Exploiting Rich Context: An Incremental Approach to Context-Based Web Search*

David Leake, Ana Maguitman, and Thomas Reichherzer

Computer Science Department, Indiana University, Lindley Hall 215
150 S. Woodlawn Avenue, Bloomington, IN 47405, U.S.A.
{*leake, anmaguit, treichhe*}@*cs.indiana.edu*

Abstract. Proactive retrieval systems monitor a user's task context and automatically provide the user with related resources. The effectiveness of such systems depends on their ability to perform context-based retrieval, generating queries which return context-relevant results. Two factors make this task especially challenging for Web-based retrieval. First, the quality of Web retrieval can be strongly affected by the vocabulary used to generate the queries. If the system's vocabulary for describing the context differs from the vocabulary used in the resources themselves, relevant resources may be missed. Second, search engine restrictions on query length may make it difficult to include sufficient contextual information in a single query. This paper presents an algorithm, IACS (Incremental Algorithm for Context-Based Search), which addresses these problems by building up, applying, and refining partial context descriptions incrementally. In IACS, an initial term-based context description is the starting point for a cycle of mining search engines, performing context-based filtering of results, and refining context descriptions to generate new rounds of queries in an expanded vocabulary. IACS has been applied in a system for proactively supporting concept-map-based knowledge modeling, by retrieving resources relevant to target concepts in the context of the rich information provided by "in progress" concept maps. An evaluation of the system shows that it provides significant improvements over a baseline for retrieving context-relevant resources. We expect the algorithm to have broad applicability to context-based Web retrieval for rich contexts.

1 Introduction

Many systems have been developed to aid users as they work, by performing automatic Web search for information to support tasks such as Web browsing, query generation, and document authoring, by mining the Web and other resources (e.g., [1, 2]). Reflecting context has long been recognized as important to realizing the potential of Web search in general [3], and context-sensitivity plays an especially crucial role in proactive retrieval systems: The extent to which the system can provide context-relevant information determines whether the system will be an aid or an annoyance. Unfortunately, fully

* This material is based upon work supported by NASA under award No NCC 2-1216. We would like to thank our collaborators Alberto Cañas and the IHMC CmapTools team for their many contributions to this project.

exploiting contextual information during Web search is challenging. In current search engines, there are strong limits on query length (e.g., Google's query length limit of ten terms), making it difficult to provide enough terms to describe rich contexts. Even if an adequate context description can be included within the limits, there is no guarantee that the vocabulary used to describe the context will match the vocabulary by which the resource is indexed.

This paper describes an approach which simultaneously addresses the problems of overcoming the variations in term-based context descriptions and reflecting rich context when mining search engines. It presents IACS (Incremental Algorithm for Context-Based Search), an algorithm which takes a novel incremental approach to mining search engines for context-relevant textual resources (such as html pages, pdf files, Word files, etc.), in light of continually refined context descriptions. IACS uses a cycle of characterizing context, generating search engine queries, performing context-based filtering of the results, and refining the context descriptions to emphasize terms discovered to be important, in order to describe the context for new rounds of queries and to accumulate resources relevant to the context as a whole.

We have tested our approach in the domain of proactive support for knowledge modeling. For some time, we have been investigating the development of intelligent support systems for aiding knowledge capture using concept maps, in collaboration with the CmapTools team at the Institute for Human and Machine Cognition [4]. Concept mapping has been extensively used for knowledge construction and sharing in education, and for the capture of expert knowledge by the experts themselves. Part of the CmapTools project focuses on facilitating this knowledge capture by generating context-relevant suggestions and aiding context-relevant search, to help the user decide which concepts to include in a concept map, to identify propositions to include about those concepts, and to find relevant resources to link to the current knowledge model [5–7]. When users request suggestions relevant to a selected concept in a concept map, the surrounding knowledge model—which may include hundreds of concepts—provides a rich source of contextual information to exploit during retrieval. IACS starts from this information and combines it with context-relevant information gathered incrementally to determine new query terms, extending the retrieval vocabulary beyond the terms in the concept map. Thus IACS mines search engines for resources at the same time it incrementally formulates and refines a context description to improve future search results.

