Effective Retrieval of Audio Information from Annotated Text Using Ontologies[1]

Latifur Khan and Dennis McLeod
Department of Computer Science and
Integrated Media Systems Center
University of Southern California
Los Angeles, California 90089
[latifurk, mcleod]@usc.edu

ABSTRACT

To improve the accuracy in terms of precision and recall of an audio information retrieval system we have created a domain-specific ontology (a collection of key concepts and their interrelationships), as well as a novel, pruning algorithm. Taking into account the shortcomings of keyword-based techniques, we have opted to employ a concept-based technique utilizing this ontology. The key problem in the retrieval of audio information is to achieve high precision and high recall. Typically, in traditional approaches, high recall is achieved at the expense of low precision, and vice versa. Through the use of a domain-specific ontology appropriate concepts can be identified during metadata generation (description of audio) or query generation, thus improving precision. In case of the association of irrelevant concepts to queries or documents there is a loss of precision. On the other hand, if relevant concepts are discarded, a loss of recall will ensue. Therefore, in conjunction with the use of a domain specific ontology we have proposed a novel, automatic pruning algorithm which prunes as many irrelevant concepts as possible during any case of query generation. By associating concepts in the ontology through techniques of correlation, this algorithm presents a method for the selection of concepts in the query generation. To improve recall, controlled and correct query expansion mechanism is proposed. This guarantees that precision will not be lost. Moreover, we present a way for the query generation in which domain-specific ontology can be used to generate information selection requests in terms of database queries in SQL. In trial implementations we have demonstrated that our ontology-based model outperforms keyword-based technique (vector space model) in terms of precision and recall.

Keywords

Metadata, Ontology, Audio, SQL, Precision, and Recall.

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. 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 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).

The traditional solution to the problem of recall and precision in information retrieval employs keyword-based search techniques. 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. Additional search terms are added to the original query based on the statistical co-occurrence of terms [20]. Recall will be expanded, but at the expense of deteriorating precision [17, 24]. 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 [12]. This model, which employs a domain dependent ontology, is presented in this paper. An ontology is a collection of concepts and their interrelationships which can collectively provide an abstract view of an application domain [5, 8].

There are two distinct questions for ontology-based model: 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

[1] This research has been funded [or funded in part] by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152.

recall will be preserved during concept selection for documents or user requests. In this paper, we propose an automatic mechanism for the selection of these concepts from user requests by addressing the first problem. This mechanism will prune irrelevant concepts while allowing relevant concepts to become associated with user requests. Furthermore, a novel, scalable disambiguation algorithm for concept selection from documents using domain specific ontology is presented in [13].

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 [4]. 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. 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.

In order to demonstrate the effectiveness of our disambiguation model we have explored and provided a specific solution to the problem of retrieving audio information. 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 by relying on speech recognition technology which converts speech to text. After generating transcripts we can deploy our ontology-based model to facilitate information selection requests. At present, an experimental prototype for the implementation 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 and Fox Sports audio, along with closed captions. 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. Nine 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.

The remainder of this paper is organized as follows. In Section 2, we review related work. In Section 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 Section 4, we introduce our domain dependent ontology. In Section 5, we present metadata management issues that arise for our ontology based model in the context of audio information unit. In Section 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 along with pruning algorithm. In Section 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 Section 8 we present our conclusions and plans for future work.

2 Related Works

Historically ontologies have been employed to achieve better precision and recall in the text retrieval system [9]. 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 [15], 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 [22] 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. The notion of conceptual distance between query and document provides an alternative approach to modeling relevance. Smeaton et al. [20] and Gonzalo et al. [7] 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 pruning algorithm, is fully automatic.

Although we use audio, here we show related work in the video domain which is closest to and which complements our approach in the context of data modeling for the facilitation of information selection requests. Key related work in the video domain for selection of video segments includes [1, 11, 16]. Of these, Omoto et al. [16] use a knowledge hierarchy to facilitate annotation, while others use simple keyword based techniques without a hierarchy. The model of Omoto et al. fails to provide a mechanism that automatically converts a generalized description into a specialized one(s). Further, this annotation is manual and does not deal with the disambiguation issues related to concepts.

