

Guiding People to Information: Providing an Interface to a Digital Library Using Reference as a Basis for Indexing

Shannon Bradshaw, Andrei Scheinkman, and Kristian Hammond

Intelligent Information Laboratory

Department of Computer Science

Northwestern University

1890 Maple Avenue

Evanston, IL 60201, USA

{bradshaw, andrei, hammond}@cs.northwestern.edu

1. ABSTRACT

We describe Rosetta, a digital library system for scientific literature. Rosetta makes it easy for people to find the information for which they are looking even when using short, imprecise queries. Rosetta indexes research articles based on the way they have been described when cited in other documents. The concise descriptions that occur in citations are similar to the short queries people typically form when searching; therefore, citations make a better basis for indexing than do the words used within a research article itself. Using this indexing technique we are able to provide a user interface that presents users with an automatically generated directory of the information space surrounding a query. Our objective with this interface is to present people with the information for which they have asked as well as the information for which they may have intended to ask.

2. Keywords

information retrieval, citation analysis, reference directed indexing

3. INTRODUCTION

In recent years, several digital library projects have established large repositories of scientific literature [6, 8, 9, 18, 20]. These libraries provide coverage of fields such as Medicine, Engineering, and Computer Science by means of research articles published in a variety of books, journals, and conference proceedings.

People access these digital libraries using some sort of keyword-based query interface often provided through the Web. Unfortunately, subject searching is implemented using technology that is not designed to handle the kinds of queries people typically pose to the system. This technology is based on the vector-space model [12]. In the vector-space model, documents are represented by a list or vector of terms selected from among the most frequently used words in the document. Term selection algorithms in these systems are designed to choose terms that uniquely identify a document as much as possible within a given collection. The problem is that this technology was designed to facilitate the comparison of one document to another, but current digital libraries use it as a

means of comparing requests for information (queries) to documents. When a search is performed, the description of the query is compared to term-vectors that represent the information content of documents in the collection. Those documents with term vectors matching the query are retrieved as documents that contain the requested information. However, the documents retrieved are often far from what the user is looking for, because the technique of matching queries to term-vectors is not adequate to handle the type of queries users typically pose when searching. Studies of the searching behavior of people who use digital libraries [8] and other information systems [7, 17] show that people rarely form queries longer than three words while term vectors are usually longer than twenty. As a result, queries and term-vectors are very different types of objects and comparing them is at best an imprecise means of retrieving the requested information. In addition, people rarely use features of the query interface such as the Boolean operator “and” or phrase delimiters such as quotation marks to indicate how they intend query words to be grouped together. Current indexing systems do nothing to resolve the ambiguity introduced by such queries; instead, they simply assume that people will form unambiguous queries, an assumption invalidated by the same studies on searching behavior [7, 8, 17].

In an effort to resolve these problems, we are developing a technique in which research articles are indexed by the way they have been cited in other papers. We are coupling this technology with an interface in which a query retrieves a directory of the information space surrounding the query. This directory is generated automatically from the indices of documents that have been referenced using wording similar to that of the query. Our test-bed for this technology is Rosetta, an indexing and retrieval system for Computer Science research articles.

4. ROSETTA

4.1 References to Research Papers

Citations from one research article to another occur for a variety of reasons. Some citations are made in support of ideas presented; others are made to cite examples of a particular viewpoint. Still others are made to name people with whom the author takes exception. Recognizing and using information about the purpose of a citation is an interesting topic in itself and one that we plan to explore with future research. In our current work; however, we are concerned solely with the feature of citations that ties the different uses together, that being that

the text surrounding a citation (the reference) is usually a concise description of the information the cited document provides. For example, the paper *Agents That Reduce Work and Information Overload* by Pattie Maes is referenced by the following two sentences:

1. "There is also the question of trust, as discussed by Maes (1994)."¹
2. "In addition, users may not trust intelligent agents since they often lack the ability to respond to user requests for clarification (Maes 1994)."²