The paper begins by examining the role of context in concept maps and presenting our goals for a proactive, context-relevant resource suggestion system to aid concept mapping. It next presents the IACS algorithm itself, followed by an evaluation comparing its performance to a baseline, non-incremental method. These results suggest that IACS provides significant improvements, both in terms of maintaining focus on context-relevant resources (measured by a generalization of precision) and in terms of retrieving resources providing good coverage of the context (measured by a generalization of recall). Because the algorithm itself relies only on the availability of a set of terms characterizing the context, and does not depend on any specific properties of concept maps, we consider the approach promising for exploiting rich contexts for other retrieval tasks as well.

2 Concept Maps and Concept Mapping

Concept maps [8, 9] are collections of propositions (simplified natural language sentences) displayed as a two-dimensional, visually-based representation of concepts and their relationships. Concept maps depict concepts as labeled nodes and inter-concept relations as labeled links, as illustrated in the sample concept map “Mars myth and science fiction” shown in Figure 1. Unlike semantic networks and other graph-based structures commonly used in artificial intelligence to perform automatic reasoning on the encoded knowledge, concept maps are “informal” knowledge representations that facilitate knowledge capture for human examination and sharing and enable students to learn “meaningfully” by connecting concepts held in long-term memory with new concepts and propositions.

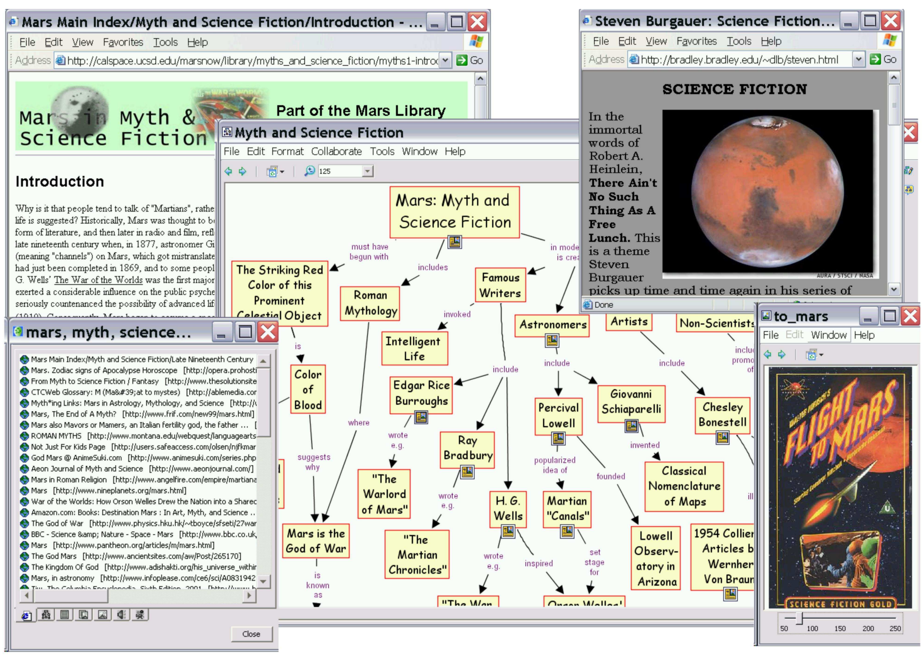

Fig. 1. The CmapTools Interface with the IACS resource suggestion window and related resources.

Concept mapping is widely used in educational settings, in which teachers assign students to draw concept maps to encourage them to organize their knowledge and to make their understanding explicit for knowledge assessment and sharing. Studies show that students in a wide range of age groups, as early as in elementary school, can generate concept maps successfully. The naturalness of the concept mapping process makes it promising as a method for direct knowledge capture by experts themselves, and the conciseness and structure of concept maps assists understanding the captured information.

To facilitate electronic concept map construction and sharing, the Institute for Human and Machine Cognition (IHMC) has developed CmapTools, publicly-available tools to support generation and modification of concept maps in an electronic form [4]. The CmapTools software enables interconnecting and annotating maps with material such as other concept maps, images, diagrams, and video clips, providing rich, browsable knowledge models available for navigation and collaboration across geographically-distant sites. CmapTools has been used for numerous projects including a large-scale initiative in modeling and sharing the knowledge of NASA experts on the planet Mars [10]. Figure 1 illustrates the interface's display of a sample concept map from that domain.