3 Research Context: Audio

Audio is one of the most powerful and expressive of the non-textual media. 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 [6]. 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 [2]. 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 [18]:

$$En = \sum_{m=-\infty}^{m=\infty} [x(m)w(n-m)]^2 = \sum_{m=n-N+1}^{m=n} x(m)^2$$

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 [10], and selected content extraction [23]. In fully automated content extraction, speech is converted to equivalent text (e.g., Informedia). 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. 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 object itself from fox sports and CNN web site in our case (see Section 7).

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. An audio-object O_i is defined by five tuple (id_i, S_i, E_i, V_i, A_i) where Id_i is an object identifier which is unique, S_i is the start time, E_i is the end time, V_i (description) is a finite set of tag or label, i.e., $V_i=\{v_{1i}, v_{2i}, \ldots, v_{ji}, \ldots, v_{ni}\}$ for a particular j where v_{ji} is a tag or label name, and 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}, *}. Of the information in the five tuple, the first four items (identifier, start time, end time, and description) are called *metadata*.

4 Ontologies

An ontology is a specification of an abstract, simplified view of the world that we wish to represent for some purpose [5, 8]. Therefore, an ontology defines a set of representational terms that we call *concepts*. Interrelationships among these concepts describe a target world. An ontology can be constructed in two ways, domain dependent and generic. CYC [14], WordNet [15], or Sensus [21] are examples of generic ontologies. For our purposes, we choose a domain dependent ontology. First, this is because a domain dependent ontology provides concepts in a fine grain, while generic ontologies provide concepts in coarser grain. Second, a generic ontology provides a large number of concepts that may contribute large speech recognition error.

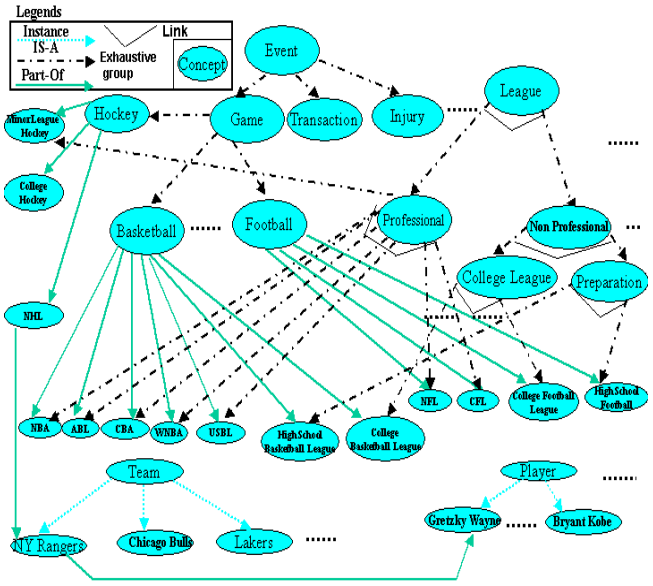

Figure 1. A Small Portion of an Ontology for Sports Domain

Figure 1 shows an example ontology for sports news. This ontology is usually obtained from generic sports terminology and domain experts. This ontology is described by a directed acyclic graph (DAG). Here, each node in the DAG represents a concept. In general, each concept in the ontology contains a label name and a synonyms list. Note also that this label name is unique in the ontology. Further, this label name is used to serve as association of concepts with audio objects. The synonyms list of a concept contains vocabulary (a set of keywords) through which the concept can be matched with user requests. Formally, each concept has a synonyms list $(l_1, l_2, l_3, \ldots, l_i, \ldots, l_n)$ where user requests are matched with this l_i what we call *element* of list. Note that a keyword may be shared by multiple concepts' synonyms lists. For example, player "Bryant Kobe," "Bryant Mark," "Reeves Bryant" share common word "Bryant" which may create ambiguity problem.

4.1 Interrelationships

In the ontology, concepts are interconnected by means of interrelationships. If there is a interrelationship R, between concepts C_i and C_j, then there is also a interrelationship R′ between concepts C_j and C_i. In Figure 1, interrelationships are represented by labeled arcs/links. Three kinds of interrelationships are used to create our ontology: IS-A, Instance-Of, and Part-Of. These correspond to key abstraction primitives in object-based and semantic data models [3].

IS-A: This 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 kind of C_i. For example, "NFL" is a kind of "Professional" league. In other words, "Professional" league is the generalization of "NFL." In Figure 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. The IS-A interrelationship 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. For example, "Professional" relates to a set of concepts, "NBA", "ABL", "CBA", ..., by exhaustive group (denoted by caps in Figure 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, then this generalized concept will not participate in the metadata generation and SQL query generation explicitly. 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 1 "Professional" concept is 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, the union of specialized concepts is not equal to the generalized concept.

