles annihilated phoenix 87-65 advancing to the western conference finals. after the game, shaquille o neal and kobe bryant discuss the lakers performance and their prospects against an equally-motivated blazers squad. coach ☑

Figure 7.3 User Interface with Result Sets and Check Boxes Marked for Narrow Down Search

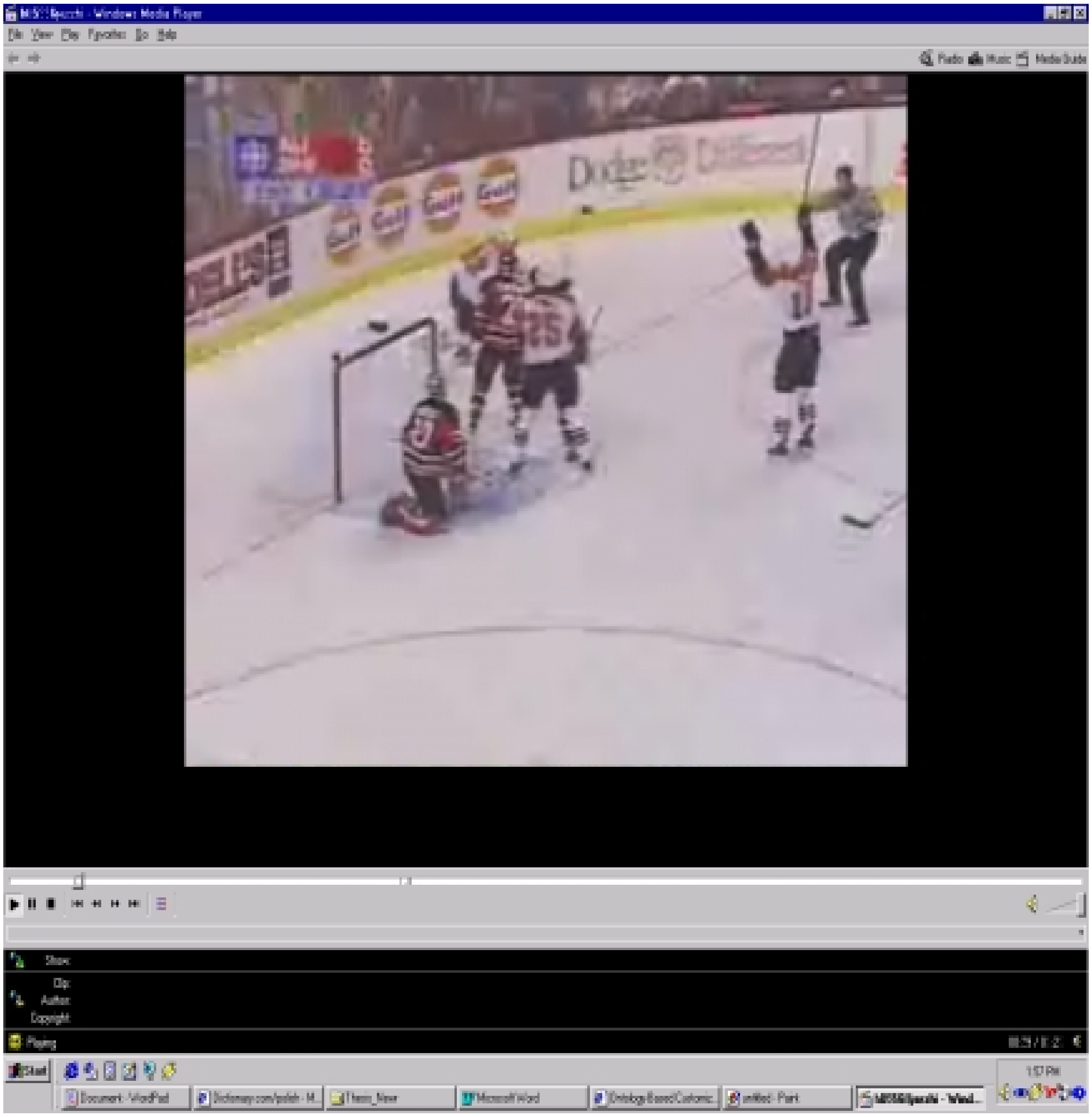


Figure 7.4 Presentation of a Clip by Windows Player

7.2.1 Advanced Search Interface

In conjunctive query the interface returns only objects that include all the query terms. The + operator enforces "and" behavior in our interface (i.e. the user types "Los Angeles Lakers + Portland Trail Blazers"). Sometimes it is helpful to choose words to be excluded from a search. That is, we want all relevant result objects except those conveying certain keywords. We support this "not" functionality with the "-" operator (i.e. the user types "hockey- college hockey." Furthermore, our interface does not discriminate between whether or not the query terms are found in close proximity.

7.2.2 Narrow Down Search Interface

Users can narrow down a search through selecting a certain number of objects by marking a check in a box as in Figure 7.3. When the user presses the "show me button" only relevant clips are displayed in the html pages, with a smaller result set.

7.3 Results

7.3.1 Effectiveness of Disambiguation Algorithm

We have completed study of the performance of our disambiguation algorithm by considering what percentage of audio objects it can successfully disambiguate. Furthermore, we studied the impact of levels of threshold values on pruning irrelevant concepts associated with audio objects while retaining those which are relevant. For study data, we ran our disambiguation algorithms over the audio clips' closed captions. We then inspected the concepts associated with various audio objects. In Figure 7.5 the X axis represents the value of threshold, γ, while the Y axis represents the percentage of instances in which objects are annotated with only correct concepts (category I), with wrong concepts (category II), and with no concept at all (category III). In category II, showing wrong concepts, some correct concepts may also be present (mixed).

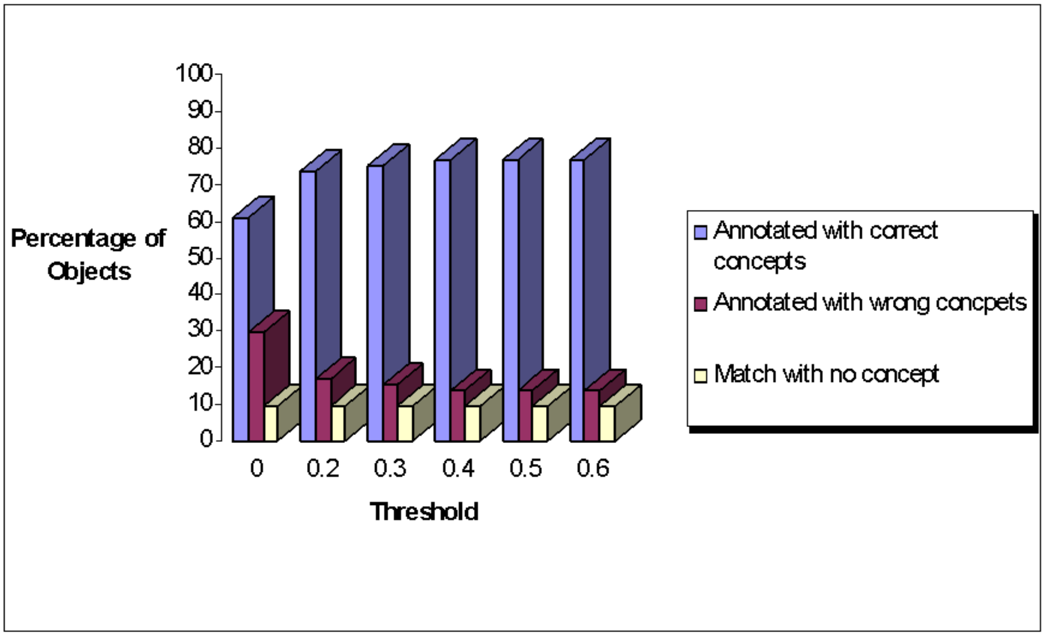


Figure 7.5 Effect of Threshold on Audio Objects' Associated Concepts

For *γ=0* when the disambiguation algorithm works among different regions, we observed that 9.5% of the objects failed to associate with any concept of the ontology (category III). This is because our ontology is incomplete. For example, an audio object includes reference to a famous hockey player whose career ended ten years ago, and who recently passed away. There is no concept for this player in our ontology, so our algorithm fails to associate a concept with this object. Thus, recall will be hurt. On the other hand, 90.5% of the objects are associated with at least some concepts of the ontology (category I & II). Among these, 60.8% objects are all associated with relevant concepts (category I). In other words, in 60.8% of the cases there is no association with an irrelevant concept. Nor in these cases, have we missed any relevant concept. 29.7% objects are associated with at least one irrelevant concept along with relevant concepts (category II). In this case, precision is hurt due to the annotation of irrelevant concepts. Note that in this case these irrelevant concepts for an audio object are distributed in several regions or a particular region.

With an increasing value of γ, the threshold-constant, ambiguous concepts will be discarded from category II. Furthermore, this threshold-constant strives to resolve ambiguity for an audio object in a particular region, rather than in several regions. Recall that an audio object might be associated with several concepts. From there the S_{max} score is calculated and ambiguous concepts whose propagated-score, S_i, falls below $S_{max} * \gamma$ are simply discarded. Note that S_{max} varies from object to object. Thus, some objects will be rid of irrelevant concepts and will now be associated with correct concepts (category I). However, as emphasized earlier, there is a chance that with the increasing value of γ, for a given audio object, we may lose a relevant concept as we shed those which are irrelevant. Thus, recall will be diminished at the expense of improving precision. For a particular γ in Figure 7.5, the first, second, and third bars represent category I, II and III respectively. Hence, with γ equal to the values 0.2, 0.3, 0.4, 0.5, and 0.6 respectively, 73.7%, 75%, 76.9%, 76.9%, and 76.9% of the objects are associated with relevant concepts (category I). Further, 16.8%, 15.5%,13.6%, 13.6% and 13.6% of the objects are associated with irrelevant concept(s), along with relevant concept(s), and/or are missing some relevant concepts that are selected a priori (as compared to γ=0). Note also that category III is independent of the increasing value of γ.