2.1 Adding Intelligent Suggesters

A goal of the CmapTools initiative is to empower experts to construct knowledge models of their domains without the need for a knowledge engineer's intervention, or to actively participate in knowledge modeling led by a knowledge engineer. While users find the interface itself natural and intuitive, part of the challenge of concept mapping is to determine the "right" concepts and relationships to include in the concept map. Informal studies show that users building concept maps often stop for significant amounts of time, wondering how to extend their models, and in some cases searching the Web to jog their memories or find new material to link to the current map. To support this process, a current effort augments the CmapTools interface with a family of "intelligent suggesters" to start from a concept map under construction, and propose context-relevant information to aid the user's knowledge capture and knowledge construction [11]. This paper focuses on one of those suggesters, a system that explores external resources on the Web to find related text documents that can be linked to the concept map or examined for additional information to be included into the concept map.

2.2 Contexts for Concept Mapping

In formal methods for knowledge capture, a goal is to associate each expression with a unique, context-independent meaning. Considerable effort and expertise may be required to train people to capture knowledge in such carefully-crafted forms. On the other hand, concept mapping tools are intended for "human-centered" knowledge capture, in which people express their knowledge informally, without a controlled vocabulary. Concept maps offer no assurance of unambiguous labels, but instead rely on the rich context of the rest of the map for disambiguation. For example, the concept label "Mars" might designate the planet Mars, the god Mars from mythology, or the Mars candy bar; the relevant meaning would be suggested by the context in which it was found. Consequently, to develop a suggester that retrieves resources relevant to a concept label, it is necessary for retrieval to reflect that concept's context in the knowledge model.

In concept map-based knowledge models, each concept can be seen as contained within several layers of context, as illustrated in Figure 2. We define the inner-most context layer C_1 of a target concept to be all concepts directly linked to the target

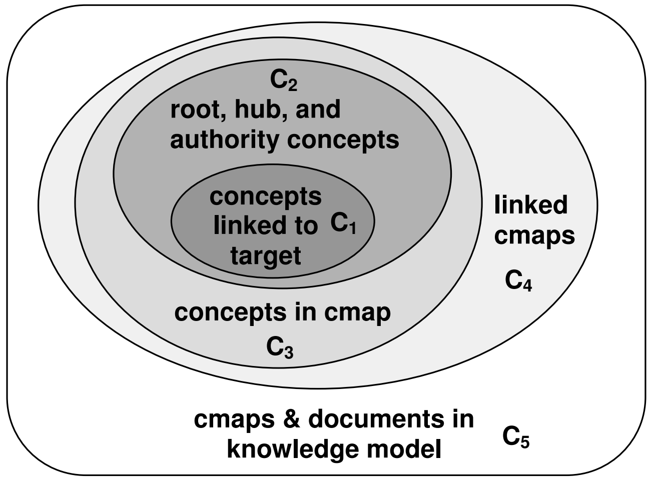

Fig. 2. Layers of context for a concept in a concept map.

concept in the concept map graph. This is the set of all concepts participating in propositions in which the target concept is directly involved. The second layer C_2 adds other concepts that play a key role in describing the topic of the concept map as a whole. In previous research, we developed and tested a set of candidate models for predicting topic-important concepts, to select a model to use for weighting the importance of particular concept labels in generating a topic description [12, 13]. Our models assess each concept's role in describing a topic, based on the topological structure of the map. Human-subjects experiments showed a statistically-significant agreement between the predictions of our best model and the actual judgments of subjects who predicted concept map topics from the maps' structure.

For humans, the root concept, typically located at the top of a map, serves as a starting point to explore a map, thus providing a first hint as to what the map discusses. Important concepts for describing the context of a target concept in a concept map include the root concept and concepts with many incoming or outgoing links (*authority* and *hub* concepts, respectively).

Next, the layer C_3 of a concept's context is the set of all the concepts connected to the target concept within the boundary of the concept map. Fully developed concept maps contain a well connected set of concepts in which each concept is explained in terms of the relations and concepts directly connected to the concept. Because interpretations of each of these concepts are influenced by their own connections, any concept in the map may influence interpretation of the target concept.