These references indicate that one facet of Maes' paper is a discussion of the issue of trust between a user and an intelligent agent. Other references indicate that the agents described in this paper are agents that act to assist a person in day-to-day tasks:

1. "...computerized personal assistants which deal with meeting scheduling, e-mail filtering and re-ordering, flight booking, selection of books etc"³
2. "An agent that acts as a personal assistant is called an interface agent".⁴

To the human, these references serve as excellent descriptions of the ideas Maes discusses in her paper. This is because the terms "trust", "intelligent agent", and "personal assistant", are familiar and touch on specific topics, especially when used in combinations such as (trust and "intelligent agents").

References are extremely valuable as a foundation for indexing in an information system, because they pair concise, on-point descriptions of information with the documents that contain that information. As a result, the information system is much better equipped to deal with the brief and often incomplete way people typically describe an information need, because a few words is often enough to eliminate from consideration many irrelevant documents that would be retrieved by standard retrieval techniques based on content.

4.2 Building the Collection

Rosetta indexes a collection composed of the full text of Computer Science research papers as well as publication information for documents not on-line. Thus the collection contains journal articles, conference papers, etc. as well as bibliographic information for other documents such as books which are not usually available on-line. The system gathers research articles from Web and converts them to text using Prescript, a postscript to text converter built as part of the New Zealand Digital Library Project [20]. After a paper is converted to text, Rosetta extracts the title, authors, and bibliography entries. It then locates each citation and pairs it with the

¹ In *Edited Adaptive Hypermedia: Combining Human and Machine Intelligence to Achieve Filtered Information* by Kristina Hook, Asa Rudstrom, and Annika Waern.

² In *Instant TEA - Instant Traveling Expert Advice* by Tod Sedbrook.

³ In *The Evolution of Intelligent Agent and Game Theory: Towards the Future of Intelligent Automation* by N. K. Khoo and Denise J. J. Chen.

⁴ In *Intelligent Agents for Internet-Based Military Training* by Niraj Joshi and V. C. Ramesh.

bibliography entry to which it refers. For each citation, the parser extracts a window of text around the citation to use as a reference to the cited document during indexing. Finally, the paper and each bibliography entry are added to the collection.

4.3 Building Indices from References

Rosetta indexes each document in the collection by the words and phrases used in references to that document. We refer to these words and phrases as terms. Documents may be indexed both by individual terms as well as combinations of terms. So Maes' paper used in the example above would be indexed both by the label "intelligent agents" and by the label ("intelligent agents" and trust).

To index documents Rosetta parses the references to them and extracts "phrases" ranging in length from one to four words. Since queries are typically composed of nouns and noun phrases, Rosetta extracts noun phrases from the references to a document. Noun phrases are extracted using a simple algorithm that employs the Moby Lexicon [19] as means of determining parts of speech. This lexicon contains 230,000 English words and lists the part of speech in which that word is most often used as well as other parts of speech for which it can be used. The noun phrases Rosetta extracts are those in which a noun is modified by pre-noun modifiers such as adjectives and other nouns and/or by post-noun modifiers in the form of prepositional phrases. Examples of these kinds of phrases are: "digital library interfaces" and "interfaces to digital libraries".

For the purpose of extracting noun phrases from references, Rosetta considers as a noun any word that can be used as a noun according to the lexicon and any word in a reference that is not found in the lexicon. Similarly Rosetta considers as an adjective any word that can be used as an adjective. For words that can be used as both a noun and an adjective Rosetta treats the word as a noun. To extract noun phrases from a reference Rosetta steps through the reference evaluating word sequences of length one, then two, and so on up to four. Any sequence consistent with the syntax of a noun phrase is extracted and used as an index for the document described by the reference.

When all single-term indices have been extracted from references, Rosetta then builds additional indices composed of multiple terms. For each document the system finds all two and three term combinations of indices used to label that document and further indexes it using these combinations.