Instance-Of: This is used to show membership. 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 a concept, "Player." In general, all players and teams are instances of the concepts, "Player" and "Team" respectively.

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 "Football" concept and player, "Wayne Gretzky" is Part-Of "NY Rangers" concept.

4.2 Disjunctness

When a number of concepts are associated with a parent concept through 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 metadata or query generation phases. Similarly, concept "College Football", "College Basketball" are disjoint concepts due to their associations with parent concept, "College League" through IS-A. Furthermore, "Professional," and "Non Professional" are disjoint. Thus, we can say that "NBA," "CBA," "ABL," "College Basketball," and "College Football," are disjoint. Each of these league and its team and player form a boundary what we call *region* (see Figure 2). During annotation of concepts with an audio object we strive to choose a particular region. This is because an audio object can be associated with only one disjoint-type concept. However, it may be possible that a particular player may play in several leagues. In that case, we make multiple instances of the player. In other words, for each league he plays, we maintain a separate concept for him. This way we preserve disjoint-property.

Concepts are not disjoint, on the other hand, when they are associated with a parent concept through 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 Ranger" and player "Gretzky Wayne," where "Grezky Wayne" is Part-Of "NY Rangers."

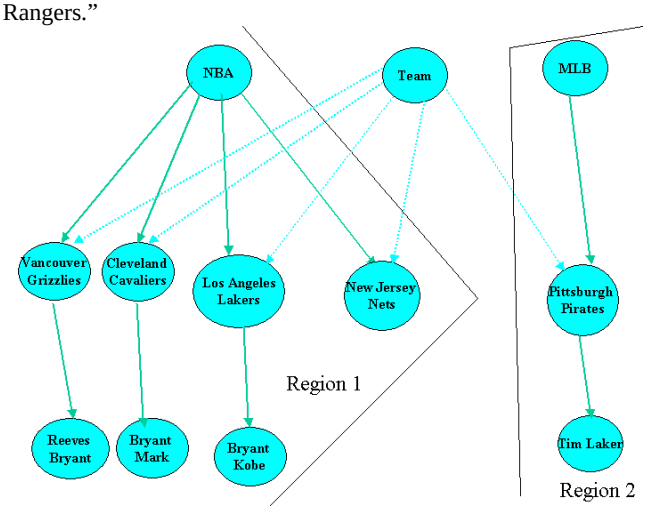

Figure 2. Different Regions of an Ontology

5 Metadata Acquisition and Management of Metadata

Metadata acquisition is the name for the process through which descriptions are provided for audio objects. For each audio object we need to find the most appropriate concept(s). Recall that using content extraction (see Section 3.2) we get a set of keywords which appear in a given audio object. 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 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. 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 (see [13] for more details).

Effective management of metadata facilitates efficient storing and retrieval of audio information. To this end, in our model most specific concepts are considered as metadata. Several concepts of the ontology, for example, can become the candidate for the metadata of an audio object. However, some of these may be 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 (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 can simply store the general concept, "Professional" for this object. When user request comes in terms of 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, for example, an audio object becomes the candidate for the concepts "NHL", "Hockey", and "Professional." During the annotation process the object will only 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 query comes in terms of "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 recall problem, user requests are first passed through ontology on the fly and expressed in terms of most specific concepts. Even so, the audio object, in the above example, can still be retrieved through querying the system by "NHL", "Hockey", and "Professional."

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. 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 the scheduler. 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. Furthermore, each object's description is stored as a set of rows or tuples in the Meta_News table for normalization purpose.

6 Query Mechanisms

We now focus specifically on our techniques for utilizing an ontology-based model for processing information selection requests. 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 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 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 based on semantic closeness. 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 other selected concepts will have a higher score, and a greater probability of being retained than ambiguous concepts which are not correlated.

For example, if a query is specified by "Please tell me about team Lakers," QConcepts "Team," "Los Angeles Lakers," and major league baseball player, "Tim Laker" (of team "Pittsburgh Pirates") are selected. 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.

