Expert Systems with Applications 29 (2005) 408–423

Expert Systems with Applications

www.elsevier.com/locate/eswa

Task-based $\mathcal{K}$-Support system: disseminating and sharing task-relevant knowledge

Duen-Ren Liu*, I-Chin Wu, Kuen-Shieh Yang

Institute of Information Management, National Chiao Tung University, 1001 Ta Hseuh Road, Hsinchu 300, Taiwan, ROC

Abstract

In task-based business environments, effective knowledge management relies on providing task-relevant information to fulfill the information needs of knowledge workers. This work proposes an adaptive task-based profiling approach to model workers' task needs, namely information needs (profiles) on tasks. Meanwhile, a fuzzy analytical method is proposed to identify peer-groups with similar task-needs based on workers' profiles. Accordingly, a Knowledge Support ($\mathcal{K}$-Support) system is developed to provide a collaborative task-based workplace facilitating knowledge retrieval and sharing among peer-groups. The proposed $\mathcal{K}$-Support system is grounded in a research institute to stimulate task-based knowledge retrieval and sharing. Experimental results show that the proposed system can support task-relevant knowledge effectively.

Keywords: Task-based portal; Knowledge retrieval; Knowledge sharing; Adaptive task profile; Task-needs

1. Introduction

Deploying knowledge management systems (KMSs) is an important strategy for organizations to gain sustainable advantages. Organizations try to maximize the use of knowledge assets to increase an organization's profitability and productivity with the support of contemporary knowledge management tools. Furthermore, the critical role of Information Technologies (ITs) is to assist knowledge workers to reuse valuable knowledge assets to carry out business tasks successfully (Andrade, Ares, Garcia, Rodriguez, & Suárez, 2003; Davenport & Prusak, 1998; Liebowitz, 1999).

Generally, ITs focus on explicit and tacit dimensions in knowledge management activities (Gray, 2001a; Kankanhalli, Tanudidjaja, Sutanto, & Tan (Bernard), 2003). The former is achieved by a codified approach. Intellectual content codified into explicit form can facilitate knowledge retrieval and reuse (Zack, 1999). Knowledge repository, knowledge-based systems, and knowledge maps are the supports for knowledge storage, organization and dissemination. The latter puts emphasis on dialoging via social networks to facilitate knowledge sharing. Knowledge expert directories, yellow pages, communities of practices and talk rooms, support interpersonal communication for knowledge sharing (Agostini, Albolino, De Michelis, De Paoli, & Dondi, 2003; Koh & Kim, 2004).

Effective knowledge management relies on understanding workers' information needs on tasks, for brevity, task-needs. As the operations and management activities of enterprises are mainly task-based, KMSs focus on providing task-relevant knowledge to workers engaged in knowledge-intensive tasks (Abecker, Bernardi, Maus, Sintek, & Wenzel, 2000; Fenstermacher, 1999, 2002; Fischer & Ostwald, 2001). The Kabiria system supports knowledge-based document retrieval in office environments, allowing users to conduct document retrieval according to the operational context of task-associated procedures (Celentano, Fugini, & Pozzi, 1995). The KnowMore system maintains task specifications (profiles) to enumerate the process-context of tasks and associated knowledge items (Abecker et al., 2000). Context-aware delivery of task-specific knowledge can then be facilitated based on the task specifications and current execution context of the process. Although providing an appropriate view for designing task-based knowledge support, existing works focus on specifying the process-context of tasks. This provides support for context-aware or process-aware knowledge retrieval, rather

* Corresponding author. Tel.: +886 3 5712 121; fax: +886 3 572 3792.
E-mail address: dliu@iim.nctu.edu.tw (D.-R. Liu).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.04.036

than on a systematic approach to construct task profiles. Moreover, the adaptation of profiles to track workers' dynamic information needs is not addressed.

KMSs rely on an effective approach to construct a community of practice to promote knowledge sharing. Koh and Kim (2004) investigated knowledge sharing in virtual communities from an e-business perspective. Their result revealed that knowledge sharing in virtual communities can increase the loyalty of Internet-based service providers. The Milk system supports informal communication and knowledge sharing for knowledge workers performing tasks in different work practices (Agostini et al., 2003). OntoShare, an ontology-based KMS, models the interests of users and provides automatic knowledge sharing in communities of practice with the aid of profiles (Davies, Duke, & Stonkus, 2003). Although user profiles had been employed to stimulate knowledge disseminations in communities of practice, they did not consider the identification of peer-groups with similar task-needs to form communities in task-based business environments.

This work proposes a task-based knowledge support ($\mathcal{K}$-Support) system to acquire, organize, and disseminate an organization's knowledge resources from the aspect of task to support task-relevant knowledge. Task-based information is conceptualized into ontology, which is used as a conceptual backbone for organizing knowledge resources and supporting knowledge access. An adaptive task-based profiling approach is proposed to tackle workers' dynamic information needs on tasks by analyzing workers' access behaviors or relevance feedbacks based on the domain ontology. Furthermore, a fuzzy analytical method is proposed to identify task-based peer-groups according to workers' profiles, namely, task needs. With the aid of task-based profiles and peer-groups, the $\mathcal{K}$-Support system effectively provides task-relevant knowledge and knowledge sharing among task-based peer-groups.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 presents the architecture of the $\mathcal{K}$-Support system. The approach of adaptive task-based profiling is described in Section 4. Section 5 describes the fuzzy analytical method for identifying task-based peer groups. The proposed $\mathcal{K}$-Support portal with associated system evaluation is presented in Sections 6 and 7, respectively. Conclusions and future works are stated in Section 8.

2. Literature review

2.1. Task-based knowledge retrieval

The repository of structured, explicit knowledge, especially document form, is a codified strategy to manage knowledge (Davenport & Prusak, 1998; Gray, 2001b). However, with the growing amount of information in organizational memories, KMSs face the challenge to help users find pertinent and needed information. Accordingly, knowledge retrieval is considered a core component to access knowledge items in knowledge repository (Kwan & Balasubramanian, 2003; Fenstermacher, 2002). Moreover, domain ontology, a shared conceptualization of a specific domain, is often used to specify the working domain of an organization. Organizing knowledge items into ontological structure based on the domain ontology is promising to support knowledge retrieval in business environments (Fensel, Staab, Studer, van Harmelen, & Davies, 2003).

