Proceedings of the National Conference
On Undergraduate Research (NCUR) 2006
The University of North Carolina at Asheville
Asheville, North Carolina
April 6 - 8, 2006

Ontology-Based Search Engine for a Real-World Decision Support System

Brian C. Becker
Christina M. Vargas
Department of Electrical and Computer Engineering
University of Central Florida
4000 Central Florida Blvd.
Orlando, FL 32816. USA

Faculty Advisor: Dr. Avelino J. Gonzalez, PhD, P.E.

Abstract

As the amount of digital information increases, finding a specific piece of information becomes increasingly difficult. Historically, search engines have endeavored to use a search term or question to sift through large amounts of information to find relevant data. Traditional search engines usually have difficulties finding relevant information unless the search term occurs explicitly within the information. While this approach oftentimes yields valid search results, it fails to account for variations or related words not specified in the search term, forcing users to rephrase their question and perform another search. In a decision support system, searching becomes more critical as users will expect an intelligent response on the first try. Utilizing a tree of word relationships, such as an ontology, strengthens linkage between words and phrases. An ontology contains commonly used words within a domain of knowledge and links them with other words in the ontology through a combination of five relationships: hypernym, hyponym, holonym, meronym, and synonym. An ontology is best used in a localized domain, such as a Decision Support System (DSS), since it allows a search engine to perform more in-depth matching between the user's question and the knowledge base that contains the answer. This paper investigates the application and effectiveness of a search engine designed for a real-world domain-specific DSS called AlexDSS. The knowledge within AlexDSS is categorized into contextual graphs (CxG) with each graph describing a specific practice, process, or fact. When selected through the ontology-aided search engine, a contextual graph interacts with the user through a dialogue to tailor the information to the user's unique situation. Comparisons between a traditional keyword-only search engine and an ontology-based search engine in AlexDSS will determine the effectiveness of utilizing an ontology.
Keywords: Ontologies, Search Engines, Decision Support Systems, Knowledge Acquisition

1. Introduction

Searching involves locating information based on a small subset of that knowledge, usually referred to as a search term or query. Historically, with regard to Internet search engines or text-processing applications, searching could only locate information that exactly matched the words contained in the search query. Slight variations in the search query, such as pluralizing a word or using a synonym, would return completely different results. When searching within a Decision Support System (DSS), exact matching is not ideal since words typically have highly interconnected meanings. In specialized domains, words may take on completely different meanings and there is often an overabundance of acronyms. In such instances, exact word matches fail to retrieve all the relevant information. To solve this problem, additional tools can be used to augment exact word searches. This paper focuses on using an ontology as a basis for a search engine in a localized domain. The following section provides background information addressing the need for the research. In section 3, previous work is investigated for potential application to the problem at hand. The methods and implementation of the ontology and the search engine are examined in section 4 followed by an evaluation of the effectiveness of the work. Finally, the paper concludes with some lessons learned and a discussion of future research.

2. Background

The domain of interest for this search engine is NSF's Industry/University Cooperative Research Center (I/UCRC) program. The I/UCRC program encourages and funds the combination of universities and industries into partnership-based research centers. The long-standing I/UCRC program manager Dr. Alex Schwarzkopf is regarded as the expert in managing I/UCRCs and serves as the resource all individual I/UCRC directors turn to with questions or issues. In preparation of his retirement, NSF has funded the development of a Decision Support System named AlexDSS in honor of Dr. Schwarzkopf. The goal of AlexDSS is to capture, preserve, and allow the reuse of the knowledge and expertise Dr. Schwarzkopf has acquired in his position as program manager. AlexDSS is also a response to NSF's need to be able to answer the numerous questions posed about the I/UCRC program without overburdening personnel dedicated to other I/UCRC related tasks. However, AlexDSS is not designed to replace Dr. Schwarzkopf but to help answer commonly asked questions in a timely fashion. As an online computer program available anytime from anywhere, research center directors or Dr. Schwarzkopf's successor can use AlexDSS for guidance regarding issues or problems in the I/UCRC domain and expect quality answers.

The knowledge within AlexDSS is organized into a set of activities represented as Contextual Graphs (CxG)[Brezillon]. Each activity addresses a specific topic and represents a self-contained piece of knowledge such as a definition, explanation, or advising session. Activities can interactively query the user in a question and answer (Q&A) fashion to provide the proper context for the issue at hand. Upon finishing an activity, AlexDSS presents the user with an answer based on the Q&A session. Additionally, activities can be nested so that broader topics encompass more specific topics, allowing smaller activities to be combined to form more comprehensive activities. Currently, AlexDSS contains over 100 activities with more being added as the system matures. Since scanning through over a hundred activities is cumbersome, a method to aid users in locating activities was needed. Although a topical organization was considered, it was deemed that a user would find a search engine interface more familiar.