After every document has been fully indexed by both single-term and multi-term indices, the system calculates weights for each index that represent its importance as a descriptor for each document it is used to describe. Index weights are calculated using the following metric, which is based on TFIDF [13], a standard term-weighting measure from the Information Retrieval research community:

$$w_{id} = \frac{n_{id}}{1 + \log N_i}$$

where w_{id} is the weight of index i as a label for document d , n_{id} is the number of times index i was used in reference to d , and N_i is the number of documents for which index i is used as a label.

5. ROSETTA'S USER INTERFACE

Recent studies of the searching behavior of people who use information retrieval systems indicate that people tend to search

for information using remarkably simple queries. Jansen, et al. evaluated the usage of Excite, a popular Web search engine [7]. Excite provides a keyword search interface, in which users list the terms they expect to find in the documents they are seeking. Users may make a query more specific using the Boolean operators “and”, “or”, and “not” and by delimiting phrases using quotation marks. However, in an evaluation of the transaction logs of over fifty thousand queries, the authors found that people rarely make use of query language features. Instead, users typically enter a simple list of keywords as queries. Furthermore, on average these queries are less than three words in length. In a related study of the usage of Excite, Spink, et al. surveyed over three hundred users to collect, among other data, information on search strategies [17]. In this study, Spink et al. found that not one searcher used quotation marks to indicate that some or all of the terms in their query should be interpreted as a phrase, even though many searchers clearly intended that some search terms be interpreted as such. These two studies show that most users of Excite search for information by simply listing two or three terms that touch on the topics of interest. Since Excite is one of the most heavily used Web search engines it is likely that the behavior of users of Excite is representative of searching behavior on the Web in general.

Jones et al. [8] found that this behavior is not limited to the use of Web search engines, but is exhibited by users of other types of information systems as well. This study revealed that less than 30% of people searching the New Zealand Digital Library use Boolean operators and less than 20% pose queries longer than three words.

We can draw two conclusions from the results of these studies. One is that people for the most part do not make use of query language features when searching for information. They enter simple lists of words and expect the information system to interpret them correctly. The other is that most users do not describe information needs in sufficient detail. For some people this is because they are inexperienced in using information retrieval systems. For others it is because they are not sure exactly how to describe the information for which they are looking. This may be due to lack of familiarity with the jargon of a particular field or due to the complexity of the topic of interest. Therefore, to effectively satisfy user requests for information the system must be able to interpret the query as the user intended it. It should also guide the user to those documents that satisfy an information need that is not specified in sufficient detail.

5.1 Searching for Information in Rosetta

People typically search for information by simply listing the terms that form the words and phrases that describe an information need. Therefore, the information system must infer the intended parsing of the query and respond accordingly. Rosetta accepts queries in natural language and responds based on the most reasonable interpretations of the list of words that compose a query.

When a query is posed to Rosetta, it generates all possible interpretations of the query by parsing the query into words and phrases (terms). For example, given the query:

intelligent agents trust

Rosetta finds the following parses:

1. “intelligent agents trust”

2. “intelligent agents”, trust
3. intelligent, “agents trust”
4. intelligent, agents, trust

Phrases are the best tools for disambiguating one topic from another, because they are a commonly used by people for the same purpose. Therefore, Rosetta uses the phrases found in the various interpretations of a query as the basis for searching for information. The system sorts the query interpretations by the length of the longest phrase they contain and then steps through the resulting list using each as a query to the retrieval system until either a match is found or every query interpretation has been tried. If a match is found, the system tries the remaining query interpretations having a longest phrase of the same length as the one that matched. This is done to collect all matches at a given level of specificity.

In the retrieval system, matches to queries can be either complete or partial. A complete match is one in which an entire query matches some document index. A partial match occurs when one or more terms in a query match some index in the database. For example, a partial match for the query,

“intelligent agents”, trust

is any document indexed by the label,

“intelligent agents”.