6.1.1 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 user request. Recall that in an ontology each concept (QC_i) has a complementary list of synonyms ($l_1, l_2, l_3, ..., l_j, ..., l_n$). Keywords in the user request 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 (semantic distance) in the ontology. Let us formally define each of these scores.

Definition 1: Element-score (Escore): The Element-score of an element l_j for a particular QConcept QC_i is the number of keywords of l_j matched with keywords in the user request divided by total number of keywords in l_j.

$$Escoreij \equiv \frac{\#\ of\ \ keywords\ of\ \ lj\ \ matched}{\|\ \#\ of\ \ keywords\ \ in\ lj\ \|}$$

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 2: Concept-score (Score): The Concept-score for a QConcept, QC_i is the largest score of all its element-scores. Thus,

$$Score_i = max\ Escore_{ij}\ \ where\ 1 \le j \le n$$

Definition.3: Semantic distance (SD (QC_i , QC_j)): *SD(QC_i, QC_j)* between QConcepts QC_i and QC_j is defined as the shortest path between two QConcepts, QC_i and QC_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 2). This is because the two concepts are directly connected via a Part-Of interrelationship. Similarly, the semantic distance between "NBA," and "Bryant Kobe" is 2. The semantic distance between "Los Angeles Lakers," and "New Jersey Nets" is infinite.

Definition.4: Propagated-score (S_i): If a QConcept, QC_i, is correlated with a set of QConcepts *(C_j, C_{j+1},...,C_n)*, the propagated-score of QC_i is its own Score, $Score_i$ plus the scores of each of the correlated QConcepts' *(QC_k k=j, j+1, ..., n)* $Score_k$ divided by *SD (QC_i, QC_k)*. Thus,

$$S_i = Score_i + \sum_{k=j}^{k=n} \frac{Score_k}{SD(QC_i,QC_k)}$$

$$= Score_i + \frac{Score_j}{SD(QC_i,QC_j)} + \frac{Score_{j+1}}{SD(QC_i,QC_{j+1})} + ... + \frac{Score_n}{SD(QC_i,QC_n)}$$

For example, in Figure 2 let us assume that values of $Score_i$ for "Los Angeles Lakers" and "Bryant Kobe" be 0.5 and 1.0 respectively. Furthermore, these concepts are correlated with a semantic distance of 1, and their Propagated-scores are 1.5 (0.5 + 1.0/1) and 1.5 (1.0+0.5/1) respectively. The pseudo code for the pruning algorithm is as follows:

QC_1, QC_2 , …, QC_l, …, QC_r are selected with concept-score $Score_1$,…,$Score_l$,…$Score_r$

Determine correlation of selected concepts *(QC_i, QC_j, QC_{j+1}, .., QC_n)* and update their Propagated-scores using

$$S_i = Score_i + \sum_{k=j}^{k=n} \frac{Score_k}{SD(QC_i,QC_k)}$$

$$= Score_i + \frac{Score_j}{SD(QC_i,QC_j)} + \frac{Score_{j+1}}{SD(QC_i,QC_{j+1})} + ... + \frac{Score_n}{SD(QC_i,QC_n)}$$

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, ... where $S_i > S_j > S_l$, ...

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 4

//End of For loop each QConcept

Figure 3. Pseudo Code for Pruning Algorithm

Using pruning algorithm (see Figure 3), for a user request, "team Lakers," at the beginning selected QConcepts are "Team", "Los Angeles Lakers" and "Tim Laker" (see Figure 2). Note that ambiguous concepts are "Los Angeles Lakers," and "Tim Laker." In Figure 2 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 QConcepts, "Team," "Los Angeles Lakers," and "Tim Laker" 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. 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.

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 (see Figure 2). 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. 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 for this QConcept only.

6.2 Query Expansion and SQL Query Generation

We now discuss a technique for query expansion and SQL query 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. The complete algorithm is shown in Figure 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 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) 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"...)

6.2.1 Remedy of Explosion of Boolean Condition

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.

7 Experimental Implementation

In discussing implementation we will first, present our experimental setup, and then we will demonstrate power of our ontology-based over keyword-based search techniques. 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), which is an object relational database system. For the sample audio content we use CNN broadcast sports audio and Fox Sports. 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. Currently, our working ontology has around 7,000 concepts for the sports domain. 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.

7.1 Results

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 [19].

7.1.1 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:

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

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

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.

