A Knowledge Retrieval Model Using Ontology Mining and User Profiling

Xiaohui Tao, Yuefeng Li, and Richi Nayak *

October 10, 2008

Abstract

Over the last decade, the rapid growth and adoption of the World Wide Web has further exacerbated the user need for efficient mechanisms for information and knowledge location, selection and retrieval. Much research in the area of semantic web is already underway, adopting information retrieval tools and techniques. However, much work is required to address knowledge retrieval; for instance, users' information needs could be better interpreted, leading to accurate information retrieval. In this paper, a novel computational model is proposed for solving retrieval problems by constructing and mining a personalized ontology based on world knowledge and a user's Local Instance Repository. The proposed model is evaluated by applying to a Web information gathering system, and the result is promising.

*X. Tao, Y. Li, and R. Nayak are with the Faculty of Information Technology, Queensland University of Technology, Australia. Emails: {$x.tao, y2.li, r.nayak$}@qut.edu.au

1 Introduction

Over the last decade, the rapid growth and adoption of the World Wide Web has further exacerbated the user need for efficient mechanisms for information and knowledge location, selection and retrieval. Web information covers a wide range of topics and serves a broad spectrum of communities. How to gather useful and meaningful information from the Web however, becomes challenging to Web users. Many information retrieval (IR) systems have been proposed, attempting to answer the call for this challenge [6]. However, to date there has not been a satisfactory solution proposed. Existing methods suffer from the problems of information mismatching or overloading. Information mismatching means valuable information being missed, while information overloading means non-valuable information being collected during information retrieval [20].

Most IR techniques are based on the keyword-matching mechanism. In this case, the information mismatching problem may occur if one topic has different syntactic representations. For example, "data mining" and "knowledge discovery" refer to the same topic. By the keyword-matching mechanism, documents containing "knowledge discovery" may be missed if using "data mining" to search. Another problem, information overloading, may occur in the case of one phrase having different semantic meanings. A common example is the query "apple", which may mean apples, the fruit, or iMac computers. In this case, the search results may be mixed by much useless information [16, 19, 20]. If a user's information need could be better captured, say, we knew that a user needed information about "apples the fruit" but not "iMac computers", we can

deliver the user more useful and meaningful information. Thus, the current IR models need to be enhanced in order to better satisfy user information needs.

The information diagram of *data-information-knowledge-wisdom* in information science suggests the enhancement route for IR models [39]. The diagram describes the information abstraction levels. Information is the abstraction of data, and knowledge is the abstraction of information. The data retrieval systems focus on the structured data stored in a database, and attempt to solve problems on the data level [39]. Consequently, although the data retrieval systems perform sufficiently on well-structured databases, they cannot achieve the same performance on the Web, as Web information is not well-structured. Enhanced from the data retrieval systems, the IR systems focus on the semi-structured or unstructured text documents, and attempt to solve problems on the information level. However, the IR systems still suffer from the aforementioned information mismatching and overloading problems [16–20, 41], and cannot capture user information needs well [20,33]. Therefore, if the IR systems can be enhanced from solving problems on the information level to the knowledge level, better results can be expected to be retrieved for Web users.

Many concept-match approaches have been proposed to promote the IR techniques from solving problems on the information level to the knowledge level. Owei [26] developed a concept-based natural language query system to handle and resolve the problem of keyword-match. Andreasen *et al.* [1] used a domain ontology for conceptual content-based querying in IR. Some works [7, 9, 31] proposed concept-based methods to refine and expand queries. These

developments, however, are concentrated on the context of a submitted query but not a user's background knowledge, in order to capture an information need.

In this paper, we propose a computational model for knowledge retrieval using a world knowledge base and a user's Local Instance Repository (LIR). World knowledge is "*the kind of knowledge that humans acquire through experience and education*" [40]. A world knowledge base is a frame of world knowledge. While generating a search query, a user usually holds a concept model implicitly. The concept model comes from a user's background knowledge and focuses on a particular topic. A user's LIR is a personal collection of Web documents that were recently visited by the user. These documents implicitly cite the knowledge specified in the world knowledge base. In the proposed model, we attempt to learn what a user wants from the user's LIR and the world knowledge base, where the world knowledge possessed by a user is described by a subject ontology. A two-dimensional ontology mining method, *Specificity* and *Exhaustivity*, is presented for the knowledge discovery in the subject ontology and the LIR. In the conducted experiments, the proposed computational model is evaluated by comparing the retrieved knowledge to the knowledge generated manually by linguists and the knowledge retrieved from the Web, and the results are promising. The proposed knowledge retrieval model is a novel attempt to conduct retrieval tasks at knowledge level instead of information level.

The paper is organized as follows. After *Introduction*, Section 2 presents related work. Section 3 introduces related definitions used in this paper, and Section 4 presents how to discover a user's background knowledge. Section 5

summarizes the proposed knowledge retrieval model. Section 6 describes the experiments, and the experimental results are discussed in Section 7. Finally, Section 8 makes conclusions.

2 Related Work

Information retrieval (IR) systems search in a corpus to fulfil user information needs [2]. A widely used strategy in IR is keyword-matching, which computes the similarity of relevant documents to an information need, and ranks the retrieved documents according to the weights calculated based on the frequency of important terms appearing in the documents, e.g. *Euclidean distance*, *Cosine similarity*, and the use of *feature vectors* [30]. There are three groups of IR models [12]: *Statistical* models that capture the relationships between the keywords from the probability of their co-occurrence in a collection; *Taxonomical* models that use the content and relations of a hierarchy of terms to derive a quantitative value of similarity between terms; and *Hybrid* models that combine both statistical and taxonomical techniques. However, these models all suffer from the common problems of information mismatching and overloading [16–20, 41].

Yao [39] pointed out that knowledge retrieval will be the importance feature of IR systems in the future. Recently, many concept-matching approaches have been proposed. Owei [26] developed a concept-based natural language query model to handle and resolve problems that occur with keyword-matching. Andreasen *et al.* [1] proposed a method using domain ontology for conceptual

content-based querying in IR. Some works [7, 9, 31] proposed concept-based methods to refine and expand queries in order to improve search performance. These models, however, are concentrated on reformulation of given queries but not users' background knowledge.

User profiles are used by many IR systems for personalized Web search and recommendations [8, 10, 20, 37, 42]. A user profile is defined by Li & Zhong [20] as the topics of interests relating to user information needs. They further categorized user profiles into two diagrams: the data diagram for the discovery of interesting registration data, and the information diagram for the discovery of the topics of interests related to information needs. The data diagram profiles are usually generated by analyzing a database or a set of transactions; for example, user logs [8,20,23,24,27,32]. The information diagram profiles are generated by using manual techniques such as questionnaires and interviews [24,37], or by using the IR techniques and machine-learning methods [27]. In order to generate a user profile, Chirita *et al.* [4] and Teevan *et al.* [36] used a collection of the user's desktop text documents, emails, and cached Web pages for query expansion and exploration of user interests. Makris *et al.* [22] comprised user profiles by a ranked local set of categories and then utilized Web page categories to personalize search results.