To find a match for a query the system first tries to find the entire query as a document index. If such an index is found it is returned as the match for that query. If not then the system extracts from the query all partial queries containing at least the longest phrase found in that query. If more than one phrase in a query shares the distinction of longest phrase then the system extracts all partial queries containing at least one of those phrases. The list of partial queries is sorted based on the number of terms in each. The system steps through the resulting list until either a match is found or the list has been exhausted. If a match is found, Rosetta attempts to match the remaining partial queries containing the same number of words as the matching partial query. The matching indices are then returned as matches for the query.

Words used together in queries and in document indices serve to disambiguate the information described, especially when used in phrases. So by sorting query interpretations by the length of the longest phrase they contain, and partial queries by the number of terms they contain, we make the assumption that the first group of matches the system finds are the best matches for the query found in the collection.

5.2 Finding Supplementary Information

For many queries simply finding those documents which best match the information need as stated is not enough, because the information need has not been described in enough detail. For these queries it is necessary to guide the user to a more detailed description of the information for which he is looking. To do this Rosetta automatically generates a directory of the information space surrounding the information requested in the query. In Rosetta each index is further indexed by the terms of which the index is composed. For example, the index (“intelligent agents” and “personal assistant”) is itself indexed by the terms “intelligent agents” and “personal assistant”.

To build a directory Rosetta searches the index database for indices composed of some term matched by the query and at

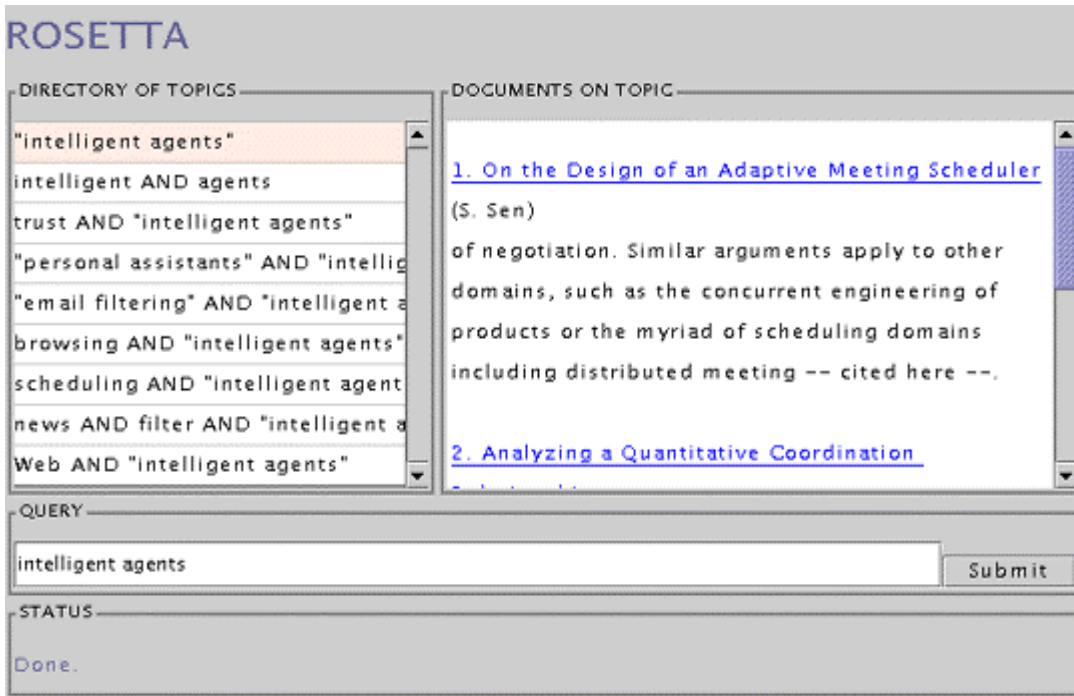


Figure 1. Rosetta interface after response to query: intelligent agents.

least one other term. So for the query “intelligent agents” indices such as (“intelligent agents” and “personal assistant”) and (“intelligent agents” and trust) would be included in the directory. The directory differentiates between several related topics touched on by the query. Rosetta presents these topics together with the possible parses of the query as a list of labels similar to those found in the Yahoo! Web search engine. Users may browse the directory by selecting a label and viewing the related documents.