Translating users' information needs into compromised queries is not an easy work. Most systems rely on Information Retrieval (IR) techniques to access organizational codified knowledge. The technique of Information Filtering (IF) with a profiling approach to model users' information needs is an effective approach to proactive delivering relevant information to users. The technique has been widely used in the areas of Information Retrieval and Recommender Systems (Herlocker & Konstan, 2001; Middleton, Shadbolt, & De Roure, 2004; Pazzani & Billsus, 1997). The profiling approach has also been addressed by some KMSs to enhance knowledge retrieval and further promote knowledge sharing among project-based or interesting groups (Abecker et al., 2000; Agostini et al., 2003; Davies et al., 2003).

The information can be delivered in a specific context of business environments. The information retrieval (IR) technique coupled with workflow management systems (WfMS) was employed to support proactively delivery of task-specific knowledge according to the context of tasks within a process (Abecker et al., 2000; Fenstermacher, 2002). Furthermore, a process meta-model specifying the knowledge-in-context is integrated with workflow systems to capture and retrieve knowledge within a process context (Kwan & Balasubramanian, 2003). Despite the subtle difference among these works, they provide an appropriate view by specifying the process-context of tasks to support context-aware knowledge retrieval. Furthermore, acquiring and disseminating role-relevant process views was considered in workflow environments (Shen & Liu, 2004). Alvarado, Romero-Salcedo, and Sheremetov, (2004) also proposed acquiring and organizing corporate memory from the perspective of role/job position, in which an Organizational Memory is modeled by adopting UML/XML to specify the ontologies for organization positions, tasks, and application domains.

2.2. Knowledge sharing in community of practices

For complex and knowledge-intensive tasks, the collaboration among knowledge workers may arise around common goals, problems and interests. Domain experts or experienced workers who hold valuable tacit knowledge play important roles in assisting knowledge workers to accomplish business tasks (McDonald & Ackerman, 2000). The ultimate goal of KM is to enable innovative activities by promoting collaboration or communication among knowledge workers in organizations (Fischer & Ostwals, 2001; Wolverton, 1999). Collaboration may take place in a formal group such as a business project or in an informal group such as a community of practice. A community of practice consists of people who share common needs of information; hence, a community of practice is an effective approach to promote knowledge creation, transfer and sharing within or across organizations (Agostini et al., 2003; Brown & Duguid, 1991; Davies et al., 2003; Koh & Kim, 2004). Although user profiles had been employed to stimulate knowledge disseminations in communities of practice, they did not consider the identification of peer-groups with similar task-needs to form communities in task-based business environments.

3. Task-based knowledge support system

This section presents an overview of the $\mathcal{K}$-Support system for providing task-relevant information. The system architecture of the proposed knowledge support is then described.

3.1. Overview of $\mathcal{K}$-Support

In task-based environments, codified knowledge and human resources are important knowledge assets for accomplishing organizational tasks. This work presents an architecture and implementation of a knowledge support system ($\mathcal{K}$-Support) in task-based workplaces. The proposed $\mathcal{K}$-Support system provides task-relevant knowledge to a worker based on his/her information needs on the *target-task*, namely the task being conducted at hand. The main concepts of the proposed system are as follows.

- An adaptive task-based profiling approach is proposed to model workers' dynamic information needs (profiles) on tasks based on their access behaviors or relevance feedbacks on knowledge items. Task-based knowledge support can then be facilitated to assist knowledge workers to access and disseminate task-relevant knowledge based on profiles. A fuzzy linguistic approach is employed to model workers' relevance feedbacks. The fuzzy linguistic approach is an approximate technique for modeling human thinking in evaluating qualitative problems (Zadeh, 1975). Moreover, a modified relevance feedback (RF) technique, adopted from the techniques proposed by Rocchio (1971) and Ide (1971), is used to adjust workers' profiles based on relevance feedbacks. Relevance feedback is a well-known technique in information retrieval for improving search effectiveness by automatic query reformulation (Salton & Buckley, 1990).
- Experienced workers with valuable task-relevant knowledge and expertise can help knowledge workers solve problems or make decisions. For complex and knowledge-intensive tasks, collaboration among knowledge workers and experts is often necessary for more effective knowledge dissemination. The proposed system can provide more effective knowledge support through knowledge sharing among peer-group members. Peer-group members engaged in common tasks or with similar task-needs can collaborate in the proposed task-based portal to accomplish their tasks. A fuzzy analytical method is proposed to determine peer-groups with similar task-needs based on the profiles. The method employs a fuzzy max–min operation to derive the similarity among workers by computing the transitive max–min closure. The method of computing transitive max–min closure is an effective approach to infer fuzzy relations which are not explicitly given (Chen & Horng, 1999; Klir & Yuan, 1995). Inherent transitive relationship among workers is inferred to derive a fuzzy similarity matrix of workers. Task-based peer-groups can then be identified by grouping members with equivalence relation determined by α-cuts applied to the fuzzy similarity matrix. More details will be addressed in Section 5.

Fig. 1 shows the proposed knowledge support model to facilitate task-based knowledge retrieval and sharing. The model comprises three main modules, namely *task-oriented repository*, *profile handler*, and *task-oriented retrieval router*.

Task-oriented repository. Task-oriented repository is constructed with the support from domain ontology to effectively organize codified knowledge. The ontology configuration generates the domain ontology, which is structured into four levels, including categories, fields, tasks, and codified knowledge items. Categories representing the main subjects of organizations are pre-defined to organize tasks and codified knowledge items. The key features of tasks are extracted from related knowledge items. Tasks are classified into categories based on fuzzy classifications. Moreover, tasks with similar subjects are grouped into fields. Details are discussed in Section 4.1.