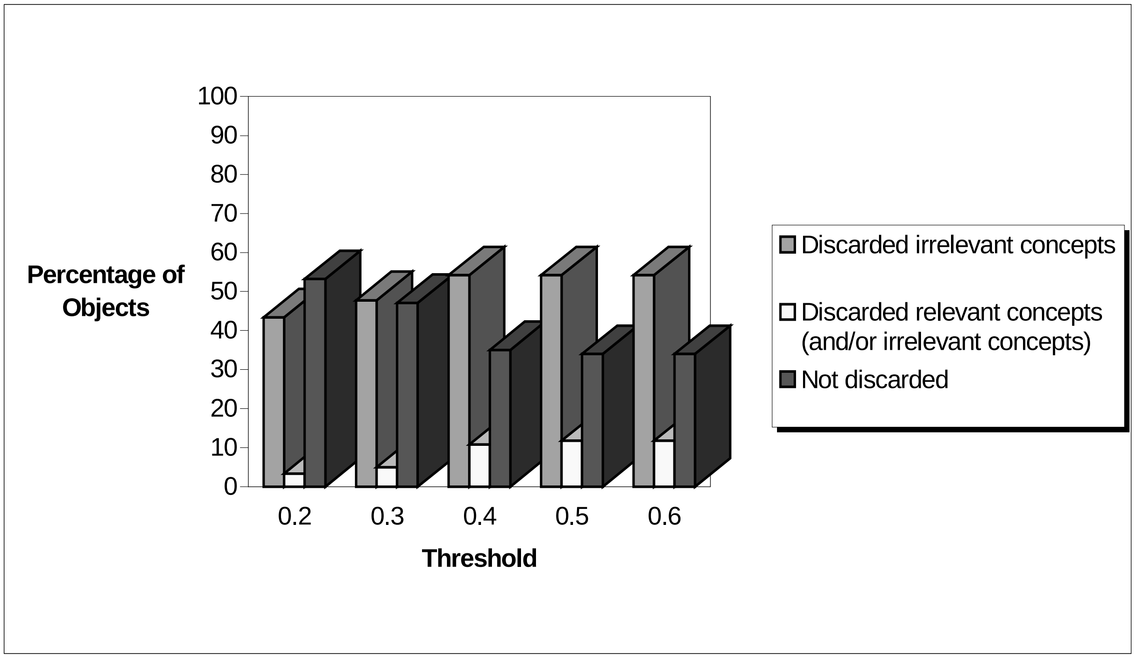


Figure 7.6 Effect of Threshold on Audio Objects' Irrelevant Concepts (Mixture)

In Figure 7.6 we show separately the results of our study of the impact of an increasing value of γ for category II. Increasing the value of γ not only leads to the discarding of irrelevant concepts from audio objects but also to the loss of relevant concepts. Here, the X axis represents the threshold value of γ, while the Y axis represents the percentage of objects in which discarded irrelevant concepts and relevant concepts occur for category II. For a particular γ, the first, second, and third bars represent the percentage of objects in which all associated irrelevant concept were discarded, the percentage of objects in which at least one relevant concept was discarded, and the percentage of objects in which no ambiguous concept was discarded out of the 29.7%

total objects of category II at γ=0. With γ=0.2, 0.3, 0.4, 0.5, and 0.6, 43.4%, 47.81%, 54.21%, 54.22% and 54.22% of the objects reflect the condition that only irrelevant concepts have been discarded, while only 3.4%, 5.05%, 10.77%, 11.78% and 11.78% of the objects reflect the condition that relevant concepts have been discarded. One important observation is that with an increasing γ, more objects discarded irrelevant concept(s) as compared to a decreasing number of objects in which correct concepts were missed. For γ=0, 60.8% objects are in category I. With γ=0.2, 0.3, 0.4, 0.5, and 0.6, out of 29.7% objects 12.89% (43.4%* 29.7%), 14.20% (47.81%* 29.7%), 16.10% (54.21%*29.7%), 16.10% (54.22%*29.7%) and 16.10% (54.22%* 29.7%) of the objects are all associated with relevant concepts respectively. These will be added to the 60.8% of the objects associated with relevant concepts at γ=0 and are in category I. Thus, with γ=0.2, 0.3, 0.4, 0.5, and 0.6, 73.69%, 75%, 76.9%, 76.9%, and 76.9% objects are in category I respectively in Figure 7.6.

Note also, with the increasing threshold-constant, γ curves of categories I, and II (in Figure 7.5) and all curves in Figure 7.6 become flat. This is because at γ=0.4 or higher the disambiguation algorithm is unable to throw any new irrelevant/relevant concepts from category II. It is important to note that in our data set the propagated-scores of non-correlated concepts do not fall into this range. Moreover, in our data set, the semantic distance of most of the correlated selected concepts is 1. After the propagation of scores among these concepts their propagated-scores are equal, and they participate in the selection based on the largest scores principle. On the other hand, the propagated-scores of non-correlated concepts are low. When we cannot disambiguate concepts due to unavailability of context (i.e., usually selected concepts' propagated-scores are equal) we simply keep all the concepts, both those which are relevant and those which are not.

Now, the question is what threshold constant should we choose? This will depend on the dataset. However, inability to achieve further progress in distinguishing relevant from irrelevant concepts dictates a limit on increasing the threshold-constant value. This is seen when categories II and III cease changing. Recall that category I is always independent of threshold-constant. With regard to categories II and III the threshold-constant governs the way in which concepts associated with an object are used as metadata to handle user requests. For example, in the above result we increased the threshold-constant 0.1 in each successive iteration. When we observe that categories II and III have become flat we simply stop and use the value 0.4, which is the maximum value in the set of threshold-constants (γ) from successive iterations. After that the process ceases and no further improvement is possible with the increasing threshold-constants (e.g., 0.5, 0.6, and so forth).

7.3.2 Theoretical Foundations of Ontology-based Model

We would like to demonstrate the power of our ontology over the keyword-based search technique. For an example of keyword-based technique we have used the most widely used model-vector space model [77, 78].

7.3.2.1 Vector Space Model

Here, queries and documents are represented by vectors. Each vector contains a set of terms or words and their weights. The similarity between a query and a document is calculated based on the inner product or cosine of two vectors' weights. The weight of each term is then calculated based on the product of term-frequency (*TF*) and inverse-document frequency (*IDF*). *TF* is calculated based on number of times a term occurs in a given document or query. *IDF* is the measurement of inter-document frequency. Terms that appear unique to a document will have high *IDF*. Thus, for *N* documents if a term appears in n documents, *IDF* for this term *=log(N/n) +1*. Let us assume query (Q_i) and document (D_j) have t terms and their associated weights are WQ_{ik} and WD_{ik} respectively for k = 1 to t. Similarity between these two is measured using the following inner product:

$$Sim(Q_i,D_j) = Cosine(Q_i,D_j)$$

$$= \frac{\sum_{k=1}^{k=t} WQ_{ik} * WD_{jk}}{\sqrt{\sum_{k=1}^{k=t} (WQ_{ik})^2 * \sum_{k=1}^{k=t} (WD_{jk})^2}}$$

The denominator is used to nullify the effect of the length of document and query and ensure that the final value is between 0 and 1.

7.3.2.2 Types of Queries

Sample queries are classified into 3 categories, with each category containing 5 queries. The first category is related to broad/general query formulation such as "tell me about basketball" which is associated with an upper level concept of the ontology. The second category is related to narrow query formulation such as "tell me about Los Angeles Lakers," which is associated with a lower level concept of the ontology. The third category is context query, in which a user specifies a certain context in order to make the query unambiguous, such as Laker's Kobe, Boxer Mike Tyson, and Team

Lakers. For example, for keyword “Lakers” in our ontology, two concepts are selected: major league baseball player “Tim Laker” who plays for the team “Pittsburgh Pirates” and NBA team “Los Angeles Lakers.” Although we collected data for fifteen queries, results are reported for only nine. The reason for this is that the nine are chosen in a way which give the worst result from an ontology-based model perspective.

7.3.2.3 Analytical Results

Unlike standard database systems, such as relational or object-oriented systems, audio objects retrieved by a search technique are not necessarily of use to the user [104]. This is due mainly to inaccuracies in the way these objects are presented, in the interpretations of the audio objects and the users' queries, and through the inability of users to express their retrieval needs precisely. A relevant object is an object of use to the user in response to his or her query, whereas an irrelevant object is one of little or no use. The effectiveness of retrieval is usually measured by the following two quantities, recall and precision [75]:

$$Recall = \frac{The\ number\ of\ relevant\ objects\ that\ are\ retrieved}{The\ number\ of\ relevant\ objects}$$

$$Precision = \frac{The\ number\ of\ relevant\ objects\ that\ are\ retrieved}{The\ number\ of\ retrieved\ objects}$$

This can be illustrated by means of a Venn diagram (see Figure 7.7) in which *Rel* represents the set of relevant objects and Ret represents the set of retrieved objects. The above measure can also be redefined in the following manner.

$$Recall = \frac{\left|Rel \cap Ret\ \right|}{\left|Rel\ \right|} \qquad \textbf{(7.1)}$$

$$Precision = \frac{|Rel \cap Ret\ |}{|Ret\ |} \qquad \textbf{(7.2)}$$

For example, if 10 documents are retrieved of which 7 are relevant, and the total number of relevant documents is 20, then recall =7/20 and precision = 7/10.

