Mining for Topics to Suggest Knowledge Model Extensions

CARLOS LORENZETTI and ANA MAGUITMAN, Universidad Nacional del Sur
DAVID LEAKE and FILIPPO MENCZER, Indiana University
THOMAS REICHHERZER, University of West Florida

Electronic *concept maps,* interlinked with other concept maps and multimedia resources, can provide rich *knowledge models* to capture and share human knowledge. This article presents and evaluates methods to support experts as they extend existing knowledge models, by suggesting new context-relevant topics mined from Web search engines. The task of generating topics to support knowledge model extension raises two research questions: first, how to extract topic descriptors and discriminators from concept maps; and second, how to use these topic descriptors and discriminators to identify candidate topics on the Web with the right balance of novelty and relevance. To address these questions, this article first develops the theoretical framework required for a "topic suggester" to aid information search in the context of a knowledge model under construction. It then presents and evaluates algorithms based on this framework and applied in EXTENDER, an implemented tool for topic suggestion. EXTENDER has been developed and tested within CmapTools, a widely used system for supporting knowledge modeling using concept maps. However, the generality of the algorithms makes them applicable to a broad class of knowledge modeling systems, and to Web search in general.

CCS Concepts: • **Human-centered computing → Human computer interaction (HCI)**; • **Applied computing → Document management and text processing**; • **Information systems → Information systems applications**; **Information retrieval**; *Data extraction and integration;*

Additional Key Words and Phrases: Concept mapping, web mining, knowledge construction, knowledge discovery, intelligent suggesters

ACM Reference Format:
Carlos Lorenzetti, Ana Maguitman, David Leake, Filippo Menczer, and Thomas Reichherzer. 2016. Mining for topics to suggest knowledge model extensions. ACM Trans. Knowl. Discov. Data 11, 2, Article 23 (December 2016), 30 pages.
DOI: http://dx.doi.org/10.1145/2997657

1. INTRODUCTION

Knowledge modeling is the process of representing a body of knowledge so that this knowledge can then be used and shared. Knowledge acquisition has long been considered to be a bottleneck in the development of knowledge-based systems [Hayes-Roth et al. 1983], and knowledge modeling remains a difficult task [Cairó and Guardati 2012; Shadbolt and Smart 2015]. One impediment is selecting the right knowledge

---

This work was supported by NASA under Award no. NCC 2-1216, CONICET (PIP 11220120100487), MinCyT (PICT 2014-0624), and Universidad Nacional del Sur (PGI-UNS 24/N029).
Authors' addresses: C. Lorenzetti and A. Maguitman, ICIC CONICET-UNS, Depto. de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Bahía Blanca 8000, Argentina; emails: {cml, agm}@cs.uns.edu.ar; D. Leake and F. Menczer, School of Informatics and Computing, Indiana University, Bloomington, IN 47405; emails: {leake, fil}@cs.indiana.edu; T. Reichherzer, Department of Computer Science, University of West Florida, Pensacola, FL 32514; email: treichherzer@uwf.edu.

DOI: http://dx.doi.org/10.1145/2997657

to encode; another is representing the selected knowledge. Typically, domain experts are unfamiliar with the formalisms used to encode knowledge, requiring the intervention of knowledge engineers. This process may cause communication problems, and the requisite knowledge engineering is expensive. To address such problems, several initiatives have attempted to alleviate the knowledge acquisition bottleneck by developing knowledge modeling tools that allow experts to enter descriptions of their expertise without the intervention of knowledge engineers (e.g., Gil [1994], Blythe et al. [2001], Aiken and Sleeman [2003], and Gunning et al. [2010]).

In a recent article [Leake et al. 2014], we have described methods and tools for supporting an approach in which experts themselves build and extend knowledge models using concept maps [Novak and Gowin 1984; Hoffman et al. 2001; Basque et al. 2004; Briggs et al. 2004]. The present article provides a deeper description of one of the developed tools, Extender, an implemented system that mines search engines to generate topic suggestions for extending concept-map-based knowledge models.

The proposed approach to supporting knowledge modeling is in the spirit of human-centered computing, in that it is aimed at supporting human knowledge sharing and examination, rather than supporting autonomous machine reasoning. Easy-to-use computer tools exist for supporting the construction, examination, and sharing of concept-map-based knowledge models. Although such tools alleviate representational burdens, the content selection burden remains. This motivates the effort to include mechanisms for information access and delivery within the knowledge modeling tools.

To support knowledge extension, we have developed methods that extract information from the current knowledge model and mine a Web search engine to identify novel but relevant topics, to suggest those topics for possible inclusion in the knowledge model. Topics are commonly defined as pieces of information that have been grouped together as a result of having a common theme [Mohr and Bogdanov 2013; Wang et al. 2013; Caballero Barajas and Akella 2013]. To enable topics to be generated flexibly, according to possibly specialized user interests, our techniques dynamically generate topics taking concept maps as the initial search context. Each generated topic is presented to the user as a small collection of terms, intended to evoke a particular theme. The set of terms for each topic is supplemented with a ranked list of topic-relevant Web pages together with their descriptions and URLs, to help the user extend the knowledge model beyond information that has already been captured. The methods developed to accomplish the topic suggestion task automatically generate queries by using the concept map labels and by incorporating novel terms identified through an incremental search process. The excerpts of the top-retrieved search results are filtered based on their similarity to the concept map. These excerpts are analyzed to find a set of topics, which are represented by patterns of tightly co-occurring terms. These topics are presented as suggestions to the user. The developed interface enables users to easily start the construction of a concept map from the terms that constitute the suggested topic.

This application of Web mining is in the spirit of numerous intelligent support tools that examine the user's task context or information access behavior, to suggest related resources (e.g., Rhodes and Starner [1996], Budzik et al. [2001], Palmisano et al. [2008], Kraft [2011], Fuxman et al. [2014], and Livne et al. [2014]). However, our topic-suggestion task contrasts with this task in at least two ways. First, our goal is not to suggest individual *resources*, but rather, to make suggestions at the higher level of *topics*. A single topic may be partially reflected by a number of different Web pages, with none focusing solely on the topic or summarizing the topic in its entirety. Second, our goal is not to suggest the topics *most similar* to the current knowledge model, but rather to suggest new topics that are *related but novel*.

To the best of the authors' knowledge, the work presented in this article is the first to develop methods for supporting human-centered knowledge extension by generating topic suggestions. Developing topic suggestion tools depends on addressing three central questions:

(1) How to characterize topics?
(2) Given an initial search context defined by a concept map, how to generate new topics to suggest? and
(3) How to evaluate the appropriateness of candidate topics?

The following sections address those questions, focusing especially on how they apply to suggestions for knowledge modeling via concept mapping. Sections 2 and 3 provide an overview of concept mapping and the topic suggestion task. Section 4 proposes a theoretical framework for topic generation. Section 5 shows how the proposed framework is applied in the EXTENDER system, providing new methods and algorithms for topic extension. Section 6 evaluates the proposed methods and the system's ability to generate novel, related, and cohesive topic suggestions. The evaluations focus on examining which terms from a concept map should be used to form initial queries and on assessing the overall quality of EXTENDER's automatically-generated topics. The article closes with a review of related work and a discussion.

2. SUPPORTING CONCEPT MAPPING

Concept maps were developed by Joseph D. Novak in the 1970s for use in education [Novak 1977]. Concept maps are collections of simplified natural language sentences displayed as a two-dimensional, visually-based representation of concepts and their relations. In concept maps, concepts are depicted as labeled nodes, and relations between concepts as labeled links. Figure 1 shows a concept map on the topic of space missions to Mars. This concept map is part of CMEX Mars, a joint effort by a group of researchers from the Center for Mars Exploration at NASA Ames Research Center and the Institute for Human & Machine Cognition (IHMC) [Briggs et al. 2004], available at http://cmex.ihmc.us/CMEX/.

Concept mapping techniques have been applied by a diverse population ranging from elementary school students to adult experts, to model knowledge in many different domains (for a recent sample of uses, see Cañas et al. [2012] and Correia et al. [2014]). For knowledge capture, concept mapping offers map creators the ease and flexibility of natural language, but also induces them to organize captured knowledge in a structured fashion, in which concepts and their connections are made explicit. The rich structure of concept maps facilitates their understanding by other humans, and helps to make them more tractable than plain text for automated systems. Thus, concept maps provide an intermediate representation between natural language text and traditional AI representations [Leake and Wilson 2001]. In addition, electronic concept maps are browsable and sharable, making them an effective vehicle for aiding human understanding [Freeman and Jessup 2004; Garcia Castro et al. 2006].

The Institute for Human and Machine Cognition (IHMC) has conducted a multi-year initiative to support knowledge modeling by means of concept maps. The CmapTools system [Cañas et al. 2004; Cañas and Novak 2014], developed by the IHMC, is a suite of publicly available software tools for knowledge acquisition, construction, and sharing based on concept maps. The CmapTools system has been downloaded hundreds of thousands of times worldwide. It has been used in educational settings, by both students and teachers, as well as by experts to construct knowledge models of their domains without needing a knowledge engineer's intervention or to actively participate in the knowledge elicitation if a knowledge engineer leads the process. Examples of successful applications of electronic concept mapping for knowledge capture include

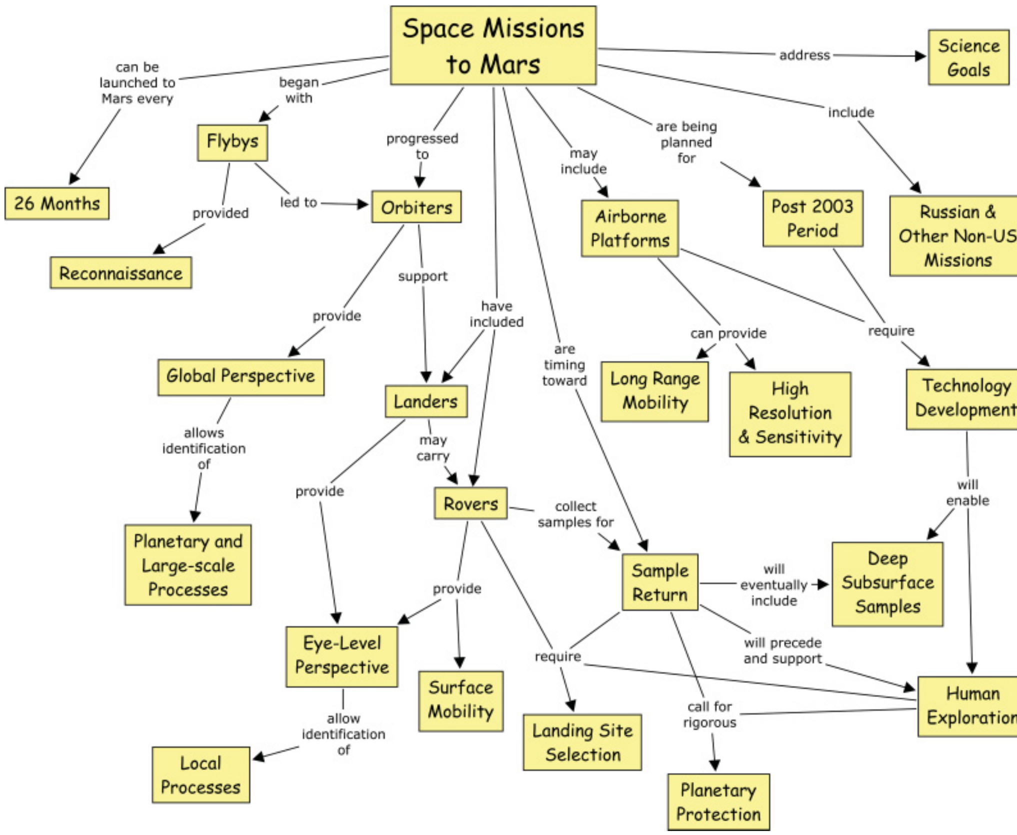

Fig. 1. A concept map created by a NASA expert.

tasks such as maintaining Navy equipment [Cañas et al. 1998], local weather prediction [Hoffman et al. 2001], explaining the design of rocket engines [Coffey 1999], recording knowledge about the planet Mars [Briggs et al. 2004], describing Web services in Service-Oriented Architectures (SOA) composite applications [Coffey et al. 2012], and conveying information about software security assurance cases to novices [Snider et al. 2014]. An overview of a number of applications of concept mapping tools is presented in Moon et al. [2011].