Ontologies have been utilized by many models to improve the performance of personalized Web information gathering systems. Some reports [8,37] demonstrate that ontologies can provide a basis for the match of initial behavior information and the existing concepts and relations. Li & Zhong [19, 20] used

ontology mining techniques to discover interesting patterns from positive documents, and ontologized the meaningful information to generate a user profile. Navigli *et al.* built an ontology called *OntoLearn* [25] to mine the semantic relations among the concepts from Web documents. Gauch *et al.* [8] used a reference ontology based on the categorization systems of online portals and learned a personalized ontology for users. Such categorizations were also used by Chirita *et al.* [5] to generate user profiles for Web search. Liu *et al.* [21] proposed a model to map a user's query to a set of categories in order to discover the user's search intention. Sieg *et al.* [29] modelled a user's context as an ontological profile and assigned interest scores to the existing concepts in a profile. Middleton *et al.* [24] used ontologies to represent a user profile for online recommendation systems. Developed by King *et al.* [13], *IntelliOnto* uses the Dewey Decimal Code system to describe world knowledge and generate user profiles. Unfortunately, these works cover only a small number of concepts and do not specify the semantic relationships of *partOf* and *kindOf* existing in the concepts, but only "super-class" and "sub-class".

In summary, the existing IR models need to be enhanced from the current information level to knowledge level. The enhancement can be achieved by using user profiles to capture the semantic context of a user's information needs. A user profile can be better generated using an ontology to formally describe and specify a user's background knowledge. According to the related work, however, how to use ontologies to specify a user's background knowledge still remains a research gap in the IR development. Filling this gap motivates our research

work presented in this paper.

3 Definitions

3.1 World Knowledge Base

A world knowledge base is a knowledge frame describing and specifying world knowledge. In a knowledge base, knowledge is formalized in a structure and the relationships between the knowledge units are specified. The Library of Congress Subject Headings[1] (LCSH), a taxonomic classification system originally developed for organizing and retrieving information from the large volumes of library collections, suits the requirements of constructing a world knowledge base. The LCSH system is comprised of a thesaurus containing about 400,000 subject headings that cover an exhaustive range of topics. The LCSH aims to facilitate users' perspectives in accessing the information items stored in a library, and has proved excellent for the study of world knowledge [3]. In this paper, we build a world knowledge base using the LCSH system.

We transform each subject heading in the LCSH into a knowledge unit in the world knowledge base, and name a primitive knowledge unit as a *subject* in this paper. The LCSH structure is transformed into the taxonomic backbone of the knowledge base. The backbone specifies the semantic relationships of subjects. Three types of semantic relations are specified in the world knowledge base. *KindOf* is a directed relationship for two subjects describing the same

[1]The Library of Congress, http://www.loc.gov/.

entity on different levels of abstraction (or concretion); e.g. "Professional Ethic" is a kind of "Ethics", etc. The $kindOf$ relationships are transformed from the BT (*Broader Term*) and NT (*Narrower Term*) references specified in the LCSH. *KindOf* relationships are transitive and asymmetric. Let s be a subject, transitivity means if s_1 is a kind of s_2 and s_2 is a kind of s_3, then s_1 is a kind of s_3 as well. Asymmetry means if s_1 is a kind of s_2, s_2 may not be a kind of s_1.

PartOf is a directed relationship used to describe the relationships for a compound subject and its component subjects or a subject subdivided by others. A component subject forms a part of a compound subject; e.g. "Economic Espionage" is part of "Business Intelligence". The *partOf* relationships are transformed from the *UF* (*Used-For*) references specified in the LCSH. The *partOf* relationships also hold the transitivity and asymmetry properties. If s_1 is a part of s_2 and s_2 is a part of s_3, then s_1 is also a part of s_3. If s_1 is a part of s_2 and $s_1 \neq s_2$, s_2 is definitely not a part of s_1.

RelatedTo[2] is a relationship held by two subjects related in some manner other than by hierarchy. The semantic meanings referred by the two subjects may overlap. One example of *relatedTo* relations is "Ships" to "Boats and boating". The *kindOf* relationships in the world knowledge base are transformed from the *RT* (*Related term*) references specified in the LCSH. *RelatedTo* holds the property of symmetry but not transitivity. Symmetry means if s_1 is related to s_2, s_2 is also related to s_1. *RelatedTo* relationships are not transitive. If s_1

[2] Although the *relatedTo* references are specified in the LCSH system, we are not focused on this semantic relationship in this paper. The utilization of the *KindOf* and *partOf* semantic relationships is challenging and the solution is a significant contribution to the related areas.

is related to s_2 and s_2 related to s_3, s_1 may not necessarily be related to s_3, if s_1 and s_3 do not overlap at all.

The taxonomic knowledge base constructed in our knowledge retrieval model is formalized as follows.

Definition 1 *Let $\mathcal{KB}$ be a taxonomic world knowledge base. It is formally defined as a 2-tuple $\mathcal{KB} :=< \mathbb{S}, \mathbb{R} >$, 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 $s :=< label, \sigma >$, where label is a label assigned by linguists to a subject s and is denoted by $label(s)$, and $\sigma(s)$ is a signature mapping defining a set of subjects that hold direct relationship like $partOf$, $kindOf$, or $relatedTo$ with 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 $r := < type, r_\nu >$, where type is a relation type of $kindOf, partOf$, or $relatedTo$ and $r_\nu \subseteq \mathbb{S} \times \mathbb{S}$. For each $(s_x, s_y) \in r_\nu$, s_y is the subject that holds the type of relation to s_x, e.g. s_x is $kindOf$ s_y.*

3.2 Subject Ontology

A personalized subject ontology formally describes a user's background knowledge focusing on an individual need of information. While searching for information online, a user can easily determine if a Web page is interesting or not by scanning through the content. The rationale behind this is that users implicitly possess a concept model based on their background knowledge [20]. A user's

personalized subject ontology aims to rebuild his (or her) concept model.

A subject ontology may be built based on a user's feedback and the world knowledge base. In IR, a query $\mathcal{Q}$ is usually a set of terms generated by a user as a brief description of an information need. After receiving a query from a user, some potentially relevant subjects can be extracted from the world knowledge base using the syntax-matching mechanism. A subject s and its ancestor subjects in the world knowledge taxonomy are extracted if the $label(s)$ matches (or partially matches) the terms in the query. The extracted subjects are displayed to the user in a fashion of taxonomy, and the user then selects positive and negative subjects considering the information need [33, 35]. With the user identified subjects, we can extract the semantic relationships existing between the subjects and then construct a subject ontology to simulate the user's implicit concept model.

A subject ontology is formalized by the following definition:

Definition 2 *The structure of a subject ontology that formally describes and specifies query $\mathcal{Q}$ is a 4-tuple $\mathcal{O}(\mathcal{Q}) := \{\mathcal{S}, \mathcal{R}, tax^{\mathcal{S}}, rel\}$, where*

- *$\mathcal{S}$ is a set of subjects ($\mathcal{S} \subseteq \mathbb{S}$) which includes a subset of positive subjects $\mathcal{S}^+ \subseteq \mathcal{S}$ relevant to $\mathcal{Q}$, a subset of negative subjects $\mathcal{S}^- \subseteq \mathcal{S}$ non-relevant to $\mathcal{Q}$, and a subset of unlabelled subjects $\mathcal{S}^? \subseteq \mathcal{S}$ that have no evidence of appreciating any site of positive or negative;*
- *$\mathcal{R}$ is a set of relations and $\mathcal{R} \subseteq \mathbb{R}$;*
- *$tax^{\mathcal{S}}$: $tax^{\mathcal{S}} \subseteq \mathcal{S} \times \mathcal{S}$ is called the backbone of the ontology, which is con-*