The CmapTools software enables concepts to be linked to other concept maps, with a link analogous to a Web link, enabling users to jump from one map to another. However, unlike links in Web documents, links in concept maps may also allow users to navigate to the same concept discussed in different maps, with each map providing a different context for the concept. For example, the concept "rocket engine" may occur in a map on rocket architecture or in a map on rocket propulsion systems. Layer C_4

reflects this, extending the context of layer C_3 by also considering all the concepts in concept maps that are directly linked to the target concept. Finally, layer C_5, the most general context of a target concept, is the entire knowledge model consisting of a set of concept maps and annotations such as text documents, images, or other multi-media resources. All the concepts in the concept maps of a knowledge model share the same C_5 context.

Each of the layers of context could influence human judgments of resources' relevance to a target concept. Our current work focuses on exploiting information extracted automatically from C_2, in order to provide the user with suggestions of resources relevant to that context. IACS is applied in a system which describes contexts using a weighted set of terms, with term weights reflecting estimates of the terms' importances to characterizing the context. Initially, term weights for a given concept map are computed based on the structural analysis methods summarized previously. The next section describes how this initial context description can be incrementally refined and used to focus retrieval as new relevant material is retrieved from the Web.

3 An Incremental Strategy for Exploiting Rich Context

A limitation of current search engines is their restriction on query length, enabling only a small set of terms to be contained in any query. Consequently, to take advantage of the rich contextual information provided by a knowledge model, incremental approaches are needed to allow multiple queries to build up context-relevant information. In an incremental approach to Web search, contextual information can help to guide the exploration and discovery of relevant resources both at the moment a query is constructed (pre-query stage) and after an initial set of results have been obtained (post-results stage).

To retrieve resources relevant to a target concept in a concept map, IACS exploits the rich context of a surrounding concept map in three ways. First, it uses terms extracted from the concept map context C_2 to augment the initial search engine query. This is achieved by analyzing the concept map, identifying important terms and ranking them using the topological analysis methods sketched in section 2.2. The most highly-rated candidate terms are added to the terms of the concept label, reflecting the context in which the label occurs. This enables the use of limited context, but because of query length limits, few terms can be included, so it provides a coarse-grained starting point. Second, the context of the concept map is exploited after the initial set of results has been obtained, for filtering irrelevant material and ranking retrieved results based on their estimated context-relevance. This enables the rich context to help select relevant material.

Third, IACS exploits the context to generate new queries that go beyond the initial query, and that may even go beyond the vocabulary of the initial concept map. After the first set of results has been obtained, the search context is used to refine/extend the set of terms used for the context description. Terms that appear "often" in search results similar to the context tend to be good *descriptors* of the user's information needs. In addition, because these descriptors are expected to occur in a large fraction of the relevant material, they are useful as query terms when high recall is desirable. Likewise, terms that tend to occur "only" in results similar to the search context can serve as *discriminators*. When used as query terms, topic discriminators help restrict the set of search results to mostly similar material and therefore can help achieve high precision. A formal characterization of topic descriptors and discriminators as well as an evaluation of their usefulness as query terms can be found in [13].

IACS identifies topic descriptors and topic discriminators by analyzing the terms in retrieved documents. Consequently, descriptors and discriminators are not restricted to terms occurring in the originating search context, and if novel terms have high descriptive or discriminating power, they expand the initial vocabulary used to describe the context. Therefore, while the initial context only reflects the vocabulary of the originating concept map, new terms weighted as a function of their descriptive and discriminating power will be incrementally added to the search context. In IACS's incremental search process, the generation of second-round and subsequent queries can significantly benefit from a search context refined by the addition of good descriptors and discriminators.

Table 1 presents an outline of the incremental algorithm for context-based search. The algorithm starts by applying topological analysis to a concept map to identify the most salient terms in the map. These terms define the initial search context, which is used to start the incremental Web search and context expansion/refinement process. Terms in the retrieved results are analyzed in light of the search context to refine the search context description, and the highest-ranked terms in the search context are used as query terms in subsequent Web queries.

For efficiency, IACS bases its processing on the short "snippets" of text returned for each page in the search engine results summary, rather than full pages. Results are filtered and weighted according to context. Filtering is done by comparing the set of keywords occurring in the snippets against the set of keywords associated with the current context. If the cosine similarity between the two sets is above a threshold (defined in terms of a "curiosity mechanism" described in detail in [14]) the results are added to the set of relevant material. Terms found in the search results are weighted according to their descriptive and discriminating power and used to refine the search context. The extended search context is clustered by a soft term clustering algorithm which we developed to facilitate the generation of cohesive queries in subsequent iterations [15]. Soft clustering algorithms generalize hard clustering algorithms by allowing cluster overlap (i.e. the same term may be part of more than one cluster). After all iterations have been completed, the collected search results are cleaned to eliminate redundancies and sorted and returned to the user.