To support knowledge modeling in CmapTools, in collaboration with Alberto Cañas and the IHMC CmapTools research team, we have developed a number of intelligent aides [Leake et al. 2003, 2014]. These systems take as their starting point a concept map under construction, and propose information to aid the user's knowledge capture and construction efforts by proactively suggesting relevant concept maps, propositions, resources, concepts, and topics. The focus of this article, the EXTENDER system, uses information automatically extracted from the current knowledge model to guide mining a Web search engine to identify and suggest novel but relevant topics.

3. THE TOPIC SUGGESTION TASK

The specific goal of the EXTENDER project is to aid experts building knowledge models by "jogging the user's memory," providing suggestions for new topics to include in a knowledge model. EXTENDER provides its topic suggestions as small sets of terms, meant to convey the sense of a topic. This task is an instance of a more general one: suggesting novel topics related to a user's focus. For example, topic suggestions could also be useful to a researcher, to provide related but distinct areas to consider for connections and synergies or to help assure that relevant areas have been considered.

The effectiveness of a specific topic-generation strategy may be hard to assess, because the usefulness of suggested topics is highly subjective. However, the performance of a topic suggester depends on the following general criteria:

—*Global coherence*: The generated topics must be *relevant* to the initial knowledge model.
—*Local coherence*: Each generated topic must be of high quality according to a general set of domain-independent criteria. Such criteria might include measures for conciseness (the topic is summarized in a few terms, for easy comprehension) and term coherence (each topic description is locally coherent and is constituted of tightly related terms and documents).
—*Coverage*: A good topic-generation strategy should be able to generate a sufficient subset of the topics considered to be relevant.
—*Novelty*: Some generated topics must go beyond previously captured information.
—*Diversity*: Topics must be sufficiently diverse from each other for each topic in the suggested set to be useful.

In the following section, we discuss the theoretical framework we have developed for topic generation.

4. A FRAMEWORK FOR TOPIC GENERATION

Our approach to generating novel but related topics is iterative. The first step identifies topic terms to serve as a starting point to generate queries and retrieve additional material. The results of those retrievals are used to identify new terms to produce subsequent queries.

The first set of terms is generated by analyzing an initial concept map chosen by the user; the aim is to identify novel topics related to those in the concept map. A key question is how to select the terms to use. We hypothesize (and confirm through the evaluation presented in Section 6) that using concept labels as the initial terms, rather than linking phrases or a combination, results in better precision and recall.

The selection of terms from the retrieved material to generate subsequent queries requires some additional analysis of the usefulness of these terms to represent the topic at hand. Some terms may have strong descriptive power, enabling a small set to convey the topic to a human. Other terms may be effective cues for retrieving topic-relevant documents, but may not be good descriptors. Consider for example a topic involving exploration of Mars, described by the following set of terms occurring in documents related to Mars exploration:

Mars	Exploration	Rover	Landing	Site
Planet	Curiosity	Nasa	Images	Global
Surveyor	Orbiter	Camera	MAHLI	MAVEN

The terms *Mars* and *Exploration* are good descriptors of the topic for a general audience, whereas the term *MAHLI*—which stands for "Mars Hand Lens Imager"—may not be a good descriptor of the topic for that audience, because it is used by specialists, but it is effective for retrieving information similar to the topic because it is a good topic discriminator.

Intuitively, we can characterize topic descriptors and discriminators as follows:

—Terms are *good topic descriptors* if they answer the question "What is this topic about?"
—Terms are *good topic discriminators* if they answer the question "What are good query terms to access similar information?"

As experimentally shown in previous work [Maguitman et al. 2004; Lorenzetti and Maguitman 2009], a good strategy for finding good topic descriptors is to (1) find documents that are similar to other documents already known to have that topic, and (2) select non-stopword terms that occur frequently in those documents. On the other hand, a term is a good discriminator for a topic if most documents that contain that term are topically related. Thus, finding good topic discriminators requires finding terms that tend to occur only in the context of the given topic.

Both topic descriptors and discriminators are important as query terms. Because topic descriptors occur often in relevant pages, using them as query terms may improve recall. Because good topic discriminators occur primarily in relevant pages, using discriminators as query terms may improve precision. The following sections develop the above informal characterizations of topic descriptors and discriminators into precise definitions and describe their application to the task of searching the Web for topics related to a concept map.

4.1. Document Descriptors and Discriminators

Let $\mathbf{H}$ be a document–term matrix such that $\mathbf{H}[i, j]$ stands for number of occurrences of term t_j in document d_i. We model term descriptive power in a document with a function $\lambda : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$ that maps a document–term pair into a value in the unit interval. It is defined as follows:

$$\lambda(d_i, t_j) = \frac{\mathbf{H}[i, j]}{\sqrt{\sum_{k=0}^{n-1} (\mathbf{H}[i, k])^2}}.$$

The structure $[\lambda(d_i, t_1), \ldots, \lambda(d_i, t_{n-1})]$ is the normalized term-frequency vector for document d_i. The $\mathbf{H}[i, j]$ values are squared to penalize small values relatively more severely than larger values.

To model the discriminative power of a term in a document, we define a function $\delta : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$ as follows:

$$\delta(d_i, t_j) = \frac{\mathrm{s}(\mathbf{H}[i, j])}{\sqrt{\sum_{k=0}^{m-1} \mathrm{s}(\mathbf{H}[k, j])}},$$

where $\mathrm{s}(\cdot)$ is the step function, returning 1 if its argument is greater than zero and 0 otherwise. The function δ maps a document–term pair into a value in the unit interval. If term t_j does not occur in document d_i, then $\delta(d_i, t_j) = 0$. On the other extreme, if term t_j occurs in no document other than d_i, then $\delta(d_i, t_j) = 1$, and we say that t_j fully discriminates d_i. The discriminating power of a term in a document is independent of the number of occurrences of the term in the document. Hence, δ considers only whether or not a term occurs in a document. The step function is used for that purpose, disregarding the collection frequency of a term and focusing on the document frequency, defined to be the number of documents in the collection that contain the term.

As is the case with other Information Retrieval (IR) characterizations of descriptors and discriminators [Salton 1979; Yu et al. 1982], the notions discussed above only allow discovering terms that are good descriptors or discriminators of a *document*, as opposed to good descriptors or discriminators of the *topic of a document*. Traditionally, a non-stopword term that occurs often in a document is a good descriptor of that document. In this case, the term's descriptive value is commonly reflected by the term-frequency score [Salton 1971]. On the other hand, a term that occurs in a document without being common in the whole corpus is a good document discriminator and is traditionally captured by the inverse-document-frequency score [Jones 1972]. The combination of descriptive and discriminative scores gives rise to well-known term weighting schemes, such as TF–IDF (term frequency–inverse document frequency) [Salton and Yang 1973].

In the next sections, we build on the notions of document descriptors and discriminators to identify higher-order relations between documents and terms and to provide new definitions of descriptors and discriminators. These new definitions make the notions of descriptors and discriminators topic-dependent rather document-dependent.

4.2. Similarity and Co-occurrence

We treat topics as defined by either a collection of similar documents or a collection of terms that tend to co-occur. Thus, the notions of document similarity and term co-occurrence play important roles in identifying topics. The similarity between documents d_i and d_j can be computed using the well-known cosine measure as follows:

$$\sigma(d_i, d_j) = \frac{\sum_{k=0}^{n-1}[\lambda(d_i, t_k) \cdot \lambda(d_j, t_k)]}{\sqrt{\sum_{k=0}^{n-1}[\lambda(d_i, t_k)]^2 \cdot \sum_{k=0}^{n-1}[\lambda(d_j, t_k)]^2}} = \sum_{k=0}^{n-1}[\lambda(d_i, t_k) \cdot \lambda(d_j, t_k)].$$

The idea of *term co-occurrence* captures a relation between terms that is dual to the notion of document similarity. If two terms tend to occur in the same documents, we expect that their meanings are related. An expression for the co-occurrence of terms t_i and t_j can be written as

$$\kappa(t_i, t_j) = \frac{\sum_{k=0}^{m-1}[\delta(d_k, t_i) \cdot \delta(d_k, t_j)]}{\sqrt{\sum_{k=0}^{m-1}[\delta(d_k, t_i)]^2 \cdot \sum_{k=0}^{m-1}[\delta(d_k, t_j)]^2}} = \sum_{k=0}^{m-1}[\delta(d_k, t_i) \cdot \delta(d_k, t_j)],$$

which is similar to the cosine correlation formulas proposed in Salton and Lesk [1968] and Wong et al. [1987]. The fact that the λ and δ values are normalized allows omitting the denominator for both the similarity and co-occurrence formulas.

4.3. Topic Discriminators and Topic Focus

By examining the document–term duality, we can develop higher-order notions useful for identifying good topic descriptors and discriminators. A term is a *good discriminator of a document's topic* if those documents discriminated by the term are similar to the given document. This intuition can be formally expressed by the function

$\Delta : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$ defined as follows:

$$\Delta(d_i, t_j) = \frac{\sum_{\substack{k=0\\k\neq i}}^{m-1} [\delta(d_k, t_j)]^2 \cdot \sigma(d_i, d_k)}{\sum_{\substack{k=0\\k\neq j}}^{m-1} [\delta(d_k, t_j)]^2}$$

$$= \sum_{\substack{k=0\\k\neq i}}^{m-1} [\delta(d_k, t_j)]^2 \cdot \sigma(d_i, d_k).$$

We can think of the discriminating power of term t_j for the topic of document d_i as the average of the similarity of d_i to other documents discriminated by t_j. Note that even in the case when d_i does not contain t_j, the value of the function $\Delta(d_i, t_j)$ will not necessarily be 0. On the other hand, if no document similar to d_i contains t_j, i.e., $\sigma(d_i, d_k) = 0$ or $\delta(d_k, t_j) = 0$ for all documents d_k with $k \neq i$, then t_j has no discriminating power over the topic of d_i and as a consequence $\Delta(d_i, t_j) = 0$. The use of δ values in this formula allows giving higher weight to those documents for which the term has good discriminative power, whereas the use of σ values strengthens the impact of those documents that are similar to the document under analysis. The δ terms in the definition of Δ are squared to penalize small values relatively more severely than larger values.

We have previously discussed the dual notions of document similarity and term co-occurrence. At this stage, we might ask what would be the dual notion to "term discriminating power in a topic." This would be a function comparable to Δ but applicable to documents rather than terms. We can think of *document focus* as a property of documents that plays a role dual to that of term discriminating power. A document is focused on the topic associated with a term if the terms describing the document tend to co-occur with the given term. Formally, we can compute the degree of focus of a document on the topic identified by a term as a function $\Phi : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$ defined as follows:

$$\Phi(d_i, t_j) = \frac{\sum_{\substack{k=0\\k\neq j}}^{n-1} [\lambda(d_i, t_k)]^2 \cdot \kappa(t_k, t_j)}{\sum_{\substack{k=0\\k\neq j}}^{n-1} [\lambda(d_i, t_k)]^2}$$

$$= \sum_{\substack{k=0\\k\neq j}}^{n-1} [\lambda(d_i, t_k)]^2 \cdot \kappa(t_k, t_j).$$

The λ terms are also squared in this case to more strongly penalize small values.

Note that we have defined the higher-order dual notions of topic discriminators and topic focus by means of more basic dual notions. Term discriminating power in a topic has been defined using the notions of term discriminating power in a document and document similarity. Analogously, the measure of document focus on a topic has been defined via term descriptive power in a document and term co-occurrence.

4.4. Topic Descriptors and Topic Exhaustivity

The notion of *topic descriptors* was informally described earlier as terms that occur *often* in the context of a topic. More precisely, the *descriptive power* of a term in a topic is a measure that can be computed using the previously defined measures of document similarity and term descriptive power in documents. We define the *term descriptive power in the topic of a document* as a function $\Lambda : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \rightarrow [0, 1]$.

If $\sum_{\substack{k=0\\k\neq i}}^{m-1} \sigma(d_i, d_k) = 0$, then we set $\Lambda(d_i, t_j) = 0$. Otherwise we define $\Lambda(d_i, t_j)$ as follows:

$$\Lambda(d_i, t_j) = \frac{\sum_{\substack{k=0\\k\neq i}}^{m-1} \sigma(d_i, d_k) \cdot [\lambda(d_k, t_j)]^2}{\sum_{\substack{k=0\\k\neq i}}^{m-1} \sigma(d_i, d_k)}.$$

Descriptive power of a term t_j in the topic of a document d_i is a measure of the quality of t_j as a descriptor of documents similar to d_i. If no document is similar to d_i or t_j does not occur in any document similar to d_i, then the descriptive power of t_j in the topic of d_i is equal to 0. We can think of the descriptive power of term t_j in the topic of document d_i as the average document descriptive power of t_j in documents similar to d_i. The λ values in this formula increase the impact of those documents for which the term has good descriptive power, whereas the σ values reinforce the effect of those documents that are similar to the document under analysis.