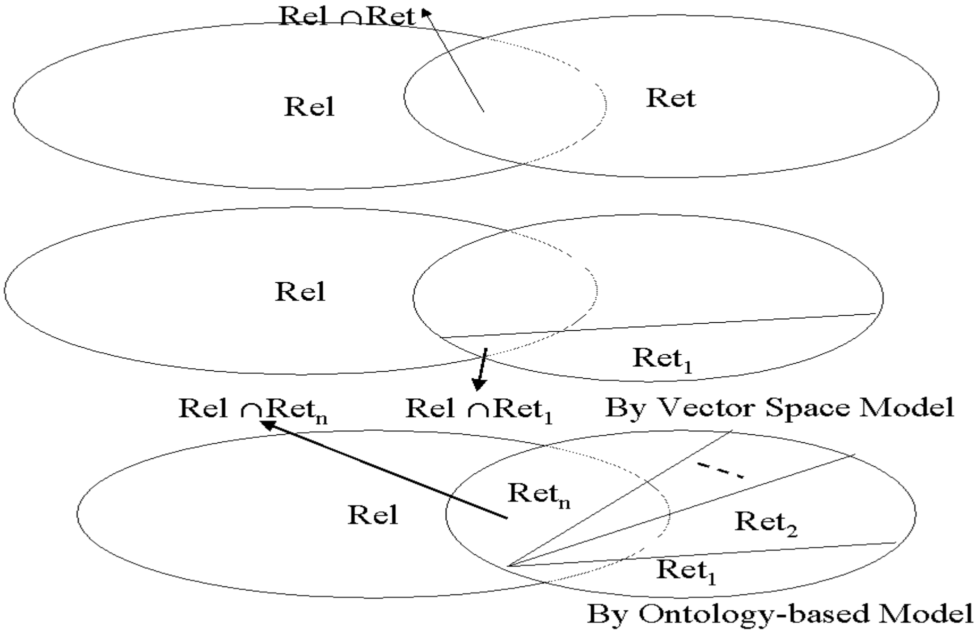


Figure 7.7 Diagram of Precision and Recall for Two Search Techniques

Therefore, recall and precision denote, respectively, completeness of retrieval and purity of retrieval. A common phenomenon is that as recall increases, precision decreases unfortunately. This means that when it is necessary to retrieve more relevant audio objects a higher percentage of irrelevant objects will usually also be retrieved

In order to evaluate the retrieval performance of two systems, we can employ an F score [88]. The F score is the harmonic mean of recall and precision, a single measure that combines recall and precision. The function ensures that an F score will have values within the interval [0, 1]. The F score is 0 when no relevant documents have been retrieved, and it is 1 when all retrieved documents are relevant. Furthermore, the harmonic mean F assumes a high value only when both precision and recall are high. Therefore, determination of the maximum value for F can be interpreted as an attempt to find the best possible compromise between recall and precision.

$$F\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7.3}$$

Let us assume that a user request is expressed by the single keyword W_1. A keyword based search technique retrieves Ret_1, the set of objects where *Rel* is the set of relevant objects (see Figure 7.7). Precision (P_k) and recall (R_k) for keyword-based search technique are defined by:

$$R_k = \frac{|Rel \cap Ret_1|}{|Rel|} \tag{7.4}$$

$$P_k = \frac{|Rel \cap Ret_1|}{|Ret_1|} \tag{7.5}$$

Now we turn to the ontology-based model. Without loss of generality we assume that keyword W_1 chooses the NPC concept C_1 in the ontology, and for this concept a given set of objects, Ret_1, is retrieved. Furthermore, this concept can be expanded to include its sub-concepts $C_2, C_3, C_4, C_5, \ldots, C_n$ for which the objects $Ret_2, Ret_3, Ret_4, Ret_5, \ldots, Ret_n$ are retrieved. Note that for purposes of simplification, we assume that concept C_1 and keyword W_1 each retrieve the same set of objects, Ret_1.

Precision (P_o) and recall (R_o) for the ontology-based search technique are defined by:

$$R_o = \frac{\sum_{i=1}^{n} \left|Rel \cap Ret_i\right|}{\left|Rel\right|}$$

$$= \frac{\left|Rel \cap Ret_1\right| + \left|Rel \cap Ret_2\right| + ... + \left|Rel \cap Ret_n\right|}{\left|Rel\right|} \qquad \textbf{(7.6)}$$

$$P_o = \frac{\sum_{i=1}^{n} \left|Rel \cap Ret_i\right|}{\sum_{i=1}^{n} \left|Ret_i\right|}$$

$$= \frac{\left|Rel \cap Ret_1\right| + \left|Rel \cap Ret_2\right| + ... + \left|Rel \cap Ret_n\right|}{\left|Ret_1\right| + \left|Ret_2\right| + ... + \left|Ret_n\right|} \qquad \textbf{(7.7)}$$

Therefore, from Equations 7.4 and 7.6 we get,

$$\frac{R_o}{R_k} = \frac{\left|Rel \cap Ret_1\right| + \left|Rel \cap Ret_2\right| + ... + \left|Rel \cap Ret_n\right|}{\left|Rel \cap Ret_1\right|} \qquad \textbf{(7.8)}$$

$$\frac{R_o}{R_k} = 1 + \frac{\left|Rel \cap Ret_2\right| + ... + \left|Rel \cap Ret_n\right|}{\left|Rel \cap Ret_1\right|} \qquad \textbf{(7.9)}$$

$$\frac{R_o}{R_k} \geq 1 \text{, since } \frac{\left|Rel \cap Ret_2\right| + ... + \left|Rel \cap Ret_n\right|}{\left|Rel \cap Ret_1\right|} \geq 0$$

Therefore, query expansion always guarantees that recall for the ontology-based model will be higher or equal to recall of keyword based technique. Note that if expanded concepts retrieve nothing or C_1 is itself a leaf concept then the recall result will be the same in either case.

Similarly, using Equations 7.5 and 7.7, we get

$$\frac{P_o}{P_k} = \frac{\dfrac{\sum_{i=1}^{n} \left| Rel \cap Ret_i \right|}{\sum_{i=1}^{n} \left| Ret_i \right|}}{\dfrac{\left| Rel \cap Ret_1 \right|}{\left| Ret_1 \right|}} \qquad \textbf{(7.10)}$$

$$= \frac{\left| Rel \cap Ret_1 \right| + \left| Rel \cap Ret_2 \right| + ... + \left| Rel \cap Ret_n \right|}{\left| Ret_1 \right| + \left| Ret_2 \right| + ... + \left| Ret_n \right|} \times \frac{\left| Ret_1 \right|}{\left| Rel \cap Ret_1 \right|} \qquad \textbf{(7.11)}$$

$$= \frac{\left| Rel \cap Ret_1 \right| + \left| Rel \cap Ret_2 \right| + ... + \left| Rel \cap Ret_n \right|}{\left| Rel \cap Ret_1 \right|} \times \frac{\left| Ret_1 \right|}{\left| Ret_1 \right| + \left| Ret_2 \right| + ... + \left| Ret_n \right|}$$

$$\frac{P_o}{P_k} = \left(1 + \frac{\left| Rel \cap Ret_2 \right| + ... + \left| Rel \cap Ret_n \right|}{\left| Rel \cap Ret_1 \right|} \right) \times \left(1 - \frac{\left| Ret_2 \right| + ... + \left| Ret_n \right|}{\left| Ret_1 \right| + ... + \left| Ret_n \right|} \right) \qquad \textbf{(7.12)}$$

The first term in Equation 7.12 is greater than or equal to 1,

i.e., $\left(1 + \dfrac{\left| Rel \cap Ret_2 \right| + ... + \left| Rel \cap Ret_n \right|}{\left| Rel \cap Ret_1 \right|} \right) \geq 1$

On the other hand, the second term in Equation 7.12 is less than or equal to 1, i.e.,

$\left(1 - \dfrac{\left| Ret_2 \right| + ... + \left| Ret_n \right|}{\left| Ret_1 \right| + ... + \left| Ret_n \right|} \right) \leq 1$

Therefore, it is not trivial problem to say which case is better.

Assume best case; each C_i returns only relevant object, so $Rel \cap Ret_i = Ret_i$ and $\left| Rel \cap Ret_i \right| = \left| Ret_i \right|$ for $\forall$i . Then using Equation 7.11,

$$\frac{P_o}{P_k} = \frac{|Rel \cap Ret_1| + |Rel \cap Ret_2| + ... + |Rel \cap Ret_n|}{|Ret_1| + |Ret_2| + ... + |Ret_n|} \times \frac{|Ret_1|}{|Rel \cap Ret_1|}$$

$$= \frac{|Ret_1| + |Ret_2| + ... + |Ret_n|}{|Ret_1| + |Ret_2| + ... + |Ret_n|} \times \frac{|Ret_1|}{|Ret_1|}$$

$$= 1 \times 1$$

$$= 1$$

Therefore, maximally, P_o=P_k.

Assume worst case; each C_i *(i >1)* returns only irrelevant concepts. Then $Rel \cap Ret_i = 0$ and $|Rel \cap Ret_i| = 0$ for $\forall \mathrm{i} > 1$. Then using Equation 7.11,

$$\frac{P_o}{P_k} = \frac{|Rel \cap Ret_1| + |Rel \cap Ret_2| + ... + |Rel \cap Ret_n|}{|Ret_1| + |Ret_2| + ... + |Ret_n|} \times \frac{|Ret_1|}{|Rel \cap Ret_1|}$$

$$= \frac{|Rel \cap Ret_1| + 0 + ... + 0}{|Ret_1| + |Ret_2| + ... + |Ret_n|} \times \frac{|Ret_1|}{|Rel \cap Ret_1|}$$

$$= \frac{|Rel \cap Ret_1|}{|Ret_1| + |Ret_2| + ... + |Ret_n|} \times \frac{|Ret_1|}{|Rel \cap Ret_1|}$$

$$= \frac{|Ret_1|}{|Ret_1| + |Ret_2| + ... + |Ret_n|} << 1$$

Therefore, at best P_0=P_k, at worst P_0<<*1*.