Profile handler. Two kinds of profiles, task profile and work profile, are maintained to model workers' information needs on the target task at hand. Task profile describes the key features of a task and is the kernel for discovering and disseminating task-relevant information to knowledge workers. Work profile models a worker's information needs on the target task, and is represented as a set of relevant tasks or fields of the target task with associated relevance degrees. Workers' information needs may change during the progress on performing the target task. The *user behavior tracker* is an on-line module to capture workers' dynamic behaviors, including access behaviors on the task-based domain ontology and relevance feedbacks on knowledge items. The *profile handler* uses an adaptive task-based profiling approach, which is described in Section 4.2, to

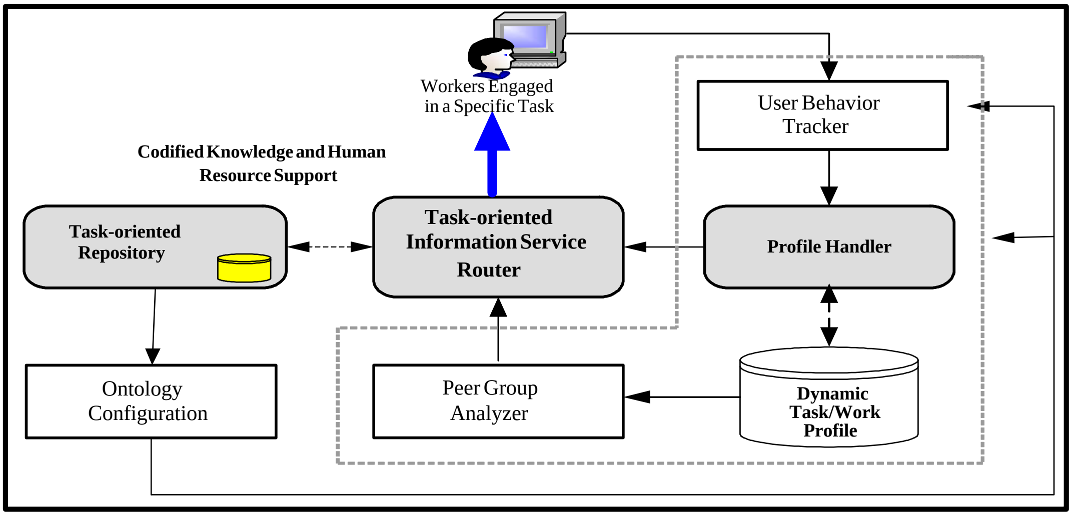


Fig. 1. Task-based knowledge support.

adjust workers' profiles. The *peer-group analyzer* employs a fuzzy analytical method described in Section 5 to identify peer-groups with similar task needs (information needs on the target task) based on work profiles.

Task-oriented information service router. The task-oriented information service router helps knowledge workers gather appropriate information from the task-oriented repository and task-based peer-groups. The router fetches task-relevant information according to the worker's task profile. Moreover, each worker has his/her own view of task-relevant information, namely, personalized ontology, which is derived from his/her work profile on the target task and is organized according to the domain ontology. Knowledge sharing from other peer-group members is derived by retrieving each peer-group member's personalized ontology. The retrieval and sharing of task-relevant knowledge is demonstrated in Section 6.

3.2. System architecture

Fig. 2 depicts the system architecture comprising four implementation layers, including knowledge resource

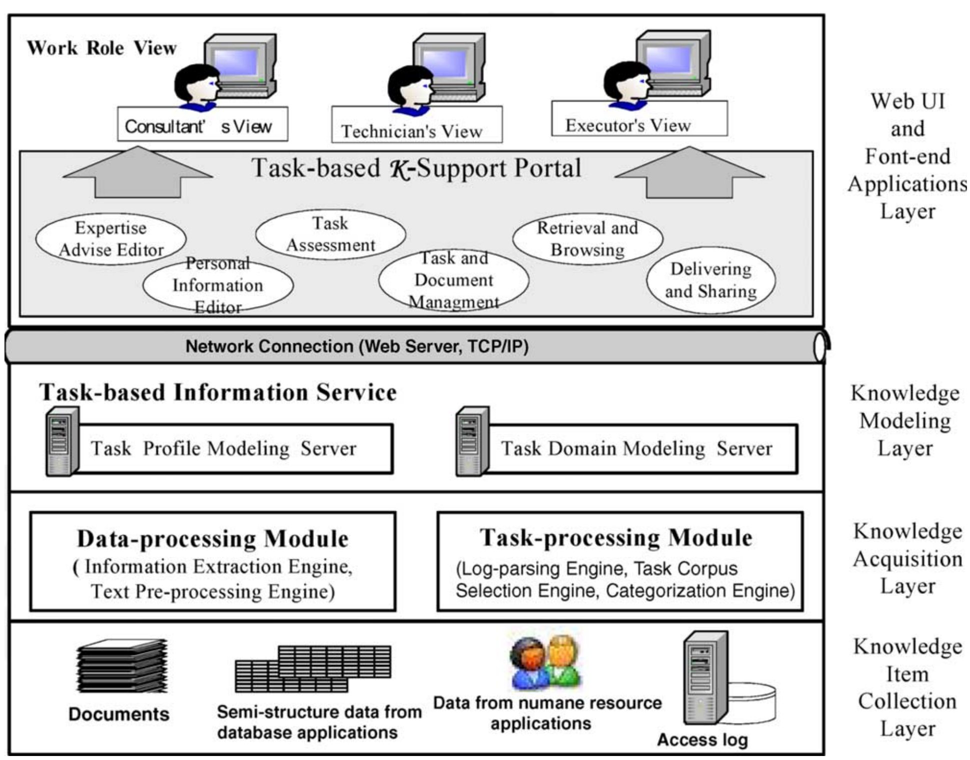


Fig. 2. System architecture.

collection, knowledge acquisition, knowledge modeling, and Web-based front-end application.

Knowledge resource collection layer. The unstructured or semi-structured information embedded in records such as documents, presentation slides, reports, lesson-learned, database entries, etc., are valuable knowledge items. This layer collects information expressed in various forms from different knowledge sources that are generated and accessed during task executions. Meanwhile, the system collects data from human resource applications to provide a platform for gathering and exchanging task-relevant knowledge among workers.

Knowledge acquisition layer. This layer extracts explicit (codified) knowledge and tacit (human resource) knowledge within the organization. Two modules are responsible to handle and process task-relevant knowledge items: one is the *data-processing module* and the other is the *task-processing module*. This layer employs information retrieval, text mining and database techniques to process and organize task-relevant information.

- The *data-processing module* deals with textual data represented in different formats. The information extraction engine retrieves meaningful information such as title, abstract, and author name from documents. The text pre-processing engine employs *term transformation*, *term weighting*, and *feature selection* steps (Baeza-Yates & Ribeiro-Nero, 1999; Porter, 1980; Salton & Buckley, 1988) to extract meaningful information (metadata) of textual-based knowledge items.
- The *task-processing module* comprises three processing units, including log-parsing, task corpus selection, and task categorization engines to handle task-relevant data. The log-parsing engine analyzes log-files to track user's interaction with the system. The task corpus selection engine generates the task corpus of a task t_r by analyzing the contents of textual data accessed by t_r. The task corpus represents the key features of a task. The task categorization engine is responsible for ontology configuration via a seed-based fuzzy classification technique.