7.1.2 Types of Queries

Sample queries are classified into 3 categories, with each category containing 3 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. The comparison metrics used for these two search techniques are precision, recall, and F score. We discuss precision, recall, and F score for individual queries.

7.1.3 Empirical Results

In Figures 5, 6, and 7, 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. In Figures 5, 6, and 7 for each query the first and second bars represent the recall/precision/F score for ontology-based and keyword-based search techniques respectively.

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.

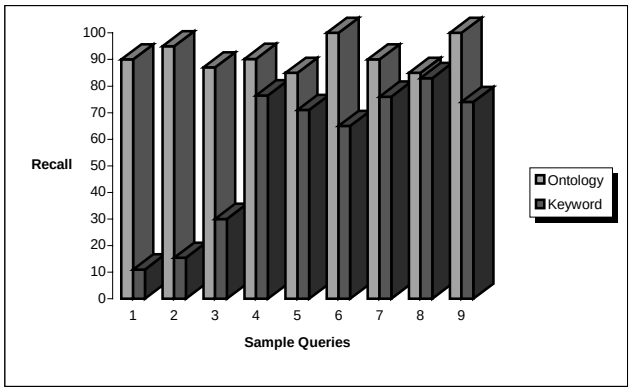

Figure 5. Recall of Ontology-based and Keyword-based Search Techniques

In Figure 5, 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.

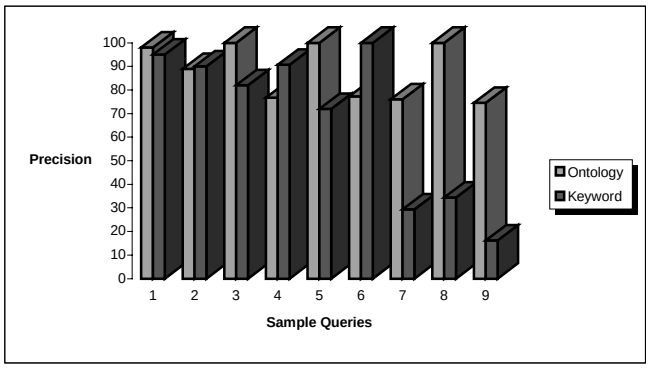

Figure 6. Precision of Ontology-based and Keyword-based Search Techniques

In Figure 6, 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.

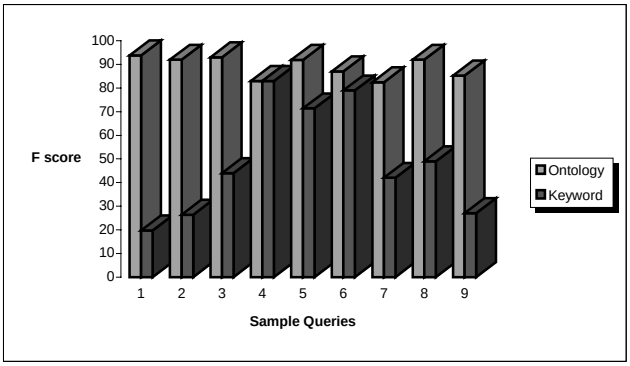

Figure 7. F score of Ontology-based and Keyword-based Search Techniques

Finally, the F score of our ontology-based model outperforms (or at least equals) that of a keyword-based technique (see Figure 7). 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%).

8 Conclusions

In this paper we have proposed a potentially powerful and novel approach for the retrieval of audio information. The crux of our innovation is the development of an ontology-based model for the generation of metadata for audio, and the selection of audio information in a user customized manner. We have shown how the ontology we propose can be used to generate information selection requests in database queries. We have used a domain of sports news information for a demonstration project, but our results can be generalized to fit many additional important content domains including but not limited to all audio news media. Our ontology-based model demonstrates its power over keyword based search techniques by providing many different levels of abstraction in a flexible manner with greater accuracy in terms of precision, recall and F score. Although we are confident that the fundamental conceptual framework for this project is sound, and its implementation completely feasible from a technical standpoint, some questions remain to be answered in future work. These include detailed work on user studies and evaluation. In this connection, we are confident that we will ultimately be able to develop an intelligent agent that will dynamically update user profiles. This will provide a level of customization that can have broad application to many areas of content and user interest.

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

[3] 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.

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

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

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

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

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

[16] 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.

[17] 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.

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

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

[20] 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.

[21] 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.

[22] 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.

[23] 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.

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