An Ontology-based Framework for Knowledge Retrieval

Xiaohui Tao, Yuefeng Li, Ning Zhong*, Richi Nayak
Faculty of Information Technology, Queensland University of Technology, Australia
*Department of Systems and Information Engineering, Maebashi Institute of Technology, Japan
{x.tao, y2.li, r.nayak}@qut.edu.au, *zhong@maebashi-it.ac.jp

Abstract

Retrieving accurate information from the Web is a great challenge to users. The existing information retrieval systems are mostly term-based and thus need to be enhanced toward knowledge-based. User information needs need to be better captured in order to deliver personalized search results. In this paper, an ontology-based framework is proposed for capturing user information needs using a world knowledge base and the user's local instance repository. The framework aims to discover a user's background knowledge for knowledge retrieval. The evaluation result is encouraging, in which the proposed model achieved the same performance as a manual user model.

1. Introduction

In the past decades, the Web information has exploded rapidly. How to gather useful information from the Web nowadays becomes a challenging issue. Attempting to solve this problem, many information retrieval (IR) systems have been proposed and made great achievements. However, there is still not a solution to the challenge eventually [2]. The IR systems are mostly based on keyword-match techniques and suffer from the problems of information mismatching and overloading. The information needs may be expressed in different queries because of different user perspectives, background knowledge, terminological habits and vocabulary. Thus, if a user's background knowledge is discovered, more accurate information can be retrieved.

However, discovering user background knowledge through a given query is difficult. When users read through the content of a document, they can easily find out if it is interesting or not. This is because users implicitly have a knowledge system built based on their background knowledge [7]. By re-building this knowledge system, a user's background knowledge can expect to be discovered, and thus better IR performance can be achieved. This route is suggested by [15] as future knowledge retrieval systems.

In this paper, we introduce a knowledge retrieval framework for ontology-based information gathering, and propose an approach to rebuild users' knowledge systems. A user's mental model and querying model are formalized, aiming to describe the process of how an information need is transformed into a query. In response to the query, the user background knowledge is discovered from the world knowledge base and user's local instance repository. Based on these, a personalized ontology is constructed to simulate the user's mental model and capture the information need. The semantic relations of hypernym/hyponym, holonym/meronym and synonym are specified in the ontology. The proposed approach is evaluated by comparing to a manual model that discovers knowledge by linguists, and the evaluation result is promising. The ontology-based knowledge retrieval framework is a novel contribution to knowledge engineering and Web information retrieval.

The paper is organized as follows. Section 2 presents related work. In Section 3, we introduce the ontology-based knowledge retrieval framework. The evaluation of our proposal is described in Section 4, and the results are discussed in Section 5. Finally, Section 6 makes conclusions.

2. Related Work

Ontologies have been used by many groups for personalized information retrieval. Tran et al. [14] introduced an approach to translate keyword queries to DL conjunctive queries and used ontologies to describe a user's background knowledge. Gauch et al. [4] learned ontologies for users in order to specify their personalized preferences and interests in Web search. King et al. [6] developed an ontology based on the Dewey Decimal Classification for distributed IR systems. However, these works have problems that either the volume of knowledge covered in ontologies are limited or the knowledge specified is not clear adequately.

Learning an ontology to specify knowledge is challenging. Maedche [9] formally defined an ontology as a 5-tuple of concepts, hierarchical relations, plain relations, instances and axioms. He also proposed an ontology learning framework using semi-automatic ontology construction tools with human intervention. However, the human intervention increases the cost in this framework. Aiming to reduce the cost, Liu and Singh [8] developed *ConceptNet* ontology by smartly using free contributions from Web users. However, as a trade-off, the knowledge specified in *ConceptNet* is not expert evaluated. These works need to be improved for knowledge acquisition.

Many other works attempted to learn ontologies automatically. Li and Zhong [7] mined patterns from Web contents and used association rules for ontology learning. Web content mining techniques were also used by Jiang and Tan [5] to discover semantic knowledge from domain-specific documents for ontology learning. Dou et al. [3] proposed a framework for developing domain ontologies using pattern decomposition, clustering/classification, and association rules mining techniques. However, as pointed out by [7], these works cover only a limited number of concepts and specify only simple "super-class" and "sub-class" relations. They suffer from the problem of inadequate knowledge specification.

In summary, there still remains a research gap in learning an ontology to describe user background knowledge in IR. This motivates our research work presented in this paper.

3. Ontology-based Retrieval Framework

The ontology-based knowledge retrieval framework consists of four models: a user's mental model and querying model, a computer model, and an ontology model. A user's mental model is her (his) background knowledge system. A querying model is a user's translation of an information need generated from her (his) mental model. The computer model constructs an ontology for the user. The constructed ontology is the ontology model aiming to simulate the user's mental model in IR.

3.1. Mental Model

A search task starts from a user's information need. From observations, when a user was in need of some information and commencing a search task, we found that the user usually fell into one of the following situations:

- she (he) knew nothing about that information;
- she (he) had tried but failed to infer that information from his (her) already possessed knowledge;
- she (he) might know something about that information but was not sure, so she (he) needed to confirm it.