Knowledge modeling layer. This layer is responsible for modeling task-relevant information and workers' information needs. The *domain modeling module* is responsible for representing domain ontology from the aspect of task. The module handles the grouping of similar tasks into fields. The ontology is used to represent the organization's domain-specific knowledge. Task related information (e.g. skills, knowledge, workers, and documents, etc.) is also conceptualized into the agreed ontology to provide knowledge support. The *profile modeling module* provides mechanisms such as profile creation, modification, and integration to conduct profile management. Moreover, the module implements the profile handler described in Section 3.1. Profile modeling is the kernel to support knowledge retrieval and sharing.

Web-based GUI and front-end application layer. An integrated platform is built upon the *profile modeling server* and *domain modeling server* to construct the task-based knowledge support portal. This layer mainly provides the function of a task-oriented retrieval router described in Section 3.1. Moreover, the proposed system considers the task perspective to acquire and disseminate task-relevant knowledge. Different knowledge management applications are available for workers. For example, the function of *task assessment editor* assists a worker to conduct task assessment to create his/her own task profile. A worker may use the *personal information editor* to organize his/her own knowledge. In addition, a worker can enter the *task-based workspace* to browse, access, and organize task-relevant knowledge. The proposed system not only delivers task-relevant knowledge based on task profile but also identifies peer-groups with similar task-needs (or similar projects) based on work profiles. Workers engaged in the same task or with common task needs can solve the encountered problem together, thereby realizing collaborative task-based knowledge support.

4. Adaptive task-based profiling

Section 4.1 describes the configuration of task-based information into an ontological structure, while Section 4.2 describes the proposed profiling approach which models and adapts to workers' dynamic information needs on the target task.

4.1. Domain ontology

A pertinent ontology to conceptualize the domain information of an organization is required. The domain ontology (DO) is structured into four levels, including categories, fields, tasks and knowledge items, as shown in Fig. 3. Categories representing the main subjects of organizations are pre-defined to organize tasks and codified knowledge. Tasks with similar subjects are grouped into fields. The procedures to construct the domain ontology are as follows. The key contents of codified knowledge (textual data; documents) can be represented as a vector of weighted terms, using a term weighting approach that considers term frequency, inverse document frequency, and normalization factors (Salton & Buckley, 1988). Task corpus describes the key features of tasks and can be constructed by extracting key contents from textual data gathered during task execution. Categorizing tasks is then undertaken via a fuzzy classification approach. The categorization derives fuzzy relationships between tasks and categories. The final step groups similar tasks into fields.

Task corpus extraction. The task corpus of a task t_r is represented as a feature vector of weighted terms

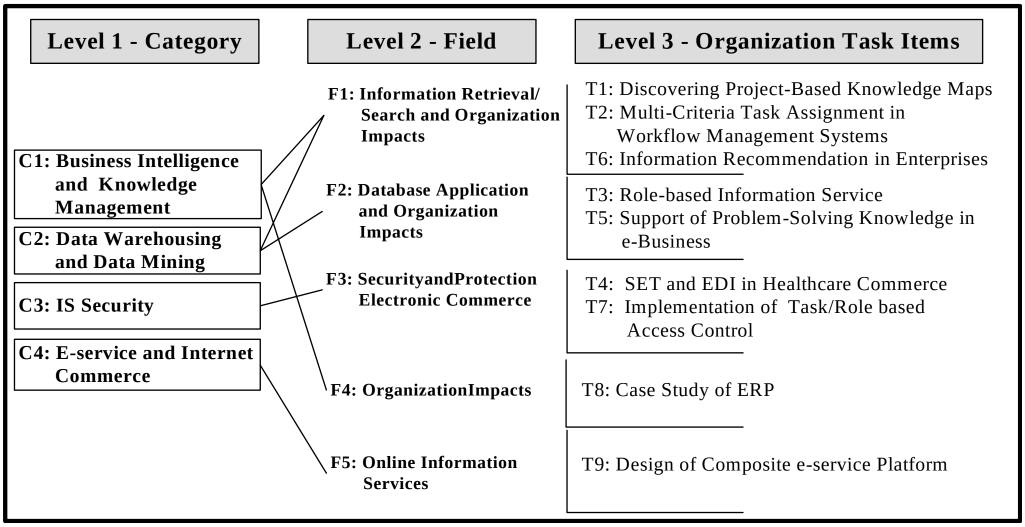


Fig. 3. Example of domain ontology.

(keywords). The task corpus of task t_r is defined as the centroid vector $\vec{t}_r$, which is the vector, obtained by averaging the feature vectors of documents generated and accessed by t_r.

Task categorization. This work employs a seed-based fuzzy classification method to categorize tasks into categories. Fuzzy classification extends the traditional crisp classification notation by associating objects in every category with a membership function so that each object can belong to more than one category (Zadeh, 1965). A category can be represented as a feature vector of weighted terms derived from the task corpora of its seed tasks. The degree of relevance between task and categories is calculated based on term distributions under the vector space model. A fuzzy relation matrix $\boldsymbol{R}$ is defined to represent the relevance between tasks and categories. $\boldsymbol{R}_{\text{k-by-m}}=[\mu_{\text{Ci}}(t_r)]$ denotes an k-by-m fuzzy relation matrix, whereas an element $\mu_{\text{Ci}}(t_r)$ ($\mu_{\text{Ci}}(t_r)\in[0,1]$) in the matrix denotes the relevance degree of rth task to the ith category. The notation m represents the number of categories while k represents the number of tasks. Accordingly, the relevance degree of task t_r to categories can be modeled as a vector characterized by membership grades to categories, $\vec{\mathbf{t}}_r^C = <\mu_{c_1}(t_r), \mu_{c_2}(t_r), \ldots, \mu_{c_m}(t_r)>$. The details of task categorization are presented in (Wu & Liu, 2003).