To augment the search engine, a domain-specific ontology was used. An ontology is a natural language processing concept that relates words and phrases. Chandrasekaran defines an ontology as a "representation vocabulary often specialized to some domain or subject matter." The important thing about an ontology is the conceptualizations it represents and the relations between words [Chandrasekaran]. Although many standard relationships such as synonym exist, any custom relationship may be defined and used. One example of a well known ontology that uses two standard relationships, synonyms and antonyms, is a thesaurus. Each word in a thesaurus or ontology links to other words via these relationships. In a search engine, the purpose of an ontology is to more concretely define words and phrases and provide an additional measure of similarity between a search query and the information being searched. Although words in the query might not exactly match words in the information being searched for, an ontology might be able to assist the process of matching them by inferring connections based not on word similarity but conceptual similarity.

3. Related Work

Several previous works have been considered as starting points for this research involving ontology-based search engines. In a similar research project, Pinheiro and Moura implemented a web portal search engine called TOSS. This search engine utilizes an ontology to expand the search query to include words of similar meaning for the purpose of achieving more relevant search results [Pinheiro and Moura]. Similarly, Sim developed a filtering agent that uses an ontology for the same goal as TOSS, although the problem was approached in a different manner. Instead of enhancing the search query, the Information Filter Agent (IFA) performs full-text analysis of the returned results (web pages) and re-filters and re-ranks them using heuristics that take the ontology relationships into account. Interestingly the IFA uses the popular WORDNET ontology, developed by Princeton University, and several relationships, each with different weight, to calculate a relevancy factor [Sim]. When compared to keyword only searching, IFA and TOSS increase search result relevancy by approximately 5%.

While both approaches deal with ontology-based search engines, neither specifically limits the search to a particular domain. In an expert system such as a DSS, the terminology being used is restricted to a unique domain. Furthermore, the knowledge residing in an expert system is highly specialized. Both qualities emphasize the necessity of creating a custom ontology rather than a using pre-existing general-purpose ontology such as WORDNET, especially since a lot of domain specific terminology would be missing. Although an existing ontology could be extended with the specialized terms found in the I/UCRC domain, it was decided to forgo the additional complexity that approach would entail. Although ontologies can be compiled by hand, one novel approach is to

automatically generate an ontology based on user search or browsing history as is done by Pretschner and Gauch [Pretschner and Gauch]. This approach of automatically constructing the ontology was considered, but is unsuited to AlexDSS because the knowledge is not sufficiently wide-spread in use Furthermore, not enough user searches exist to form a complete and statistically valid ontology. However, instead of using automation to generate an ontology, automated analysis assists in validating the ontology.

4. Methodology

This section details the development of both the ontology and the search engine. Although designed and implemented concurrently, the ontology will be discussed first to provide a background to ontologies. Section 4.2 discusses the integration of the ontology with the search engine.

4.1 ontology development

Many tools exist as an aid to ontology creation. During initial research we tested 3 ontology editors: the Karlsruhe Ontology and Semantic Web tool suite (KAON), Protégé, and Ontolingua. KAON is the product of collaboration between the Institute for Applied Informatics and Formal Description Methods at the University of Karlsruhe, Germany and Forschungszentrum Informatik (FZI). Protégé was developed by Stanford Medical Informatics at the Stanford University School of Medicine. An online editor, Ontolingua Ontology Editor was created by the Stanford Knowledge Systems AI Laboratory (KSL) Network Services. Upon evaluation, Protégé proved itself more user-friendly and better supported in terms of documentation. Its easy-to-use slots combined with a drag and drop system seemed optimal for this project. However, it lacked full featured inverse relationships. For instance, assuming two inverse relationships PART-OF and HAS-A, creating the relationship “car HAS-A engine” in Protégé would not automatically add “engine PART-OF car.” This functionality is available when using instances instead of classes, but instances cannot be added hierarchically as classes can, a feature used extensively for organizing the ontology.

Because of limitations found in other ontology editing programs, a new in-house ontology editor program called nawsOntology was developed specifically for the AlexDSS search engine project. Based on Protégé, it contained similar features and added automatic inverse relationship capabilities. By implementon only the functionality required by the AlexDSS ontology, complexity found in KAON, Protégé, and Ontolingua was reduced. To avoid being locked into using nawsOntology and to allow for future growth, nawsOntology can export files to the popular OWL Web Ontology Language file format. This way the ontology constructed in nawsOntology could easily be transferred to Protégé for further development or analysis if needed.