5.3 Presenting Search Results

Rosetta presents search results to the user in a format that should be familiar to anyone who has used the Web. (See Figure 1.) The topic labels activated by a query, both those matched by the query itself as well as those labeling related topics, are presented as a directory at the left of the results page. For each label, the top twenty matching documents are selected on the basis of the weight calculated at indexing time. The labels matched by the query are listed first and are sorted according to weight of the index as a label for the top document among the twenty retrieved for that index. The related topics are listed next and are sorted by the frequency with which they are used in the collection.

To the right of the directory, the documents indexed by the first label on the list are displayed. For each document, the title and authors are listed as well as a sample of the references to that document. The references provide a clear picture of how people write about a document and therefore, help the user to quickly decide which documents will satisfy his information need. Similarly, the labels listed in the directory help the user to find the group of documents most likely to contain the information for which he is looking. The user can switch to another group of documents by simply selecting the label describing that group (See Figure 2). The topic labels serve to suggest to the user how

information is indexed in the system. If the first query is unsuccessful they may point him to alternative ways of describing the needed information to retrieve the documents of interest.

While we have not yet performed any serious user study for this interface we believe it will prove to be a valuable research tool for three reasons. First, by suggesting less ambiguous topics this interface helps users find information sought even in the face of poorly formed queries. Second, it helps users understand how their topic of interest fits into a body of research. Finally, by browsing the information space previously unknown work can be easily discovered.

6. SYSTEM EVALUATION

We have performed a preliminary evaluation of Rosetta’s indexing system coupled with a simple user interface. We present here the precision with which Rosetta is able to satisfy a query in the top twenty documents retrieved. For this test the query results were composed of the top twenty documents matching some combination of the words in the query. Forthcoming is a more detailed evaluation of Rosetta.

6.1 Experiment Design

Ideally, an evaluation of Rosetta would be an analysis of system performance when deployed as a publicly available research tool. Unfortunately, the system is not yet mature enough for release. Therefore, we evaluated Rosetta in an environment that simulates the kind of use we expect the system to receive. We selected several abstracts from papers in the collection and asked eight graduate students in Computer Science to use Rosetta to find the papers those abstracts described. The collection used in this evaluation includes over ten thousand papers and bibliography entries indexed using over twenty thousand references.