Field configuration. Tasks with similar subjects are grouped into fields. Notably, the relevance degrees to categories represent the subjects of a task. The similarity between tasks can thus be calculated based on their relevance degrees to categories. Based on the fuzzy relationship matrix $\boldsymbol{R}$, similar tasks are grouped together to form a field, as follows. A threshold value, $thres\theta$, is defined to transform the fuzzy relation matrix $\boldsymbol{R}$ into a binary relation matrix $\boldsymbol{B}$. The threshold value is determined by the max–min operation, as shown in Eq. (1).

$$thres\,\theta = \max(\mu_{\min}(t_1), \mu_{\min}(t_2), \ldots, \mu_{\min}(t_k));$$
$$\text{where}\quad \mu_{\min}(t_r) = \min(\mu_{C_1}(t_r), \mu_{C_2}(t_r), \ldots, \mu_{C_m}(t_r)) \tag{1}$$
$$\text{for}\quad r = 1, 2, \ldots, k$$

According to Eq. (2), the fuzzy relation matrix $\boldsymbol{R}$ is transformed into a binary relation matrix $\boldsymbol{B}$.

$$\mu_{C_i}(t_r) = \begin{cases} 1 & \mu_{C_i}(t_r) > thres\theta \\ 0 & \mu_{C_i}(t_r) \le thres\theta \end{cases} \tag{2}$$

Tasks that have the same relationship with respect to each category in B, are similar tasks to be grouped into a field labeled by a field name. This work defines fields according to the schema of ACM Computing Classification Systems. The result generates a l-by-k field-to-task relation matrix $\boldsymbol{F}=[f_j(t_r)]$ such that $f_j(t_r)$ is one if task t_r is grouped into field f_j; and is zero otherwise; where l denotes the number of fields. Fig. 3 shows an example of domain ontology.

4.2. Profile modeling

Knowledge retrieval and sharing rely on profile modeling to capture workers' information needs on the target task. A systematic approach to assessing task relevance for appropriately constructing task profiles is proposed in our previous work (Wu & Liu, 2003). The proposed approach generates task profiles, namely feature vectors of weighted terms, by analyzing similar historical tasks in organization memory instead of conducting evaluations on a tremendous amount of documents.

In this work, the adaptation of profile is proposed to model workers' dynamic information needs on the target task based on users' access behaviors or relevance feedback on knowledge items. A modified relevance feedback technique is employed to adjust workers' dynamic information needs. The traditional relevance feedback technique requires the user to rate information items explicitly. However, in practice, users are unwilling to conduct tedious relevance feedback. This work proposes an adaptive task-based profiling approach to model workers' dynamic information needs via considering both explicit feedback, where our system collects workers' linguistic ratings, and implicit feedback, where our system monitors user access behavior. Meanwhile, a worker's relevance feedback is modeled by a fuzzy linguistic approach, as described in the following.

Perception modeling through fuzzy linguistic approach. The fuzzy linguistic approach is a technique for approximating human perception, and provides easier assess to qualitative problems. Linguistic assessment uses words rather than numbers. For example, the linguistic variable '*Relevance*' is defined to assess the degree of relevance between objects (such as document, task, etc.). Notably, a linguistic variable is characterized by a quintuple (S, $E(S)$, U, G, M) as defined in Definition I (Zadeh, 1975). The semantic meaning of a linguistic term can be formulated as a fuzzy number, which represents the approximate value of each linguistic term.

Definition I. A *linguistic variable* is expressed as a quintuple (S, $E(S)$, U, G, M) where S denotes the *name* of the variable; $E(S)$ is the *linguistic term* of S, namely the set of its linguistic values range over a universe of discourse U; G is a *syntactic rule* (a grammar) which generates the linguistic term set in $E(S)$; and M is a *semantic rule* that assigns meaning, $m(e)$, to each linguistic term e in E with a fuzzy set on U.

From Definition I, a linguistic variable, *Relevance*, is defined to represent the degree of relevance between items (tasks or categories) assessed by evaluators. $E(Relevance)$ is characterized using a fuzzy set of a universe of discourse $U=[0,1]$, in which six linguistic terms r_j and their associative semantic meanings $m(\mathrm{r}_j)$ are defined as follows:

$$E(\text{Relevance}) = \{\mathrm{r}_0 = \text{Very Low}(VL),$$

$$\mathrm{r}_1 = \text{Low}(L), \quad \mathrm{r}_2 = \text{Normal }(N), \quad \mathrm{r}_3 = \text{High}(H),$$

$$\mathrm{r}_4 = \text{Very High }(VH), \quad \mathrm{r}_5 = \text{Perfect }(P)\}$$

where $m(\mathrm{r}_i) < m(\mathrm{r}_j)$, for $i<j$, and all $m(\mathrm{r}_j)$ are distributed in [0,1].

The fuzzy linguistic approach models the meaning of each term using fuzzy numbers, as defined in Definition II (Dubis & Prade, 1978). The fuzzy number plays a fundamental role in formulating the semantic meaning of the linguistic term, which represents an approximate value of the linguistic variable.

Definition II. A fuzzy number $\tilde{Z}$ is a 'normal' and 'convex' fuzzy set defined on the set $\mathbb{R}$ and $\tilde{Z}$ is a closed interval for every $\alpha \in (0,1]$. The membership function $f\tilde{Z}(x)$ of the triangular fuzzy number (TFN), $\tilde{Z} = (l, m, r)$, is presented in Eq. (3)

$$f\tilde{Z}(x) = \begin{cases} (x-l)/(m-l) & l \le x \le m \\ (r-x)/(r-m) & m \le x \le r \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

This work adopts the center of area (COA) method to calculate fuzzy numbers, owing to its simplicity and practicability. The COA method calculates the fuzzy mean under uniform probability distribution assumption (Lee & Li, 1988). If the fuzzy number $\tilde{U}$ is triangular, where $\tilde{U} = (l, m, r)$, the crisp rating can be derived by the equation: $CV(\tilde{U}) = [(r-l)+(m-l)]/3 + l$.