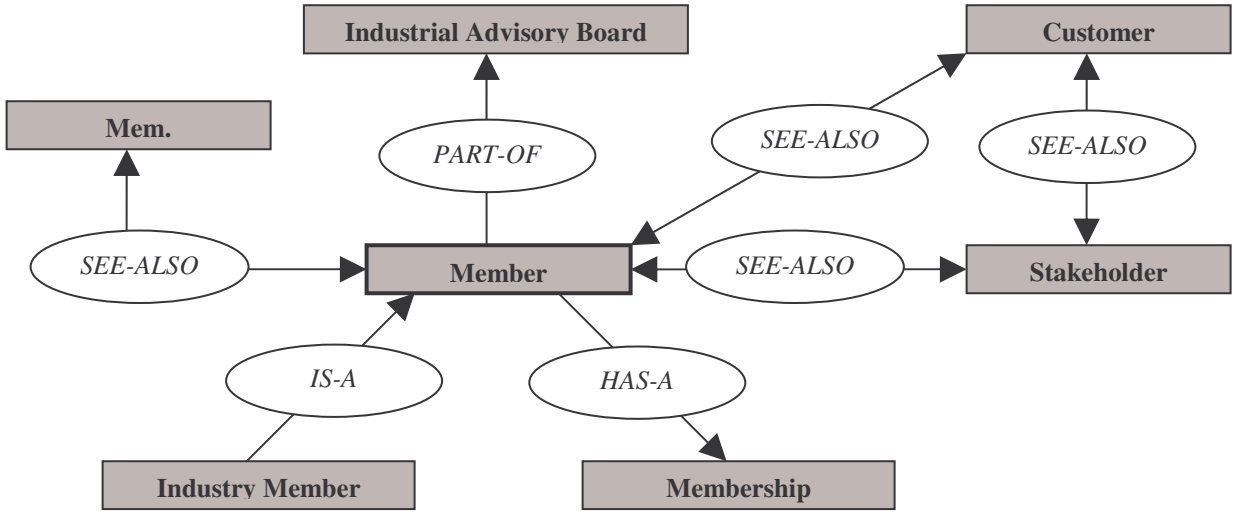

Figure 1. Sample ontology term “member” and relationships.

NawsOntology defines five ontological relationships that cover the basic linkages in the I/UCRC domain: IS-A (both hyponym and hypernym), HAS-A (meronym), PART-OF (holonym), and SEE-ALSO (synonym). The IS-A relationship is broken down into hyponym, a more generic word, and hypernym, a more specific word. The ontology was built from several sources. First, official written information on I/UCRCs such as the official book entitled *Managing the Industry/University Cooperative Research Center* were considered. Secondly, the knowledge acquisition engineer who shadowed Dr. Schwarzkopf made lists of words and relationships that needed to be added to the ontology based on the knowledge being entered in the system. In this way, both the ontology and the knowledge were incrementally constructed and kept in sync with each other. The contents of the ontology include definitions, acronyms, and common terms used within the I/UCRC domain. A standard entry in the ontology might be "Program Director ALSO-SEE Alex Schwarzkopf," or "discussion PART-OF semi-annual meeting." For a graphical representation of the term "member" and the associated relationships, see Figure 1.

4.2 search engine development

In order to most effectively utilize the search engine, each activity in the AlexDSS was indexed with a set of keywords that summarize the knowledge contained within the activity. This method was preferred over full-text searching for two reasons. First, full-text searching is slower. More importantly, a finer degree of control can be exerted over the activities by manually indexing them according to categories. By modifying the keywords attached to an activity, the activity can be filed or re-filed under a different category, ensuring that all the knowledge is correctly organized. To aid the user experience, the title of each activity is considered as a keyword phrase as well. Not only is the title indicative of the activity's topic, but this practice increases the probability that an activity will be retrieved if a user searches for part of the title. The assumption underlying this decision underscores the fundamental difference in how the user perceives relevance compared to how the system calculates relevance. The system calculates relevance primarily on keywords and activity categorizations. The user perceives relevance based on the information they see, namely the title and description of the activity. A search considering the title as part of the search criteria is far more likely to be regarded as effective by the user because the user never sees the other factors that the search engine bases its ranking on.

Because of its prevalence, a keyword search was implemented first. In addition to providing a working search engine, it supplied a baseline from which to benchmark the final ontology-based search engine. The keyword-only search engine works by splitting the input query into individual words (combining words where appropriate, such as merging I/UCRC to IUCRC) and comparing them to the keywords associated with all the activities. For simplicity's sake, the initial version did not take into account any weights or ranking. If one of the words in the search query matched one of the keywords for a particular activity, the activity was included in the result list and displayed to the user.