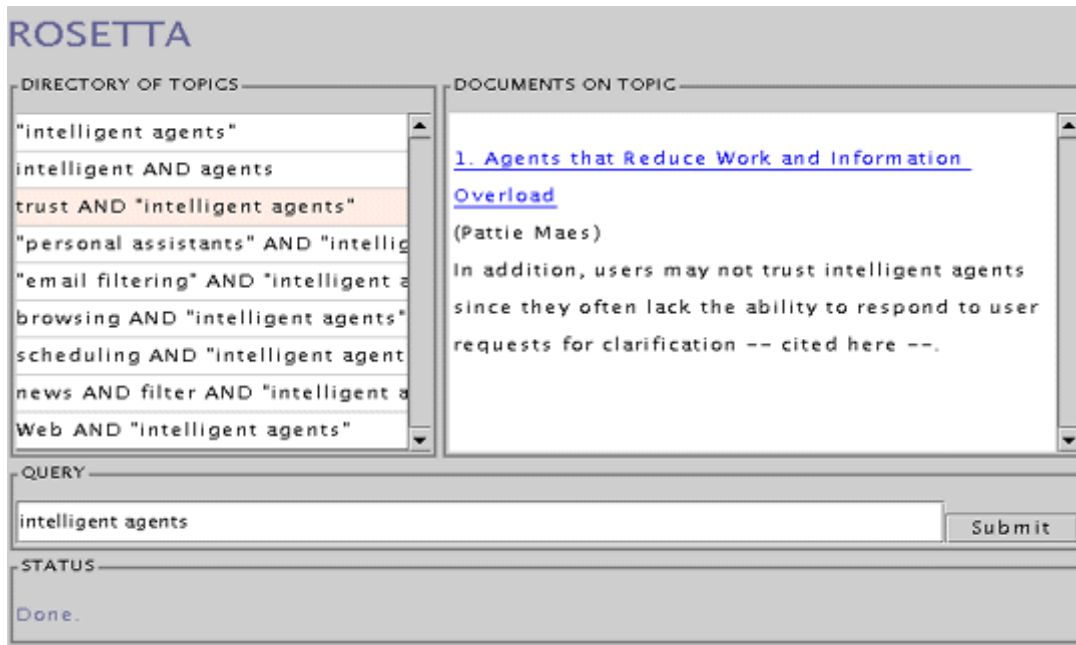


Figure 2. Selection of the topic: trust and “intelligent agents”.

We set up the experiment as follows: Forty papers were selected at random from Rosetta's collection. From this group of forty, we selected twenty abstracts that we deemed comprehensible and informative enough to present a clear picture of what the paper is about. Each of the abstracts was assigned a number and eighteen were randomly assigned to one of six groups. Two of the abstracts, selected at random, were assigned to every group, making six groups of five abstracts each. The title and any terms coined by the author (e.g. “Rosetta”) were eliminated from each abstract to force the users to query based on the topics mentioned in an abstract.

Each of the eight users was given one group of five abstracts. For each abstract, the user was asked to find that paper or a paper on the same topic by the same author using Rosetta. The users were asked to find each paper on the basis of the abstract alone (use of author names was not allowed) and to use no more than three queries in searching for each paper. The users gauged whether or not they had found the paper using the author names and the example references presented with each paper in the results. They were asked to answer three questions about the query session for each abstract in the group they were given:

1. Did you find the paper?
2. What queries did you use? (no more than three)
3. What number was the paper ranked in the results?

When the users had finished testing, we double-checked their results. Three users mistakenly thought they had found one of the papers when in fact they had not. This happened on the first query attempt in each case, and the users did not try other queries in an effort to find the paper. Therefore, we dropped these three query sessions from the evaluation. So for the evaluation we have the results from eight users in a total of thirty-seven query sessions.

6.2 Evaluation Results

Our objective in performing this evaluation was to measure the performance of Rosetta along two dimensions: precision over a variety of topics and precision on the same topic for a variety of users. To test Rosetta's ability to perform consistently well across a range of topics we calculated the average ranking assigned to each paper in searches by the users. We found that for an average of 1.26 queries per paper, Rosetta retrieved and ranked the correct paper in the top twenty for 70.0% of the papers, in the top fifteen for 60.0% of the papers, and in the top ten for 55.0% of the papers. See Figure 3 for a graph of these results.

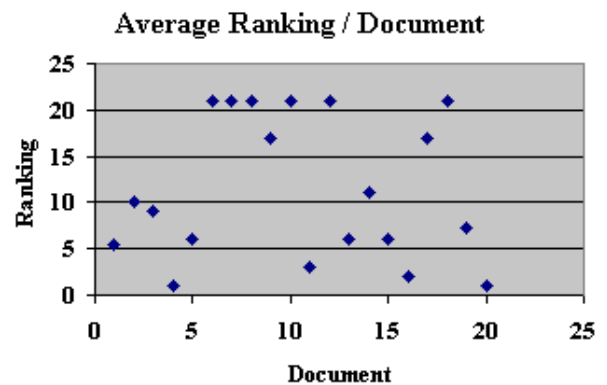


Figure 3. Average Ranking of Each Document

To test the performance of Rosetta for a variety of users searching for the same information, we compared the results of each user in searching for the two papers (numbers 13 and 19)

assigned to every group. For each of these papers every user was able to find it in the first attempt. In queries for paper 13, Rosetta ranked the correct paper in the top ten for 100% of the queries and in the top five for 50% of the queries. In searches for 19, Rosetta ranked the paper in the top fifteen for 100% of the users, in the top ten for 75% of the users. (See Figure 4).