4.2.1. Profile structuring

Two kinds of profiles, task profile and work profile, are maintained in our system. Both profiles are used to represent a worker's current information needs on the target task at hand. The task profile of a task t_r is a feature vector of weighted keywords, denoted as $\vec{t}_\mathrm{r} = <w_{\mathrm{kw}\ 1}, w_{\mathrm{kw}\ 2}, \ldots, w_{\mathrm{kwn}}>$. The work profile of a worker u, denoted as $WP_u = \{<topic_j, w_p(topic_j)>\}$, contains a set of topics (fields or tasks in domain ontology) with associated degree of relevance to the target task at a specific time period. $w_p(topic_j)$ represents the relevance degree of $topic_j$ to the target task at time p, from the aspect of u. The associated degree of relevance indicates a similarity measure between a topic and the target task at a specific time period. The similarity measure is initially obtained from a worker's relevance assessment, and will be updated via analyzing a worker's access behaviors and explicit feedback as described in Section 4.2.2. Notably, category level is not considered since the topics in category are too general to differentiate workers' task needs. Let FS denote the set of topics in field level and TS denote the set of topics in task level. A work profile represents a worker's task-needs expressed as a set of relevant fields or tasks in domain ontology, and can be used to derive a worker's personalized ontology (WPO) on the target task. An ontology threshold value δ can be defined by a worker to generate a WPO on the target task by filtering out irrelevant fields or tasks with relevance degrees below the threshold value. Accordingly, $WPO_u = \{<topic_j, \ w_p(topic_j)> | w_p(topic_j) \ge \delta$ and $topic_j \in \boldsymbol{FS} \cup \boldsymbol{TS}\}$. The result forms a worker u's personalized ontology on the target task.

4.2.2. Feedback analysis

In the proposed system, each worker's feedback is collected and analyzed by the *user behavior tracker* which is activated periodically. A temporal profile, denoted as $\vec{T}emp_{u,p}$, is generated by the profile handler to represent a worker u's current information needs on the target task.

The temporal file is derived from the feature vectors of those documents accessed by worker u during time period p, as shown in the Eq. (4).

$$\vec{T}emp_{u,p} = \frac{1}{|D^{\exp}_{u,p}|} \times \sum_{\forall dj \in D^{\exp}_{u,p}} (A_u \times (d_j) \times \vec{d}_j)^j + \frac{1}{|D^{imp}_{u,p}|} \times \sum_{\forall dj \in D^{imp}_{u,p}} (CV(\tilde{H})^u \times \vec{d}_j) \tag{4}$$

$D^{\exp}_{u,p}$ denotes the set of documents which had been explicitly rated by worker u in conducting the target task during the time period p. $A_u(d_j)$ denotes worker u's crisp rating on the relevance of document d_j to the target task. The crisp rating is derived from the linguistic rating according to the COA method described previously. $D^{\mathrm{imp}}_{u,p}$ denotes the set of documents which had been browsed and accessed but not been rated by worker u during time period p. A linguistic rating 'High' is given by default to represent the relevance degree of unrated documents (implicit feedback). $CV(\tilde{H})^u$ denotes the corresponding crisp value of relevance rating 'High' of worker u. Notably, our system will show the description of a document. Thus, we assume that a worker will read the description first to decide if the document is relevant, and then access and browse the document. Accordingly, a linguistic rating 'High' is assigned to unrated documents that had been accessed and browsed by the worker.

The similarity between the temporal profile and a topic t_j in the domain ontology can be derived by cosine measure, namely $\mathrm{sim}(\vec{T}emp_{u,p}, \vec{t}_j)$. Notably, $\mathrm{sim}(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y} / |\vec{x}||\vec{y}|)$.

4.2.3. Profile adaptation

The system will adjust the work profile based on the result of feedback analysis. Notably, a work profile records topics (tasks or fields) with associated relevance degree to the target task. The associated relevance degree of each topic is adjusted as follows. The system will increase or decrease the relevance degree (associated weight) of a task t_j (a topic in the task level of domain ontology) gradually, where $w_{p+1}(t_j) = w_p(t_j) \pm \Delta w$. The adjustment Δw of a task t_j is derived based on the proportion of feedbacks and the similarity between the temporal profile and t_j. If $\mathrm{sim}(\vec{T}emp_{u,p}, \vec{t}_j)$ is above a relevance-adjustment threshold θ, the system will increase the associated weight of task t_j. Meanwhile, if $\mathrm{sim}(\vec{T}emp_{u,p}, \vec{t}_j)$ is below θ, the system will decrease the associated weight of task t_j. The adjustment is given by Eq. (5).

$$\Delta w(t_j) = \frac{N^d_{u,p}}{N^d_u + N^d_{u,p}} \times |\mathrm{sim}(\vec{T}emp_{u,p}, \vec{t}_j) - \theta| \tag{5}$$

where N^d_u denotes the number of documents accessed and browsed by worker u in conducting the target task prior to time p, while $N^d_{u,p}$ denotes the number of documents accessed and browsed by worker u in conducting the target task during time p. Notably, $w_{p+1}(t_j) = 1$, if $w_p(t_j) + \Delta w > 1$; $w_{p+1}(t_j) = 0$, if $w_p(t_j) - \Delta w < 0$. Moreover, a field contains a set of tasks. Thus, the value of $w_p(field_i)$ is set to the maximum value of $w_p(t_j)$ for any *task* t_j belongs to $field_i$. Namely, the weight of $field_i$ will be adjusted at time $p+1$, where $w_{p+1}(field_i) = \max_{tj \in fieldi}(w_{p+1}(t_j))$.

Meanwhile, the adjustment may change the information structure of a WPO. The personalized ontology of worker u is adjusted by removing an irrelevant topic t_j, if $w_{p+1}(t_j)$ is below the ontology threshold δ, and adding a relevant topic t_j, if t_j did not exist at time p and $w_{p+1}(t_j) \geq \delta$.

Furthermore, the task profile of the target task can be adapted based on the adjustment of work profile. The system generates a set of top-k relevant tasks (denoted as TRTs) and a set of top-k irrelevant tasks (denoted as TIRTs) based on the work profile. Relevant tasks are those tasks with associated relevance degree $w_{p+1}(t_j)$ higher than the relevance-adjustment threshold θ, whereas irrelevant tasks are those tasks with $w_{p+1}(t_j)$ lower than θ. Only the feature terms and the associated relevance degrees of topics in the task level are used to adjust the task profile of the target task, since a field is a generic view of similar tasks; hence, the feature terms of fields are not as representative as the feature terms of tasks for the target task. The new task profile of the target task, denoted as $\vec{S}_{p+1}$ is generated based on Eq. (6), which is modified from standard Rocchio (1971); Ide (1971) algorithms presented in Appendix I. The modification considers the associated relevance degrees of relevant/irrelevant tasks to the target task and the temporal profile derived form the feedback analysis.