Before the ontology was integrated into the search engine, several non-ontology related algorithms were used to refine the search query and improve results. First, certain words, commonly referred to as "stop words" [Luján-Mora and Palomar], were eliminated from the search query. These include words such as articles, prepositions, commonly used adjectives, and other words that exist to provide grammatical structure to a sentence. This process reduces potential mismatching and prevents time-consuming attempts to match words that possess no functional value. To save processing cycles and avoid skewed results, words that are ambiguous are also removed. For instance, the word "IUCRC" or "center" does little to narrow down the search to a specific topic since nearly all topics relate to the word "IUCRC" or "center." Secondly, all of the words in the search query are analyzed and a set of root words is extracted. Through a process called stemming, affixes are stripped from the word, leaving a root word. Reducing a word to its root word negates the effects of pluralization and other slight variations between words. Third, a spelling check is used to verify that the search query is spelled correctly. In AlexDSS, a word is defined misspelled if the word does not occur in the activity keywords, in the ontology, or in a standard English dictionary. When such a word is encountered, the user is notified of the possible misspelling and is given a chance to correct the mistake in a fashion similar to Google's "Did you mean" feature. Using this definition of a misspelled word alleviates the problem of flagging domain-specific words or acronyms as misspelled.

Once the query string has been cleaned up and improved, the search engine performs a preliminary search. In most cases it was found that a keyword search resulted in fast and relevant search results. However, if the keyword search matched little to no activities, an ontology-based search could infer additional connections between words and expand the search results. This can be likened to using a dictionary to look up the definition of a particular word. For a common word, a dictionary is not needed; however, for a less common word whose definition is not clearly known, a dictionary aids understanding. The ontology is used in a similar fashion. Each word in the search term is analyzed for its five ontological relations: synonym, hypernym, hyponym, holonym, and meronym. These words are

added to the search query and matched to the activity keywords with a reduced weight. A weight represents the strength of the relationship. For instance, a synonym naturally has a stronger weight than a holonym (HAS-A) because the relationship defined by a synonym is a more exact. Especially valuable is the acronym section within the ontology. Using the synonym relationship, the word "MIPR" can easily be expanded to "Military Interdepartmental Procurement Request" and vice-versa. This relieves the burden on the knowledge engineer manually indexing of the activities. Instead of specifying all variations of the word MIPR, the knowledge engineer can specify only one of the variants and let the ontology figure out the remaining relationships. Moreover, this saves additional time if multiple activities contain the keyword MIPR. Once the search term's ontological relationships have been matched, the process can be repeated with the ontological relationships themselves. By looking up a synonym of a synonym or a holonym of a synonym, more related words can be found. Of course, the law of diminishing returns limits the usefulness of this iterative method. Each relationship leads you further and further away from the original word and deeper and deeper into the ontology. To counteract skewed rankings, the weight for each successive linked relationship is reduced. To prevent excessively long search times, a limit is placed on the depth the search engine can analyze the ontology.

Finally, two additional steps are performed to enhance the search engine. First, the search time is tracked such that the total time to search through the ontology and the activities will never exceed a specified time (such as half a second). This improves the user experience by creating the feel of a snappy search engine and thus a responsive application. This approach costs little in terms of relevancy since typically only a deep ontology search will exceed the time limit, and the deeper the ontology is searched, the less relevancy is encountered. Secondly, activities can be categorized by the type of information they contain. Currently, AlexDSS defines six categories: what, who, how, how long, why, and optimization. These categories correspond to information that provides descriptive information, contact information, procedural or step-by-step information, timeframe information, purpose information, and streamlining information. For instance, when asking "What is a MIPR?" the word "what" indicates the category is "about," specifying that the knowledge is largely informational. When the search returns a list of activities pertaining to MIPRs, activities that contain descriptions of what MIPRs are will be given more weight during the ranking phase because they are probably more relevant. Conversely, a search term containing the word "who" will weigh activities containing contact information as more relevant.

5. Evaluation

In order to evaluate the performance of the methods discussed in this paper, two versions of the search engine were developed: a simple keyword-only search engine and an enhanced ontology-based one with the additional algorithms designed to improve the search results. A list of about 100 common questions was compiled to form a test data set. To automate the evaluation process, a human processed the list and manually identified the most relevant activity for each question. This list could then be compared to the results of each version of the search engine. Three tiered statistics were calculated: exact matches, top 3 matches, and page matches. An exact match occurs if the search engine's top search result was the correct activity, i.e. the activity the human picked. Likewise, top 3 and page matches are recorded if the search engine returns the correct activity in the top 3 and or within the page results, respectively. The analysis stopped at one full page of results (6 results) because it was assumed users would be less inclined to scroll to a second page.