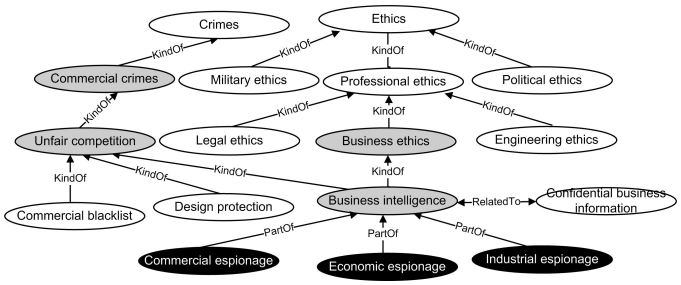

Figure 1: A Constructed Ontology (Partial) for Query "*Economic Espionage*".

structed by two directed relationships $kindOf$ *and* $partOf$*;*

- rel *is a relation between subjects, where* $rel(s_1, s_2) = True$ *means* s_1 *is* $relatedTo$ s_2 *and* s_2 *is* $relatedTo$ s_1 *as well.*

One assumption of a constructed subject ontology is that no any loop or cycle exists in the ontology. Fig. 1 presents a partial subject ontology constructed for query "Economic espionage", where the white nodes are positive subjects, the black are the negative, and the gray are the unlabelled subjects. The unlabelled subjects are those subjects extracted by the syntax-matching mechanism but not selected by the user for either positive or negative. We call this subject ontology "personalize", since the knowledge related to an information need is identified by a user personally. A constructed subject ontology could have multiple roots, depending on the domains that a user's given query covers.

4 Discovering User Information Needs

In this section, we present how a user's information needs are discovered from the constructed subject ontology and the user's Local Instance Repository (LIR).

4.1 Local Instance Repository

An LIR is a collection of information items (instances) that are recently visited by a user, e.g. a set of Web documents. The information items cite the knowledge specified in a subject ontology. To evaluate the proposed model in this paper, we use the information summarized in a library catalogue to represent a user's LIR, since the catalogue information is assigned with subject headings and cites the knowledge specified in the LCSH. The catalogue information of an item stored in a library and recently visited by a user is collected as an instance in the user's LIR. Such catalogue information includes title, table of contents, summary, and a list of subject headings. Each instance is represented by a vector of terms $i = \{t_1, t_2, \ldots, t_n\}$ after text pre-processing including stopword removal and word stemming.

A semantic matrix can be formed from the relations held by the instances in a user's LIR and the subjects in the user's personalized subject ontology. By using the subject headings assigned to an instance, each instance in an LIR can map to some subjects in the world knowledge base. Let $2^{\mathcal{S}}$ be the space referred to by $\mathcal{S}$ in a subject ontology $\mathcal{O}(\mathcal{Q})$, and 2^I be the space referred by I in an LIR and $I = \{i_1, i_2, \cdots, i_p\}$. The mapping of an i to the subjects in $\mathcal{S}$ can be described as follows:

$$\eta : I \rightarrow 2^{\mathcal{S}}, \quad \eta(i) = \{s \in \mathcal{S} | s \text{ is used to describe } i\} \subseteq \mathcal{S}. \tag{1}$$

and the reverse mapping η^{-1} of η, specifying the mappings of a $s \in \mathcal{S}$ to the

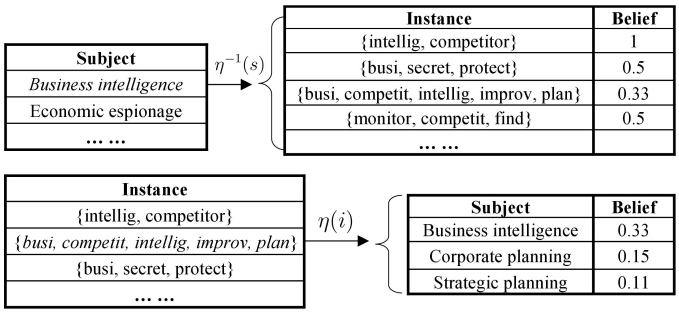

Figure 2: Mappings of Subjects and Instances Related to "*Economic Espionage*".

instances in the LIR:

$$\eta^{-1} : \mathcal{S} \to 2^I, \quad \eta^{-1}(s) = \{i \in I | s \in \eta(i)\} \subseteq I. \tag{2}$$

Figure 2 displays a sample of the mappings. The "*Business intelligence*" subject maps to a set of instances, "{*intellig, competitor*}", "{*busi, secret, protect*}", "{*busi, competit, intellig, improv, plan*}", "{*monitor, competit, find*}", and so on. Whilst, the "{*busi, competit, intellig, improv, plan*}" instance maps to a set of subjects of "*Business intelligence*", "*Corporate planning*", and "*Strategic planning*". These mappings aim to explore the semantic matrix existing between the subjects and instances. Each i is relevant to one or more subjects in $\mathcal{S}$, and each s refers to one or more instances in I.

The referring belief of an instance to the cited subjects (see Fig. 2) may be at different levels of strength. Belief is affected by many things. Usually, the subject headings assigned to an instance are in the fashion of a sequence, e.g. "*Business intelligence – Data processing*". The tail, "*Data processing*", is to

further restrict the semantic extent referred to by the head, "*Business intelligence*". While extracting the referred subject classes from the world knowledge base, we treat each sequence as one subject heading. It is perfect if a subject class in the world knowledge base matches the entire subject heading sequence. There is no information lost in the process of knowledge extraction. However, sometimes we cannot have such a perfect match and have to cut the tail in order to find a matching subject in the world knowledge base. In that case, some information is lost. As a consequence, the instance's belief to the extracted subject class is weakened.

In many cases, multiple subject headings are assigned to one instance, for example, the subject headings:

Business intelligence – Management;

Business intelligence – Data processing;

Telecommunication – Management;

are assigned to an instance titled "*Business intelligence for telecommunications*" in the catalogue of the Queensland University of Technology (QUT) library[3]. These subjects headings are indexed by their importance to the instance. Thus, if a subject referred by the top subject heading, we can assume that it receives stronger belief from the instance than a subject referred by the bottom heading, e.g. *Business intelligence – Management* vs. *Telecommunication – Management*. Moreover, more subject headings assigned to an instance will weaken the belief shared by each subject.

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

We denote $\varpi(s)$ as the level of information lost in matching a subject heading sequence to a subject class in the world knowledge base. For a perfect match, we set $\varpi(s) = 1$. Each time the tail is cut, $\varpi(s)$ increases by 1. Thus, the greater $\varpi(s)$ value indicates more information lost. We also denote $\xi(i)$ as the number of subject headings assigned to an instance i and $\iota(s)$ as the index (starting with 1) of an assigned s. By counting the best belief an instance could deliver as 1, we can have the belief of an i to a s calculated by:

$$bel(i, s) = \frac{1}{\xi(i) \times \iota(s) \times \varpi(s)}. \tag{3}$$

In the aforementioned example and case of s referring to "*Business intelligence – Data processing*", we can have $\xi(i) = 3$, $\iota(s) = 2$, $\varpi(s) = 2$ and $bel(i, s) = 0.083$.

4.2 User Information Needs Analysis

An LIR is a set of documents describing and referring to the knowledge related to a user's interests. An instance in an LIR may support a user's information need (represented by a query) at different levels. In Section 3.2, we have discussed that a user's background knowledge is formally specified by a subject ontology. The ontology is constructed by focusing on a specific information need, and contains a subject set consisting of a subset of positive and a subset of negative subjects. Therefore, the support level of an instance to a user's information need depends on its referring positive and negative subjects. If an instance refers to more positive subjects than negative, it supports the information need.