The last property we define is *document exhaustivity* with respect to a topic. A document is exhaustive with regard to a topic if most terms that co-occur with a term identifying that topic are good discriminators of that document. We propose a measure of document exhaustivity as a function $\Xi : \{d_0, \ldots, d_{m-1}\} \times \{t_0, \ldots, t_{n-1}\} \to [0, 1]$. If $\sum_{\substack{k=0\\k\neq j}}^{n-1} \kappa(t_k, t_j) = 0$, we set $\Xi(d_i, t_j) = 0$. Otherwise we define $\Xi(d_i, t_j)$ as follows:

$$\Xi(d_i, t_j) = \frac{\sum_{\substack{k=0\\k\neq j}}^{n-1} \kappa(t_k, t_j) \cdot [\delta(d_i, t_k)]^2}{\sum_{\substack{k=0\\k\neq j}}^{n-1} \kappa(t_k, t_j)}.$$

By this definition, if term t_j does not co-occur with any other term or d_i does not contain any term that co-occurs with t_j, then the exhaustivity of d_i with respect to the topic associated with t_j is 0. We can think of $\Xi(d_i, t_j)$ as the average discriminating power for document d_i of terms that co-occur with t_j.

In this case, we have defined the higher-order dual notions of topic descriptors and topic exhaustivity by means of more basic dual notions. On the one hand, term descriptive power in a document and document similarity were used to define term descriptive power in a topic. On the other hand, term discriminating power in a document and term co-occurrence were used to define document exhaustivity with respect to a topic.

4.5. An Illustrative Example

Consider the following example (adapted from Lorenzetti and Maguitman [2009]) to illustrate the introduced notions, in which scientific and science fiction documents about Mars were analyzed. We show the transpose of matrix **H** to facilitate readability:

$$\begin{array}{r c} & \begin{array}{ccccc} d_0 & d_1 & d_2 & d_3 & d_4 \end{array} \\ \begin{array}{r} \texttt{mars} \\ \texttt{science} \\ \texttt{jpl} \\ \texttt{planet} \\ \texttt{nasa} \\ \texttt{book} \\ \texttt{fiction} \\ \texttt{fantasy} \\ \texttt{maven} \\ \texttt{mahli} \end{array} & \left(\begin{array}{ccccc} 4 & 2 & 5 & 5 & 2 \\ 2 & 6 & 3 & 2 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 & 1 \\ 3 & 0 & 2 & 2 & 0 \\ 0 & 3 & 0 & 0 & 3 \\ 0 & 4 & 0 & 0 & 2 \\ 0 & 4 & 0 & 0 & 1 \\ 0 & 0 & 2 & 1 & 0 \\ 0 & 0 & 3 & 3 & 0 \end{array}\right) \end{array}$$

Documents:
d_0: mars.nasa.gov
d_1: scifan.com
d_2: space-fact.com
d_3: jpl.nasa.gov
d_4: sf-encyclopedia.com

In our example, terms *mars*, *science*, *jpl*, *planet*, and *nasa* are all good descriptors in the topic of documents d_0, d_2, and d_3. However, although terms *jpl* and *nasa* are good discriminators in that topic, the term *mars* is not—*mars* occurs often in that topic but not only in that topic. On the other hand, note that a term may have a high discriminative power in the topic of a document even if the term is not in the document itself—it is sufficient for the term to occur only in documents on that topic to have a high discriminative value. For instance, terms *maven* and *mahli* have high discriminative values in the topic of d_0 despite not occurring in that document:

$$
\begin{array}{r cccc}
 & \lambda(d_0,t_j) & \delta(d_0,t_j) & \Lambda(d_0,t_j) & \Delta(d_0,t_j) \\
\begin{array}{r}
\texttt{mars}\\ \texttt{science}\\ \texttt{jpl}\\ \texttt{planet}\\ \texttt{nasa}\\ \texttt{book}\\ \texttt{fiction}\\ \texttt{fantasy}\\ \texttt{maven}\\ \texttt{mahli}
\end{array}
&
\begin{pmatrix} 0.718\\0.359\\0.180\\0.180\\0.539\\0.000\\0.000\\0.000\\0.000\\0.000 \end{pmatrix}
&
\begin{pmatrix} 0.447\\0.500\\0.577\\0.500\\0.577\\0.000\\0.000\\0.000\\0.000\\0.000 \end{pmatrix}
&
\begin{pmatrix} 0.385\\0.158\\0.014\\0.040\\0.055\\0.089\\0.064\\0.040\\0.032\\0.124 \end{pmatrix}
&
\begin{pmatrix} 0.493\\0.524\\0.566\\0.517\\0.566\\0.385\\0.385\\0.385\\0.848\\0.848 \end{pmatrix}.
\end{array}
$$

4.6. Comparing Topic Descriptors and Discriminators to Other Weighting Schemes

Like the well-known TF and IDF measures, the Λ and Δ functions allow discovering terms that are good descriptors and good discriminators. However, there are important differences between these definitions. Terms rarely occur more than once in a concept map and as a consequence TF is not useful as a term weighting scheme to determine term importance in concept maps. It is necessary to highlight that the notion of topic descriptor attempts to determine term importance at the higher level of topic. For a term to be a good topic descriptor, it is not sufficient that it occurs often in a concept map or in a document: it must occur often in other documents that are similar to the target concept map.

The notion of topic discriminator is substantially different from the notion of IDF in several senses. The weighting scheme based on the notion of topic discriminator takes a small set of dynamically retrieved documents and computes the topic discriminating power as a function of how similar to the target concept map are the documents that contain that term. A good topic discriminator is a term that tends to occur exclusively in similar documents. In other words, a good topic discriminator occurs in some documents that are related to the target concept map but hardly occurs in unrelated documents. On the other hand, a term with high IDF is a term that rarely occurs in an entire corpus. The IDF weighting scheme does not distinguish between related and unrelated documents to penalize common terms, whereas this distinction is crucial for the topic discriminator weighting scheme. We regard this as a key conceptual advantage of our weighting scheme over IDF.

Extensive experimental evidence published as part of our previous work [Lorenzetti and Maguitman 2009] shows that the performance of different term-weighting schemes that account for the notions of topic discriminators and topic descriptors is superior to other term weighting schemes including a baseline method and the Bose–Einstein statistics model (Bo1-DFR) [Amati 2003], which is known to be superior to TF–IDF.

5. APPLYING THE FRAMEWORK

We have applied the theoretical framework in the EXTENDER system, a program that iteratively searches the Web to generate term-based topic descriptions, which are presented to the user as small sets of keywords. This section introduces EXTENDER's

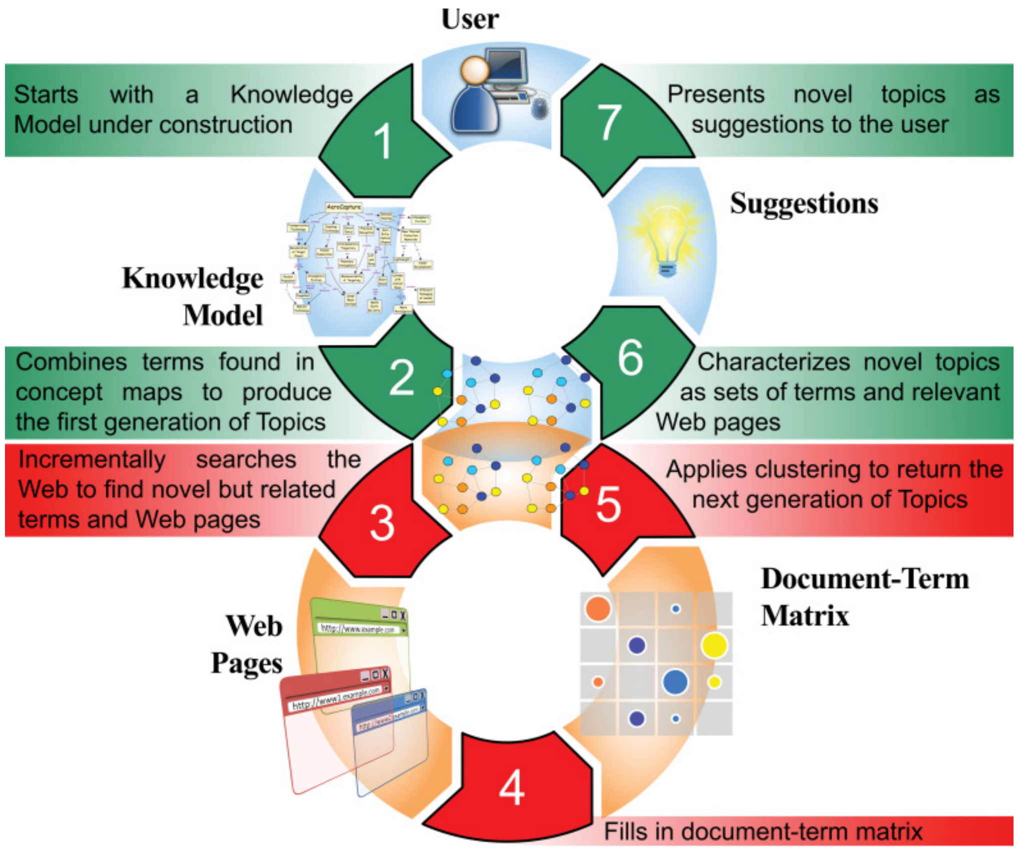

Fig. 2. EXTENDER's Cycle.

processing cycle, then discusses how the system's design addresses the desiderata for topic generation presented in Section 3, and finally describes the full algorithm used by the system.

5.1. EXTENDER's Processing Cycle

Figure 2 outlines EXTENDER's processing cycle. The system starts from a concept map and iteratively searches the Web for novel information. EXTENDER's interface allows the user to highlight a concept or set of concepts from the starting concept map in order to bias the system's search toward topics related to the highlighted concepts. Alternatively, the search can be initiated from the full map, without introducing any additional bias.

At each iteration, the collected material is represented internally by document–term matrices; clustering is applied to identify topics in the collection; and unimportant material, such as terms with low descriptive and discriminating power, is discarded. This process is repeated until topics converge or a user-selected limit on iterations is reached. In our tests, three iterations are usually sufficient to generate a rich variety of topics. Details on clustering and topic convergence are given in Section 5.3.

Once EXTENDER completes its iterations, it presents the generated topics as suggestions to the user. Figure 3 shows the CmapTools interface. EXTENDER's suggestions are presented in a suggestions panel that is attached to a concept map window from which

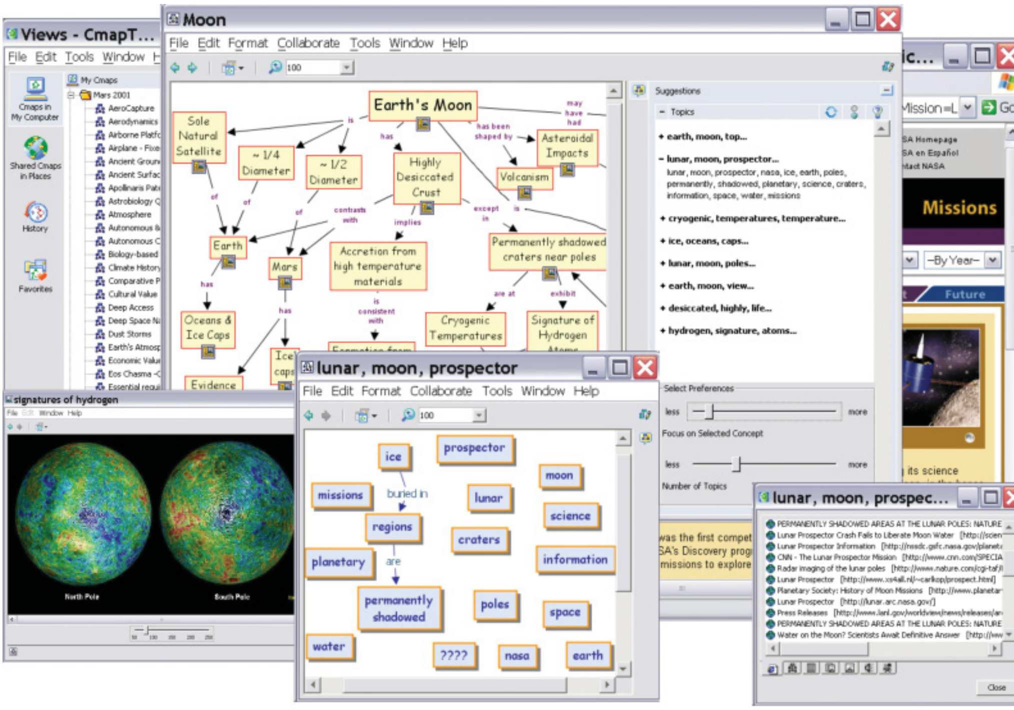

Fig. 3. Portion of a knowledge model with EXTENDER suggesting new topics.

the topic search started. The panel is visible at the upper right in the figure. To avoid distracting the user, the suggestions panel is only visible when the user chooses to open it. When the panel is closed, an unobtrusive icon shows that new suggestions are available. In addition, the interface presents the Web pages that gave rise to those topics, grouped by topic, to facilitate access to topic-relevant information. EXTENDER's preferences panel, shown at the bottom of the suggestion window in Figure 3, allows the user to adjust the tool's level of focus on the chosen concepts and the range of topics generated. The interface enables users to easily import a generated topic into an in-progress concept map as a set of concepts named by the terms in the topic, from which the user can start the mapping process. The lower central window of the figure shows a concept map being started from some concepts that the user selected from a set of topic terms suggested by EXTENDER, as well as additional concepts not yet included in the map.