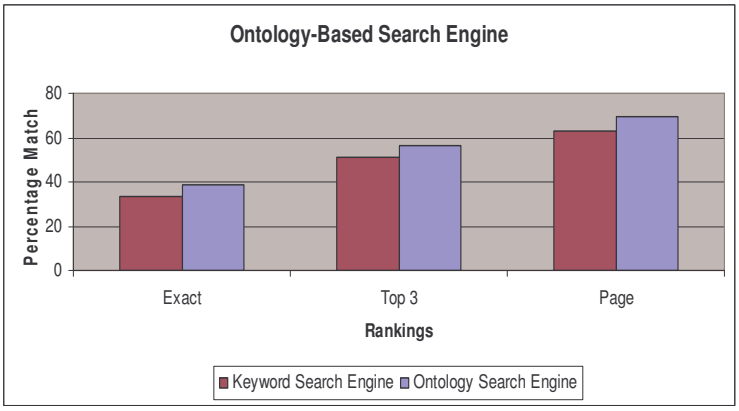

Figure 2. Search engine performance on 114 queries.

Table 1. ontology search engine settings.

Ontology Search Engine Settings	
Ontology depth limit	1
Keywords weight	4
Rootwords weight	3
Category weight	1
Hypernym weight	0.1
Hyponym weight	0.1
Holonym weight	0.1
Meronym weight	0.1
Synonym weight	0.1

Table 2. search engine evaluation results.

Match	Keyword Search	Ontology Search	Increase
Exact	33.33%	38.60%	5.26%
Top 3	50.88%	56.14%	5.26%
Page	63.16%	69.30%	6.14%

Once the search engine had processed the list, a percentage match based on how many of the questions matched exactly, in the top 3, or on the first page was created. The results of these three percentage matches are displayed in graph form in Figure 2 and in numeric form in Table 2. The parameters used for the search engine and ontology to produce these results can be found in Table 1. The results showed that that although neither search engine achieved spectacular accuracy results, largely due to the lack of a true natural language processing system in the search engine, the ontology-based search engine outperformed the keyword-only search by about 5 to 6%.

6. Conclusion

Through the evaluation process, the ontology-based search engine for a decision support system achieved improvements similar to ontology-based search engines in more general domains. The improvement is slightly less than originally hypothesized; the original reasoning was that higher accuracy could be achieved because a specialized domain contains fewer ambiguous terms and the ontology can cover a higher degree of the terminology. However, several issues may be preventing the search engine from achieving higher accuracy. First the search query and information are analyzed individually by words rather than as a phrase, possibly leading to wrong results. Secondly, the ontology terms can have a tendency to overwhelm the search query, resulting in skewed search results. Finally, only semi-normalization is performed on the terms contained in the knowledge, so words that link to a great many different activities have a tendency to return more superfluous results than words that link to a few activities. As part of future research in this area, these areas will need to be addressed. Overall, the results of this research demonstrate the applicability of a search engine that utilizes an ontology in a decision support system.

7. References

1. Brezillon, P. Using Context for Supporting Users Efficiently. Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS '03) , IEEE (2002).
2. Sim, K. M. "Toward an Ontology-Enhanced Information Filtering Agent." SIGMOD RECORD 33, no. No. 1 (March 2004): 95-100.
3. Chandrasekaran, B., J. R. Josephson, and V. R. Benjamins. "What Are Ontologies, and Why Do We Need Them?" *Intelligent Systems and Their Applications, IEEE [see also IEEE Intelligent Systems]* 14, no. 1 (1999): 20.
4. Pinheiro, W. A., and A. M. C. Moura. "An Ontology Based-Approach for Semantic Search in Portals." Paper presented at the Database and Expert Systems Applications 2004.
5. Alexander Pretschner and Susan Gauch. “Ontology Based Personalized Search.” Proceedings of *Tools with Artificial Intelligence, 1999.* (1999): 391-398.
6. Denis O. Gray and S. George Walters. *Managing the Industry/University Cooperative Research Center: A Guide for Directors and Other Stakeholders*. (Columbus: Battelle Press, 1998).
7. Sergio Luján-Mora and Manuel Palomar. “Reducing Inconsistency in Integrating Data from Different Sources,” *Database and Engineering & Applications, 2001 International Symposium* (July 2001): 209-218.