Otherwise, it is against the need. Based on these, we can calculate the belief of an instance i to a query $\mathcal{Q}$ in an ontology $\mathcal{O}(\mathcal{Q})$ by:

$$bel(i, \mathcal{Q}) = \sum_{s \in \eta(i) \cap s \in \mathcal{S}^+} bel(i, s) - \sum_{s \in \eta(i) \cap s \in \mathcal{S}^-} bel(i, s). \tag{4}$$

The instances associated to an unlabelled subject count nothing to the query because there is no evidence that they appreciate positive or negative.

With the beliefs of instances to a query calculated, the belief of a subject to a query can also be determined by:

$$bel(s, \mathcal{Q}) = \sum_{i \in \eta^{-1}(s)} bel(i, \mathcal{Q}). \tag{5}$$

For a subject $s \in \mathcal{S}^+$, if $bel(s, \mathcal{Q}) > 0$, the subject supporting the query is confirmed. Greater $bel(s, \mathcal{Q})$ value indicates stronger support. If $bel(s, \mathcal{Q}) < 0$, using that subject to interpret the semantic meaning of a given query is actually confusing, and the subject should be moved from $\mathcal{S}^+$ to $\mathcal{S}^-$. For a subject $s \in \mathcal{S}^-$, $bel(s, \mathcal{Q}) < 0$ confirms its negative. If $bel(s, \mathcal{Q}) > 0$, it makes the interpretation confusing, and should be removed from the $\mathcal{S}^-$. The unlabelled subjects again hold belief value of 0 to the query because their beliefs are not clarified.

4.3 Exhaustivity and Specificity of Subjects

Ontology mining means discovering knowledge from the backbone and the concepts that construct and populate an ontology. Two schemes are introduced

here for mining an ontology: *Specificity* (*spe* for short) describes the semantic focus of a subject corresponding to a query, whereas *Exhaustivity* (*exh* for short) restricts the semantic extent covered by a subject. The terms of specificity and exhaustivity were used by information science originally to describe the relationship of an index term with the retrieved documents [11]. They are assigned new meanings in this paper in order to measure how a subject covering or focusing on what a user wants.

input : the ontology $\mathcal{O}(\mathcal{Q})$; a subject $s \in \mathbb{S}$; a parameter θ between (0,1).
output: the specificity value $spe(s)$ of s.

1 If s is a leaf then let $spe(s) = 1$ and then return;
2 Let S_1 be the set of direct child subjects of s such that $\forall s_1 \in S_1 \Rightarrow type(s_1, s) = kindOf$;
3 Let S_2 be the set of direct child subjects of s such that $\forall s_2 \in S_2 \Rightarrow type(s_2, s) = partOf$;
4 Let $spe_1 = \theta,\ spe_2 = \theta$;
5 **if** $S_1 \neq \emptyset$ **then** calculate $spe_1 = \theta \times min\{spe(s_1)|s_1 \in S_1\}$;
6 **if** $S_2 \neq \emptyset$ **then** calculate $spe_2 = \frac{\sum_{s_2 \in S_2} spe(s_2)}{|S_2|}$;
7 $spe(s) = min\{spe_1, spe_2\}$.

Algorithm 1: $spe(s)$: Assigning Specificity Value to a Subject

The specificity of a subject increases if the subject is located on a lower level of an ontology's taxonomic backbone. Algorithm 1 presents a recursive method $spe(s)$ for assigning the specificity value to a subject in an ontology. We assign the leaf subjects the highest *spe* value of 1, since they are primitive and cannot be further decomposed. From the leaf subjects bottom-up, if a subject is decomposed into a set of child subjects and holds the *kindOf* relationship with them, the subject takes the least *spe* value from its child subjects, as a parent subject is the abstractive refinement of the child subjects. If a parent subject holds the *partOf* relationship with a set of child subjects, it is assigned

the average *spe* value of its component subjects, because its referring semantic space is the combination of sematic meanings referred by the component subjects and all the component subjects should be considered. If the child subjects are mixed by *kindOf* and *partOf* relationships to their parent subject, the least specificity value of *kindOf* or *partOf* child subjects should take place for the parent subject.

By concentrating on specificity, the support value of a subject, being the knowledge referring to and supporting a user's information need, can be measured by:

$$sup_{spe}(s, \mathcal{Q}) = spe(s) \times bel(s, \mathcal{Q}) \times \sum_{i \in \eta^{-1}(s)} sup(i, \mathcal{Q}); \tag{6}$$

where $\sum_{i \in \eta^{-1}(s)} sup(i, \mathcal{Q})$ refers to the total support from other related subjects, and is calculated by:

$$sup(i, \mathcal{Q}) = \sum_{s \in \eta(i)} bel(i, s) \times bel(s, \mathcal{Q}) \tag{7}$$

By concentrating on exhaustivity and modifying Eq. (7), we can also determine the certainty level of a subject being the knowledge related to a user information need. The extent of knowledge is extended if more relevant subjects appear in its volume. Based on this assumption, the $sup_{exh}(s, \mathcal{Q})$ concentrating on exhaustivity is determined by the number of relevant subjects covered in the

volume of s ($vol(s)$):

$$sup_{exh}(s, \mathcal{Q}) = bel(s, \mathcal{Q}) \times \sum_{s' \in vol(s)} \sum_{i \in \eta^{-1}(s')} sup(i, \mathcal{Q}). \quad (8)$$

A subject with higher exhaustivity value covers more relevant knowledge referring to a user's information need.

The knowledge to interpret the user information need $\mathcal{Q}$ can finally be represented by a set of subjects:

$$\mathcal{RK}(\mathcal{Q}) = \{s | sup_{spe}(s, \mathcal{Q}) \geq min_{spe}, sup_{exh}(s, \mathcal{Q}) \geq min_{exh}\}. \quad (9)$$

A subject in $\mathcal{RK}(\mathcal{Q})$ needs to satisfy both of the conditions of greater than min_{spe}, the minimum value of sup_{spe}, and min_{exh}, the minimum value of sup_{exh}. The min_{spe} and min_{exh} are used to prune the weak subjects representing a user's background knowledge focusing on a given query.

5 Framework

The knowledge retrieval model proposed in this paper aims to acquire and analyze a Web user's background knowledge so that his (her) information need can be better captured and satisfied.

Two knowledge resources are used in the model: (i) *World Knowledge Base*, which provides a frame of world knowledge for a user to identify the positive and negative knowledge corresponding to an information need. The world knowledge

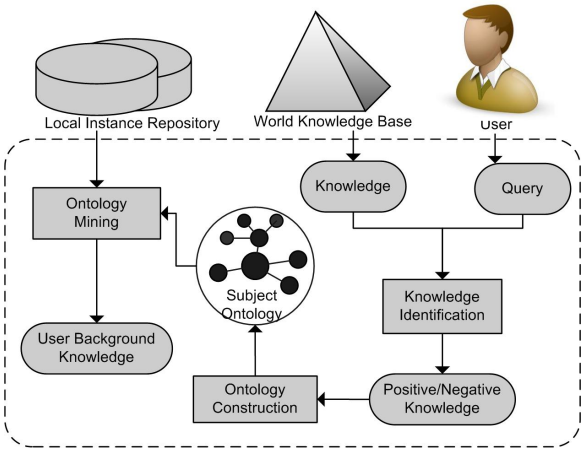

Figure 3: Framework of the Knowledge Retrieval Model

base also defines the backbone of a user's personalized subject ontology; (ii) *Local Instance Repository*, which provides a resource to discover a user's real information need.