5.2. Preserving Global Coherence

For EXTENDER's task, topics are considered globally coherent to the extent they relate to the user's initial concept map, which may provide rich information to exploit as search context. Because search engines restrict queries to a small number of terms (e.g., the 32-term limit for Google), a single query can only reflect limited information. Consequently, EXTENDER uses a multi-step approach to focus its topics by collecting and filtering descriptors and discriminators over multiple retrievals. Although the concept map offers a good initial search context, the vocabulary characterizing this search context can be incrementally refined by identifying new terms that are good descriptors or good discriminators of the topic of the concept map. During its cycle, EXTENDER maintains the relations between candidate topic terms and the initial concept map in three ways:

—*Term-weight reinforcement.* Terms collected during EXTENDER's retrievals are associated with weights summarizing the terms' goodness as topic descriptors or topic discriminators. Two lists of terms and their corresponding descriptive and discriminating values are maintained and updated throughout the iterations to keep track of good topic descriptors and good tropic discriminators. At the start of its cycle, EXTENDER calculates the descriptive power of terms directly from the topology of the user's concept map. The motivations for using the map's topology can be summarized by the following hypotheses: (1) concepts that are closer to the root of a concept map are considered better descriptors of the topic of the map, and (2) concepts with higher connectivity are considered better descriptors of the topic of the map (see Cañas et al. [2001], Leake et al. [2004], and Reichherzer and Leake [2006b] for a detailed description and an evaluation of three different topological analysis methods). If the user has selected focus concepts to bias the topic search, the weights of the terms in the labels of the selected concepts are adjusted by a constant weighting factor greater than one. For subsequent iterations, weights are adjusted according to the quantities defined in Section 4, to reinforce the weights of terms that have proven to be good descriptors or discriminators for the topic represented by the search context.

—*Context-based filtering.* For a document's terms to be considered candidates for inclusion as part of a new topic, the document has to survive a selection process that requires a minimum similarity between the document and the search context. Novel terms that are not good descriptors or discriminators of the topic reflected by the search context are also discarded.

—*Query refinement.* The first query terms generated for a Web search may not provide the definitive results. However, initial search results can help to automatically refine subsequent queries. To incrementally refine search queries, EXTENDER prioritizes the use of those query terms that are good topic descriptors and good topic discriminators.

EXTENDER's search context is initially defined using the knowledge model under construction, and it is then progressively updated as the focus shifts through a connected series of topics. Figures 4 and 5 illustrate the importance of exploiting the search context to keep global coherence. The first figure presents a concept map from a knowledge model on *Mars*, describing the topic *Ancient Surface Water Environments*. The second figure presents a concept map on the topic of *Rivers*. In both examples, the user highlighted the concept *Water* to initiate the search. However, the topics produced in the two maps are different, reflecting the corresponding contexts. The two sliders at the bottom right of EXTENDER's suggestion window allow the users to control the extent of focus on the selected concept and the maximum number of topics the system will return. The first slider adjusts the system's weightings of the highlighted concepts. The second slider determines the number of system iterations before returning the final set of topics and the number of topics produced after each iteration.

In EXTENDER's final stage, when the system presents the final set of topics to the user, the terms with highest descriptive value are used to produce labels for the suggested topics. A small set of terms with high descriptive or discriminated value provides a concise but informative representation of the topic. A combination of focus and exhaustivity is used to rank documents in a topic.

5.3. Generating Cohesive Topics

Local coherence reflects the degree to which each generated topic is composed of tightly related terms. Our approach measures cohesiveness by the ability of the topic to prompt retrieval of documents that are similar to each other. EXTENDER uses only short text

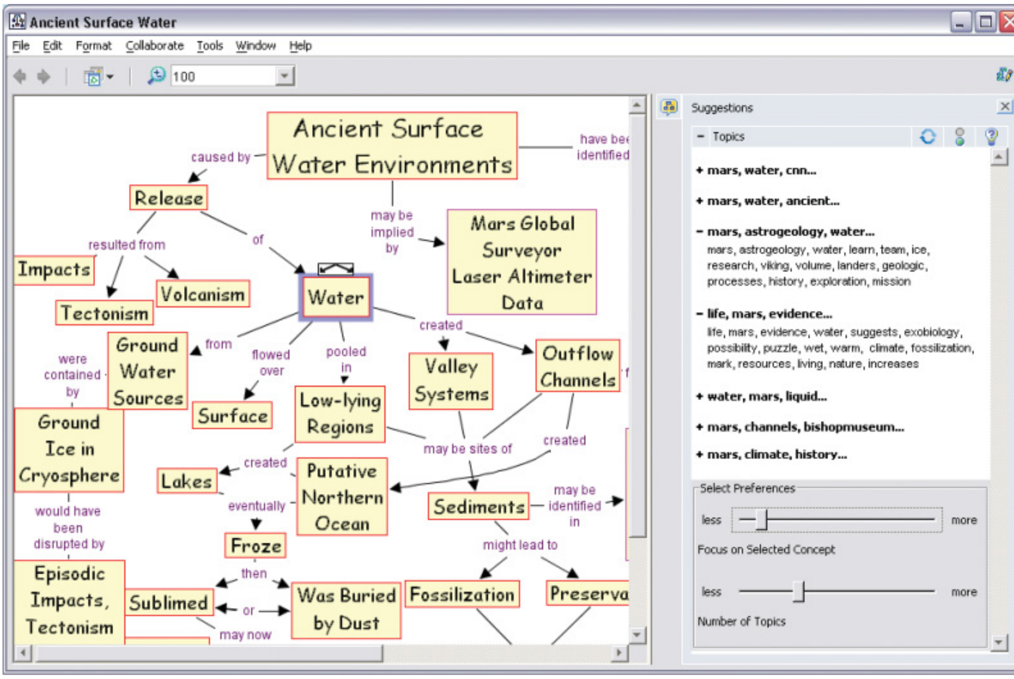

Fig. 4. EXTENDER suggesting topics for the concept *Water* in the context of *Mars' Ancient Surface Water Environments*.

excerpts (the text "snippets" provided by search engines, which are readily available from the search results) to represent documents. The need to group a collection of short text excerpts from highly related documents contrasts with common clustering scenarios. When full access to document text is available, document clustering appears preferable over term clustering, to give the clustering algorithm greater discerning power to identify topics. However, we have observed that when documents are represented by a small number of terms (as is the case for the text excerpts collected by EXTENDER), and the collection under analysis consists of material that shares a common general theme (which is a consequence of EXTENDER's attempt to preserve global coherence), terms may be as informative as documents for identifying topics within the collection.

With a few exceptions (e.g., Dhillon [2001]), most existing clustering algorithms apply single-purpose clustering—they cluster documents and terms separately. EXTENDER applies a medoid-based co-clustering algorithm to cluster documents and terms simultaneously. A medoid is the most appropriate point within a cluster that represents it [Kaufman and Rousseeuw 1989]. Assuming the set of medoids is given, then the clustering problem reduces to selecting the subsets of items "close" to the respective medoids. For each document, EXTENDER finds the terms that best characterize the document's topic. In a subsequent step, it uses the selected terms to identify the documents that best specify the topic of those terms, based on the notions of topic focus and topic exhaustivity discussed in Section 4. This process is repeated until either (1) two consecutive iterations produce the same set of terms and documents (a fixed point has been reached), or (2) the same result is detected for two non-consecutive iterations (a cycle has been detected). Because after each iteration the sizes of the sets containing selected terms and documents decrease or remain the same (as explained in Section 5.4), the

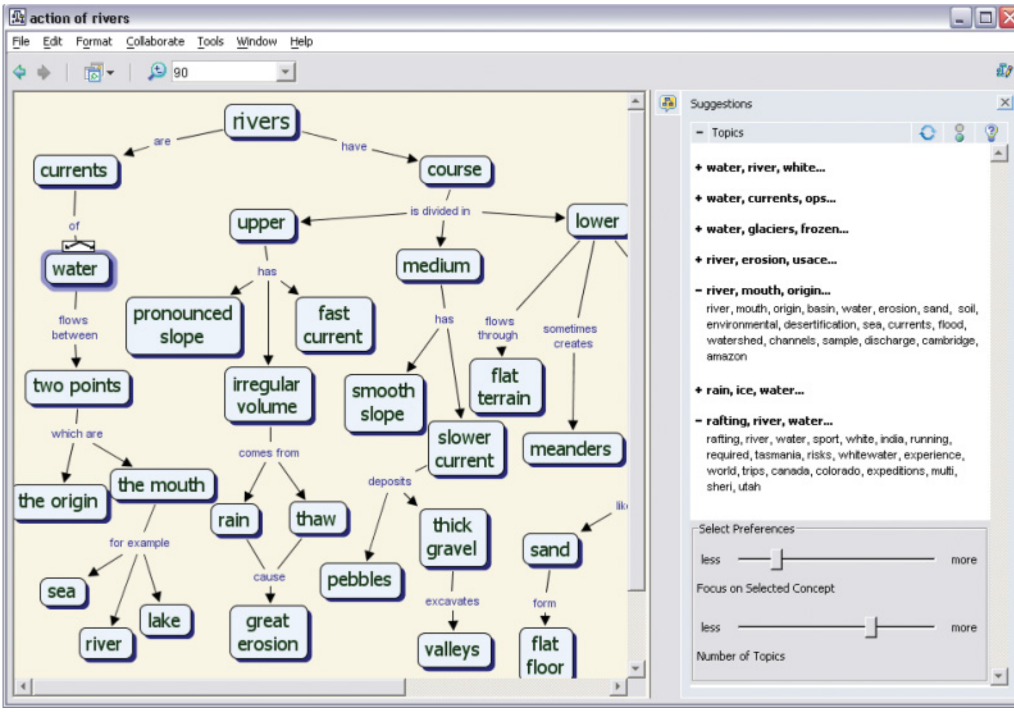

Fig. 5. EXTENDER suggesting topics for the concept *Water* in the context of *Rivers*.

selection process is guaranteed to converge in a finite number of iterations. The sets of selected terms and documents are used as term medoids and document medoids to define a set of cohesive topics. This clustering algorithm addresses the goal of achieving local coherence for EXTENDER's generated topics.

5.4. Coverage, Novelty, and Diversity

A reasonable topic-generation strategy should be able to produce topics with a suitable balance of coverage, novelty, and diversity. EXTENDER uses a "curiosity mechanism" to diversify during initial processing stages and to focus towards the end. The application of EXTENDER's curiosity mechanism is in the spirit of searching and learning techniques in which a temperature factor is used to favor exploration at the beginning and exploitation during the final stages (e.g., simulated annealing and reinforcement learning).

Throughout the system's iterations, while attempting to extend a given topic, newfound terms are collected. Because the number of collected terms grows rapidly, novel terms to retain are selected. For each term, the system tracks the goodness of the term in describing and discriminating the current topic. It retains those that surpass a threshold for the survival of descriptors/discriminators, where the threshold is a function of the number of iterations. For iteration $\textsc{i}$, the threshold for the survival of descriptors is computed by means of a function $\tau_\Lambda : \{0, \ldots, s-1\} \to [a, b]$:

$$\tau_\Lambda(\textsc{i}) = (b-a) \cdot \left(\frac{\textsc{i}}{s-1}\right)^c + a,$$

where a stands for the "starting threshold" parameter, b for the "stopping threshold" parameter, c is a curiosity decay parameter, and s is the total number of iterations.

The parameter a (resp. b) reflects the initial (final) stage of exploration (exploitation), when many (few) new terms are collected. Default values for these parameters were set based on empirical observations carried out on test instances. The threshold for discriminators, τ_{Δ}, is defined similarly.