Based on the findings, apparently a user holds a repository in her (his) brain that stores the knowledge she (he) possesses, that is why a user can check if knowing something or not. The second finding suggests that the knowledge possessed by a user may be linked each other, so that a user can perform an inference task from what is known to what is unknown. The last finding indicates that a user actually holds a confidence rate to the knowledge she (he) possesses, so that the user knows that she (he) is certain or uncertain about it. However, the confidence rate may be implicit because a user may not be able to express it clearly. Apparently, a user has an implicit knowledge system in brain for information search.

Thus, although the mechanism of a user's brain-working in Web search has not yet been clearly understood, we can at least have the following assumption:

Assumption 1 *A user has a knowledge repository, in which*

- *the stored knowledge is embedded in an association structure;*
- *the stored knowledge is associated with implicit confidence rates.*

Based on the assumption, by calling a user's implicit knowledge system as a *mental model*, we formalize it as follows:

Definition 1 *A user's mental model is a 3-tuple* $\mathcal{U} :\approx \langle \mathcal{K}, \widehat{\mathcal{B}}, \mathcal{G} \rangle$*, where*

- $\mathcal{K}$ *is a non-empty set of pairs* $\{\langle k, w_k \rangle\}$*, where* k *is a primitive knowledge unit possessed by the user and* w_k *is the user's confidence rate on* k*;*
- $\widehat{\mathcal{B}}$ *is the backbone of the mental model that frames the association structure of knowledge units;*
- $\mathcal{G}$ *is a set of gaps* $\{g_1, g_2, \ldots, g_i\}$ *existing on* $\widehat{\mathcal{B}}$*, in which each gap* g *is a knowledge unit that the user does not possess.*

Note that we use $:\approx$ instead of $:=$ because this definition is given under Assumption 1, which is based on observations and cannot be proved in laboratories currently.

3.2. Querying Model

Filling a knowledge gap in the mental model triggers a user's search task and becomes the user's information need. This implicit g in $\mathcal{U}$ is expressed in an explicit short phrase by the user through her (his) own language, e.g. English. In IR, we call this phrase as a query, which is a set of terms. We formally describe a user's query as a querying model in our knowledge retrieval framework:

Definition 2 *A user querying model* $\mathcal{Q}$ *is a set of terms* $\{t | t \in \mathcal{L}_u\}$*, in which elements are primitive units in the user's language* $\mathcal{L}_u$*.*

Generating a query means the process of translating an implicit knowledge gap, $g \in \mathcal{G}$ in a user's mental model $\mathcal{U}$, to a set of explicit terms in $\mathcal{Q}$. In contrast, capturing an information need means the inverse process of translation from a $\mathcal{Q}$ back to a $g \in \mathcal{G}$ in $\mathcal{U}$. However, users may use different terms to generate queries, even for the same information need, because of user perspectives, terminological habits and vocabulary. Thus, capturing a user's information need through a given query only is extremely difficult. However, if the user background knowledge can be discovered, capturing the accurate information need is possible. In the following sections we discuss how this can be done.

3.3. Computer Model

The computer model aims to simulate a user's mental model. The computer model discovers a user's background knowledge associated with an information need. For the sake of explanation, we first formalize the computer model:

Definition 3 *A computer model $\mathcal{C}$ is a 3-tuple $\mathcal{C} := \langle \mathbb{WKB}, \mathbb{LIR}, \mathbb{F} \rangle$, where*

- *$\mathbb{WKB}$ is a world knowledge base that frames a user's background knowledge;*
- *$\mathbb{LIR}$ is a user's local instance repository, in which the elements cite the knowledge in $\mathbb{WKB}$;*
- *$\mathbb{F}$ is a set of functions, inferences and algorithms that build an ontology for a user using $\mathbb{WKB}$ and $\mathbb{LIR}$.*

To simulate a user's mental model $\mathcal{U}$, an ontology is constructed based on the $\mathbb{WKB}$ and personalized using the $\mathbb{LIR}$, co-responding to a querying model $\mathcal{Q}$ for a $g \in \mathcal{G}$.

3.3.1 Constructing Ontologies using $\mathbb{WKB}$

The world knowledge base is a knowledge frame describing and specifying the background knowledge possessed by humans. In this paper, we assume the existence of a world knowledge base and use a *subject* as a primitive knowledge unit in the $\mathbb{WKB}$. Subjects in the $\mathbb{WKB}$ are linked by semantic relations. Two semantic relations are postulated existing in the $\mathbb{WKB}$: *hypernym/hyponym* and *holonym/meronym*. Hypernym/hyponym (usually called *is-a*) relations describe the situation that the semantic range referred by a hyponym is within that of its hypernym, e.g. "car" is a hyponym of "automobile", and "automobile" is a hypernym of "car", they are on different levels of abstraction (or concretion). In this paper, we treat hypernym/hyponym as one single semantic relation, as they are just the two sides of one coin. Differently, holonym/meronym (usually called *part-of*) relations define the relationship between a (holonym) subject denoting the whole and a (meronym) subject denoting a part of, or a member of, the whole, e.g. a "tyre" is a meronym of its holonym "car". Again, we treat holonym/meronym as one relation. The world knowledge base is formalized as:

Definition 4 *Let $\mathbb{WKB}$ be a world knowledge base, which is a directed acyclic graph consisting of a set of subjects linked by their semantic relations, $\mathbb{WKB}$ is formally defined as a 2-tuple $\mathbb{WKB} := \langle \mathbb{S}, \mathbb{R} \rangle$, where*

- *$\mathbb{S}$ is a set of subjects $\mathbb{S} = \{s_1, s_2, \cdots, s_m\}$, in which each element is a 2-tuple $s := \langle label, \sigma \rangle$, where label is the name of s and $label(s) = \{t_1, t_2, \ldots, t_j | t \in \mathcal{L}_u\}$; and σ is a signature mapping defining a set of subjects that directly link to s, and $\sigma(s) \subseteq \mathbb{S}$;*
- *$\mathbb{R}$ is a set of relations $\mathbb{R} = \{r_1, r_2, \cdots, r_n\}$, in which each element is a 2-tuple $r := \langle r_\nu, r_\tau \rangle$, where $r_\nu \subseteq \mathbb{S} \times \mathbb{S}$ and r_τ is a relation type of hypernym/hyponym or holonym/meronym. For each $(s_x, s_y) \in r_\nu$, s_x is the subject who holds the r_τ of relation to s_y, e.g. s_x is a hypernym of s_y.*

The relevance of a subject to a query in $\mathbb{WKB}$ is determined using the syntax-matching mechanism, because they are both represented by a set of terms in the user's language $\mathcal{L}_u$ (see Definition 2 and 4). We use $sim(s, \mathcal{Q})$ to specify the relevance of a subject $s \in \mathbb{S}$ to $\mathcal{Q}$:

$$sim(s, \mathcal{Q}) = |label(s) \cap \mathcal{Q}|. \tag{1}$$

A subject with $sim(s, \mathcal{Q}) > 0$ is a positive subject relevant to $\mathcal{Q}$, otherwise a non-relevant negative subject.

Let $\mathcal{S}$ be a subject set dealing with $\mathcal{Q}$, $\mathcal{R}$ be a relation set specifying the semantic relations existing in $\mathcal{S}$. The positive subjects in $\mathbb{S}$ are first extracted for $\mathcal{S}$. For each $s \in \mathcal{S}$, the subjects in its $\sigma(s)$ are also extracted for $\mathcal{S}$, along with their associated semantic relations $r \in \mathbb{R}$ to s extracted for $\mathcal{R}$. The extraction is iteratively conducted for three times, as we believe that any subjects out of that range from a positive subject are no longer significant and can be ignored. We can then decompose $\mathcal{S}$ into to two sets $\mathcal{S}^+$ and $\mathcal{S}^-$, based on the extracted subjects' $sim(s, \mathcal{Q})$ values:

$$\begin{aligned} \mathcal{S}^+ &= \{s | sim(s, \mathcal{Q}) > 0, s \in \mathbb{S}\}; \\ \mathcal{S}^- &= \{s | sim(s, \mathcal{Q}) = 0, s \in \mathcal{S}\}; \end{aligned} \tag{2}$$

and have their semantic relations specified as:

$$\mathcal{R} = \{r | r \in \mathbb{R}, r_\nu \subseteq \mathcal{S} \times \mathcal{S}\}. \tag{3}$$

The subjects in $\mathcal{S}$ specify the implicit knowledge k in $\mathcal{K}$ in a user's mental model $\mathcal{U}$, associated with an information need $g \in \mathcal{G}$, which is expressed by the user as a query $\mathcal{Q}$. $\mathcal{S}^+$ specifies the knowledge relevant to g, and $\mathcal{S}^-$ specifies the subjects paradoxical or non-relevant to g. The relations in $\mathcal{R}$ link the subjects in $\mathcal{S}$, and thus provide the backbone of $\widehat{\mathcal{B}}$ for $\mathcal{U}$. By these, we have the user's ontology constructed.

3.3.2 Personalizing Ontologies using $\mathbb{LIR}$s

A user's constructed ontology is personalized using the user's $\mathbb{LIR}$. An $\mathbb{LIR}$ is a collection of information items that are recently visited by the user. These items have tags assigned with subjects that cite the knowledge specified in the $\mathbb{WKB}$. A user's personal background knowledge related to an information need can be discovered from these citations. The discovered background knowledge personalizes the constructed ontology.

The subjects assigned to items in an $\mathbb{LIR}$ are the ties connecting the $\mathbb{LIR}$ to the $\mathbb{WKB}$. We call an element in $\mathbb{LIR}$s as an instance and denote it by i. Let $I = \{i_1, i_2, \cdots, i_p\}$ be an $\mathbb{LIR}$, and $S \subseteq \mathbb{S}$ be a set of subjects assigned to the instances in I. The relationships between S and I can be described as the following mappings:

$$\begin{aligned}&\eta : I \to 2^S,\ \ \eta(i) = \{s \in S | s \text{ is cited by } i\};\\ &\eta^{-1} : S \to 2^I,\ \ \eta^{-1}(s) = \{i \in I | s \in \eta(i)\}; \quad (4)\end{aligned}$$

where $\eta^{-1}(s)$ is a reverse mapping of $\eta(i)$. These mappings aim to explore the semantic matrix existing between the subjects and instances.