The framework of the knowledge retrieval model is presented in Fig. 3. The model takes a query from a user, say, "Economic espionage", extracts a set of potentially relevant subjects from the world knowledge base, and displays the subjects to the user, as described in Section 3.2. The user identifies the related knowledge including positive and negative subjects from the present subjects. Finally, based on the user identified knowledge, the model constructs a subject ontology, as the partial ontology illustrated in Fig. 1. Once a user's subject ontology is constructed, the knowledge for user information needs can then be mined from the user's LIR and the constructed ontology. The knowledge mining methods are discussed in Section 4. The proposed model produces a set of subjects related to a user's interests and helping to interpret the user's

information need.

Our proposed knowledge retrieval model uses ontologies to specify a user's background knowledge and to capture a user's information need. This model attempts to enhance existing IR techniques by solving problems on the knowledge level, and to fill the related research gap in the IR development as specified in Section 2.

6 Evaluation

The proposed model aims to discover knowledge to what a user wants, in response to a given query. Such knowledge is also commonly called a *user profile* in IR [20]. The evaluation of the proposed model is then concentrated on the quality of its generated user profiles.

6.1 Experiment Design

The techniques of generating a user profile can be categorized into three groups of interviewing, non-interviewing, and pseudo-relevance feedback. The interviewing mechanism usually involves user efforts. The profiles generated by interviewing techniques can be technically called "perfect", as they are generated manually and reflect a user's interests perfectly. One example is the training sets in TREC-11 Filtering Track[4]. Linguists read each document in the TREC training sets and provide a judgement of positive or negative to the document

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

against a given query [28].

The techniques using non-interviewing mechanisms do not involve user efforts directly. Instead, they observe and mine knowledge from a user's activity and behavior in order to generate a training set to describe a user's interests [37]. One representative of these implicit techniques is the OBIWAN model proposed by Gauch et al [8].

Different from the interviewing and non-interviewing mechanisms, pseudo-relevance feedback profiles are generated by semi-manual techniques. The pseudo-relevance feedback techniques assume a certain number of top documents on an initially extracted list as the positive information feedback from a user. One of these techniques is the Web training set acquisition method [34], which analyzes the retrieved Web documents using a belief based method.

Our proposed knowledge retrieval model is compared to the aforementioned mechanisms in the evaluation experiments. For this, four experimental user profiling models have been implemented. The implementation of the proposed model is called "KRM", standing for "Knowledge Retrieval Model". Three competitor models are: the TREC model generating perfect user profiles and representing the manual interviewing techniques; the Web model for the Web training set acquisition method [34] and representing the semi-automated pseudo-relevance feedback methods; and the Category model for the OBIWAN [8] and representing the automated non-interviewing profiling mechanism.

Figure 4 illustrates the experiment design. The experimental queries go into the four user profiling models, and produce different profiles. A produced user

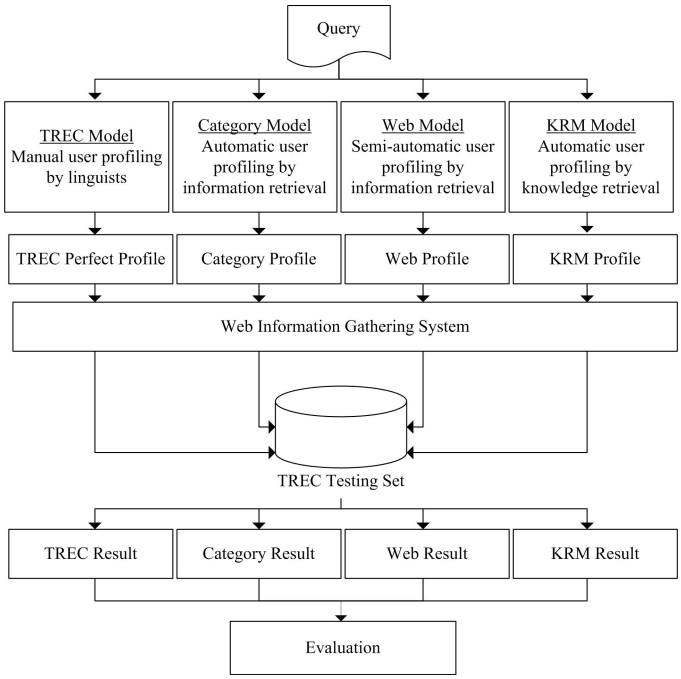

Figure 4: The Dataflow of the Experiments

profile is represented by a training set consisting of a positive subset and a negative subset of documents. Each document in a training set is assigned a value indicating the support level of the document to a given query. The user profiles (training sets) are used by the same Web information gathering system to retrieve relevant documents from the testing data set. The retrieval results are compared and analyzed for evaluation of the proposed model.

6.2 The Experimental Models

6.2.1 Proposed Model: KRM Model

A user profile is represented in this model by a training set consisting of positive and negative documents. Since a user could come from any domain, we treat each incoming query as a Web user. For example, "Economic espionage" is a query coming from a user who may have background of "Economy" and "Intelligence". The related LIR is a collection of documents visited by this user, and his (or her) background knowledge is underlying from the LIR and related to the background of "Economy" and "Intelligence". This user's profile is mined from the LIR, and is a description of his (or her) background knowledge.

In the experiments, a user's LIR is obtained through searching the subject catalogue of the QUT Library (see http://library.qut.edu.au). The content of a document in an extracted LIR is the catalogue information of an information item stored in the library, including title, table of contents, and summary. These data and information are available to the public on the QUT library's Web site.

The world knowledge base is constructed based on the LCSH classification system, which contains 394,070 topical subjects. As described in Section 3, *BT* (*Broader Term*) and *NT* (*Narrower Term*) references in the LCSH are transformed into *kindOf* relationships, *UF* (*Used-For*) references are transformed into *partOf* relationships, and *RT* (*Related Term*) references are transformed into *relatedTo* relationships in the experiments.

For a given query, e.g. the aforementioned "Economic espionage", the KRM

model extracts a set of potentially relevant subjects from the world knowledge base and displays to a user, as described in Section 3.2. The user identifies the positive and negative subjects from the present subjects. Based on the identified knowledge, the KRM model constructs a subject ontology, as the partial ontology illustrated in Fig. 1. The knowledge for user information need is mined from the user's LIR and the constructed ontology, as discussed in Section 4. The discovered knowledge is represented by a set of subjects $\mathcal{RK}(\mathcal{Q})$, as described in Eq. (9).

The training set documents are generated from a user's LIR based on the $\mathcal{RK}(\mathcal{Q})$. By treating each instance as a document and representing it by a vector of terms after text pre-processing including stopword removal and word stemming, we can have a set of positive documents generated by:

$$D_{\mathcal{Q}}^{+} = \{d_i | i \in \eta^{-1}(s), s \in \mathcal{RK}(\mathcal{Q})\}. \tag{10}$$

A support value *sup* is assigned to a document $d_i \in D_{\mathcal{Q}}^{+}$, indicating the support level of d_i containing the relevant knowledge corresponding to an information need referred by $\mathcal{Q}$. The support value is calculated by:

$$sup(d_i, \mathcal{Q}) = \sum_{s \in (\eta(i) \cap \mathcal{RK}(\mathcal{Q}))} bel(i, s) \times sup_{spe}(s, \mathcal{Q}). \tag{11}$$

The experimental model appreciates specificity more than exhaustivity. We assume that specificity contributions to the precision performance of a model, whereas exhaustivity contributes to the recall. Thus, using the semantic focus of

a subject may make the model having better precision performance than using the semantic extent of a subject.

A negative document set D^- is generated by:

$$D_{\mathcal{Q}}^- = \{d_i | i \in \eta^{-1}(s), s \in (\mathcal{S} - \mathcal{RK}(\mathcal{Q}))\}. \tag{12}$$

The support value of these documents set as 0.