Another curiosity threshold is used by EXTENDER to filter irrelevant documents according to the search context. This is implemented by a similarity threshold function τ_{σ} defined analogously to the other curiosity mechanism functions. During the initial steps EXTENDER collects documents on diverse topics, which must preserve the global theme of the originating concept map. After each iteration is completed, the current topic gives rise to a new set of descendant topics. As the system moves its focus through the new set of topics, the search context is updated and the curiosity threshold required for term retention is increased. Because the threshold increases with the number of iterations, novel terms and documents are seldom collected during the final stages. Consequently, the final stages are an exploitation phase that primarily reinforces the weights associated with a particular material that has already been added to the collection.

5.5. The EXTENDER Algorithm

The previous techniques form the core of EXTENDER's topic extension algorithm. However, because retrieving and processing large numbers of Web pages is costly, EXTENDER's process begins with a less expensive *distillation phase*, in which a series of queries is submitted to a search engine and only the information that is readily available from the search results (e.g., title, "snippet" of text) is used to identify promising topic descriptors and discriminators. The initial queries are formed using the concept labels of the given concept map. (Section 6 will provide experimental evidence showing that using concept labels results in better performance than using linking phrases to form queries.) After this preliminary step, the best topic descriptors and discriminators are used as query terms in a *search phase* to search for additional material on the Web. The new set of search results is filtered according to the search context and then clustered to produce the next generation of topics. Finally, each of the topics is refined, keeping only those documents and terms that are good topic representatives. The topics resulting from different branches usually have significant overlap of coverage. To ensure diversity, after an iteration is completed, EXTENDER merges similar topics. This is done by applying a simple single-linkage clustering procedure [Sibson 1973]. A parameter r defines the similarity threshold between two topics. If the similarity between two topics is at least r, then the two topics are merged. The system avoids submitting the same query more than once; therefore, the system may stop after no more novel queries can be produced. Otherwise, EXTENDER's cycle will be repeated at most s times, where s is a parameter supplied by the user through a slider in the topic suggestion panel. Algorithms 1 and 2 provide a high-level description of this procedure.

6. EVALUATION

Our evaluation examines two questions. The first is which terms from a concept map should be used as query terms input to the process for extracting topic descriptors and discriminators. We hypothesized that using concept labels only, disregarding linking phrases, would provide the best results. Our evaluation compared the effectiveness of using concept labels only versus using linking phrases to generate the initial set of queries, using both traditional IR metrics and ad hoc metrics formulated for EXTENDER's specific task. The second question concerns the overall quality of EXTENDER's automatically generated topics. We assessed this by comparing the set of suggested topics with expert-generated concept maps.

ALGORITHM 1: Extend-Topic

Input: M: source concept map, s: total number of iterations, q_d: number of queries submitted for distillation, q_s: number of queries submitted for search.
Output: *Topics*: A set of topics related to M.
```
Topics[0] = { M };
for (i = 0; i < s; i++) do
    Topics[i + 1] = ∅;
    for each Topic T ∈ Topics[i] do
        N = Next-Generation-Of-Topics(T, i);
        Topics[i + 1] = Topics[i + 1] ∪ N;
    end
end
Merge similar topics in Topics;
return Topics;
```

ALGORITHM 2: Next-Generation-Of-Topics

Input: T: topic to extend, i: iteration.
Output: N: A new set of topics.
```
/* distillation                                                    */
if (i = 0) then
    Use concept labels to form queries;
else
    Use those terms with highest descriptive value to form q_d queries;
end
Submit the queries to a search engine;
Use search result's "readily available information" to compute descriptive and discriminating
    power for each term;
/* search                                                          */
Combine best descriptors and discriminators to form q_s queries;
Submit the queries to a search engine and collect the returned document excerpts;
/* filtering                                                       */
Use the curiosity mechanism to filter the returned documents according to the map;
Use the curiosity mechanism to filter the terms according to their descriptive and
    discriminating value;
/* clustering                                                      */
Cluster remaining data to generate cohesive topics;
/* clean-up                                                        */
For each topic only keep terms that are good descriptors or discriminators;
The terms with highest descriptive power are used to produce the main labels for the suggested
    topics, while the best descriptors and discriminators are used to populate the topics;
For each topic only keep documents that are similar to the medoid of the topic. Rank these
    documents using a combination of the exhausitvity and focus values;
Collect resulting topics into set N;
return N;
```

6.1. Evaluating Query Terms Extracted from Concept Maps

We expected that concept labels are more effective than linking phrases to form queries. We tested this hypothesis using an especially built test platform. To build our test platform we used 448 topics from the Open Directory Project (ODP).[1] The topics were selected from the third level of the ODP hierarchy. A number of constraints were

[1]http://dmoz.org.

Table I. Concept Maps, Abbreviated Reference Names, and Associated Relevant Topics from ODP

Concept map (abbreviation)	Topic
AeroCapture (Aer)	SCIENCE/TECHNOLOGY/SPACE
BacterialCharacteristics (BCh)	HEALTH/CONDITIONS_AND_DISEASES/INFECTIOUS_DISEASES
BacterialClassification (BCl)	HEALTH/CONDITIONS_AND_DISEASES/INFECTIOUS_DISEASES
Bioinformaticbynati (Bio)	SCIENCE/BIOLOGY/BIOINFORMATICS
Comparative_Planetology (CoP)	SCIENCE/ASTRONOMY/SOLAR_SYSTEM
FungiCharacteristics (FCh)	SCIENCE/BIOLOGY/FLORA_AND_FAUNA
Moon (Moo)	SCIENCE/ASTRONOMY/SOLAR_SYSTEM
PlanetMars (PMa)	SCIENCE/ASTRONOMY/SOLAR_SYSTEM
Rabbits (Rab)	RECREATION/PETS/RABBITS
Sleep (Sle)	HEALTH/CONDITIONS_AND_DISEASES/SLEEP_DISORDERS
VirusCharacteristics (VCh)	HEALTH/CONDITIONS_AND_DISEASES/INFECTIOUS_DISEASES
Volcanism (Vol)	SCIENCE/EARTH_SCIENCES/GEOPHYSICS

imposed on this selection with the purpose of ensuring the quality of our test set. The minimum size for each selected topic was 100 URLs, and the language was restricted to English. For each topic, we collected all of its URLs as well as those in its subtopics, resulting in more than 350,000 collected pages. The Terrier framework [Ounis et al. 2007] was used to index these pages and to run our experiments.

To run our tests, it was necessary to find concept maps about some of these topics. The search for test concept maps was done by using the names of the indexed ODP topics to query the Public CmapServers accessible through CmapTools. This procedure identified 12 concept maps, each matching one of the ODP topics. The names of these concept maps and their associated ODP topics are presented in Table I.[2]

In our tests, each of these concept maps was used to form queries based on four strategies similar to those described in Reichherzer and Leake [2006a]:

—*CCC* (concept–concept–concept): These queries were formed by considering all the possible combinations of three different concepts in a concept map, where each of these three concepts is a neighbor of at least one of the other two. Two concepts are neighbors if there exists a linking phrase connecting both. The queries were built using the keywords extracted from these three concepts.
—*CLC* (concept–link–concept): These queries were formed by combining a concept label, a linking phrase and another concept label. To build the CLC-type queries, we considered all the triplets concept–link–concept with the two concepts connected to the link. The keywords from the two concept labels and the linking phrase were then used to form a query.
—*LCL* (link–concept–link): These queries were formed by combining a linking phrase, a concept label, and another linking phrase. To build the LCL-type queries, we considered each concept with at least two linking phrases connected to it. The keywords from the concept and the two linking phrases were then used to form a query. If there were more than two linking phrases connected to a concept, all the possible combinations of linking phrases pairs were considered to form different queries.
—*LLL* (link–link–link): In a form dual to CCC-type queries, the LLL-type queries were constructed by selecting all the possible combinations of three different linking phrases in a concept map, where each of these three linking phrases is a neighbor of least one of the other two. Two linking phrases are neighbors if there is a common concept to which both are connected.

[2]To see the concept maps and the generated queries used in this experiment visit http://ir.cs.uns.edu.ar/datasets.

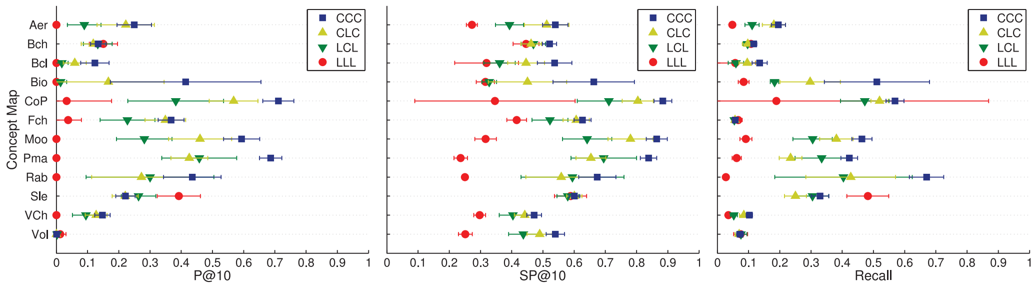

Fig. 6. Means and their 95% confidence intervals for precision at rank 10 (left), semantic precision at rank 10 (middle), and recall (right) of the query formation strategies.

Stopwords were removed from the concept map before the queries were created. As has been discussed in the concept mapping literature (e.g., Shams and Elsayed [2008] and Schwendimann [2014]), in most cases, the concepts are nouns or noun phrases, whereas the relations are verbs or prepositions. Therefore, it is natural to anticipate that concepts will have a superior performance to linking phrases. In spite of that, it is worth inquiring into the possible benefits of incorporating linking phrases in the constructed queries. Linking phrases act as “predicates,” providing rich information about the relation that holds between concepts. Therefore, it is worth evaluating whether the strategies that involve the use of these predicates offer any advantages over those strategies that disregard them.

To evaluate the different query construction strategies, we used three measures of query performance: precision at rank 10, semantic precision at rank 10, and recall. These metrics are described below.

—*Precision at rank 10 (P@10).* This metric is computed as the fraction of the top 10 retrieved documents which are known to be relevant, i.e., P@10= $|A_{10} \cap R|/|A_{10}|$, where R and A_{10} are the relevant and answer sets (restricted to the top 10 retrieved documents), respectively. The relevant set for each analyzed topic was set as the collection of its URLs as well as those in its subtopics.

—*Semantic Precision at rank 10 (SP@10).* Other topics in the ontology could be semantically similar (and therefore partially relevant) to the topic of the given context. Therefore, we use semantic precision at rank 10 defined as SP@10= $\sum_{p \in A_{10}} \sigma^S(t(\mathcal{C}), t(p))/|A_{10}|$, where $t(\mathcal{C})$ is the ODP topic associated with the description used as the initial context, $t(p)$ is the topic of page p, and $\sigma^S(t(\mathcal{C}), t(p))$ is the semantic similarity between these two topics. To compute σ^S, we used a semantic similarity measure for generalized ontologies proposed in Maguitman et al. [2005]. For a detailed analysis of the advantages and implications of using semantic precision as an alternative to the traditional precision metric, see Cecchini et al. [2011].

—*Recall.* This metric is computed as the fraction of relevant documents that are effectively retrieved, i.e., Recall= $|A \cap R|/|R|$, where R and A are the relevant and answer sets, respectively.

Figure 6 presents the means and their 95% confidence intervals for precision at rank 10, semantic precision at rank 10, and recall. According to this analysis, for most of the analyzed concept maps, the CCC-type query formation strategy is statistically significantly superior to the other three strategies. This is not surprising, considering that the guidelines for building “good concept maps” suggest that the use of links should be restricted to identifying relations between concepts rather than conveying other information about the domain [Novak and Cañas 2008]. Indeed, inspection reveals

Table II. Initial Query Formation Strategies: Mean and Confidence Intervals for the Global Performance Metrics

	CCC (1,736 queries)		CLC (887 queries)		LCL (568 queries)		LLL (296 queries)	
	Mean	95% C.I.	Mean	95% C.I.	Mean	95% C.I.	Mean	95% C.I.
P@10 at level 1	0.579	(0.563, 0.594)	0.530	(0.508, 0.552)	0.434	(0.407, 0.461)	0.294	(0.257, 0.331)
P@10 at level 2	0.353	(0.337, 0.370)	0.293	(0.271, 0.314)	0.233	(0.208, 0.258)	0.236	(0.199, 0.273)
P@10 at level 3	0.311	(0.295, 0.327)	0.253	(0.232, 0.274)	0.192	(0.168, 0.216)	0.191	(0.156, 0.226)
SP@10	0.632	(0.622, 0.643)	0.581	(0.566, 0.595)	0.517	(0.498, 0.535)	0.445	(0.418, 0.472)
Recall	0.268	(0.256, 0.281)	0.209	(0.194, 0.224)	0.194	(0.174, 0.214)	0.240	(0.205, 0.275)

that the few concept maps where linking phrases yield better performance do not meet the guidelines.