Using Equation 7.3, F scores for keyword and ontology-based models are as follows:

$$F\ score_k = \frac{2 \times P_k \times R_k}{P_k + R_k} \qquad \textbf{(7.13)}$$

$$F\ score_o = \frac{2 \times P_o \times R_o}{P_o + R_o} \qquad \textbf{(7.14)}$$

Therefore, from Equations 7.13 and 7.14 we get,

$$\frac{F\ score_o}{F\ score_k} = \frac{P_o \times R_o}{P_0 + R_o} \times \frac{P_k + R_k}{P_k \times R_k}$$

$$= \frac{\frac{1}{P_k} + \frac{1}{R_k}}{\frac{1}{P_o} + \frac{1}{R_o}} \qquad \mathbf{(7.15)}$$

Using Equations 7.4, 7.5, 7.6, and 7.7, we get from Equation 7.15,

$$\frac{F\ score_o}{F\ score_k} = \frac{\frac{|Ret_1|}{|Rel \cap Ret_1|} + \frac{|Rel|}{|Rel \cap Ret_1|}}{\frac{\sum_{i=1}^{n} |Ret_i|}{\sum_{i=1}^{n} |Rel \cap Ret_i|} + \frac{|Rel|}{\sum_{i=1}^{n} |Rel \cap Ret_i|}} \qquad \mathbf{(7.16)}$$

$$= \frac{\frac{|Ret_1| + |Rel|}{|Rel \cap Ret_1|}}{\frac{\sum_{i=1}^{n} |Ret_i| + |Rel|}{\sum_{i=1}^{n} |Rel \cap Ret_i|}}$$

$$\frac{F\ score_o}{F\ score_k} = \frac{|Ret_1| + |Rel|}{|Rel \cap Ret_1|} \times \frac{\sum_{i=1}^{n} |Rel \cap Ret_i|}{\sum_{i=1}^{n} |Ret_i| + |Rel|} \qquad \mathbf{(7.17)}$$

$$= \frac{\sum_{i=1}^{n} |Rel \cap Ret_i|}{|Rel \cap Ret_1|} \times \frac{|Ret_1| + |Rel|}{\sum_{i=1}^{n} |Ret_i| + |Rel|} \qquad \mathbf{(7.18)}$$

$$\Rightarrow \frac{F\ score_o}{F\ score_k} = \left(1 + \frac{|Rel \cap Ret_2| + ... + |Rel \cap Ret_n|}{|Rel \cap Ret_1|} \right) \times \frac{1}{1 + \frac{|Ret_2| + |Ret_3| + ... + |Ret_n|}{|Ret_1| + |Rel|}}$$

$$\mathbf{(7.19)}$$

The first term in Equation 7.19 is greater than or equal to 1 (i.e., $1+\frac{|Rel \cap Ret_2|+....+|Rel \cap Ret_n|}{|Rel \cap Ret_1|} \geq 1$) and the second term in Equation 7.19 is less than or equal to 1 (i.e., $\frac{1}{1+\frac{|Ret_2|+|Ret_3|+...+|Ret_n|}{|Ret_1|+|Rel|}} \leq 1$).

Therefore, we cannot say in a straightforward manner that one outperforms the other between $F\ score_k$ and $F\ score_o$. However, in special cases we can say which is better such as:

Again assume best case; each C_i returns only relevant object, so $Rel \cap Ret_i = Ret_i$ and $|Rel \cap Ret_i| = |Ret_i|$ for $\forall$i . Then using Equation 7.19,

$$\frac{F\ score_o}{F\ score_k} = \left(1+\frac{\sum_{i=2}^{i=n} |Rel \cap Ret_i|}{|Rel \cap Ret_1|} \right) \times \frac{1}{1+\frac{\sum_{i=2}^{i=n} |Ret_i|}{|Ret_1|+|Rel|}}$$

$$= \left(1+\frac{\sum_{i=2}^{i=n} |\mathrm{Re}t_i|}{|Ret_1|} \right) \times \frac{1}{1+\frac{\sum_{i=2}^{i=n} |Ret_i|}{|Ret_1|+|Rel|}} \qquad \textbf{(7.20)}$$

$$= \frac{1+\frac{X}{A}}{1+\frac{X}{A+|Rel|}} \quad where\ X = \sum_{i=2}^{i=n} |Ret_i| \ and\ A = |Ret_1|$$

And it is obvious that

$$\frac{X}{A} > \frac{X}{A+|Rel|}$$

$$\Rightarrow 1+\frac{X}{A} > 1+\frac{X}{A+|Rel|}$$

$$\Rightarrow \frac{1+\frac{X}{A}}{1+\frac{X}{A+|Rel|}} > 1$$

Then using Equation 7.20, we get,

$$\frac{F\ score_o}{F\ score_k} > 1 \Rightarrow F\ score_o > F\ score_k$$

Assume worst case; each C_i *(i >1)* returns only irrelevant concepts. Then $Rel \cap Ret_i = 0$ and $|Rel \cap Ret_i| = 0$ for $\forall \mathrm{i} > 1$. Then using Equation 7.19,

$$\frac{F\ score_o}{F\ score_k} = \left(1+\frac{\sum_{i=2}^{i=n} |\ Rel \cap Ret_i\ |}{|Rel \cap Ret_1\ |} \right)\times\frac{1}{1+\frac{\sum_{i=2}^{i=n} |\ Ret_i\ |}{|Ret_1\ |+|Rel\ |}}$$

$$= \left(1+\frac{0}{|Rel \cap Ret_1|} \right)\times\frac{1}{1+\frac{\sum_{i=2}^{i=n} |\ Ret_i\ |}{|Ret_1\ |+|Rel\ |}}$$

$$= \frac{1}{1+\frac{\sum_{i=2}^{i=n} |\ Ret_i\ |}{|Ret_1\ |+|Rel\ |}} < 1$$

$$\frac{F\ score_o}{F\ score_k} < 1 \Rightarrow F\ score_o < F\ score_k$$

7.3.3 Empirical Results

Table 3 Recall/Precision/F score for Two Search Techniques

Types of Queries		Recall		Precision		F score	
		Ontology	Keyword	Ontology	Keyword	Ontology	Keyword
Generic /broader queries	Query 1	90	11	98	95	94	20
	Query 2	95	15	89	90	92	26
	Query 3	87	30	100	82	93	44
Specific/ narrow queries	Query 4	90	76	76	90	83	83
	Query 5	85	71	100	72	91	71
	Query 6	100	65	77	100	87	79
Context queries	Query 7	90	76	76	29	82	42
	Query 8	85	83	100	34	92	49
	Query 9	100	74	74	16	85	27

The comparison metrics used for these two search techniques are precision, recall, and F score. First, we discuss precision, recall, and F score for individual queries. These queries are then grouped into the three categories: broad query, narrow query, and context query. Next, we present average precision, recall, and F score for each category, and then for all the categories taken together.

In Figures 7.8, 7.9, and 7.10, the X axis represents sample queries. The first three queries are related to broad query formulation, the next three to narrow query formulation, and the last three queries to context queries. Thus, results are reported for only nine queries. In Figures 7.8, 7.9, and 7.10 for each query the first and second bars represent the recall/precision/F score for ontology and keyword-based search technique respectively. Corresponding numerical values are reported in Table 3. Although, the vector space model is ranked-based and our ontology-based model is a Boolean retrieval model, in the former case we report precision for maximum recall in order to make a fair comparison.

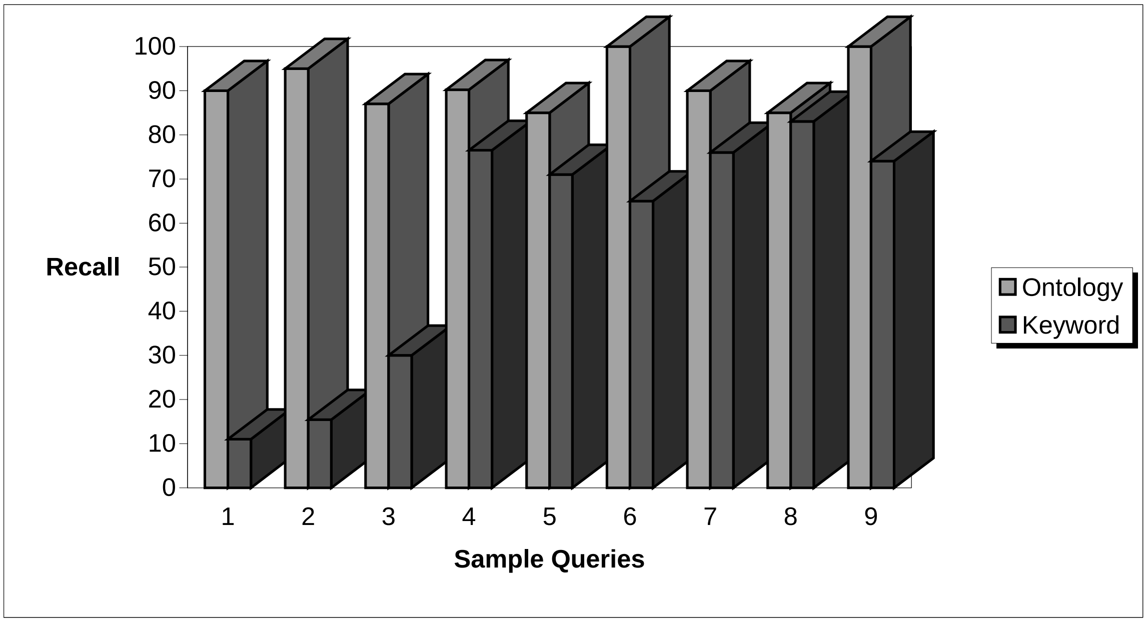


Figure 7.8 Recall of Ontology-based and Keyword-based Search Techniques

In Figure 7.8, the data demonstrates that recall for our ontology-based model outperforms recall for keyword-based technique. Note that this pattern is pronounced related to broader query cases. For example, in query 1, 90% verses 11% recall is achieved for ontology-based as opposed to keyword-based technique whereas for query 4, 90% and 76% recall are obtained. This is because in the case of a broader query, more children concepts are added, as compared to narrow query formulation or a context query case. Furthermore, in a context query case, it is usual for broader query terms to give context only. In an ontology-based model these terms will not participate in the query expansion mechanism. Instead, broader query terms will be subsumed under specific concepts. For example, in query 7, the user requests "tell me about team Lakers." Concepts referring to "team" will not be expanded. Therefore, the gap between the two techniques is not pronounced.