6.2.2 Goal Model: TREC Model

The training sets are manually generated by the TREC linguists. For a coming query, the TREC linguists read a set of documents and marked either positive or negative against each document [28]. Since the queries are also generated by these linguists, the TREC training sets perfectly reflect a user's background knowledge and concept model, and the support value of each positive document is assigned with 1, and negative with 0. These training sets are thus deemed as "perfect" training sets.

The "perfect" model marks the research goal that our proposed model attempts to achieve. A successful retrieval of user background knowledge can be confirmed if the performance achieved by the proposed model can match or is close to the performance of the "perfect" TREC model.

6.2.3 Baseline Model: Category Model

This experimental model represents a typical model using the non-interviewing techniques to generate user profiles. In this model, a user profile is a set of

topics related to the user's interests. Each topic is represented by a vector of terms trained from a user's browsing history using the $tf \cdot idf$ method. While searching, the cosine similarity value of an incoming document to a user profile is calculated, and higher similarity value indicates that the document is more interesting to the user.

In the experiments, we used the same LIRs in the KRM model as the collection of a user's Web browsing history in this model in order to make the comparison fair.

6.2.4 Baseline Model: Web Model

This model represents a typical model using the pseudo-relevance feedback mechanism to generate a user's profile. As with the KRM model, a user profile is represented by a training set, including a sub-set of positive and a sub-set of negative documents. In this experimental model, the training sets (user profiles) are automatically retrieved from the Web by employing a Web search engine.

For each incoming query, a set of positive concepts and a set of negative concepts are identified manually. By using *Google*, we retrieved a set of positive and a set of negative documents (100 documents in each set) using the identified concepts. The support value of a document in a training set is defined based on (i) the precision of the chosen search engine; (ii) the index of a document on the result list, and (iii) the belief of a subject supporting or against a given query. This model attempts to use Web resources to benefit information retrieval. The technical details can be found in [34].

6.2.5 Web Information Gathering System

The common information gathering system is implemented, based on a model that tends to effectively gather information by using user profiles [20]. This model uses patterns to represent positive documents, where each document is viewed as a pattern P which consists of a set of terms (T) and the distribution of term frequencies w in the document ($\beta(P)$).

Let PN be the set of discovered patterns. Using these patterns, we can have a probability function:

$$pr_\beta(t) = \sum_{P \in PN, (t,w) \in \beta(P)} support(P) \times w \tag{13}$$

for all $t \in T$, where $support(P)$ is used to describe the percentage of positive documents that can be represented by the pattern.

In the end, for an incoming document d, its relevance can be evaluated as

$$\sum_{t \in T} pr_\beta(t)\tau(t,d), \quad \text{where} \quad \tau(t,d) = \begin{cases} 1 & \text{if } t \in d \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

6.3 Testbed and Queries

The Reuters Corpus Volume 1 (RCV1) [15] is used as the test data set in the experiments. The RCV1 collections are a large data set (an archive of 806,791 documents) of XML (Extensible Markup Language) documents with great topic coverage. The data in RCV1 has been processed by substantial verification and validation of the content, attempting to remove spurious or duplicated

documents, normalization of dateline and byline formats, addition of copyright statements, and so on. RCV1 is also the testbed used in the TREC-11 2002 Filtering track. The TREC-11 Filtering track aims to evaluate the methods of persistent user profiles for separating relevant and non-relevant documents in an incoming stream. TREC-11 provides a set of searching topics defined and constructed by linguists. These topics are associated with the positive and negative documents judged by the linguists [28]. In the experiments, the titles of 40 topics (R101-140) were used as the experimental queries. For example, the aforementioned query "Economic espionage" is the title of topic R101.

6.4 Performance Assessment Methods

The performance of the system by applying the four models is compared and analyzed to find out if the KRM model outperforms other models. The performance is assessed by two methods: the precision averages at eleven standard recall levels, and F_1 Measure. The former is used in TREC evaluation as the standard for performance comparison of different information filtering models [38]. A recall-precision average is computed by summing the interpolated precisions at the specified recall cutoff and then dividing by the number of queries:

$$\frac{\sum_{i=1}^{N} precision_{\lambda}}{N}. \tag{15}$$

N denotes the number of experimental queries, and $\lambda = \{0.0, 0.1, 0.2, \ldots, 1.0\}$ indicates the cutoff points where the precisions are interpolated. At each λ point, an average precision value over N queries is calculated. These average

precisions then link to a curve describing the precision-recall performance. The other method, F_1 Measure [14], is well accepted by the community of information retrieval and Web information gathering. F_1 Measure is calculated by:

$$F_1 = \frac{2 \times precision \times recall}{precision + recall} \tag{16}$$

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

7 Results and Discussions

7.1 Experimental Results

The experiments attempt to compare the knowledge retrieved and specified by the KRM model to the goal and baseline models. Some experimental results are displayed in the Fig. 5, which is the chart for the precision averages at eleven standard recall levels. As shown on the figure, the TREC model has better precision than the KRM model before recall cutoff 0.5, and the KRM has better precision after that point. Both the TREC and KRM models outperform the Web model and Category model in terms of precision and recall results.

In terms of F_1 measure, Table 1 presents the results. According to the F-

	Macro-F1 Measure				Micro-F1 Measure			
Query	**TREC**	**Web**	**Category**	**KRM**	**TREC**	**Web**	**Category**	**KRM**
R101	0.733	0.652	0.614	0.598	0.666	0.592	0.556	0.542
R102	0.728	0.529	0.551	0.563	0.671	0.492	0.509	0.521
R103	0.360	0.347	0.345	0.388	0.324	0.315	0.313	0.346
R104	0.644	0.647	0.448	0.628	0.585	0.594	0.415	0.582
R105	0.555	0.570	0.566	0.584	0.509	0.521	0.517	0.534
R106	0.232	0.256	0.233	0.281	0.222	0.239	0.220	0.260
R107	0.230	0.207	0.215	0.228	0.206	0.189	0.199	0.210
R108	0.179	0.150	0.144	0.159	0.168	0.140	0.135	0.150
R109	0.451	0.653	0.645	0.662	0.420	0.599	0.594	0.609
R110	0.218	0.156	0.279	0.280	0.202	0.146	0.256	0.257
R111	0.108	0.101	0.064	0.132	0.102	0.096	0.062	0.126
R112	0.194	0.195	0.167	0.201	0.180	0.179	0.156	0.184
R113	0.315	0.213	0.297	0.353	0.287	0.195	0.276	0.326
R114	0.413	0.427	0.412	0.437	0.373	0.392	0.376	0.399
R115	0.506	0.552	0.532	0.537	0.452	0.494	0.477	0.481
R116	0.632	0.512	0.567	0.576	0.578	0.466	0.518	0.527
R117	0.361	0.374	0.330	0.334	0.331	0.344	0.305	0.309
R118	0.111	0.177	0.214	0.221	0.108	0.168	0.203	0.208
R119	0.410	0.249	0.270	0.290	0.380	0.236	0.255	0.273
R120	0.673	0.656	0.666	0.666	0.615	0.590	0.601	0.601
R121	0.471	0.465	0.340	0.403	0.416	0.412	0.317	0.360
R122	0.449	0.434	0.451	0.473	0.401	0.397	0.411	0.427
R123	0.184	0.172	0.163	0.169	0.172	0.161	0.157	0.162
R124	0.236	0.386	0.348	0.357	0.224	0.356	0.327	0.336
R125	0.465	0.474	0.425	0.454	0.423	0.420	0.387	0.403
R126	0.772	0.689	0.609	0.653	0.720	0.645	0.574	0.609
R127	0.483	0.505	0.499	0.487	0.446	0.467	0.462	0.450
R128	0.331	0.309	0.339	0.346	0.308	0.290	0.317	0.324
R129	0.337	0.358	0.282	0.354	0.301	0.323	0.261	0.317
R130	0.169	0.204	0.151	0.166	0.163	0.193	0.144	0.158
R131	0.615	0.628	0.602	0.601	0.564	0.573	0.557	0.555
R132	0.117	0.171	0.163	0.170	0.110	0.161	0.152	0.158
R133	0.266	0.245	0.182	0.263	0.245	0.231	0.173	0.249
R134	0.454	0.336	0.415	0.421	0.416	0.307	0.383	0.391
R135	0.627	0.524	0.511	0.497	0.583	0.496	0.489	0.475
R136	0.307	0.309	0.371	0.403	0.286	0.286	0.337	0.363
R137	0.138	0.134	0.134	0.135	0.131	0.129	0.128	0.130
R138	0.406	0.293	0.379	0.376	0.368	0.270	0.348	0.347
R139	0.247	0.286	0.254	0.292	0.231	0.268	0.240	0.273
R140	0.417	0.405	0.480	0.496	0.378	0.367	0.431	0.442
Avg.	0.389	0.374	0.367	0.391	0.357	0.343	0.338	0.359