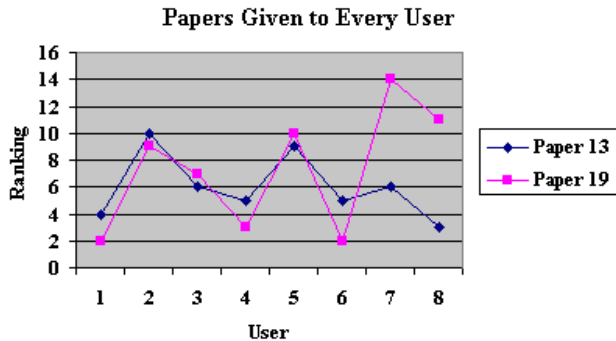


Figure 4. Ranking of papers 13 and 19 in queries by every user.

We received some promising results from this experiment. Rosetta was able to locate the needed paper in the top twenty or what would be the first two pages of results for 70% of the papers and in the first page of results for 55% percent of the papers. In addition, Rosetta showed consistently good performance for several users searching for the same information.

7. RELATED WORK

The value of using citations in documents has been explored extensively in IR research. (See [11], [15], and [21] for examples.) However, this work has been concerned mainly with what can be inferred about the similarity of documents based on an analysis of citation networks. One intuition supporting this research is that the similarity of two documents varies directly as the number of documents that cite both of them.

Within the past few years, researchers in Artificial Intelligence (AI) have done work that exploits the hyper-link structure of Web pages. Spertus uses heuristics to determine the relationship between the topics of two pages based on the type of link that exists between them [16]. She describes one such heuristic as the following: "starting at an index, any page reached by following a single outward link is likely to be on the same topic."

In other work, referential text has been used as a means of summarizing a document when presenting query results to a user of an information retrieval system. In [10], the authors describe using the text surrounding hyperlinks to Web pages as descriptions of those pages when presenting search results to users of Lycos. Similarly, Citeseer [6], a research paper indexing and retrieval system presents results to queries with a list of citations describing each document retrieved. In this respect Rosetta is similar to Citeseer; however, Citeseer indexes documents by content using an indexing system based on the vector-space model and uses a Boolean query interface.

The use of referential text as a basis for indexing has not been explored to a great extent. One exception is the Google Web search engine [2]. Google indexes a Web page using the text enclosed in anchor tags that define a hyperlink to it on other Web pages. The value of a word as an index for a Web page is determined by the importance of pages that link to it using that word in the hypertext. A page is considered important if a large number of pages have a hyperlink to it or if a few important pages have a hyperlink to it. As a result, queries to Google retrieve popular sites that have been described using the same words.

The issue of adding context to ambiguous queries has been explored in a variety of systems; however, these systems are embedded in a document creation or browsing task and are not queried in the absence of such a task. One system is a version of Rosetta embedded in Emacs for use with the LaTeX document preparation system [1]. In this version of Rosetta context is gathered from the user as he types a document. The system retrieves supporting documents on topics related to those discussed in a small window of text surrounding the cursor. Another system is Watson [3]. In Watson, Budzik has developed a similar tool for use Microsoft Word; however, Watson searches for related documents using existing Web search engines. Watson automatically generates queries from the text of a document that are similar to the term vectors used to index Web pages.

8. CONCLUSION

Our work with Rosetta suggests that using reference as a basis for indexing is an effective approach to building searchable digital libraries of scientific literature. Using this technique Rosetta is able to provide users with the documents they need in response to simple queries. Furthermore, the labels for documents that Rosetta extracts from references make it possible to automatically construct an interface in which users can browse the information space when in doubt as to how to query for a given piece of information or when simply poking around in the collection. These features make Rosetta a valuable research tool with which needed information is easily found.