4 Evaluation

4.1 Evaluation Criteria

To evaluate the performance of context-based retrieval for supporting concept mapping, we first had to develop evaluation criteria suitable for this task. We developed two criterion functions for evaluating retrieval performance: *global coherence* and *coverage* [14].

PROCEDURE INCREMENTAL CONTEXT-BASED SEARCH
INPUT:
  M: source concept map
  s: number of iterations.
  n: number of search queries.
OUTPUT:
  A ranked list of resources related to **M**
BEGIN
  Use topological analysis to weight terms in M.
  Generate a set C of weighted terms (initial search context).
  $T[0] = \{C\}$ % $T[i]$ *is a set of sets of weighted terms.*
  $R = \emptyset$ % *Search results*.
  for (i=0; i $<$ s; i++)
  do
    $T[i+1] = \emptyset$.
    for each set of terms $C \in T[i]$
    do
      Use the most important terms in C to form n search queries.
      Submit queries to a search engine.
      Use C to filter results and add them to R.
      Compare search results to C to identify best descriptors and discriminators.
      Weight terms as a function of their descriptive and discriminating power.
      Use best descriptors and discriminators to expand C.
      Use C to generate a set N of overlapping term clusters.
      T[i+1]= T[i+1] $\cup$ N.
    end do
  end do
  Clean and sort R.
  return R.
END

Table 1. Pseudocode of the incremental algorithm for context-based search.

These two functions generalize the well known IR measures of precision and recall. However, in contrast to precision and recall, the measures of global coherence and coverage do not require that all relevant resources be precisely identified. Instead, these measures are applicable as long as an approximate description of the potentially relevant material is available. The relaxation of the requirement of a precise set of relevant resources makes these novel criterion functions suitable for the evaluation of context-based search on the Web, where a precise characterization of relevant resources is usually unavailable.

Assume $R = \{r_1, \ldots, r_m\}$ is a set containing approximate descriptions of potentially relevant material, where each r_i is a collection of keywords. Let $A = \{a_1, \ldots, a_n\}$ be the set of retrieved resources, with a_i also represented as a collection of keywords. A measure of *similarity* between a retrieved resource a_i and a relevant r_j can be computed using, for example, the *Jaccard coefficient*, defined as:

$$\textbf{Similarity}(a_i, r_j) = \frac{|a_i \cap r_j|}{|a_i \cup r_j|}.$$

Then, we can define the *accuracy* of resource a_i in R as follows:

$$\textbf{Accuracy}(a_i, R) = \max_{r_j \in R} \textbf{Similarity}(a_i, r_j).$$

When measuring the accuracy of a retrieved resource a_i, we obtain an estimate of the precision with which the terms in a_i replicate those of relevant resources.

We use the **Accuracy** function to define **Global_Coherence** as follows:

$$\textbf{Global\_Coherence}\ (A, R) = \frac{\sum_{a_i \in A} \textbf{Accuracy}(a_i, R)}{|A|}.$$

The **Global_Coherence** function measures the degree to which a retrieval mechanism succeeded in keeping its focus within the theme defined by a set of relevant resources. This is similar to the IR notion of precision, except that we use a less restrictive notion of relevance.

We note that a high global coherence value does not guarantee acceptable retrieval performance. For example, if the system retrieves only a single resource that is similar to some relevant resource, the global coherence value will be high. Because context-based suggesters should also maximize the number of relevant resources retrieved, we introduce a *coverage* factor to favor those strategies that retrieve many resources similar to a target set of relevant resources. We define a criterion function able to measure coverage as a generalization of the standard IR notion of recall:

$$\textbf{Coverage}\ (A, R) = \frac{\sum_{r_i \in R} \textbf{Accuracy}(r_i, A)}{|R|}.$$

4.2 The Performance Evaluation

A performance evaluation based on our criterion functions requires access to a set of terms taken to characterize the relevant resources (a target set R). For our task of suggesting information relevant to a concept-map-based knowledge model, we can define such a set based on an existing corpus of concept maps as follows.