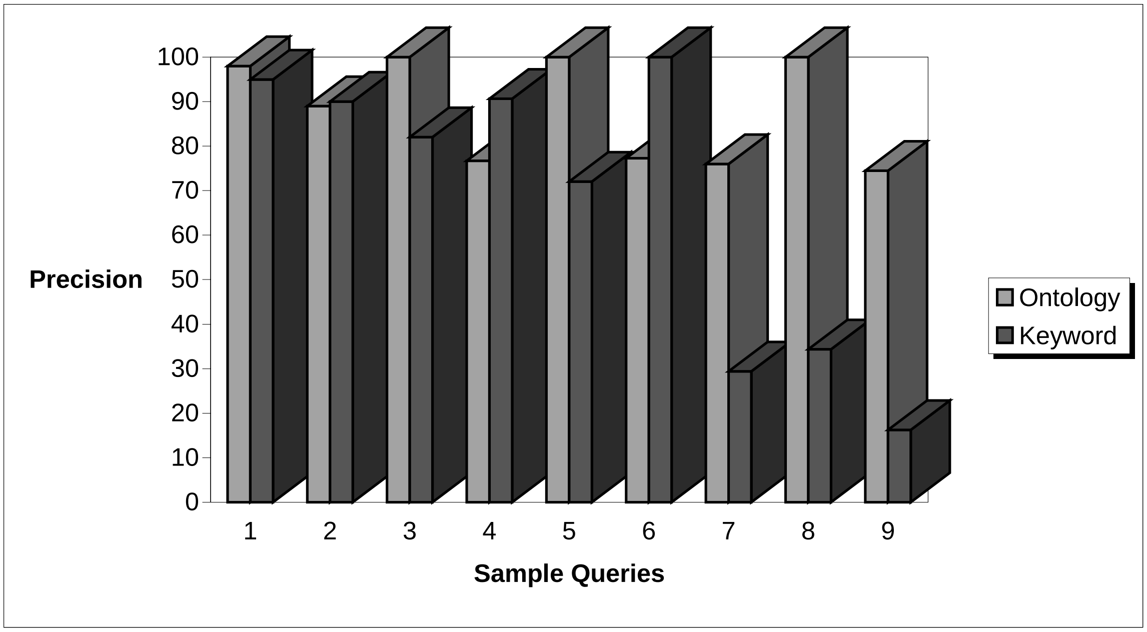


Figure 7.9 Precision of Ontology-based and Keyword-based Search Techniques

In Figure 7.9, for broader query cases, usually the precision of the ontology-based model outperforms the precision of the keyword-based technique. This is because our disambiguation algorithm disambiguates upper level concepts with greater accuracy compared to lower level concepts. For example, the disambiguation algorithm for metadata acquisition chooses the most appropriate region for each audio object. Recall that a region is formed by a league, its team, and its players. Thus if a query is requested in terms of a particular league, that is related to upper concept in this region, precision will not be hurt. However, the algorithm might fail to disambiguate lower level concepts in that region (e.g. players). For a narrow query formulation case, the precision obtained in the ontology-based model may not be greater than that obtained through use of the keyword-based technique. In query 4, the user requests "tell me about Los Angeles Lakers." In the ontology-based model the query is expanded to include all this team's players. It might be possible during disambiguation in metadata acquisition for some of these players to be associated with audio objects as irrelevant concepts; in particular when disambiguation fails. Some relevant concepts, such as other players, are also associated with these audio objects. Thus, for our ontology-based model these objects will be retrieved as a result of query expansion, leading to a deterioration in precision. In a keyword-based case, we have not expanded "Lakers" in terms of all of the players on the Lakers team. Therefore, we just look for the keyword "Lakers" and the abovementioned irrelevant objects associated with its group of players will not be retrieved. Thus, in this instance we observed 76% and 90% precision for ontology-based and keyword-based technique respectively.

In the case of the context query, it is evident that the precision of the ontology-based model is much greater than that of the keyword-based model. Since in the ontology-based model some concepts subsume other concepts, audio objects will only be retrieved for specific concepts. On the other hand a search using keyword-based technique looks for all keywords. If the user requests "team Lakers" the keyword-based technique retrieves objects with the highest rank when the keywords "team" and "Lakers" are present. Furthermore, in order to facilitate maximum recall, we have observed that relevant objects will be displaced along with irrelevant objects in this rank. Note that some irrelevant objects will also be retrieved that only contain the keyword "team." Thus, for query 7, levels of precision of 76% and 29% have been achieved.

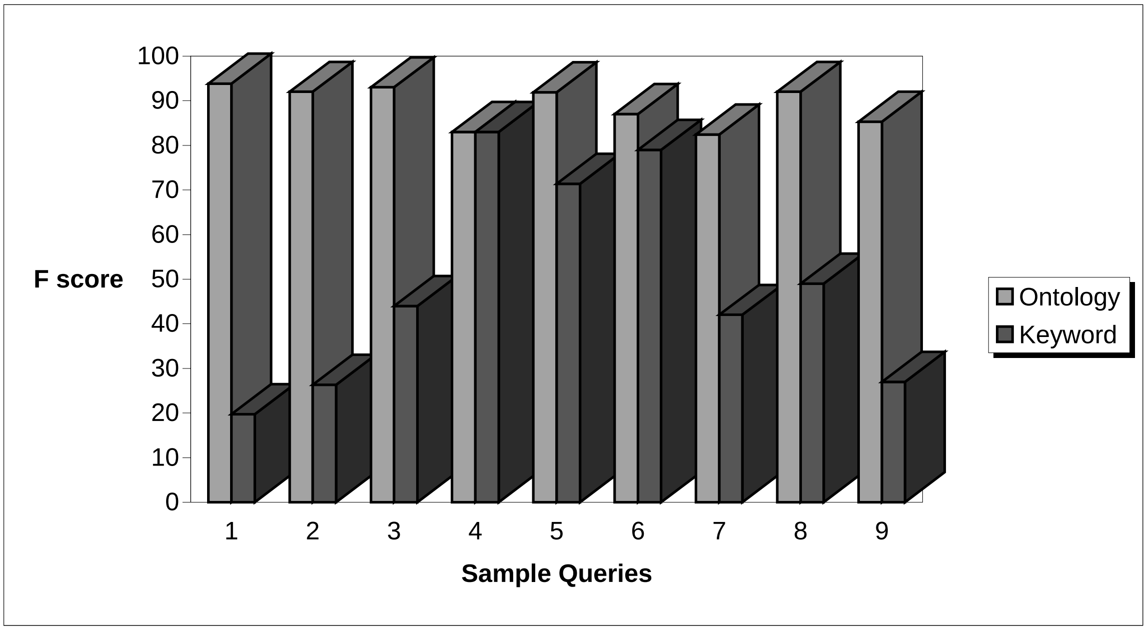


Figure 7.10 F score of Ontology-based and Keyword-based Search Techniques

Finally, the F score of our ontology-based model outperforms (or at least equals) that of a keyword-based technique (see Figure 7.10). For the broader and context query case, precision and recall are usually high for the ontology-based model in comparison with keyword-based technique. Therefore, F scores differences, for the ontology-based model are also pronounced. For example, for query 1, the F scores for ontology-based and keyword-based technique are 94% and 20% respectively. For the narrow query case, the F score of our ontology-based model is slightly better or equal to that of the keyword-based technique. For example, in query 4, we observed a similar F score (83%) in both cases; however in queries 5 and 6 we observed that the F score of the ontology-based model (91%, 87%) outperformed the keyword-based technique, (71%, 79%).

Table 4 Illustration of Power of Ontology over Keyword-based Technique

Types of Queries	Recall %			Precision %			F score %		
	Ontology	Keyword	Gain	Ontology	Keyword	Gain	Ontology	Keyword	Gain
Generic	91	19	379	96	89	8	93	30	210
Specific	92	71	30	85	88	-3	87	78	12
Context	91	78	17	84	26	223	86	39	121
Overall	91	56	63	88	68	20	89	49	81

In Table 4, the data for average precision, recall, and F score for each type of query has been reported. We have also reported on the effectiveness of our ontology-based model over keyword based search by measuring the difference in scores between these two for recall, precision, and F score in each of the query types. Formally, for this we define

$$Gain = \frac{(A_o - A_k)}{A_k} \qquad \textbf{(7.21)}$$

where A can be recall/precision/ F score. Thus, the gain in precision

$$= \frac{(Precision_o - Precision_k)}{Precision_k}. \qquad \textbf{(7.22)}$$

For broad queries, the gain in recall is very high compared with the gain for specific and context queries. This is because, as we have already pointed out, more additional concepts are added in the case of the former. For narrow query, the gain in precision is negative. This is because for narrow query our disambiguation algorithm sometimes fails to disambiguate lower level concepts, and also because during the phase of query expansion the addition of new concepts related to these lower level concepts can hurt precision. On the other hand, for context queries the gain in precision is very high. This is because our ontology-based model chooses the most appropriate concept from the context and subsumes concepts which are more generic. Finally, the overall gains in recall, precision, and F score respectively are 63%, 20%, and 81%, for the different types of queries combined which proves our claim on behalf of our ontology-based model.