Table 1: The Detailed F-1 Measure Results

	Avg. Macro F-Measure		Avg. Micro F-Measure	
Comparison	Improvement	% Change	Improvement	% Change
KRM vs. TREC	0.002	0.50%	0.002	0.60%
KRM vs. Web	0.017	4.50%	0.016	4.70%
KRM vs. Category	0.024	6.50%	0.021	6.20%

Table 2: Comparisons of the F-Measure Performance

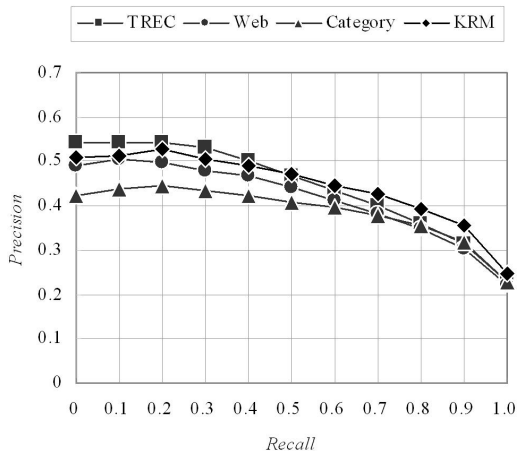

Figure 5: The Precision Averages at 11 Standard Recall Levels

Measure results, the proposed KRM model has achieved the best performance, followed by the TREC model, the Web model, and last the Category model. Table 2 presents the comparison results between the KRM model and the baseline models. The figures in "Improvement" are calculated by using the average F_1-Measure results of the KRM to minus the others. The percentages displayed in "% Change" indicate the percentage change in performance achieved by the proposed KRM model over the baseline models, which is calculated by:

$$\%\ Change = \frac{\mathcal{F}_{KRM} - \mathcal{F}_{Competitor}}{\mathcal{F}_{Competitor}} \times 100\%. \tag{17}$$

where $\mathcal{F}$ denotes the average F_1 Measure result of an experimental model. These percentage changes are also illustrated in Fig. 6. The improvement achieved by the KRM model over the Web model and Category model are relatively

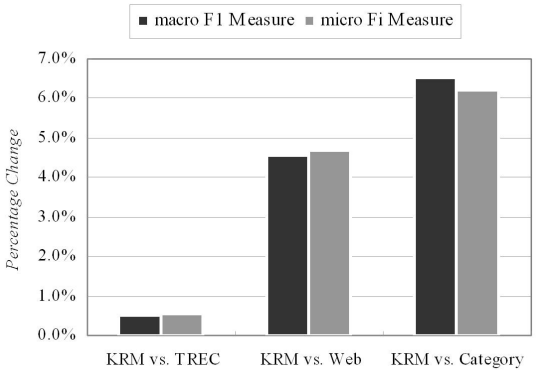

Figure 6: The Significance of the Percentage Change in Performance

significant, but compared to the TREC model is just slight.

7.2 Discussions

The experiments for the KRM and TREC model is to compare the knowledge retrieved by the proposed model to the knowledge acquired by linguists manually. As the results shown on Fig. 5, the perfect TREC model slightly outperforms the KRM model and keeps the performance until over the recall cutoff point 0.4. After that, the KRM model catches up and performs better than the TREC model. As shown in Table. 1 and Fig. 6, the F_1 Measure results and the related comparisons, the KRM model slightly outperforms the TREC model by only about 0.002 in both $Macro$and $Micro$ F_1 Measure. The KRM model has over 1000 documents per query on average for knowledge retrieval in one user profile. In contrast, the number of documents included in each TREC training set is very limited (about 60 documents per query on average), and

some semantic meanings referred by a given query are not fully covered by the TREC training sets. Consequently, the KRM training sets cover much broader semantic extent in comparison to the TREC training sets, although the expert knowledge contained in the TREC sets is more precise. Considering that the TREC model employs the human power of linguists to read every single document in the training sets, which reflects a user's concept model perfectly, it is not realistic to expect that the TREC model can be defeated. Therefore, the close performance of the KRM model to the TREC model is promising.

The experiments for the KRM and Category models is to compare the proposed model to the state-of-the-art automated user profiling techniques. According to the experimental results, the KRM model outperforms the Category model and has improved the performance of the Category by 6.5% in terms of $Macro\ F_1$ Measure and by 6.2% in terms of $Micro\ F_1$ Measure. The KRM model specifies the retrieved knowledge in a subject ontology by using the complex semantic relationships of *kindOf*, *partOf* and *relatedTo*, and analyzes the subjects by using the multi-dimensional ontology mining schemes of specificity and exhaustivity. In contrast, the Category model specifies only the simple relationships of "super-" and "sub-class". The KRM performs in more technical depth in comparison with the Category model, and moves far beyond the simple "super-" and "sub-class" specification. Based on these, we may conclude that the KRM model enhances the retrieval performance from existing state-of-the-art automated user profiling techniques.

The comparison of the KRM and Web model is to compare the world knowl-

edge and the background knowledge retrieved by the proposed method to only the world knowledge extracted by the Web model. According to the experimental results, the KRM outperforms the Web model. The percentage change in performance achieved by the KRM over the Web model is 4.5% in $Macro\text{-}F_1$ and 4.7% in $Micro\text{-}F_1$ Measure. The Web model's training sets are extracted from the Web. The Web documents, however, could be contributed by anyone. Comparing to the Web model training sets, the KRM training sets integrate the world knowledge and a user's background knowledge from his (or her) LIR. The world knowledge and background knowledge retrieved by the KRM model leverages its performance. Based on these, we conclude that the proposed model can integrate world knowledge and background knowledge and improves the performance of Web information gathering.

Based on the discussions, the proposed knowledge retrieval model is evaluated and the results are promising.

8 Conclusions

In this paper, a computational model for knowledge retrieval is proposed. Two knowledge resources are used by the proposed model: a world knowledge base constructed based on the LCSH classification and a Local Instance Repository containing documents visited by a user. Based on a user's constructed subject ontology corresponding to a given query, the knowledge for user information need is discovered and analyzed. In order to analyze the discovered knowledge, a

two-dimensional scheme of specificity and exhaustivity is presented to assess the knowledge units and the related semantic relationships in an ontology. A user profile is finally generated from a user's LIR, which is a training set consisting of a subset of positive and subset of negative documents. Each training document is assigned with a value indicating the support level of the document to a given query. The experimental results are promising.