Let $K = \{c_1, \ldots, c_m\}$ be a concept-map-based knowledge model, where each c_k is a set of keywords representing a concept map. Suppose c is a concept map in K and c is used for context-based retrieval. If the knowledge model K has been built by a reliable source and is sufficiently extensive, then, for evaluation purposes, the set K could act as a surrogate for R, the set of relevant resources. In our evaluations we use an expert-generated knowledge model on the Mars domain as our "gold standard" [10]. This knowledge model contains 118 concepts map, presenting an extensive description of the Mars domain.

In our tests the top-level concept map from the Mars knowledge model was used as the starting point (corresponding to the concept map under construction, for which related suggestions were sought) and IACS was used to search for resources on the Web, without access to any of the other maps in the knowledge model. As a baseline method for comparison, we implemented a simple non-incremental algorithm which constructs queries from the concept labels of the same concept map used as IACS's starting point, after stopword elimination. It submits these as individual queries to the Google Web API. For each query submitted by IACS, the baseline creates a query of equal size, using terms extracted from concept labels selected randomly from the source map. The baseline's queries include full concept labels when possible, but may use subsets to reduce query size or terms from additional concept's labels when needed, in order to assure that neither method benefits from differences in query length. In contrast to IACS's incremental approach, the baseline constructs all its queries using terms that occur in the originating concept map. We expected IACS's incremental mechanism to provide results with superior global coherence and coverage for equal number of Web queries. When comparing the performance of our incremental search strategy against the baseline, we set the number of iterations to 3. Our evaluation involved 48 trials. Figures 3(a) and 3(b) compare the performance of the IACS algorithm to the baseline method in terms of global coherence and coverage. Each trial is represented by a point. The point's vertical coordinate corresponds to the performance of IACS for that trial, while the horizontal coordinate corresponds to the performance of the baseline method. The trials in which IACS outperforms the baseline can be identified as those points above the diagonal.

Method	N	MEAN	STDEV	95% C.I.
IACS	48	0.086	0.045	**(0.073, 0.099)**
Baseline	48	0.036	0.021	**(0.030, 0.042)**

Table 2. Confidence intervals for the mean global coherence of the incremental algorithm for context-based search (IACS) and baseline.

In Tables 2 and 3 we present the number of trials (N), mean, standard deviation (STDEV), and mean confidence interval (CI) resulting from computing the performance criterion functions for IACS and the baseline. These comparison tables show that the proposed method results in statistically significant improvements over the baseline method.

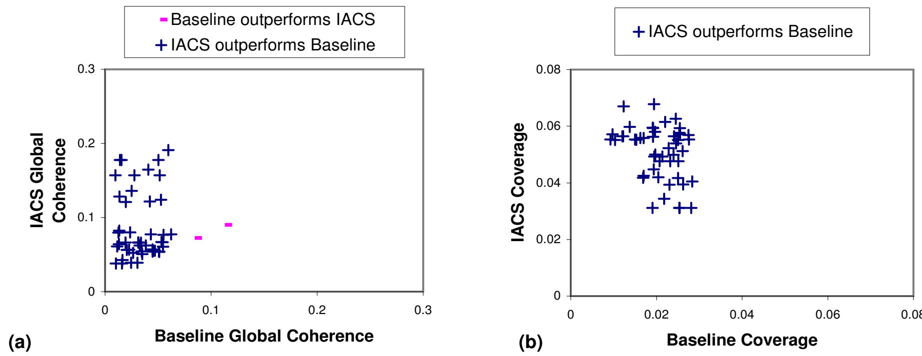

Fig. 3. IACS vs. Baseline: (a) Global Coherence and (b) Coverage.

Method	N	MEAN	STDEV	95% C.I.
IACS	48	0.051	0.009	**(0.048, 0.054)**
Baseline	48	0.021	0.005	**(0.020, 0.022)**

Table 3. Confidence intervals for the mean coverage of the incremental algorithm for context-based search (IACS) and baseline.