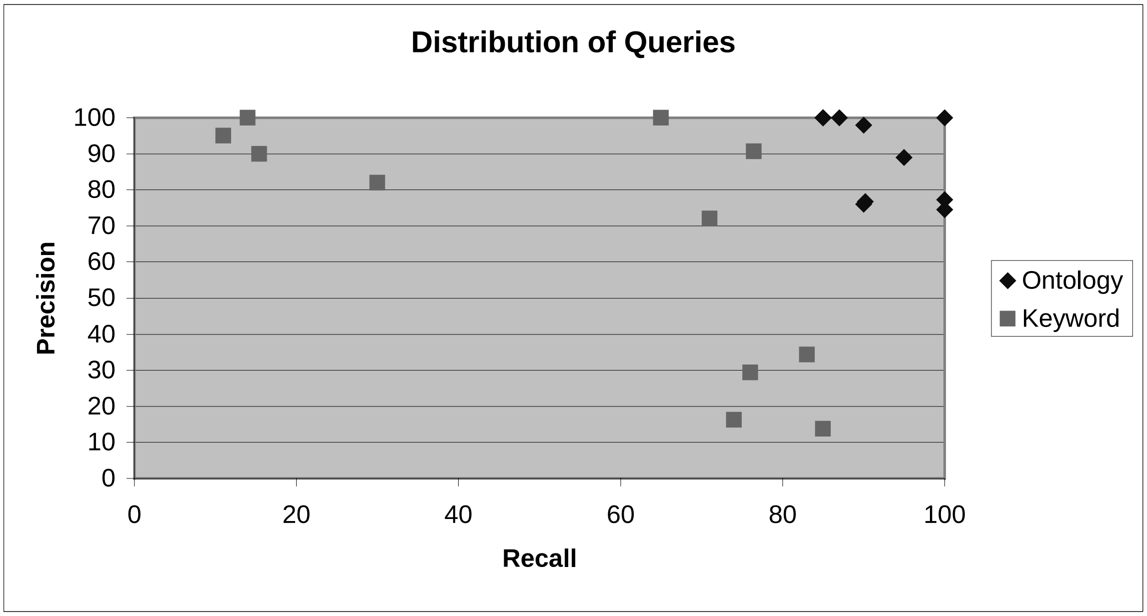


Figure 7.11 Distribution of Queries in Terms of Precision and Recall for Ontology-based and Keyword-based Search Techniques

In Figure 7.11 we display the distribution of different queries in terms of precision and recall for two search techniques. Here, the X axis represents recall and the Y axis represents precision. We can observe that the results of all queries for the ontology-based model reflect a high level of precision and recall compared to keyword-based search.

Chapter 8 Conclusions and Future Work

In this dissertation we have proposed a potentially powerful and novel approach for the selection of information. The crux of our innovation is the deployment of an ontology-based model that facilitates responses to requests for information with high precision and high recall. This ontology-based model uses concept to concept matching between user requests and documents rather than keyword to keyword matching. Therefore, the key problem in the use of this technique is to identify and match appropriate concepts which describe and identify documents on the one hand, and on the other, the language employed in user requests. In this it is critically important to make sure that irrelevant concepts will not be associated and matched, and that relevant concepts will not be discarded. In other words, it is important to insure that high precision and high recall will be preserved during concept selection for both documents in the database and the natural language employed in user requests.

In this dissertation, we have developed an automatic mechanism for concept selection from these two sources, documents and user requests. Furthermore, this concept selection mechanism includes a novel, scalable disambiguation algorithm which uses domain specific ontology, and which will prune irrelevant concepts while allowing relevant concepts to become associated with documents and to participate in query generation. We have also proposed an automatic query expansion mechanism which deals with natural language user requests. This mechanism generates database queries which allow only appropriate and relevant expansion through taking into account knowledge encoded in the ontologies. We have further devised a framework for allowing user requests expressed in natural language to be automatically mapped into SQL database queries, with no user knowledge of the database or SQL query procedures. We have also demonstrated that some novel optimization techniques which rewrite the SQL query with the help of knowledge that comes from the ontology can be employed without a loss of precision and recall. We have specifically demonstrated the effectiveness of our model in the domain of audio information for sports news.

Given the current state of the art speech recognition technology, the selection of audio information is necessarily based upon the description of audio segments, or metadata generation. We claim that ontology can be employed to facilitate metadata generation by using word-spotting technique that reduces the chance of speech recognition error. At present an experimental prototype of the model has been developed. For sample audio content we use CNN broadcast sports and Fox Sports audio, along with closed captions. We have demonstrated the superiority, analytically and empirically, of the retrieval effectiveness of our ontology-based model over traditional keyword-based search techniques.

Although we have used a domain of sports news information for a demonstration project, our results can be generalized to fit many additional important content domains including but not limited to all audio news media. We are confident that the fundamental conceptual framework for this project is sound, and that its implementation is completely feasible from a technical standpoint.

Now the question is: can we adopt this ontology-based model to the task of web document retrieval? Since the web is a collection of an enormous amount of information which cannot possibly be described by a single ontology, modeling the web will require several domain dependent ontologies. Furthermore, cross-references will probably exist among these domain dependent ontologies. Clearly, the success of an ontology-based model depends entirely on the availability of such ontologies. Through the efforts of the knowledge engineering community the possibility of using ontologies successfully to solve web document retrieval problems in the near future has been enhanced [2, 39, 54, 92, 95]. Researchers in this arena have made several inspiring contributions:

- Techniques for the construction of domain dependent ontologies from existing ontologies,
- Techniques for merging different existing ontologies, and
- Tools for the maintenance of ontologies, such as GUI for the construction of ontology and applications of XML for the creation of an ontology markup language.

The above innovations and developments enable us to boost our claim that an ontology-based model will be adopted as a search mechanism in a web setting in the near future.

8.1 Future Work

We propose extending this work in five directions: evolving ontologies, extracting highlighted sections of audio, dynamic updates of user profiles, addressing retrieval questions in the video domain, and facilitation of cross-media indexing.

It is impossible to construct an ontology that is sufficient for all purposes and domains. Furthermore, it may not even be desirable to build a comprehensible, stable ontology in what is sure to be a rapidly changing environment. We would like, rather, to build ontology that is easy to update, open and dynamic both algorithmically and structurally for easy construction and modification, and fully capable of adapting to changes and new developments in a domain [66]. For example, suppose player "Bryant Kobe" switches from team "Los Angeles Lakers" to team "Portland Trail Blazers." In this case, we need to remove the interrelationship link between concepts "Bryant Kobe" and "Los Angeles Lakers" and add a new link between the concepts "Bryant Kobe" and "Portland Trail Blazers." In this connection, we would like to address the problem of how to create useful ontology by minimizing the cost of initial creation, while allowing for novel concepts to be added with minimum intervention and delay. For this, we would like to combine techniques from knowledge representation, natural language processing, and machine learning. In this connection we can mention some works [2, 39, 40, 92].

Users may be interested in highlights of the news. For this, we need to identify and store these highlights. By analyzing the pitch of the recorded news we can identify sections to be highlighted. This is because as is well known in the speech and linguistics communities there are changes in pitch under different speaking conditions [37, 83]. For example, when a speaker introduces a new topic the range of pitch will be increased. On the other hand, sub-topics and parenthetical comments are often associated with a compression of pitch range. Since pitch varies considerably between speakers, it is also necessary to find an appropriate threshold for a particular speaker.

We would like also to address the problem of customization through assessing ways to dynamically update user profiles. Users express their interests in terms of keywords, topics, and so on. Therefore, user profiles may be generated to the extent that a user might specify concepts by browsing the ontology through a user interface. However, human interests change as time passes. For example, people are interested in earthquake information just after a big earthquake, but this interest gradually disappears. It is cumbersome for users to have to modify keywords often. Moreover, people cannot necessarily specify what they are interested in because their interests are sometimes unconscious [17, 46, 48, 56]. One way to anticipate user preference is to register the time spent listening to each news item. This approach is also intuitively reasonable because users spend more time reading interesting news items than uninteresting ones. For this, we need an intelligent agent [61, 90] which can perhaps capture implicit user preferences via learning. The agent observes the manner in which the user interacts with the news items, based on the time spent, then tries to estimate the user's interests and to suitably modify the user's profile.

We would also like to extend work in the domain of video. Video is an information-intensive medium. Beside its temporal property, shared with audio, video has a spatial property which makes the problem more challenging [4, 60, 67]. For example, a user might request "give me all video clips in which President Clinton and President Yeltsin are shaking hands." To respond to such a query we will need a data modeling technique which can support both spatial and temporal requests.

We would like to consider cross-modal queries that go beyond the current modes of textual and visual query formulation to more advanced methods that capture a user's intention from natural language text initiated requests and recast these into appropriately reformulated and expanded image, audio, and video queries. Further advanced cross-media indexing mechanisms will be required to support cross-modal queries [16]. These will be built upon the combined analysis of audio, video, and text content.

8.2 Concluding Remarks

The field of digital media continues to be heavily impacted by significant and rapidly expanding technical advances. These advances are changing the nature of the information generated by these media. A great deal of textual information is now being augmented by ever increasing amounts of non-textual information: images, video, and audio streams. Therefore, the potential for the exchange and retrieval of information is vast, and at times daunting. In general, users are easily overwhelmed by the amount of information available via electronic means. For example, when we enter a search into a web browser, we receive pages of links--only some which are relevant, and many of which are not. Therefore, it is essential if the use of the web is to continue to expand as a source of information, both in the context of search for knowledge and with regard to commercial transactions, for query and retrieval mechanisms to become increasing sophisticated and efficient. An efficient search mechanism will guarantee the delivery of a minimum of irrelevant information (high precision), as well as insuring that relevant information is not overlooked (high recall). Organizing the database through the use of ontology is by far the most promising avenue to creating a search mechanism that will greatly enhance the possibility of obtaining high precision and high recall. But doing so is not an easy task. In order to use ontology as a search mechanism there are several non trivial problems that must be addressed. In this dissertation we have addressed these problems and put forward some robust and significant directions for the effective retrieval of information based on the actual meaning of documents rather than relying on the mere simultaneous occurrence of keywords in the document and the query.

References

[1] S. Adali, K. S. Candan, S. Chen, K. Erol, and V. S. Subrahmanian, "Advanced Video Information System: Data Structures and Query Processing," *ACM-Springer Multimedia Systems Journal*, vol. 4, pp. 172-186, 1996.

[2] E. Agirre, O. Ansa, E. Hovy and D. Martinez, "Enriching Very Large Ontologies Using the WWW," in *Proc. of the Ontology Learning Workshop*, ECAI, Berlin, Germany, 2000.

[3] E. Agirre and G. Rigau, "Word Sense Disambiguation using Conceptual Density," in *Proc. of the Coling-ACL'96 Workshop*, pp.16-22, Copenhagen, Denmark, 1996.