The knowledge retrieval model attempts to enhance the existing IR systems from solving problems on the information level to the knowledge level. The proposed computational model contributes to the development of the next generation of retrieval systems.

Acknowledgements

We would like to thank the *Library of Congress* and *Queensland University of Technology Library* for authorizing the use of MARC and the catalogue records. We also like to thank Prof. N. Zhong, for his valuable suggestions to this present research work, and the anonymous reviewers of this paper for their comments. We also extend our thanks to Mr. M. Carry-Smith and Mr. P. Delaney for proofreading this paper.

References

[1] T. Andreasen, P. A. Jensen, J. F. Nilsson, P. Paggio, P. S. Pedersen, and H. E. Thomsen. Content-based text querying with ontological descriptors. *Data & Knowledge Engineering*, 48(2):199–219, February 2004.

[2] C. Baumgarten. A probabilistic solution to the selection and fusion problem in distributed information retrieval. In *Proc. of the 22nd annual intl. ACM SIGIR conf. on Res. and development in inf. retr.*, pages 246–253. United States, 1999.

[3] L. M. Chan. *Library of Congress Subject Headings: Principle and Application.* Libraries Unlimited, 2005.

[4] P. A. Chirita, C. S. Firan, and W. Nejdl. Personalized query expansion for the web. In *Proc. of the 30th intl. ACM SIGIR conf. on Res. and development in inf. retr.*, pages 7–14, 2007.

[5] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschütter. Using ODP metadata to personalize search. In *Proc. of the 28th annual intl. ACM SIGIR conf. on Res. and development in inf. retr.*, pages 178–185. ACM Press, 2005.

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

[7] B. M. Fonseca, P. Golgher, B. Possas, B. Ribeiro-Neto, and N. Ziviani. Concept-based interactive query expansion. In *Proc. of the 14th ACM intl. conf. on Information and knowledge management*, pages 696–703, New York, NY, USA, 2005.

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

[9] F. A. Grootjen and Th.P. van der Weide. Conceptual query expansion. *Data & Knowledge Engineering*, 56(2):174–193, February 2006.

[10] J. Han and K.C.-C. Chang. Data mining for Web intelligence. *Computer*, 35(11):64–70, 2002.

[11] K. S. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[12] I. Kaur and A. J. Hornof. A comparison of LSA, WordNet and PMI-IR for predicting user click behavior. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 51–60, New York, USA, 2005. ACM Press.

[13] 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.

[14] D. D. Lewis. Evaluating and optimizing autonomous text classification systems. In *Proc. of the 18th intl. ACM SIGIR conf. on Res. and development in inf. retr.*, pages 246–254, 1995.

[15] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. RCV1: A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research*, 5:361–397, 2004.

[16] Y. Li and Y. Y. Yao. User profile model: a view from Artificial Intelligence. In *3rd International Conference on Rough Sets and Current Trends in Computing*, pages 493–496, 2002.

[17] Y. Li and N. Zhong. Ontology-based Web mining model. In *Proceedings of the IEEE/WIC International Conference on Web Intelligence, Canada*, pages 96–103, 2003.

[18] Y. Li and N. Zhong. Capturing Evolving Patterns for Ontology-based Web Mining. In *Proceedings of the IEEE/WIC/ACM intl. Conf. on Web Intelligence, WI2004.*, pages 256–263, 2004.

[19] Y. Li and N. Zhong. Web Mining Model and its Applications for Information Gathering. *Knowledge-Based Systems*, 17:207–217, 2004.

[20] 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.

[21] F. Liu, C. Yu, and W. Meng. Personalized web search by mapping user queries to categories. In *Proc. of 11th intl. conf. on Information and knowledge management, CIKM '02*, pages 558–565, New York, USA, 2002. ACM Press.

[22] C. Makris, Y. Panagis, E. Sakkopoulos, and A. Tsakalidis. Category ranking for personalized search. *Data & Knowledge Engineering*, 60(1):109–125, January 2007.

[23] S. E. Middleton, N. R. Shadbolt, and D. C. De Roure. Capturing interest through inference and visualization: ontological user profiling in recommender systems. In *Proc. of the 2nd intl. conf. on Knowledge capture*, pages 62–69, 2003.

[24] S. E. Middleton, N. R. Shadbolt, and D. C. De Roure. Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.*, 22(1):54–88, 2004.

[25] R. Navigli, P. Velardi, and A. Gangemi. Ontology learning and its application to automated terminology translation. *Intelligent Systems, IEEE*, 18:22–31, 2003.

[26] V. Owei. An intelligent approach to handling imperfect information in concept-based natural language queries. *ACM Trans. Inf. Syst.*, 20(3):291–328, 2002.

[27] A. Pretschner and S. Gauch. Ontology based personalized search. In *ICTAI*, pages 391–398, 1999.

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

[29] A. Sieg, B. Mobasher, and R. Burke. Learning ontology-based user profiles: A semantic approach to personalized web search. *The IEEE Intelligent Informatics Bulletin*, 8(1):7–18, Nov. 2007.

[30] E. Sormunen, J. Kekalainen, J. Koivisto, and K. Jarvelin. Document text characteristics affect the ranking of the most relevant documents by expanded structure queries. *Journal of Documentation*, 57(3):358–374, 2001.

[31] N. Stojanovic. Conceptual query refinement: The basic model. In *Proceedings of WISE2005*, pages 404–417, 2005.

[32] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proc. of the 13th intl. conf. on World Wide Web*, pages 675–684, USA, 2004.

[33] X. Tao, Y. Li, and R. Nayak. Mining ontology for semantic interpretation of information needs. In *Accepted by the second international conference on Knoweldge Science, Engineering and Management*, Australia, 2007.

[34] X. Tao, Y. Li, N. Zhong, and R. Nayak. Automatic Acquiring Training Sets for Web Information Gathering. In *Proc. of the IEEE/WIC/ACM Intl. Conf. on Web Intelligence*, pages 532–535, HK, China, 2006.

[35] X. Tao, Y. Li, N. Zhong, and R. Nayak. Ontology mining for personalzied web information gathering. In *Proc. of the IEEE/WIC/ACM intl. conf. on Web Intelligence*, pages 351–358, Silicon Valley, USA, Nov. 2007.

[36] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *Proc. of the 28th intl. ACM SIGIR conf. on Res. and development in inf. retr.*, pages 449–456, 2005.

[37] J. Trajkova and S. Gauch. Improving ontology-based user profiles. In *Proc. of RIAO 2004*, pages 380–389, France, 2004.

[38] E.M. Voorhees. Overview of TREC 2002. In *The Text REtrieval Conference (TREC)*, 2002. Retrieved From: http://trec.nist.gov/pubs/trec11/papers/OVERVIEW.11.pdf.

[39] Y. Y. Yao, Y. Zeng, N. Zhong, and X. Huang. Knowledge retrieval (kr). In *Proceedings of the 2007 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 729–735, Silicon Valley, USA, Nov 2007.

[40] L.A. Zadeh. Web intelligence and world knowledge - the concept of Web IQ (WIQ). In *Processing of NAFIPS '04.*, volume 1, pages 1–3, 27-30 June 2004.

[41] N. Zhong. Representation and construction of ontologies for Web intelligence. *International Journal of Foundation of Computer Science*, 13(4):555–570, 2002.

[42] N. Zhong. Toward web intelligence. In *Proc. of 1st Intl. Atlantic Web Intelligence Conf.*, pages 1–14, 2003.