5 Related Work

The use of context to select and filter information plays a vital role in proactive retrieval systems. Such systems observe user interactions, infer user needs for additional information resources, and search for relevant documents on the Web or other online electronic libraries. Traditionally, such systems find documents relevant to a target by augmenting terms from the target with indexing keywords selected from the context, to improve recall and precision. A variety of recent systems pursuing this approach have obtained encouraging results. For example, Watson [2] uses contextual information from documents that users are manipulating to automatically generate Web queries from the documents, using a variety of term-extraction and weighting techniques to select suitable query terms. Watson then filters the matching results, clusters similar HTML pages, and presents the pages to the user as suggestions. Another such system is the Remembrance Agent [1] which operates inside the Emacs text editor and continuously monitors the user's work to find relevant text documents, notes, and emails previously indexed. Other systems such as Letizia [16] and WebWatcher [17] use contextual information compiled from past browsing behavior—searches within the locus of a currently viewed Web page—to provide suggestions on related Web pages or links to explore next.

CALVIN [18, 19] is a context-aware system which monitors the user's Web browsing activity to generate a model of the user's task to use to retrieve relevant resources indexed in similar contexts. In addition, versions of the system provide capabilities for users to manually enter information about a variety of resources, such as descriptions of books or articles, and data on useful personal contacts. The gathered material is stored as contextualized cases recording information that users consult during their decision-making, and is suggested when the user context is similar to the one associated with the stored cases.

Except for Watson, these systems either suggest information previously indexed by the system or crawled from the currently viewed pages. In contrast, our system, like Watson, potentially considers the entire Web, using widely available search engines such as Google to search for related documents. The IACS approach differs from Watson in its incremental search, which refines the Web queries to find documents more closely related to the concept map in progress.

SenseMaker [20] is an interface that facilitates the navigation of information spaces by providing task specific support for consulting heterogeneous search services. The system helps users to examine their present context, move to new contexts or return to previous ones. SenseMaker presents the collection of suggested documents in bundles (their term for clusters), which can be progressively expanded, providing a user-guided form of incremental search. Our EXTENDER system [14], like IACS, also applies an incremental technique to build up context descriptions. Its task, however, is to generate brief descriptions of new topics relevant to the current concept map. Rather than providing documents, EXTENDER aims to jog the user's memory during the concept mapping process by presenting a set of keywords suggesting novel, diverse and relevant topics to start new concept maps that extend the knowledge model under construction.

While our work explores the use of the rich context provided by the structure and labels of a knowledge model under construction, other work has pursued retrieval based on other types of contextual information. For example, Suitor [21] is a collection of "attentive agents" that gather information from the users by monitoring users' behavior and context, including eye gaze, keyword input, mouse movements, visited URLs and software applications on focus. This information is used to retrieve context relevant material from the Web and databases. Outside of proactive retrieval systems, IACS' learning of new context-related terms may be seen as related to learning semantic correspondences, studied in Semantic Web research (e.g., [22]).

6 Conclusion and Future Directions

When rich contextual information is available, it provides a potential resource for improving the performance of proactive retrieval systems. However, it may be difficult to select terms to describe a context, and the descriptions may be difficult to apply in single search queries. This paper describes research on addressing these problems through an incremental algorithm, IACS, which successively retrieves relevant resources and refines the context description. IACS has been applied to the task of retrieving Web pages relevant to a concept in the context of a concept map, in order to aid the concept mapping process. In an evaluation using an expert-generated knowledge model as the basis for assessing relevance, the IACS approach outperformed a baseline in both coherence and coverage of the resources retrieved.

As discussed in section 2.2, concept-map-based knowledge models provide many different layers of context. The study reported in this paper examines the use of a single layer, the concepts judged important to the topic of the concept map. Consequently, an interesting followup study concerns developing strategies for including appropriate weightings of terms in other layers, and assessing the tradeoffs of expanded contexts in terms of global coherence and coverage.

The IACS algorithm is applicable to any domain for which it is possible to generate term-based characterizations of a context. Thus another interesting task is to study IACS for other task domains for which rich context is available. For example, IACS could be applied to retrieve resources relevant to an electronic document such as a report, an email message, a presentation, or a Web page as it is written or consulted. We expect incremental approaches to have broad potential applicability to exploiting rich contexts for context-relevant Web search.