The beliefs of an instance to its referring subjects are varying. An instance cites multiple subjects, and these subjects are indexed by their importance to the instance. Thus, the belief of an instance to a subject can be defined by

$$bel(i, s) = \frac{priority(s, i)}{n(i)}; \quad (5)$$

where $n(i)$ is the number of subjects on the citing list of instance i, $priority(s, i) = \frac{1}{index(s,i)}$ where $index(s, i)$ is the index (starting from one) of s on the citing list of i. The $bel(i, s)$ increases when less subjects occur in the citing list and the higher index of s on the list.

A user's personal background knowledge is discovered from the semantic matrix existing between subjects and instances. Based on the mappings in Eq. (4), we first define the $coverset$ for a subject aiming to discover its synonym subjects. A subject's $coverset$, referring to the extent of instances in an $\mathbb{LIR}$ citing the subject, is defined by:

$$coverset(s) = \{i | i \in \eta^{-1}(s)\}. \quad (6)$$

Assume subject $s_1 \in \mathcal{S}^+$ and $s_2 \in \mathcal{S}^-$, if $coverset(s_1) \cap coverset(s_2) \neq \emptyset$, they have something in common, because s_1 and s_2 both refer to some common instances. Thus, we may say that s_2 is relevant to s_1, and may further deduce that s_2 may also be interesting to the user because s_1 is a positive subject. Based on these, let $\widehat{S}(s) = \{s' | s' \in \mathcal{S}^+, coverset(s') \cap coverset(s) \neq \emptyset\}$ be the synonyms of $s \in \mathcal{S}^-$ in $\mathcal{S}^+$, we discover new interesting subjects from $\mathcal{S}^-$ and personalize the $\mathcal{S}^+$ and $\mathcal{S}^-$ by:

$$\begin{aligned}\mathcal{S}^+ &= \mathcal{S}^+ \cup \{s | s \in \mathcal{S}^-, \widehat{S}(s) \neq \emptyset\};\\ \mathcal{S}^- &= \mathcal{S}^- - \{s | s \in \mathcal{S}^-, \widehat{S}(s) \neq \emptyset\}. \quad (7)\end{aligned}$$

We call the positive subjects extracted by using syntax-matching mechanism as the *initial positive subjects*, and the subjects $\{s | s \in \mathcal{S}^-, \widehat{S}(s) \neq \emptyset\}$ as *newly discovered interesting subjects*, for the sake of explanation in this paper.

3.3.3 Specifying Confidence Rates

Recall back to Assumption 1 and Definition 1, a knowledge unit k possessed by a user in the mental model $\mathcal{U}$ is associated with a confidence rate w_k. In this section, we call this confidence rate as the support value $sup(s, \mathcal{Q})$ of a subject s to $\mathcal{Q}$, and present how w_k is re-produced for subjects.

The $sim(s, \mathcal{Q})$ judges the positive or negative of a subject to $\mathcal{Q}$, and has impact to the $sup(s, \mathcal{Q})$. For an initial positive subject, we have its sim value measured by Eq. (1). We also need to define the sim values for those new interesting subjects discovered in Section 3.3.2. Because these new interesting subjects are discovered based on their synonyms in initial $\mathcal{S}^+$, these synonyms also have authority to determine a new interesting subject's sim value. Thus, we measure the sim of a new interesting subject as:

$$sim(s, \mathcal{Q}) = \frac{\sum_{s' \in \widehat{S}(s)} conf(s' \to s) \times sim(s')}{|\widehat{S}(s)|}. \quad (8)$$

where s' is an initial positive subject and $sim(s')$ is determined by Eq. (1). $conf(s' \to s)$ is the confidence of s received from its synonyms and calculated by:

$$conf(s' \to s) = \frac{|coverset(s') \cap coverset(s)|}{|coverset(s')|}. \quad (9)$$

As a result, a new interesting subject overlapping more initial positive subjects would have higher sim value. Subjects with higher sim has greater $sup(s, \mathcal{Q})$ values.

A subject's locality in the backbone of ontology affects its $sup(s, \mathcal{Q})$. Subjects located toward lower bound levels of backbone are more specific and so more focused on the knowledge they refer to. Thus, they should have higher $sup(s, \mathcal{Q})$ than the subjects located toward upper bound (more abstractive) levels. For the study of a subject's locality, we use an ontology mining method *Specificity*, which is introduced by [13, 12], and describes a subject's semantic focus on its referring knowledge. Algorithm 1 presents how the specificity values are assigned to subjects in an ontology. $hyponym(s)$ and $meronym(s)$ are functions that return the set of direct hyponyms or meronyms of s satisfying $hyponym(s) \subseteq \sigma(s)$ and $meronym(s) \subseteq \sigma(s)$. θ is a coefficient defining the reducing rate of specificity for focus lost in each step up from lower bound toward upper bound levels ($\theta = 0.9$ in our experiments). A subject's specificity depends on its child subjects, as described in Algorithm 1. As $\mathbb{WKB}$ is a directed acyclic graph (see Definition 4), Algorithm 1 has complexity of only $O(n)$, where $n = |\mathcal{S}|$.

```
input  : the subject and relation set $(\mathcal{S}, \mathcal{R})$; a coefficient $\theta$
         between (0,1).
output: specificity $spe(s)$ assigned to all $s \in \mathcal{S}$.
1  let $k = 1$, get the set of leaves $S_0$ from $\mathcal{S}$;
2  for $(s_0 \in S_0)$ assign $spe(s_0) = k$;
3  get $S'$ which is the set of leaves in case that we remove the
   nodes in $S_0$ and the related relations from $(\mathcal{S}, \mathcal{R})$;
4  if $(S' == \emptyset)$ then return;//the terminal condition;
5  foreach $s' \in S'$ do
6      if $(hyponym(s') == \emptyset)$ then $spe_1 = k$;
7      else $spe_1 = \theta \times min\{spe(s) | s \in hyponym(s')\}$;
8      if $(meronym(s') == \emptyset)$ then $spe_2 = k$;
9      else $spe_2 = \frac{\sum_{s \in meronym(s')} spe(s)}{|meronym(s')|}$;
10     $spe(s') = min(spe_1, spe_2)$;
11 end
12 $k = k \times \theta, S_0 = S_0 \cup S'$, go to step 3.
```