A more general overview of the performance of each query formation strategy can be provided by averaging the performance metrics across all the queries from all the concept maps. Because the analyzed ODP topics are organized in a hierarchical structure, we decided to consider three different levels of specificity to compute precision at three different levels, as described next.

—*Level 1*: The first approach takes the first level of the ODP hierarchy to differentiate between topics with the purpose of defining the set of relevant documents for a given concept map. As a result, subtopics such as Science/Technology/Space and Science/Astronomy/Solar_System will be considered part of the same topic, namely Science. Based on this approach, any document associated with the general topic Science is considered relevant for any concept map on a Science subtopic.
—*Level 2*: The second approach considers the second level of the ODP hierarchy to distinguish a topic from another. According to this approach, subtopics such as Science/Technology/Space and Science/Astronomy/Solar_System will be considered different topics, whereas subtopics Health/Conditions_and_Diseases/Sleep_Disorders and Health/Conditions_and_Diseases/Infectious_Diseases will be considered the same.
—*Level 3*: The third approach is the most specific one. This is the approach that has been adopted to compute the results reported in Figure 6. In this case, two subtopics need to agree at the third level of the ODP hierarchy to be considered equivalent.

Table II summarizes these global results. Semantic precision at rank 10 and recall are reported as well. For all the evaluated metrics, except for recall, we can conclude that using only concept labels to form queries results in significantly superior performance. The CCC-type strategy also yields better recall, but the difference with LLL is not statistically significant.

6.2. Evaluating Topic Generation

To perform an objective test of the topic generation process, we used the Mars 2001 knowledge model [Briggs et al. 2004], an expert-generated set of concept maps, as our "gold standard" for an automatic evaluation of EXTENDER's topics. This knowledge model on Mars exploration was created by NASA experts and contains 118 concept maps, presenting an extensive coverage of topics in the field.

The top-level concept map that introduces the topics covered by the knowledge model, and serves as a starting point to navigate to the different maps in the model, was used as the starting point in our tests—corresponding to the user's map under construction when EXTENDER is invoked. EXTENDER's topic extension algorithm was used to produce a collection of topics, without access to any of the other maps in the knowledge model. We used the Google Web API with special permission from Google to carry out our evaluations; non-commercial search engines, such as Faroo (http://www.faroo.com) or Yacy (http://yacy.net), could serve as effective alternatives. As a baseline method for comparison, we implemented a simple algorithm that constructs queries using all the concepts from the given concept map as a starting point, submits them as queries to the Google Web API, and clusters the results to generate topics.

We expected EXTENDER's mechanism to provide results with superior global coherence and coverage for equal numbers of Web queries. An evaluation based on coherence and coverage requires an operational definition of *topic relevance*. Here, we consider the expert-generated Mars 2001 topics as *target topics*, with the relevance of a system-generated topic measured by the accuracy with which it replicates an expert-generated topic. Such an accuracy measure also provides an indication of topic quality, because its results depend on the similarity between EXTENDER's topics and the expert-generated set, which we expect to be of good quality for the domain.

Expert-generated topics are concept maps and therefore they have a rich structure. However, for evaluation purposes, each of these concept maps is characterized as a set of terms representing a relevant topic. This makes it possible to compare expert-generated topics with those topics generated by the evaluated methods. To measure the quality of the generated topics assume that $R = \{r_1, \ldots, r_m\}$ is a target set of relevant topics and $A = \{a_1, \ldots, a_n\}$ is a set of topics generated by the topic-generation strategy under evaluation. We can measure the overlap between EXTENDER's topics and those that an expert chose to include in the knowledge model by a rate function, which measures the proportion of terms in a generated topic that are actually part of a target concept map:

$$\mathbf{Rate}(a_i, r_j) = \frac{|a_i \cap r_j|}{|a_i|}.$$

Similarity between topics a_i and r_j can be measured using, for example, the *Jaccard coefficient*:

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

Then, we can define the *accuracy* of topic a_i with respect to topics in R as follows:

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

The **Accuracy** function measures the precision with which a given topic a_i replicates some topic in a given set of topics R.

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

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

The **Global_Coherence** function measures the fraction of relevant topics generated by the algorithm being evaluated, weighted by the algorithm's level of accuracy replicating the relevant topics.

Global coherence is a generalization of the IR notion of precision, and as such, it has its limitations. This criterion function can be maximized if the system generates a single topic identical to some relevant topic, which clearly does not guarantee acceptable topic generation performance. Hence, a *coverage* factor must be introduced to favor topic-generation strategies that cover many topics of a target set of relevant topics. To address this issue, we define a coverage function as a generalization of the standard IR notion of recall:

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

Table III. How EXTENDER's Topics Approximate Expert Concept Maps

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6
	technology penetration revolutionary systems protection	air dry composition nitrogen oxygen	entry atmospheric system esa level	landing sites specific nasa mars	mars lowell history percival planet	fluvial history erosion activity glaciers
Originating concept map	0.10	0.00	0.25	0.40	0.15	0.30
Climate history	0.00	0.15	0.15	0.05	0.10	**0.70**
Deep access	**0.75**	0.05	0.20	0.15	0.10	0.10
Earth's atmosphere	0.00	**0.50**	0.00	0.00	0.00	0.10
Geologic history	0.00	0.00	0.00	0.05	0.10	0.30
History of water	0.00	0.00	0.10	0.00	0.05	**0.50**
Landers	0.20	0.10	**0.70**	**0.55**	0.05	0.15
Myth and science fiction	0.05	0.00	0.05	0.10	**0.70**	0.05
Pathfinder	0.05	0.05	0.25	0.15	0.10	0.05
Rovers	0.35	0.00	**0.55**	**0.50**	0.10	0.20

Rate values greater or equal to 0.5 are in boldface.

EXTENDER's methods depend on parameters such as the number of iterations, the number of queries submitted from the source concept map and from each generated topic, and the maximum number of topic descendants for each topic. This results in a large parameter space. In practice, however, pragmatic constraints, such as the desire for rapid response and low memory use, suggest restricting some parameters. Our evaluation involved 48 trials, with different settings for EXTENDER's parameters. These tests limited the number of iterations to 3, the number of queries from each new topic to 20, and the number of topic descendants at each stage to 8.

An initial analysis of EXTENDER's performance for a specific trial is depicted in Table III. With three iterations, EXTENDER produced 19 artificial topics, each containing 20 terms. In the table, we only present 10 target concept maps (including the originating concept maps) from the Mars 2001 knowledge model and 6 of the generated topics, each characterized by its 5 terms ranked by the system as most relevant. We also report the values of the **Rate** performance measure for the given concept maps with respect to each of the presented topics.

The observation that different topics are similar to different maps in the Mars 2001 knowledge model is encouraging. It suggests that although EXTENDER preserved the general theme of the originating concept map, it truly created diverse topics. If a topic and some expert map (other than the originating map) are highly similar, then we have a good reason to believe that the topic is a valuable suggestion, because the information that is provided by the topic is new but highly relevant to a topic the expert chose to include.

When comparing the performance of EXTENDER against the baseline, we set the number of queries for the baseline to the total number of queries submitted by EXTENDER. For each trial, EXTENDER and the baseline method used the same similarity threshold and method for merging topics.

Figures 7(a) and (b) compare the performance of EXTENDER's topic generation algorithm to the baseline method in terms of global coherence and coverage. A particular parameter setting corresponds to a trial and is represented by a point. The point's horizontal coordinate corresponds to the performance of EXTENDER for that case, whereas the vertical coordinate corresponds to the performance of the baseline method. We see that for the majority of trials, EXTENDER outperforms the baseline with respect to

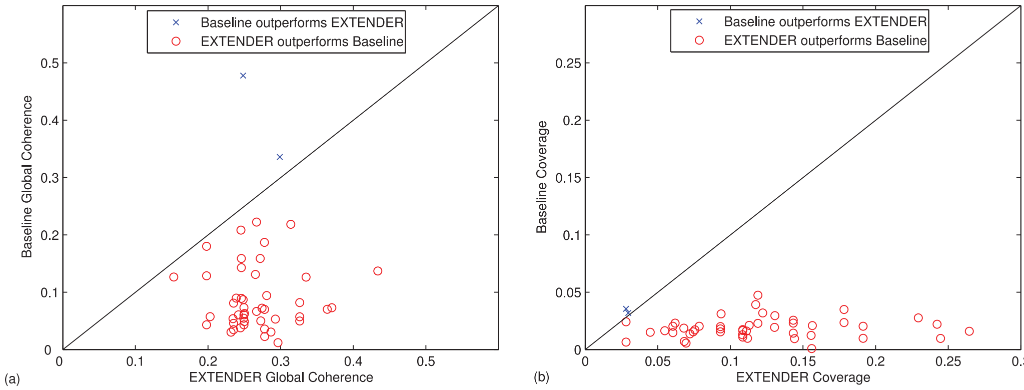

Fig. 7. EXTENDER vs. baseline: (a) global coherence; (b) coverage.

Table IV. Means and Confidence Intervals for Global Coherence and Coverage of EXTENDER and the Baseline Method

	Global coherence		Coverage	
	Mean	95% C.I.	Mean	95% C.I.
EXTENDER	0.252	(0.238, 0.266)	0.120	(0.103, 0.136)
Baseline	0.098	(0.074, 0.122)	0.015	(0.013, 0.018)

both global coherence and coverage. Table IV shows that EXTENDER yields statistically significant improvements over the baseline method across trials.

When we analyzed the relations between parameter settings and EXTENDER's results, we noticed that different parameter settings favor different aspects of EXTENDER's performance. For example, higher thresholds for the curiosity mechanism favor global coherence, whereas lower thresholds favor coverage. Therefore, the interface enables the user to adjust these parameters, focusing on topics more or less similar to the initial map.

7. RELATED WORK

Knowledge capture frameworks have focused largely on the construction of standardized representations. The knowledge modeling community has long been concerned with devising ontologies as formal specifications that machines can read and process [Gruber 1993]. More recently, Semantic Web research has prompted much interest on developing ontologies. Ontology construction is a tedious process; therefore, systems have been built to expedite the design of ontologies and to facilitate sharing and integration of different frameworks. Examples of traditional systems that facilitate collaborative development of ontologies include the *Ontolingua* server [Farquhar et al. 1997], *RiboWeb* [Altman et al. 1999], *Community Web Portals* [Staaba et al. 2000], *OntoShare* [Davies et al. 2003], and the *Protégé* family [Noy et al. 2000]. A review of more recently developed ontology tools can be found in Simperl and Luczak-Rösch [2014]. These tools provide a graphical environment for collaborative ontology development and knowledge acquisition. However, the goal of these tools is to facilitate the construction of standardized representations, whereas the goal of EXTENDER is to provide human-centered support for knowledge extension.

Other tools apply machine learning or text mining techniques in support of knowledge acquisition. For example, in Hsieh et al. [2011], text mining techniques are applied to support the extraction of concepts, instances, and relations from a handbook of a specific domain to quickly construct a basic domain ontology. There are other examples of semi-automatic construction of knowledge representation in the form of ontologies from existing data [Santoso et al. 2011; Gil and Martin-Bautista 2012]. However, these systems aim to aid the construction of formal representations rather than to capture knowledge in a human-friendly form. The CmapTools system can be applied to support ontology generation and extension, as discussed in Eskridge and Hoffman [2012]; thus, EXTENDER could serve that task as well.

Our work relates to methods that exploit context to improve search. Contextual terms can be used for query disambiguation, as discussed in Gao et al. [2010]. Other methods support the query expansion and refinement process through a query or browsing interface requiring explicit user intervention [Carpineto and Romano 2012]. Context-based search systems exploit user interaction with computer applications to determine the user's current task and contextualize information needs. A seminal context-based system is the *Remembrance Agent* [Rhodes and Starner 1996], which operates inside the Emacs text editor and continuously monitors the user's work to find relevant text documents, notes, and emails previously indexed. Another influential context-based assistant is the *Watson* system [Budzik et al. 2001]. *Watson* 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.

Kraft et al. [2006] discuss three commonly adopted approaches for searching in context. The first is a simple query rewriting technique to automatically augment the user query with terms extracted from the search context. The second is a rank-biasing technique that relies on the use of modified search engines to handle more complex query syntaxes. These queries are made up of two parts, the selection part effecting recall, and optional ranking terms only impacting the score of the selected documents. The third is a method referred to as *Iterative Filtering Meta-Search*, in which multiple simple queries are sent to a standard search engine. This method involves two main steps, to generate subqueries reflecting the context at hand and to aggregate the returned results by filtering and re-ranking them.