$$\begin{aligned} \vec{S}_{p+1} &= \alpha \vec{S}_p + \beta \vec{O} - \gamma (1 - w_{p+1}(t_j)) \sum_{\forall t_j \in T_n} \vec{t}_j \\ \vec{O} &= \lambda \sum_{\forall t_j \in T_r} w_{p+1}(t_j) \vec{t}_j + (1 - \lambda) \vec{T}emp_{u,p} \end{aligned} \tag{6}$$

where $\vec{S}_p$ denotes the task profile of the target task at time p. Notably, $\vec{S}_p$ may be an initial task profile derived from the initial assessment result. The $\vec{O}$ denotes the aggregated relevant feature vector of the target task. The aggregation of irrelevant feature vectors is derived from T_n which is the set of top-k irrelevant tasks (TIRTs). The relevant feature vector $\vec{O}$ is derived based on T_r, the set of top-k relevant tasks (TRTs), and the temporal file generated from the feedback analysis described in Section 4.2.2. Notably, $w_{p+1}(t_j)$ denotes the associated relevance degree of task t_j to the target task. $\vec{T}emp_{u,p}$ denotes the temporal profile derived from the feedback analysis. Meanwhile, α, β, γ are tuning constants. The parameter λ is used to adjust the relative importance of relevant tasks and the temporal profile.

4.3. Task-based knowledge retrieval

A worker's task profile and work profile can properly reflect a work's task-needs on the target task. The profiles can be used to further enhance the knowledge retrieval

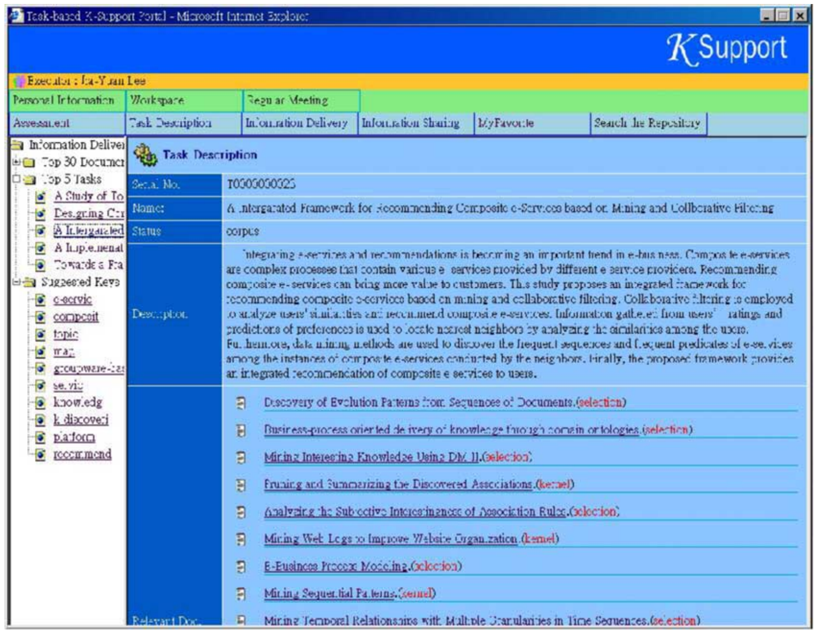


Fig. 4. Interface of knowledge delivery.

capability in the proposed system. Moreover, the adjustment of work profile across time will lead the system to refine the task profile based on the profile adaptation approach described in Section 4.2. Accordingly, $\vec{S}_{p+1}$ is used to retrieve relevant codified knowledge in the repository. Relevant task and document sets will be retrieved to provide knowledge support for task execution according to the similarity measure (e.g. cosine measure). Fig. 4 is the interface of knowledge delivery in which the system delivers task-relevant knowledge proactively based on the task profile.

5. Fuzzy peer-group analytical model

This work employs a fuzzy analytical model to identify peer-groups with similar task needs based on work profiles. The proposed method mainly contains two phases. In phase 1, a fuzzy user-user similarity matrix is constructed to record workers' similarity relationships on task-needs. In phase 2, a fuzzy inference procedure is employed to infer the implicit and transitive relationships among workers. The α-cuts approach is then applied to generate a proper set of task-based peer-groups.

Section 5.1 describes the construction of a fuzzy user-user similarity matrix based on work profiles. Notably, a work profile records a worker's task-needs on the target task which are represented as a set of topics (fields or tasks in domain ontology) with associated degree of relevance to the target task. Section 5.2 describes the fuzzy inference and α-cuts method used to derive task-based peer-groups based on user similarity matrix. The system can then stimulate knowledge sharing among peer-groups with similar task-needs.

5.1. Establishing a user-user similarity matrix

The similarity measure between workers E_x and E_y can be derived using the associated relevance degrees $w_p(topic_j)$ of topics recorded in the work profiles of E_x and E_y. The task-level/field-level relevance degrees can be used to derive the task-level/field-level similarity between workers. Notably, a field contains a set of tasks. As described in Section 4.2.3, the relevance degree in field level is derived from the task-level, namely, the value of $w_p(field_i)$ is set to the maximum value of $w_p(t_j)$ for any *task* t_j belongs to $field_i$. Thus, the task-level similarity is stricter than the field-level similarity. Very few similar users can be identified based on the task-level similarity if there are very few tasks relevant to the target task. More similar users can be identified based on the field-level similarity. However, our pilot experimental analysis shows that the field-level similarity derived from the maximum value of task-level relevance degree is too vague to measure the similarity between workers. Accordingly, we employ a compromised approach to compute field-level similarity based on the aggregation of task-level relevance degrees. An aggregated field-level relevance degree of a $field_i$ is derived from the aggregation (summation) of $w_p(t_j)$ for all *task* t_j belongs to $field_i$. Then, the field-level similarity is measured according to the aggregated field-level relevance degrees, as described in the following steps.

Step 1: Constructing a task-level user feedback matrix

An n-by-k user-feedback matrix $\boldsymbol{I}$ (*task-level*) is constructed to represent each worker's task-level relevance degrees recorded in each worker's work profile, where n denotes the number of workers, and k denotes the number of task items. The task-level relevance degrees represent workers' perspective on the relevance of tasks to the target task.

Step 2: Deriving a field-level feedback matrix.