[4] G. Ahanger and T. D. C. Little, "Automatic Composition Techniques for Video Production," *IEEE Transaction on Knowledge and Data Engineering*, vol. 10, no. 6, 1998.

[5] B. Arons, "SpeechSkimmer: Interactively Skimming Recorded Speech," in *Proc. of ACM Symposium on User Interface Software and Technology*, pp. 187-196, Nov 1993.

[6] G. Aslan and D. McLeod, "Semantic Heterogeneity Resolution in Federated Database by Metadata Implantation and Stepwise Evolution," *The VLDB Journal, the International Journal on Very Large Databas*es, vol. 18, no. 2, Oct 1999.

[7] R. Baeza and B. Neto, *Modern Information Retrieval*, ACM Press New York, Addison Wesley, 1999.

[8] J. Barnett, S. Anderson, J. Borglio, M Singh, R Hudsom, and S. Kuo, " Experiments in Spoken Queries for Document Retrieval," in *Proc. of Eurospeech 97*, vol. 3, pp. 1323-1326, Ordhes, Greece, Sept 1997.

[9] K. Bharat and A. Broder, "A Technique for Measuring the Relative Size and Overlap of Public Search Engines," in *Proc. of the Seventh WWW Conference*, pp. 379-388, Brisbane, Australia, 1998.

[10] K. Bharat, A. Broder, M. Henzinger, P. Kumar, and S. Venkatsubramanian, "The Connectivity Server: Fast Access to Linkage Information of The Web," in *Proc. of the Seventh WWW Conference*, Brisbane, Australia, 1998.

[11] K. Bharat and M. Henzinger, “Improved Algorithms for Topic Distillation in Hyperlinked Environments,” in *Proc of the Twenty First ACM SIGIR*, pp. 104-111, Melbourne, Australia, 1998.

[12] R. Bodner and F. Song, “Knowledge-based Approaches to Query Expansion in Information Retrieval,” in *Proc. of Advances in Artificial Intelligence*, pp. 146-158, New York, Springer.

[13] M. Brown, J. Foote, G. Jones, K. Jones, and S. Young, “Automatic Content-Based Retrieval of Broadcast News,” in *Proc. of ACM Multimedia*, pp. 5-9, San Francisco, CA, Nov, 1995.

[14] M. Bunge, *Treatise on basic Philosophy, Ontology I: The Furniture of the World*, vol. 3, Reidel Publishing Co., Boston, 1977.

[15] M. Carey, D. DeWitt, J. Naughton, M. Asgarian, P. Brown, J. Gehrke, and D. Shah, “The Bucky Object-Relational Benchmark,” in *Proc. of ACM SIGMOD*, pp. 135-146, May 1997.

[16] S. Chang, “Multimedia Access and Retrieval: The State of the Art and Future Directions,” in *Proc. of ACM Multimedia*, vol. 1, pp. 443-445, Orlando, FL, Nov 1999.

[17] P. Chesnais, M. Mucklo, and J. Sheena, “The Fishwrap Personalized News System,” in *Proc. of the Second IEEE International Workshop on Community Networking*, pp. 275-282, 1995.

[18] CNN Sports Illustrated, 1999, http://cnnsi.com.

[19] F. Crestani, “Effects of Word Recognition Errors in Spoken Query Processing,” in *Proc. of IEEE Advances in Digital Libraries*, pp. 39-47, Bethesda, MD, May 2000.

[20] C. Crouch and B. Yang, “Experiments in Automatic Statistical Thesaurus Construction,” in *Proc. of the Fifteenth ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 77-88, Copenhagen, Denmark, 1992.

[21] CLASSIC ESPN, The Classic Sports Network, 1999, http://www.classicsports.com.

[22] FoxSports, http://www.foxsports.com.

[23] N. Fuhr, “Probabilistic Models in Information Retrieval,” *The Computer Journal*, vol. 35, no. 3, pp. 243-255, 1992.

[24] G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L.A. Streeter, and K.E. Lochbaum, “Information Retrieval Using a Singular Value Decomposition Model of Latent Semantic Structure,” in *Proc. of the Eleventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 465-480, 1988.

[25] S. Gauch, J. Gauch, and K. M. Pua, “VISION: A Digital Video Library,” In *Proc. of ACM Digital Libraries*, pp. 19-27, Bethesda, MD, 1996.

[26] A. Ghias, J. Logan, and D. Chamberlin, “Query by Humming--Musical Information Retrieval in an Audio Database,” in *Proc of ACM Multimedia Conference*, pp. 231-235, 1995.

[27] S. Gibbs, C. Breitender, and D. Tsichritzis, “Data Modeling of Time based Media,” in *Proc. of ACM SIGMOD*, pp. 91-102, 1994, Minneapolis, USA.

[28] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran, “Indexing with WordNet Synsets can Improve Text Retrieval,” in *Proc. of the Coling-ACL'98 Workshop: Usage of WordNet in Natural Language Processing Systems*, pp. 38-44, August 1998.

[29] S. Greene, S. Devlin, P. Cannata, and L. Gomez, “No IFs, ANDs, or ORs: A Study of Database Querying,” *International Journal of Man-Machine Studies*, vol. 32, no. 3, pp. 303-326, 1990.

[30] T. R. Gruber, “Toward Principles for the design of Ontologies used for Knowledge Sharing,” in *Proc of International Workshop on Formal Ontology*, Padova, Italy, March 1993.

[31] T. R. Gruber, “A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition,” *An International Journal of Knowledge Acquisition for Knowledge-based Systems*, vol. 5, no. 2, June 1993.

[32] N. Guarino, C. Masolo, and G. Vetere, “OntoSeek: Content-based Access to the Web,” *IEEE Intelligent Systems*, vol. 14, no. 3, pp. 70-80, 1999.

[33] V. Gudivada, V. Raghavan, W. Grosky, and R. Kashanagottu, “Information Retrieval on the World Wide Web,” *IEEE Internet Computing*, pp. 58-68, Nov, 1997.

[34] A. Hampapur and R. Jain, “Video Data Management Systems: Metadata and Architecture” in *Multimedia Data Management: Using Metadata to Integrate and*

Apply Digital Media, Editors W. Klas, and A. Sheth, chapter 10, pp. 245–284, McGraw Hill, 1999.

[35] A. G. Hauptmann, “Speech Recognition in the Informedia Digital Video Library: Uses and Limitations,” in *Proc. of the Seventh IEEE International Conference on Tools with AI*, Washington, DC, Nov 1995.

[36] A. G. Hauptmann and M. Witbrock, “Informedia: News-on-Demand Multimedia Information Acquisition and Retrieval,” *Intelligent Multimedia Information Retrieval*, Editor M. T. Maybury, AAAI Press, pp. 213-239, 1997.

[37] J. Hirschberg and B. Grosz, “Intonational Features of Local and Global Discourse,” in *Proc. of the Speech and Natural Language Workshop*, pp. 23-26, Harriman, NY, Feb 1991.

[38] R. Hjelsvold and R. Midstraum, “Modeling and Querying Video Data,” in *Proc. of the Twentieth International Conference on Very Large Databases (VLDB’94)*, pp.686-694, Santiago, Chile, 1994.

[39] E. Hovy and K. Knight, “Motivation for Shared Ontologies: An Example from the Pangloss Collaboration,” in *Proc. of the Workshop on Knowledge Sharing and Information Interchange, IJCAI-93*, Chambry, France, 1993.

[40] C. Hwang, “Incompletely and Imprecisely Speaking: Using Dynamic Ontologies for Representing and Retrieval Information,” *Technical Report of Microelectronics and Computer Technology Corp, MCC*, 2000.

[41] Informix, “Informix Universal Server: Informix guide to SQL: Syntax version 9.1,” 1997.

[42] P. Jacobs and L. Rau, “Innovations in Text Interpretation,” *Artificial Intelligence*, vol. 63, no. 1-2, pp. 143-191, 1993.

[43] D. James, “The Application of Classical Information Retrieval Techniques to Spoken Documents,” *Ph.D. Dissertation*, University of Cambridge, United Kingdom, 1995.

[44] M. Jarke and J. Koch, “Query Optimization in Database Systems,” *Computing Surveys*, June 1984.

[45] Java Database Connectivity, 2000, http://java.sun.com/products/jdbc/.

[46] T. Kamba, K. Bharat, and M. C. Albers, “The Krakatoa Chronicle--An Interactive, Personalized, Newspaper on the Web,” in *Proc. of the Fourth International World Wide Web Conference*, pp. 159-170, Nov 1995.

[47] A. Kemper, G. Moerkotte, and M. Steinbrunn, “Optimizing Boolean Expressions in Object Bases,” in *Proc. of the Eighteenth VLDB Conference*, pp. 79-90, Vancouver, British Columbia, Canada 1992.

[48] W. Klippgen, T. Little, G. Ahanger, and D. Venkatesh, “The Use of Metadata for the Rendering of Personalized Video Delivery Metadata,” in *Multimedia Data Management: Using Metadata to Integrate and Apply Digital Media*, Editors W. Klas, and A. Sheth, chapter 10, pp. 287–318. McGraw Hill, 1999.

[49] J. Kupiec, “MURAX: A Robust Linguistic Approach for Question Answering Using an On-line Encyclopedia,” in *Proc. of the Sixteenth Annual International ACM SIGIR Conference*, pp. 181-190, Pittsburgh, PA, 1993.

[50] L. Khan, “Structuring and Querying Personalized Audio using Ontologies,” in *Proc. of ACM Multimedia*, vol. 2, pp. 209-210, Orlando, FL, Nov 1999.

[51] L. Khan and D. McLeod, “Audio Structuring and Personalized Retrieval Using Ontologies,” in *Proc. of IEEE Advances in Digital Libraries, Library of Congress*, pp. 116-126, Bethesda, MD, May 2000.

[52] L. Khan and D. McLeod, “Disambiguation of Annotated Text of Audio Using Onologies,” to appear in *Proc. of ACM SIGKDD Workshop on Text Mining*, Boston, MA, August 2000.