A recently developed system called *Leibiniz* [Fuxman et al. 2014] introduces the notion of "contextual insights" to provide users with information that is contextually relevant to the content that they are consuming or authoring. Some key aspects of the *Leibiniz* system include the prediction of the user's focus of attention, the addition of context terms to the query, and the adaptation of the results via post-processing. As opposed to these systems, EXTENDER's goal is not to suggest the *most similar* material, but rather to suggest topics that *go beyond* previously captured information.

Our research on topic extraction also shares insights and motivations with proposals aimed at clustering search results [Zamir and Etzioni 1999; Ferragina and Gulli 2008; Carpineto et al. 2009; Song et al. 2014]. However, in contrast to our approach, these systems provide browsing interfaces requiring explicit user intervention. In addition, their goal is to help users focus on specific information and to remove alternatives, rather than to discover novel but related material.

8. DISCUSSION

The goal of topic suggestion is to aid users in pursuing useful *new directions* relevant to their work. The World Wide Web provides a rich source of information on potential new topics to include in a knowledge model. Searching the Web to support this knowledge extension process presents new challenges, unaddressed by traditional IR systems. Specifically, making full use of the information available in these knowledge models requires:

—*Search methods that can reflect extensive contextual information* (instead of attempting to summarize context in a small number of terms). For knowledge model extension, the knowledge model under construction provides a rich context that can be exploited for information filtering, term-weight reinforcement, and query refinement.
—*Methods for topic search* (instead of document search). Users selecting topics to include in a knowledge model will be aided by search methods that directly generate characterizations of possible topics—which may span individual documents—rather than simply presenting sets of documents.
—*Methods for searching open collections of documents* (instead of a pre-defined and pre-analyzed collection). In Web-based knowledge extension tasks, the search space is the full Web, and analysis must be limited to a small collection of documents—incremental retrievals—that is built up over time and changes dynamically. Unlike traditional IR schemes, which analyze a predefined collection of documents and search that collection, Web-based knowledge extension must rely on methods that use limited information to assess the importance of documents and to manage decisions about which documents to retain for further analysis, which ones to discard, and which additional queries to generate.

This article addresses these issues, identifies general desiderata for topic generation, and presents a domain-independent topic-generation algorithm developed for supporting concept-map-based knowledge modeling. The process reflects the knowledge modeling context through an iterative process of topic generation, Web search, and context-based filtering. An experimental study shows that this approach significantly outperforms a baseline at recovering topics close to those of an expert's hand-coded knowledge model.

The evaluations reported in this article show, in an objective way, that EXTENDER is able to suggest topics similar to expert-generated concept maps. These evaluations have been designed specifically to avoid relying on a human subject confirming whether or not the generated topics are good. As part of our future work we plan to collaborate with specialists in user experience design with the purpose of carrying out an incremental human-in-the-loop experiment to evaluate the usefulness of EXTENDER's suggestions. It has been long recognized in the research literature that any evaluation that requires explicit or implicit feedback from human subjects on the usefulness of a list of suggestions may suffer from several response biases that can damage the validity of the study [Cannell et al. 1981; Hufnagel and Conca 1994]. The response-bias problem has been more recently investigated in different recommendation scenarios [Park et al. 2014; Hofmann et al. 2014], which are closely related to EXTENDER's task of recommending topics to users. As a consequence, our evaluations will give special attention to reducing different types of user biases. We foresee that these evaluations will be highly challenging. On the one hand, in an evaluation based on explicit feedback we expect a large impact resulting from the acquiescence bias, in which some human subjects have a tendency to accept most suggestions. On the other hand, we anticipate that in an evaluation based on implicit feedback the expert may not click on a specific topic because the expert wants to work on something else. However, the expert may decide later to create a concept map on the topic EXTENDER originally suggested. In other words, it is likely that some experts dismiss suggested topics but later adopt them. In that case it will be necessary to reevaluate the rating the expert has given to the suggested topics.

A limitation of the proposed approach is that the topics generated by the EXTENDER system lack the structure of human-generated concept maps. In particular, they do not provide the relations existing among concepts, which are naturally represented by linking phrases in concept maps. Discovering potentially useful links between concepts is a highly challenging task. We plan to address this problem as part of our future work, by using other information sources such as Wikipedia, WordNet, and domain specific ontologies to discover semantic relations among the suggested terms. Some approaches that address highly related problems are presented in Ruiz-Casado et al. [2005], Suchanek et al. [2008], Weikum and Theobald [2010], Panchenko et al. [2012], and Arnold and Rahm [2015], and will be used as a basis for our methods.

ACKNOWLEDGMENTS

We gratefully acknowledge the contributions of our collaborators Alberto Cañas and the IHMC CmapTools team. We thank Google for the use of their Web APIs service.

REFERENCES

Andy Aiken and Derek Sleeman. 2003. Refiner++: A knowledge acquisition and refinement tool. In *Proceedings of the KCAP Workshop on Capturing Knowledge from Domain Experts: Progress & Prospects (KCAP'03)*, Derek Sleeman and Yolanda Gil (Eds.). Retrieved from http://www.csd.abdn.ac.uk/~sleeman/published-papers/p141.pdf.

Russ B. Altman, Michael Bada, Xiaoqian J. Chai, Michelle Whirl Carillo, Richard O. Chen, and Neil F. Abernethy. 1999. RiboWeb: An ontology-based system for collaborative molecular biology. *IEEE Intelligent Systems* 14, 5 (1999), 68–76.

Giambattista Amati. 2003. *Probability Models for Information Retrieval based on Divergence from Randomness*. Ph.D. Dissertation. University of Glasgow.

Patrick Arnold and Erhard Rahm. 2015. Automatic extraction of semantic relations from Wikipedia. *International Journal on Artificial Intelligence Tools* 24, 2 (2015), 1540010.

Josianne Basque, Clément Imbeault, Béatrice Pudelko, and Michel Léonard. 2004. Collaborative knowledge modeling between experts and novices: A strategy to support transfer of expertise in an organization. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the 1st International Conference on Concept Mapping*, A. J. Cañas, J. D. Novak, and F. González (Eds.). Universidad Pública de Navarra, 75–81.

Jim Blythe, Jihie Kim, Surya Ramachandran, and Yolanda Gil. 2001. An integrated environment for knowledge acquisition. In *Proceedings of the 6th International Conference on Intelligent User Interfaces*. ACM, 13–20. DOI: http://doi.acm.org/10.1145/359784.359819

Geoffrey Briggs, David Shamma, Alberto Cañas, Roger Carff, Jeffrey Scargle, and Joseph Novak. 2004. Concept maps applied to Mars exploration public outreach. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the 1st International Conference on Concept Mapping*, A. J. Cañas, J. D. Novak, and F. González (Eds.). Universidad Pública de Navarra, 125–133.

Jay Budzik, Kristian J. Hammond, and Larry Birnbaum. 2001. Information access in context. *Knowledge-Based Systems* 14 (2001), 37–53.

Karla L. Caballero Barajas and Ram Akella. 2013. Incorporating statistical topic models in the retrieval of health care documents. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*. ELRA. Retrieved from http://clefpackages.elra.info/clefehealthtask3/workingnotes/CLEFeHealth2013_Lab_Working_Notes/TASK_3/CLEF2013wn-CLEFeHealth-CaballeroEt2013.pdf.

Osvaldo Cairó and Silvia Guardati. 2012. The KAMET II methodology: Knowledge acquisition, knowledge modeling and knowledge generation. *Expert Systems with Applications* 39, 9 (2012), 8108–8114.

Alberto Cañas, David Leake, and Ana Maguitman. 2001. Combining concept mapping with CBR: Experience-based support for knowledge modeling. In *Proceedings of the 14th International Florida Artificial Intelligence Research Society Conference*. AAAI Press, 286–290.

Alberto J. Cañas, John Coffey, Thomas Reichherzer, Greg Hill, Niranjan Suri, Roger Carff, Tim Mitrovich, and Derek Eberle. 1998. El-tech: A performance support system with embedded training for electronics technicians. In *Proceedings of the 11th International Florida Artificial Intelligence Research Society Conference*. AAAI Press, 79–83.

Alberto J. Cañas, Greg Hill, Roger Carff, Niranjan Suri, James Lott, Tom Eskridge, Gloria Gómez, Mario Arroyo, and Rodrigo Carvajal. 2004. CmapTools: A knowledge modeling and sharing environment. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the 1st International Conference on Concept Mapping*, A. J. Cañas, J. D. Novak, and F. González (Eds.). Universidad Pública de Navarra, 125–133.

Alberto J. Cañas and Joseph D. Novak. 2014. Concept mapping using CmapTools to enhance meaningful learning. In *Knowledge Cartography*. Springer, 23–45.

Alberto J. Cañas, Joseph D. Novak, and Jacqueline Vanhear. 2012. *Concept Maps: Theory, Methodology, Technology. Proceedings of the 5th International Conference on Concept Mapping*. Veritas Press.

Charles F. Cannell, Peter V. Miller, and Lois Oksenberg. 1981. Research on interviewing techniques. *Sociological Methodology* 12, 4 (1981), 389–437.

Claudio Carpineto, Stanislaw Osiński, Giovanni Romano, and Dawid Weiss. 2009. A survey of web clustering engines. *Computing Surveys* 41, 3 (2009), 17.

Claudio Carpineto and Giovanni Romano. 2012. A survey of automatic query expansion in information retrieval. *Computing Surveys* 44, 1 (2012), 1–50. DOI:http://dx.doi.org/10.1145/2071389.2071390

Rocío L. Cecchini, Carlos M. Lorenzetti, Ana G. Maguitman, and Filippo Menczer. 2011. A semantic framework for evaluating topical search methods. *CLEI Electronic Journal* 14, 1 (April 2011). Retrieved from http://www.clei.cl/cleiej/paper.php?id=211.

John Coffey, Thomas Reichherzer, Bernd Owsnicki-Klewe, and Norman Wilde. 2012. Automated concept map generation from services-oriented architecture artifacts. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the 5th International Conference on Concept Mapping*. University of Malta, Valleta, Malta.

John W. Coffey. 1999. *Institutional Memory Preservation at NASA Glenn Research Center*. Unpublished technical report. NASA Glenn Research Center, Cleveland, OH.

Paulo R. M. Correia, Maria E. I. Malachias, Alberto J. Cañas, and Joseph D. Novak. 2014. *Concept Maps: Theory, Methodology, Technology. Proceedings of the 6th International Conference on Concept Mapping*. Universidade de Sao Pãulo.

John Davies, Alistair Duke, and York Sure. 2003. OntoShare: A knowledge management environment for virtual communities of practice. In *Proceedings of the International Conference on Knowledge Capture*. ACM, 20–27. DOI:http://doi.acm.org/10.1145/945645.945652

Inderjit S. Dhillon. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM Press, 269–274. DOI:http://dx.doi.org/10.1145/502512.502550

Tom Eskridge and Robert Hoffman. 2012. Ontology creation as a sensemaking activity. *IEEE Intelligent Systems* 27, 5 (Sep. 2012), 58–65. DOI:http://dx.doi.org/10.1109/MIS.2012.101

Adam Farquhar, Richard Fikes, and James Rice. 1997. The Ontolingua server: A tool for collaborative ontology construction. *International Journal of Human-Computer Studies* 46, 6 (1997), 707–727.

Paolo Ferragina and Antonio Gulli. 2008. A personalized search engine based on Web-snippet hierarchical clustering. *Software: Practice and Experience* 38, 2 (2008), 189–225.

Lee A. Freeman and Leonard M. Jessup. 2004. The power and benefits of concept mapping: Measuring use, usefulness, ease of use, and satisfaction. *International Journal of Science Education* 26 (2004), 151–169.

Ariel Fuxman, Patrick Pantel, Yuanhua Lv, Ashok Chandra, Pradeep Chilakamarri, Michael Gamon, David Hamilton, Bernhard Kohlmeier, Dhyanesh Narayanan, Evangelos Papalexakis, and Bo Zhao. 2014. Contextual insights. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion (WWW Companion'14)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 265–266. DOI:http://dx.doi.org/10.1145/2567948.2577280

Byron J. Gao, David C. Anastasiu, and Xing Jiang. 2010. Utilizing user-input contextual terms for query disambiguation. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING'10)*. Association for Computational Linguistics, Stroudsburg, PA, 329–337.