Algorithm 1: Analyzing Semantic Relations for Specificity

The instances in an $\mathbb{LIR}$ contain a user's background knowledge, and should also count in the $sup(s, \mathcal{Q})$ values of the $\mathbb{LIR}$'s citing subjects. Considering the $bel(i, \mathcal{Q})$ calculated by Eq. (5), the support value $sup(i, \mathcal{Q})$ held by an instance i to $\mathcal{Q}$ is calculated by:

$$sup(i, \mathcal{Q}) = \sum_{s \in \eta(i)} bel(i, s) \times sim(s, \mathcal{Q}). \qquad (10)$$

Because the negative subjects have $sim(s, \mathcal{Q}) = 0$, only the positive subjects cited by i count in $sup(i, \mathcal{Q})$.

Finally, $sup(s, \mathcal{Q})$ is calculated, based on the $sim(s, \mathcal{Q})$ from Eq. (1) or (8), the $spe(s)$ from Algorithm 1, and the related $sup(i, \mathcal{Q})$ from Eq. (10):

$$sup(s, \mathcal{Q}) = spe(s) \times sim(s, \mathcal{Q}) \times \sum_{i \in \eta^{-1}(s)} sup(i, \mathcal{Q}). \qquad (11)$$

For a subject in the final $\mathcal{S}^-$, the $sup(s, \mathcal{Q}) = 0$ because its $sim(s, \mathcal{Q}) = 0$. The support value $sup(s, \mathcal{Q})$ specifies the w_k of $\langle k, w_k \rangle \in \mathcal{K}$ in $\mathcal{U}$ considering an information need, where s is for a k and $\mathcal{Q}$ is for a $g \in \mathcal{G}$.

During the personalized ontology learning, the $\mathbb{WKB}$ in the computer model $\mathcal{C}$ constructs an ontology for a user, and the user's $\mathbb{LIR}$ is used to personalize the ontology. Equations (1) to (11) and Algorithm 1 are used for knowledge extraction. They are the $\mathbb{F}$ in $\mathcal{C}$, as described in Definition 5.

3.4. Ontology Model

The ontology model is the product from the computer model, aiming to simulate a user's implicit mental model dealing with an information need. An ontology model can be formalized as follows:

Definition 5 *An ontology model associated with a $\mathcal{Q}$ is a 4-tuple $\mathcal{O}(\mathcal{Q}) := \langle \mathcal{S}, \mathcal{R}, tax^{\mathcal{S}}, rel \rangle$, where*

- *$\mathcal{S}$ is a set of subjects ($\mathcal{S} \subseteq \mathbb{S}$) consisting of a positive subset $\mathcal{S}^+$ relevant and a negative subset $\mathcal{S}^-$ non-relevant to $\mathcal{Q}$;*
- *$\mathcal{R}$ is a set of relations and $\mathcal{R} \subseteq \mathbb{R}$;*
- *$tax^{\mathcal{S}}$: $tax^{\mathcal{S}} \subseteq \mathcal{S} \times \mathcal{S}$ is a function defining the taxonomic structure of ontology containing two directed relations of hypernym/hyponym and holonym/meronym;*
- *rel is a function defining non-taxonomic relation of synonyms, e.g. overlapping.*

The ontology model simulates a user's mental model $\mathcal{U}$. The knowledge $\mathcal{K}$ in $\mathcal{U}$ is specified by $\mathcal{S}$, in which the $\mathcal{S}^+$ is relevant and $\mathcal{S}^-$ is non-relevant to a $\mathcal{Q}$ representing an information need $g \in \mathcal{G}$. The w_k for k in $\mathcal{K}$ is re-produced by $sup(s, \mathcal{Q})$ for the subjects in $\mathcal{S}$. The $\widehat{\mathcal{B}}$ in $\mathcal{U}$ is specified by $\mathcal{R}$, $tax^{\mathcal{S}}$ and rel in $\mathcal{O}(\mathcal{Q})$. The mental model $\mathcal{U}$ is rebuilt.

4. Evaluation

In the IR fields, a common batch-style experiment is to select a collection of documents (testing set), a set of topics associated with relevance judgements (training set) and then compare the performance of experimental models [11]. Our experiments follow this style, and use the standard testbed and topics as that used in the TREC-11 Filtering track[1], which aims to evaluate IR methods using relevant and non-relevant training sets.