9. REFERENCES

- Bradshaw, S. Reference Directed Indexing: Attention to the Description People Use for Information. Masters Thesis. The University of Chicago. 1998.
- Brin, S. and Page, L. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Proceedings of WWW '98* (Brisbane Australia, April 1998).
- Budzik, J., and Hammond, K. J. Watson: Anticipating and Contextualizing Information Needs. *Proceedings of the Sixty-second Annual Meeting of the American Society for Information Science*. Learned Information, November, 1999.
- Chen, H., Schatz, B., Ng, T., Martinez, J., Kirchhoff, A., and Lin, C. A parallel Computing Approach to Creating Engineering Concept Spaces for Semantic Retrieval: The Illinois Digital Library Initiative Project. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Special Section on Digital Libraries:

- Representation and Retrieval 18, 8 (August 1996). 771-782.
5. Davis, J. R. Creating a Networked Computer Science Technical Report Library. *D-lib Magazine*, September, 1995.
 6. Giles, C. L., Bollacker, K., and Lawrence, S. CiteSeer: An Automatic Citation Indexing System. *Proceedings of Digital Libraries '98* (Pittsburgh PA, June 1998). ACM Press. 89-98.
 7. Jansen, B.J., Spink, A., Bateman, J., Saracevic, Tefko. Searchers, the Subjects they Search, and Sufficiency: A Study of a Large Sample of Excite Searches. *Proceedings of Webnet '98* (Orlando FL, November 1998). 472-477.
 8. Jones, S., Cunningham, S. J., and McNab, R. An Analysis of Usage of a Digital Library. *Proceedings of ECDL '98* (Heraklion Crete Greece, September 1998).
 9. Lawrence, S., Giles, C. L., Bollacker, K. Digital Libraries and Autonomous Citation Indexing. *IEEE Computer* 32, 6. 67-71. 1999.
 10. Mauldin, M. and Leavitt, J. R. R. Web Agent Related Research at the Center for Machine Translation. *Proceedings of the ACM Special Interest Group on Networked Information Discovery and Retrieval*. 1994.
 11. Marshakova, I. V. System of Document Connections Based on References (in Russian). *Nauchno-Tekhnicheskaya Informatsiya, ser. 2, 6.* 3-8. 1973.
 12. Salton, G., Wong, A., and Yang, C. S. A Vector Space Model for Automatic Indexing. *Communications of the ACM* 18, 11. 613-620. 1971.
 13. Salton, G., and Buckley, C. Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*. 24, 5. 513-523. 1988.
 14. Schatz, B. R., Johnson, E. H., and Cochrane, P.A. Interactive Term Suggestion for Users of Digital Libraries: Using Subject Thesauri and Co-occurrence Lists for Information Retrieval. *Proceedings of Digital Libraries '96* (Bethesda MD, March 1996). ACM Press. 126-133.
 15. Small, H. Co-citation in the Scientific Literature: A New Measure of the Relationship Between Two Documents. *Journal of the American Society for Information Science*. 24. 265-269. 1973.
 16. Spertus, E. ParaSite: Mining Structural Information on the Web. *The Sixth International World Wide Web Conference*. 1997.
 17. Spink, A., Jansen, B. J., Bateman, J. Users' Searching Behavior on the Excite Web Search Engine. *Proceedings of Webnet '98* (Orlando FL, November 1998). 828-833.
 18. The U.S. National Library of Medicine. <http://www.nlm.nih.gov/nlmhome.html>.
 19. Ward, Grady. A set of lexical resources. <http://www.dcs.shef.ac.uk/research/ilash/Moby/>
 20. Witten, I. H., and McNab, R. The New Zealand Digital Library: Collections and Experience. *The Electronic Library* 15, 6. 495-503.
 21. Yaru, D. Structural Modeling of Network Systems in Citation Analysis. *Journal of the American Society for Information Science*. 48, 10. 946-952. 1997.