Alexander Garcia Castro, Philippe Rocca-Serra, Robert Stevens, Chris Taylor, Karim Nashar, Mark Ragan, and Susanna-Assunta Sansone. 2006. The use of concept maps during knowledge elicitation in ontology development processes – The nutrigenomics use case. *BMC Bioinformatics* 7, 267 (May 2006), 1–14. DOI:http://dx.doi.org/10.1186/1471-2105-7-267

Richard J. Gil and Maria J. Martin-Bautista. 2012. A novel integrated knowledge support system based on ontology learning: Model specification and a case study. *Knowledge-Based Systems* 36 (2012), 340–352. DOI:http://dx.doi.org/10.1016/j.knosys.2012.07.007

Yolanda Gil. 1994. Knowledge refinement in a reflective architecture. In *Proceedings of the 12th National Conference on Artificial Intelligence*. AAAI Press.

Thomas R. Gruber. 1993. Towards principles for the design of ontologies used for knowledge sharing. In *Formal Ontology in Conceptual Analysis and Knowledge Representation*, N. Guarino and R. Poli (Eds.). Kluwer Academic Publishers, Deventer, The Netherlands.

David Gunning, Vinay K. Chaudhri, Peter E. Clark, Ken Barker, Shaw-Yi Chaw, Mark Greaves, Benjamin Grosof, Alice Leung, David D. McDonald, Sunil Mishra, and others. 2010. Project Halo update—Progress toward digital Aristotle. *AI Magazine* 31, 3 (2010), 33–58.

Frederick Hayes-Roth, Donald A. Waterman, and Douglas B. Lenat. 1983. *Building Expert Systems*. Addison-Wesley.

Robert R. Hoffman, John W. Coffey, Kenneth M. Ford, and Mary Jo Carnot. 2001. Storm-LK: A human-centered knowledge model for weather forecasting. In *Proceedings of the 45th Annual Meeting of the Human Factors and Ergonomics Society*.

Katja Hofmann, Anne Schuth, Alejandro Bellogin Kouki, and Maarten de Rijke. 2014. User behavior and bias in click-based recommender evaluation. In *Proceedings of European Conference on Information Retrieval (ECIR'14)*, Lecture Notes in Computer Science. Springer.

Shang-Hsien Hsieh, Hsien-Tang Lin, Nai-Wen Chi, Kuang-Wu Chou, and Ken-Yu Lin. 2011. Enabling the development of base domain ontology through extraction of knowledge from engineering domain handbooks. *Advanced Engineering Informatics* 25, 2 (2011), 288–296. DOI:http://dx.doi.org/10.1016/j.aei.2010.08.004

Ellen M. Hufnagel and Christopher Conca. 1994. User response data: The potential for errors and biases. *Information Systems Research* 5, 1 (1994), 48–73.

Sparck K. Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation* 28 (1972), 11–21.

Leonard Kaufman and Peter J. Rousseeuw. 1989. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York.

Reiner Kraft. 2011. Search systems and methods using in-line contextual queries. US Patent 7,958,115.

Reiner Kraft, Farzin Maghoul, Chi Chao Chang, and Ravi Kumar. 2006. Searching with context. In *Proceedings of the 15th International World Wide Web Conference (WWW'15)*.

David Leake, Ana Maguitman, and Thomas Reichherzer. 2004. Understanding knowledge models: Modeling assessment of concept importance in concept maps. In *Proceedings of CogSci-2004*.

David Leake, Ana Maguitman, and Thomas Reichherzer. 2014. Experience-based support for human-centered knowledge modeling. *Knowledge-Based Systems* 68 (2014), 77–87. DOI:http://dx.doi.org/10.1016/j.knosys.2014.01.013

David Leake, Ana Maguitman, Thomas Reichherzer, Alberto Cañas, Marco Carvalho, Marco Arguedas, Sofia Brenes, and Tom Eskridge. 2003. Aiding knowledge capture by searching for extensions of knowledge models. In *Proceedings of International Conference on Knowledge Capture (KCAP'03)*. ACM.

David Leake and David Wilson. 2001. A case-based framework for interactive capture and reuse of design knowledge. *Applied Intelligence* 14 (2001), 77–94.

Avishay Livne, Vivek Gokuladas, Jaime Teevan, Susan T. Dumais, and Eytan Adar. 2014. CiteSight: Supporting contextual citation recommendation using differential search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'14)*. ACM, 807–816.

Carlos M. Lorenzetti and Ana G. Maguitman. 2009. A semi-supervised incremental algorithm to automatically formulate topical queries. *Information Sciences* 179, 12 (2009), 1881–1892. DOI:http://dx.doi.org/10.1016/j.ins.2009.01.029 (Including Special Issue on Web Search).

Ana Maguitman, David Leake, Thomas Reichherzer, and Filippo Menczer. 2004. Dynamic extraction of topic descriptors and discriminators: Towards automatic context-based topic search. In *Proceedings of the 13th Conference on Information and Knowledge Management (CIKM)*. ACM, New York, 463–472.

Ana G. Maguitman, Filippo Menczer, Heather Roinestad, and Alessandro Vespignani. 2005. Algorithmic detection of semantic similarity. In *Proceedings of the 14th International Conference on World Wide Web (WWW'05)*. ACM, New York, NY, 107–116. DOI:http://dx.doi.org/10.1145/1060745.1060765

John W. Mohr and Petko Bogdanov. 2013. Introduction—Topic models: What they are and why they matter. *Poetics* 41, 6 (2013), 545–569. DOI:http://dx.doi.org/10.1016/j.poetic.2013.10.001

Brian M. Moon, Robert R. Hoffman, Joseph D. Novak, and Alberto J. Cañas. 2011. *Applied Concept Mapping: Capturing, Analyzing and Organizing Knowledge*. CRC Press, New York.

Joseph Novak. 1977. *A Theory of Education*. Cornell University Press, Ithaca, IL.

Joseph Novak and Alberto Cañas. 2008. *The Theory Underlying Concept Maps and How To Construct Them*. Technical Report. Florida Institute for Human and Machine Cognition. Retrieved from http://cmap.ihmc.us/Publications/ResearchPapers/TheoryUnderlyingConceptMaps.pdf.

Joseph Novak and D. Bob Gowin. 1984. *Learning How to Learn*. Cambridge University Press.

Natalya Fridman Noy, Ray W. Fergerson, and Mark A. Musen. 2000. The knowledge model of Protégé-2000: Combining interoperability and flexibility. In *Proceedings of European Workshop on Knowledge Acquisition, Modeling and Management (EKAW'00)*.

Iadh Ounis, Christina Lioma, Craig Macdonald, and Vassilis Plachouras. 2007. Research directions in Terrier: A search engine for advanced retrieval on the web. *Novatica/UPGRADE Special Issue on Web Information Access*, Ricardo Baeza-Yates et al. (Eds.), Invited Paper VIII, 1 (Feb. 2007), 49–56.

Cosimo Palmisano, Alexander Tuzhilin, and Michele Gorgoglione. 2008. Using context to improve predictive modeling of customers in personalization applications. *IEEE Transactions on Knowledge and Data Engineering* 20, 11 (2008), 1535–1549.

Alexander Panchenko, Sergey Adeykin, Alexey Romanov, and Pavel Romanov. 2012. Extraction of semantic relations between concepts with KNN algorithms on Wikipedia. In *Concept Discovery in Unstructured Data Workshop (CDUD) of International Conference on Formal Concept Analysis*. 78–88.

Kyung-Wha Park, Byoung-Hee Kim, Tae-Suh Park, and Byoung-Tak Zhang. 2014. Uncovering response biases in recommendation. In *Workshops at the 28th AAAI Conference on Artificial Intelligence. Multidisciplinary Workshop on Advances in Preference Handling*. AAAI Press.

Thomas Reichherzer and David Leake. 2006a. Towards automatic support for augmenting concept maps with documents. In *Proceedings of the 2nd International Conference on Concept Mapping*.

Thomas Reichherzer and David Leake. 2006b. Understanding the role of structure in concept maps. In *Proceedings of the 28th Annual Conference of the Cognitive Science Society*. 2004–2009.

Bradley J. Rhodes and Thad Starner. 1996. The remembrance agent: A continuously running automated information retrieval system. In *Proceedings of the 1st International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology (PAAM'96)*. AAAI Press, 487–495.

Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. 2005. Automatic extraction of semantic relationships for WordNet by means of pattern learning from Wikipedia. In *Natural Language Processing and Information Systems*. Springer, 67–79.

Gerard Salton. 1971. *The SMART Retrieval System – Experiments in Automatic Document Processing*. Prentice Hall.

Gerard Salton. 1979. Mathematics and information retrieval. *Journal of Documentation* 35, 1 (1979), 1–29. DOI:http://dx.doi.org/10.1108/eb026671

Gerard Salton and Michael E. Lesk. 1968. Computer evaluation of indexing and text processing. *Journal of the ACM* 15, 1 (Jan. 1968), 8–36. DOI:http://dx.doi.org/10.1145/321439.321441

Gerard Salton and Chung-Shu Yang. 1973. On the specification of term values in automatic indexing. *Journal of Documentation* 29 (1973), 351–372.

Heru Agus Santoso, Su-Cheng Haw, and Ziyad T. Abdul-Mehdi. 2011. Ontology extraction from relational database: Concept hierarchy as background knowledge. *Knowledge-Based Systems* 24, 3 (2011), 457–464. DOI:http://dx.doi.org/10.1016/j.knosys.2010.11.003

Beat A. Schwendimann. 2014. Making sense of knowledge integration maps. In *Digital Knowledge Maps in Education*. Springer, 17–40.

Nigel Shadbolt and Paul R. Smart. 2015. Knowledge elicitation: Methods, tools and techniques. In *Evaluation of Human Work* (4th ed.), John R. Wilson and Sarah Sharples (Eds.). CRC Press.

Rushdi Shams and Adel Elsayed. 2008. Development of a conceptual structure for a domain-specific corpus. In *Proceedings of the 3rd International Conference on Concept Mapping, Tallinn, Estonia & Helsinki, Finland, 2008*.

Robin Sibson. 1973. SLINK: An optimally efficient algorithm for the single-link cluster method. *Computer Journal* 16, 1 (1973), 30–34.

Elena Simperl and Markus Luczak-Rösch. 2014. Collaborative ontology engineering: A survey. *The Knowledge Engineering Review* 29, 1 (2014), 101–131.

Dallas Snider, John Coffey, Thomas Reichherzer, Norman Wilde, Joe Vandeville, Northrop Grumman, Allison Heinen, and Sarah Pramanik. 2014. Using concept maps to introduce software security assurance cases. *CrossTalk* 27, 5 (2014), 4–9.

Qiuxia Song, Jin Liu, Ming Ni, Liang Chen, and Jialiang Shen. 2014. Sorting topic specific web pages based on ontology knowledge. In *Proceedings of the 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'14)*. IEEE, 880–883. DOI:http://dx.doi.org/10.1109/IIH-MSP.2014.222

Steffen Staaba, Jürgen Angeleb, Stefan Deckera, Michael Erdmanna, Andreas Hothoa, Alexander Maedchea, Hans-Peter Schnurra, Rudi Studera, and York Surea. 2000. AI for the web – Ontology-based community web portals. In *Proceedings of the National Conference on Artificial Intelligence (AAAI'00)*. MIT Press, Menlo Park.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO: A large ontology from Wikipedia and WordNet. *Web Semantics: Science, Services and Agents on the World Wide Web* 6, 3 (2008), 203–217.

Quan Wang, Jun Xu, Hang Li, and Nick Craswell. 2013. Regularized latent semantic indexing: A new approach to large-scale topic modeling. *ACM Transactions on Information Systems* 31, 1 (Jan. 2013), 5:1–5:44. DOI:http://dx.doi.org/10.1145/2414782.2414787

Gerhard Weikum and Martin Theobald. 2010. From information to knowledge: Harvesting entities and relationships from web sources. In *Proceedings of the 29th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'10)*. ACM, New York, NY, 65–76. DOI:http://dx.doi.org/10.1145/1807085.1807097

S. K. Michael Wong, Wojciech Ziarko, Vijay V. Raghavan, and P. C. N. Wong. 1987. On modeling of information retrieval concepts in vector spaces. *ACM Transactions on Database Systems* 12, 2 (Jun. 1987), 299–321. DOI:http://dx.doi.org/10.1145/22952.22957

Clement T. Yu, K. Lam, and Gerard Salton. 1982. Term weighting in information retrieval using the term precision model. *Journal of the ACM* 29, 1 (Jan. 1982), 152–170. DOI:http://dx.doi.org/10.1145/322290.322300

Oren Zamir and Oren Etzioni. 1999. Grouper: A dynamic clustering interface to web search results. *Computer Networks* 31, 11–16 (1999), 1361–1374.

Received April 2015; revised May 2016; accepted September 2016