Our proposed model (called "ONTO model" in the experiments) is compared with an implemented mental model (called "TREC model" in experiments) in the experiments. The experiment design is illustrated in Fig. 1. Against an incoming topic, the TREC model generates a training set manually, whereas the ONTO model builds a user's personalized ontology automatically and generates a training set from the user's $\mathbb{LIR}$. A training set consists of a set of positive samples D^+ and a set of negative samples D^-. Each sample is a document d holding a support value $support(d)$ to the given topic. The different training sets are used by the common information gathering system to retrieve information from the testing set. The performance of the information gathering system is then affected by the training sets input. Based upon this, we can compare the performances and evaluate our proposed model.

The Reuters Corpus Volume 1 (RCV1) [**?**] used in the TREC-11 is also used as the testbed in our experiments. The RCV1 is a large XML document set (806,791 documents) with great topic coverage. A set of 50 topics are also provided by the TREC-11. These topics are designed by linguists manually, and associated with relevance documents judged by the same linguists [10]. All 50 topics are

[1] Text REtrieval Conference, http://trec.nist.gov/.

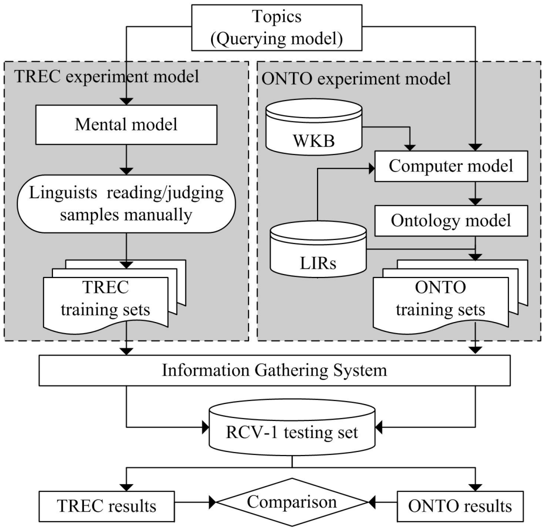

Figure 1. Experiment Design

used in our experiments, in order to maintain the high stability, as suggested by [1]. The topics have title, description and narrative. However, only the titles are used as queries, because in real world users only use short phrases to express their information needs.

4.1. Information Gathering System

An information gathering system (IGS) is implemented for common use by all the experimental models. The IGS is an implementation of a model developed by [7], which uses user profiles for information gathering. The [7]'s model is chosen because not only it is verified better than the *Rocchio* and *Dempster-Shafer* models, but also it is extensible in using support values of training documents. The input support values associated with documents would affect the IGS's performance sensitively. The technical details and the related justifications can be referred to [7].

The IGS first uses the training set to evaluate weights for a set of selected terms T. After text pre-processing of stopword removal and word stemming, a positive document d becomes a pattern that consists of a set of term frequency pairs $\hat{d} = \{(t_1, f_1), (t_2, f_2), \ldots, (t_k, f_k)\}$, where f_i is t_i's term frequency in d. The semantic space referred by $\hat{d}$ is represented by its normal form $\beta(d)$, which satisfies $\beta(d) = \{(t_1, w_1), (t_2, w_2), \ldots, (t_k, w_k)\}$, where w_i $(i = 1, \ldots, k)$ are the weight distribution of terms and $w_i = \frac{f_i}{\sum_{j=1}^{k} f_j}$.

A probability function on T can be derived based on the normal forms of positive documents and their supports for all $t \in T$:

$$pr_\beta(t) = \sum_{d \in D^+, (t,w) \in \beta(d)} support(d) \times w. \qquad (12)$$

The testing documents can be indexed by $weight(d)$, which is calculated using the probability function pr_β:

$$weight(d) = \sum_{t \in T} pr_\beta(t) \times \tau(t, d); \qquad (13)$$

where $\tau(t, d) = 1$ if $t \in d$; otherwise $\tau(t, d) = 0$.

4.2. TREC Model

The TREC model is the implementation of a user's mental model. For a given topic, the TREC linguists read a set of documents, and marked each document "positive" or "negative" against the topic. If a document d is marked "positive", it becomes a positive document in the TREC training set and $support(d) = \frac{1}{|D^+|}$; otherwise, it becomes a negative document and $support(d) = 0$. Since the linguists who marked the documents are also the people who generated the topics, following the assumption that only users know their interests and preferences perfectly, the TREC model makes a golden model to our proposed model to mark. The modelling of a user's mental model can be proven successful if the ONTO model can achieve the same or close performance to this golden model.

4.3. ONTO Model

This model is the implementation of our proposed model. As illustrated in Fig. 1 and required by the IGS, the input to this model is a topic and the output is a training set consisting of positive documents (D^+) and negative documents (D^-). Each document is associated with a $support(d)$ value indicating its support level to the topic.

The $\mathbb{WKB}$ described in Section 3.3.1 is constructed based on the Library of Congress Subject Headings[2] (LCSH) system. The LCSH system is a categorization developed for organizing the large volumes of library collections and for retrieving information from the library. The subject headings in the LCSH are transformed into the subjects in $\mathbb{WKB}$, and the LCSH structure is transformed into the backbone of $\mathbb{WKB}$. Eventually, the constructed $\mathbb{WKB}$ contains over 400,000 subjects covering various topics.