References

1. Rhodes, B., Starner, T.: The remembrance agent: A continuously running automated information retrieval system. In: The Proceedings of The First International Conference on The Practical Application of Intelligent Agents and Multi Agent Technology (PAAM '96), London, UK (1996) 487–495
2. Budzik, J., Hammond, K.J., Birnbaum, L.: Information access in context. Knowledge based systems **14** (2001) 37–53
3. Lawrence, S.: Context in Web search. IEEE Data Engineering Bulletin **23** (2000) 25–32
4. Cañas, A.J., Hill, G., Carff, R., Suri, N., Lott, J., Eskridge, T., Gómez, G., Arroyo, M., Carvajal, R.: CmapTools: A knowledge modeling and sharing environment. In Cañas, A.J., Novak, J.D., González, F., eds.: Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping. (2004)
5. Cañas, A., Carvalho, M., Arguedas, M., Eskridge, T., Leake, D., Maguitman, A., Reichherzer, T.: Mining the web to suggest concepts during concept map construction. In Cañas, A.J., Novak, J.D., González, F., eds.: Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping. (2004)
6. Leake, D., Maguitman, A., Reichherzer, T., Cañas, A., Carvalho, M., Arguedas, M., Brenes, S., Eskridge, T.: Aiding knowledge capture by searching for extensions of knowledge models. In: Proceedings of the Second International Conference on Knowledge Capture (K-CAP), New York, ACM Press (2003) 44–53
7. Leake, D., Maguitman, A., Reichherzer, T., Cañas, A., Carvalho, M., Arguedas, M., Eskridge, T.: "Googling" from a concept map: Towards automatic concept-map-based query formation. In Cañas, A.J., Novak, J.D., González, F., eds.: Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping. (2004)
8. Novak, J.: A Theory of Education. Ithaca, Illinois, Cornell University Press (1977)
9. Novak, J., Gowin, D.B.: Learning How to Learn. Cambridge University Press (1984)
10. Briggs, G., Shamma, D., Cañas, Carff, R., Scargle, J., Novak, J.D.: Concept maps applied to Mars exploration public outreach. In Cañas, A.J., Novak, J.D., González, F., eds.: Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping. (2004) 125–133
11. Leake, D., Maguitman, A., Reichherzer, T., Cañas, A., Carvalho, M., Arguedas, M., Brenes, S., Eskridge, T.: Aiding knowledge capture by searching for extensions of knowledge models. In: Proceedings of KCAP-2003, ACM Press (2003)
12. Leake, D., Maguitman, A., Reichherzer, T.: Understanding knowledge models: Modeling assessment of concept importance in concept maps. In: Proceedings of CogSci-2004. (2004)

13. Maguitman, A., Leake, D., Reichherzer, T., Menczer, F.: Dynamic extraction of topic descriptors and discriminators: Towards automatic context-based topic search. In: Proceedings of the Thirteenth Conference on Information and Knowledge Management (CIKM), New York, ACM Press (2004) 463–472
14. Maguitman, A., Leake, D., Reichherzer, T.: Suggesting novel but related topics: Towards context-based support for knowledge model extension. In: Proceedings of the 2005 International Conference on Intelligent User Interfaces. (2005) 207–214
15. Maguitman, A.: Intelligent Support for Knowledge Capture and Construction. PhD thesis, Indiana University (2005)
16. Lieberman, H.: Letizia: An agent that assists Web browsing. In: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95), San Mateo, Morgan Kaufmann (1995) 924–929
17. Armstrong, R., Freitag, D., Joachims, T., Mitchell, T.: WebWatcher: A learning apprentice for the World Wide Web. In: AAAI Spring Symposium on Information Gathering. (1995) 6–12
18. Leake, D.B., Bauer, T., Maguitman, A., Wilson, D.C.: Capture, storage and reuse of lessons about information resources: Supporting task-based information search. In: Proceedings of the AAAI-00 Workshop on Intelligent Lessons Learned Systems. Austin, Texas, AAAI Press (2000) 33–37
19. Bauer, T., Leake, D.: WordSieve: A method for real-time context extraction. In: Modeling and Using Context: Proceedings of the Third International and Interdisciplinary Conference, Context 2001, Berlin, Springer-Verlag (2001)
20. Baldonado, M.Q.W., Winograd, T.: SenseMaker: an information-exploration interface supporting the contextual evolution of a user's interests. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press (1997) 11–18
21. Maglio, P.P., Barrett, R., Campbell, C.S., Selker, T.: SUITOR: an attentive information system. In: Proceedings of the 5th international conference on Intelligent user interfaces, ACM Press (2000) 169–176
22. Doan, A., Madhavan, J., Domingos, P., Halevy, A.: Learning to map between ontologies on the semantic web. In: Proceedings of the Eleventh International WWW Conference, ACM Press (2002)