[53] L. Khan and D. McLeod, “Efficient Retrieval of Audio Information from Annotated Text Using Ontologies,” to appear in the *Proc. of ACM SIGKDD Workshop on Multimedia Data Mining*, Boston, MA, August 2000.

[54] K. Knight, I. Chander, M. Haines, V. Hatzivassiloglou, E. H. Hovy, M. Iida, S. K. Luk, R. A. Whitney, and K. Yamada, “Filling Knowledge Gaps in a Broad-Coverage MT System,” in *Proc. of the Fourteenth IJCAI Conference*, Montreal, Quebec, Canada, 1995.

[55] Y. Labrou and T. Finin, “Yahoo! as an Ontology: Using Yahoo! Categories to Describe Documents,” in *Proc. of the Eighth International Conference on Information Knowledge Management*, pp. 180-187, Nov, 1999, Kansas City, MO.

[56] K. Lang, “Newsweeder: Learning to Filter Netnews,” in *Proc. of the Twelfth International Conference on Machine Learning*, pp. 331-339, 1995.

[57] D. B. Lenat and R.V. Guha, *Building Large Knowledge-Based Systems: Representation and Interface in the CYC Project*, Addison Wesley, Reading, MD, 1990.

[58] D. B. Lenat, “Cyc: A Large-scale investment in Knowledge Infrastructure,” *Communications of the ACM*, pp. 33-38, vol. 38, no. 11, Nov 1995.

[59] H.V. Leighton and J. Srivastava, “Precision among World Wide Web Search Engines: AltaVista, Excite, Hotbot, InfoSeek, and Lycos,” http://www.winona.msus.edu/library/webind2/webind2.htm, 1997.

[60] T. D. C. Little and A. Ghafoor, “Interval based Conceptual Models for Time-Dependent Multimedia Data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 5, no. 4, 1993.

[61] P. Maes, “Agents that Reduce Work and Information Overload,” *Communications of the ACM*, pp. 30-40,vol. 37, no.7, July 1994.

[62] B. McCune, R. Tong, J. S. Dean, and D. Shaprio, “Rubric: A System for Rule-based Information Retrieval,” *IEEE Transactions on Software Engineering*, vol. 11, no. 9, 1985.

[63] G. Miller, “WordNet: A Lexical Database for English,” *Communications of the ACM*,” vol. 38, no. 11, Nov, 1995.

[64] G. Miller, “Nouns in WordNet: a Lexical Inheritance System," *International Journal of Lexicography*, vol. 3, no. 4, pp. 245-264, 1994.

[65] E. Mena, V. Keshyap, A. Illarramendi, and A. Sheth, “Improving Answers in Distributed Environments: Estimation of Information Loss for Multi-Ontology Based Query Processing,” to appear in *International Journal of Cooperative Information Systems (IJCIS)*.

[66] E. Mena, V. Keshyap, A. Illarramendi, and A. Sheth, “Domain Specific Ontologies for Semantic Information Brokering on the Global Information Infrastructure,” in *Proc. of International Conference on Formal Ontologies in Information Systems (FOIS’98)*, June 1998.

[67] M. Naphade, and B. Frey, “Probabilistic Multimedia Objects (MULTIJECTS): A Novel Approach to Video Indexing and Retrieval in Multimedia Systems,” in *Proc. of IEEE Conference on Image Processing*, Chicago, Oct 1998.

[68] E. Omoto and K. Tanaka, “OVID: Design and Implementation of a Video-Object Database System,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 5, no. 4, August 1993.

[69] N. Patel and I. Sethi, “Audio Characterization for Video Indexing,” in *Proc. of SPIE Conference on Storage and Retrieval for Still Image and Video Databases*, vol. 2670, pp. 373-384, San Jose, CA, 1996.

[70] H. J. Peat and P. Willett, “The Limitations of Term Co-occurrence Data for Query Expansion in Document Retrieval Systems,” *Journal of ASIS*, vol. 42, no. 5, pp. 378-383, 1991.

[71] M. F. Porter, “An Algorithm for Suffix Stripping Program,” Editors J. S. Karen, and P. Willet, *Readings in Information Retrieval*, San Francisco, Morgan Kaufmann, 1997.

[72] Y. Qui and H. P. Feri, “Concept Base Query Expansion,” in *Proc. of the Sixteenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 160-169, 1993.

[73] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, 1978.

[74] Real Player, http://www.rela.com.

[75] C. J. Rijsbergen, *Information Retrieval*, London, Butterworths, 1979. http://www.dcs.gla.ac.uk/Keith/Chapter.7/Ch.7.html.

[76] S. E. Roberson and K. S. Jones, “Relevance Weighting of Search Terms,” *Journal of the American Society for Information Sciences*, vol. 27, no. 3, pp. 129-146, 1976.

[77] G. Salton*, Automatic Text Processing*, Addison Wesley, 1989.

[78] G. Salton, *The SMART Retrieval System--Experiments in Automatic Document Processing*, Prentice Hall Inc., Englewood Cliffs, NJ, 1971.

[79] G. Salton, E. Fox, and H. Wu, “Extended Boolean Information Retrieval,” *Communications of the ACM*, vol. 26, no. 11, pp. 1022-1036, Nov1983.

[80] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval* McGraw Hill, 1983.

[81] T. Schoch, *Comparing Natural Language Retrieval. WIN & Freestyle. Online*, vol. 19, no. 4, pp. 83-87, 1995.

[82] P. G. Selinger, M. M. Astrahan, D. D. Chamberlin, R.A. Lorie, and T.G. Price, "Access Path Selection in a Relational Database Management System," in *Proc. of ACM SIGMOD*, June 1979.

[83] K. E. A. Silverman, "The Structure and Processing of Fundamental Frequency Contours," *PhD. Dissertation*, University of Cambridge, April 1987.

[84] R. Srihari, and W. Li, "Information Extraction Supported Question Answering, in *Proc. of Text REtrieval Conferences (TREC)*, Oct, 1999.

[85] T. G. A. Smith and N. C. Picever, "Parsing Movies in Context," in *Proc. of the 1991 Summer USENIX Conference*, Nashville, USA, 1991.

[86] A. F. Smeaton and A. Quigley, "Experiments on Using Semantic Distances between Words in Image Caption Retrieval," in *Proc. of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1995.

[87] A. F. Smeaton and V. Rijsbergen , "The Retrieval Effects of Query Expansion on a Feedback Document Retrieval System," *The Computer Journal*, vol. 26, no. 3 pp. 239-246, 1993.

[88] W. M. Shaw, R. Burgin, and P. Howell, "Performance Standards and Evaluations IR Test Collections: Cluster-based Retrieval Models," *Information Processing and Management*, vol. 33, no.1, pp.1-14, 1997.

[89] W. Small and L. Weldon, "An Experimental Comparison of Natural and Structured Query Languages," *Human Factors*, vol. 25, no. 3, pp. 253-263, 1983.

[90] B. Sheth and P. Maes, "Evolving Agents for Personalized Information Filtering," in *Proc. of the Ninth Conference on Artificial Intelligence for Applications*, Orlando, FL, March 1993.

[91] T. Strzalkowsk, *Natural Language Information Retrieval*, Kulwar Academic Publishers, 1999.

[92] B. Swartout, R. Patil, K. Knight, and T. Ross, "Toward Distributed Use of Large-Scale Ontologies," in *Proc. of the Tenth Workshop on Knowledge Acquisition for Knowledge-Based Systems*, Banff, Canada, 1996.

[93] C. Tenopir and P. Cahn, *Target & FREESTYLE: DIALOG and Mead Join the Relevance Ranks, Online*, vol. 18, no. 3, pp. 31-47, 1994.

[94] H. Turtle, and W. Croft, "Evaluation of an Inference Network-based Retrieval Model," *ACM Transactions on Information Systems*, vol. 9, no. 3, pp. 187-222, July 1991.

[95] Using XML: Ontology and Conceptual Knowledge Markup Languages, 1999, http://www.oasis-open.org/cover/xml.html.

[96] E. Voorhees, "Query Expansion Using Lexical-Semantic Relations," in *Proc. of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 61-69, 1994.

[97] S. Wartick, "Boolean Operations," Editors W. B. Frakes and R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms*, pp. 264-292, Prentice Hall, Englewood Cliffs, NJ, 1992.

[98] M. Wechsler and P. Schuble, "Metadata for Content-based Retrieval of Speech Recordings," in *Multimedia Data Management: Using Metadata to Integrate and Apply Digital Media*, Editors W. Klas, and A. Sheth, chapter 8, pp. 223–243, McGraw Hill, 1999.

[99] L. D. Wilcox and M. A. Bush, "Training and Search Algorithms for an Interactive Wordspotting System," in *Proc. of ICASSP*, vol. 2, pp. 97-100, San Francisco, CA, 1992.

[100] R. Wilkinson and P. Hilgston, "Using the Cosine Measure in a Neutral Network for Document Retrieval," in *Proc. of the Fourteenth ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 202-210, Chicago, USA, Oct 1991.

[101] Windows Media Player, http://www.microsoft.com/windows/mediaplayer.

[102] E. Wold, T. Blum, and D. Keislar, "Content-based Classification, Search, and Retrieval of Audio," *IEEE Multimedia*, pp. 7-36, Fall, 1996.

[103] W. Woods, "Conceptual Indexing: A Better Way to Organize Knowledge," *Technical Report of Sun Microsystems*, 1999.

[104] C. Yu and W. Meng, *Principles of Database Query Processing for Advanced Applications*, Morgan Kaufmann Publishers, Inc, San Francisco, CA, 1999.

[105] B. Yuwono and D. L. Lee, “Search and Ranking Algorithms for Locating Resources on World Wide Web,” in *Proc. of International Conference on Data Engineering (ICDE)*, pp. 164-171, New Orleans, USA, 1996.

[106] L. A. Zadeh, “Fuzzy Sets”, Editors D. Dubois, H. Prade, and R. R. Yager, *Readings in Fuzzy Sets for Intelligent Systems*, Morgan Kaufmann, 1993.