The semantic relations in the $\mathbb{WKB}$ are transformed from the references, *Broader term*, *Narrower term* and *Used-for*, specified in the LCSH. The *Broader term* and *Narrower term* references are transformed into hyponym/hypernym relations. *Used-for* references are usually used in two situations: to describe an action or to describe an object. When object *A* is used for an action, *A* actually becomes a part of that action, like "using turner in cooking"; when *A* is used for object *B*, *A* becomes a part of *B*, like"using wheels for a car". Hence, we transform the

[2] http://classificationweb.net/.

Used-for references in the LCSH into holonym/meronym relations in our $\mathbb{WKB}$.

In the experiments, we assume that each topic comes from an individual user. We attempt to evaluate our model in an environment that covers great range of topics. However, it is not realistic to expect a participant to hold such great range of topics in personal interests. Thus, for the 50 experimental topics, we assume each one coming from an individual user and learn her (his) personalized ontology.

An $\mathbb{LIR}$ is collected through searching the subject catalogue of Queensland University of Technology (QUT) Library[3] by using the title of a topic. Librarians have assigned title, table of content, summary, and a list of subjects to each information item (e.g. a book) stored in QUT library. The assigned subjects are treated as the tags in Web documents that cite the knowledge in the $\mathbb{WKB}$. In order to simplify the experiments, we only use the librarian summarized information (title, table of content and summary) to represent an instance in an $\mathbb{LIR}$. All these information can be downloaded from QUT's Web site and are available to the public.

Once the $\mathbb{WKB}$ and an $\mathbb{LIR}$ are ready, an ontology is learned as described in Section 3.3.1, and personalized as in Section 3.3.2. The user confidence rates on the subjects are specified as in Section 3.3.3. A document d_i in the training set is then generated by an instance i, and its support value is determined by:

$$support(d_i) = \sum_{s \in \eta(i), s \in \mathcal{S}} sup(s, \mathcal{Q}); \qquad (14)$$

where $s \in \mathcal{S}$ in $\mathcal{O}(\mathcal{Q})$ are as defined in Definition 5. As $sup(s, \mathcal{Q}) = 0$ for $s \in \mathcal{S}^-$ (according to Eq. (11)), the documents with $support(d) = 0$ go to D^-, whereas those with $support(d) > 0$ go to D^+.

4.4. Performance Measures

The performance of the experimental models are measured by three methods: the precision averages at eleven standard recall levels (11SPR), the mean average precision (MAP), and the F_1 Measure. They are all based on precision and recall, the modern IR evaluation methods [1].

The 11SPR is reported suitable for information gathering, and is used in TREC evaluations as a performance measuring standard [10]. An 11SPR value is computed by summing the interpolated precisions at the specified recall cutoff and then dividing by the number of topics:

$$\frac{\sum_{i=1}^{N} precision_\lambda}{N}; \quad \lambda = \{0.0, 0.1, 0.2, \ldots, 1.0\}. \quad (15)$$

N is the number of topics and λ are the cutoff points where the precisions are interpolated. At each λ point, an average precision value over N topics is calculated. These average precisions then link to a curve describing the recall-precision performance.

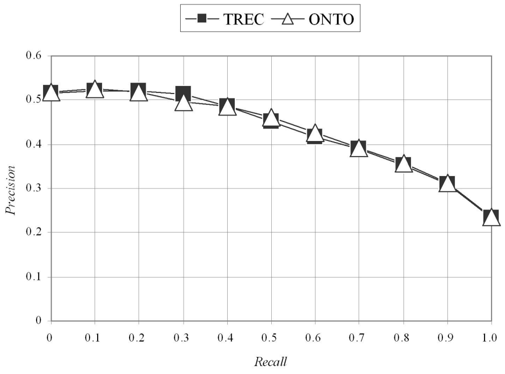

Figure 2. Experimental 11SPR Results

The MAP is a stable and discriminating choice in information gathering evaluations, and is recommended for measuring general-purpose information gathering methods [1]. The average precision for each topic is the mean of the precision obtained after each relevant document is retrieved. The MAP for the 50 experimental topics is then the mean of the average precision scores of each of the individual topics in the experiments. The MAP reflects the performance in a non-interpolated recall-precision fashion.

F_1 Measure is also well accepted by the information gathering community, which is calculated by:

$$F_1 = \frac{2 \times precision \times recall}{precision + recall}. \qquad (16)$$

Precision and recall are evenly weighted in F_1 Measure. For each topic, the *macro*-F_1 Measure averages the precision and recall and then calculates F_1 Measure, whereas the *micro*-F_1 Measure calculates the F_1 Measure for each returned result and then averages the F_1 Measure values. The greater F_1 values indicate the better performance.

5. Results and Discussions

The experiments attempt to evaluate our proposed model by comparing to an implementation of mental model. We expect that the ONTO model can achieve at least the close performance to the TREC model.

The experimental 11SPR results are illustrated in Fig. 2. At recall point 0.3, the TREC model slightly outperformed the ONTO model, but at 0.5 and 0.6, the ONTO model achieved better results than the TREC model subtly. At all other points, their 11SPR results are just the same. For the MAP results shown on Table 1, the ONTO model achieved 0.284, which is just 0.006 below the TREC model (2%

[3] http://library.qut.edu.au/.

	TREC	ONTO	*p-value*
Macro-FM	0.388	0.386	0.862
Micro-FM	0.356	0.355	0.896
MAP	0.290	0.284	0.484

Table 1. Other Experimental results

downgrade). For the average macro- and micro-F_1 Measures also shown on Table 1, the TREC model only outperformed the ONTO model by 0.002 (0.5%) in *macro-*F_1 and 0.001 (0.2%) in *micro-*F_1. The two models achieved almost the same performance. The evaluation result is promising.

The statistical test is also performed on the experimental results, in order to analyze the evaluation's reliability. As suggested by [11], we use the *Student's Paired T-Test* for the significance test. The *null hypothesis* in our T-Test is that no difference exists in two comparing models. When two tests produce substantially low *p-value* (usually $<$0.05), the null hypothesis can be rejected. In contrast, when two tests produce high *p-value* (usually $>$0.1), there is not or just little practical difference between two models [11]. The T-Test results are also presented on Table 1. The *p-value*s show that there is no evidence of significant difference between two experimental models, as the produced *p-value*s are quite high (*p-value*=0.484(MAP), 0.862(macro-FM) and 0.896(micro-FM), far greater than 0.1). Thus, we can conclude that in terms of statistics, our proposed model has the same performance as the golden TREC model, and the evaluation result is reliable.

The advantage of the TREC model is that the experimental topics and the training sets are generated by the same linguists manually. They as users perfectly know their information needs and what they are looking for in the training sets. Therefore, it is reasonable that the TREC model performed better than the ONTO model, as we cannot expect that a computational model could outperform a such perfect manual model. However, the knowledge contained in TREC model's training sets is well formed for human beings to understand, but not for computers. The contained knowledge is not mathematically formalized and specified. The ONTO model, on the other hand, formally specifies the user background knowledge and the related semantic relations using the world knowledge base and local instance repositories. The mathematic formalizations are ideal for computers to understand. This leverages the performance of the ONTO model. As a result, as shown on Fig. 2 and Table 1, the ONTO model achieved almost the same performance as that of the TREC model.

6. Conclusions

In this paper, an ontology-based knowledge retrieval framework is proposed aiming to discover a user's background knowledge to improve IR performance. The framework consists of a user's mental model, a querying model, a computer model and an ontology model. A world knowledge base is used by the computer model to construct an ontology to simulate a user's mental model, and the ontology is personalized by using the user's local instance repository. The semantic relations of hypernym/hyponym, holonym/meronym and synonym are specified in the ontology model. The framework is successfully evaluated by comparing to a manual user model. The ontology-based framework is a novel contribution to knowledge engineering and Web information retrieval.

References

[1] C. Buckley and E. M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 33–40, 2000.

[2] R. M. Colomb. *Information Spaces: The Architecture of Cyberspace*. Springer, 2002.

[3] D. Dou, G. Frishkoff, J. Rong, R. Frank, A. Malony, and D. Tucker. Development of neuroelectromagnetic ontologies(NEMO): a framework for mining brainwave ontologies. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 270–279, 2007.

[4] S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems*, 1(3-4):219–234, 2003.

[5] X. Jiang and A.-H. Tan. Mining ontological knowledge from domain-specific text documents. In *Proceedings of the Fifth IEEE International Conference on Data Mining*, pages 665–668, 2005.

[6] J. D. King, Y. Li, X. Tao, and R. Nayak. Mining World Knowledge for Analysis of Search Engine Content. *Web Intelligence and Agent Systems*, 5(3):233–253, 2007.

[7] Y. Li and N. Zhong. Mining Ontology for Automatically Acquiring Web User Information Needs. *IEEE Transactions on Knowledge and Data Engineering*, 18(4):554–568, 2006.

[8] H. Liu and P. Singh. ConceptNet: a practical commonsense reasoning toolkit. *BT Technology*, 22(4):211–226, 2004.

[9] A. D. Maedche. *Ontology Learning for the Semantic Web*. Kluwer Academic Publisher, 2002.

[10] S. E. Robertson and I. Soboroff. The TREC 2002 filtering track report. In *Text REtrieval Conference*, 2002.

[11] M. D. Smucker, J. Allan, and B. Carterette. A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 623–632, 2007.

[12] X. Tao, Y. Li, and R. Nayak. A knowledge retrieval model using ontology mining and user profiling. *Integrated Computer-Aided Engineering*, 15(4):313–329, 2008.

[13] X. Tao, Y. Li, N. Zhong, and R. Nayak. Ontology mining for personalzied web information gathering. In *Proc. of WI '07*, pages 351–358, 2007.

[14] T. Tran, P. Cimiano, S. Rudolph, and R. Studer. Ontology-based interpretation of keywords for semantic search. In *Proceedins of the 6th International Conference on Semaic Wweb*, pages 523–536, 2007.

[15] Y. Y. Yao, Y. Zeng, N. Zhong, and X. Huang. Knowedge retrieval (KR). In *Proc. of WI '07*, pages 729–735, 2007.