An n-by-l user-feedback matrix $\boldsymbol{I}$ (field-level) is derived via a matrix operation employed between the transpose of an l-by-k field-to-task binary relationship matrix $\boldsymbol{F}$ (described in Section 4.1) and task-level user-feedback matrix $\boldsymbol{I}$ (task-level), as shown in Eq. (7).

$$I(\text{field}-\text{level})_{n-\text{by}-l} = I(\text{task}-\text{level})_{n-\text{by}-k} \times F^{T}_{k-\text{by}-l} \tag{7}$$

l denotes the dimension of *Field*, k denotes the dimension of *Task*, and n denotes the number of workers.

Step 3: Determining the similarity relationship matrix.

The cosine measure (Eq. (8)) is employed to calculate the similarity among workers based on the n-by-l user-feedback matrix $\boldsymbol{I}$ (field-level).

$$\xi(E_i, E_j) = \frac{\vec{A}_{E_i} \cdot \vec{A}_{E_j}}{|\vec{A}_{E_i}||\vec{A}_{E_j}|} \tag{8}$$

$\vec{A}_{E_j}$ and $\vec{A}_{E_j}$ are workers' E_i and E_j's feedback values in field-level derived from the user-feedback matrix $\boldsymbol{I}$ (field-level).

Finally, a reflective and symmetric matrix is derived, denoted as an n-by-n fuzzy similarity relationship matrix S (Fig. 5), which represents the similarity-relationship on workers' task-needs.

5.2. Identifying task-based peer-groups

A fuzzy inference procedure is employed to infer the implicit and transitive relationships among workers. The α-cuts approach is then applied to generate a proper set of task-based peer-groups. This section demonstrates how the task-based peer-groups can be automatically identified. Notably, the fuzzy inference procedure is used to derive the inherent transitive relationships among workers. By fuzzy inference, the proposed system can effectively identify peer groups with similar task needs derived from the inherent transitive relationships. Accordingly, the system can identify similar workers with implicit and inherent transitive relationships, even if very few explicit similarity measures are found in the similarity relationship matrix S.

Step 1: Inferring user relationship by fuzzy inference

The n-by-n fuzzy similarity relationship matrix S represents the fuzzy relation among U, a set of workers. The fuzzy relation of workers is represented in terms of membership function $\xi(E_i, E_j) \in [0, 1]$. The method of transitive max–min closure (Chen & Horng, 1999; Klir & Yuan, 1995) is adopted to derive a reflective, symmetric, and transitive matrix, which is a fuzzy equivalence matrix. The definition of a transitive max–min closure $\boldsymbol{S_T}$ of the similarity matrix $\boldsymbol{S}$ is defined in Definition III, which is adopted from Klir and Yuan (1995).

Definition III. Given an n-by-n fuzzy similarity relationship matrix $\boldsymbol{S}$ which represents the fuzzy relation among U, a set of workers, where $|U| = n$. A transitive max–min closure $\boldsymbol{S_T}$ of the similarity matrix $\boldsymbol{S}$ is derived as $\boldsymbol{S_T} = \boldsymbol{S}^y$ by a sequence of max–min operations on the relation matrix until $\boldsymbol{S}^y = \boldsymbol{S}^{y+1} = \ldots = \boldsymbol{S}^{\infty}$. Notably, $\boldsymbol{S}^y = \boldsymbol{S}^{y-1^{\circ}S y-1}$, where y is an integer, $1 \le y \le n-1$ and $^\circ$ denotes a fuzzy max–min operation. The max–min composition and max operator for set unions are used to derive the transitive max–min closure $\boldsymbol{S_T}$. The fuzzy max–min operation is defined as shown in Eq. (9).

$$\xi^y(E_i, E_j) = \max_{E_u \in U} \quad \min(\xi^{y-1}(E_i, E_u), \xi^{y-1}(E_u, E_j)) \tag{9}$$

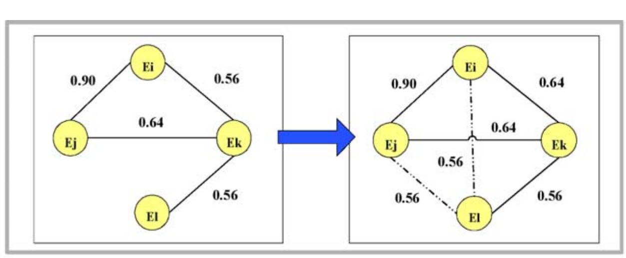
where $\xi^y(E_i, E_j)$ represents an element in S^y and $\xi^{y-1}(E_i, E_u)/\xi^{y-1}(E_u, E_j)$ represents an element in S^{y-1}.

Assume that the initial similarity relationships of workers are shown in the left part of Fig. 5. The right part shows the inferred transitive relationships among workers in matrix $\boldsymbol{S_T}$ after transitive max–min operations. The dashed line indicates the new inferred relationships after transitive max–min operations. For example, the relationship between E_i and E_l is 0.56.

Step 2: Identifying task-based peer-group by α-cuts.

The α-cuts can be applied to the equivalence matrix $\boldsymbol{S_T}$ for any α degree to group workers in $\boldsymbol{U}$, where $\alpha \in (0,1)$. Workers grouped together have equivalence relation. Several α degrees can be gradually refined to partition workers to form the subsets with equivalence relations. Different subsets of equivalence relations are derived by setting different α degrees in the matrix $\boldsymbol{S_T}$ to partition set $\boldsymbol{U}$. For example, two subsets of equivalence relations are derived by setting $\alpha = 0.64$ in the matrix $\boldsymbol{S_T}$ to partition

$$\boldsymbol{S} = \begin{bmatrix} 1.00 & 0.90 & 0.56 & 0.00 \\ 0.90 & 1.00 & 0.64 & 0.00 \\ 0.56 & 0.64 & 1.00 & 0.56 \\ 0.00 & 0.00 & 0.56 & 1.00 \end{bmatrix}$$



$$\boldsymbol{S}_T = \begin{bmatrix} 1.00 & 0.90 & \mathbf{0.64} & \mathbf{0.56} \\ 0.90 & 1.00 & 0.64 & \mathbf{0.56} \\ \mathbf{0.64} & 0.64 & 1.00 & 0.56 \\ \mathbf{0.56} & \mathbf{0.56} & 0.56 & 1.00 \end{bmatrix}$$

Fig. 5. Inferring similarity relationships based on workers' task-needs.

set $\boldsymbol{U}$.

$$\boldsymbol{S}_{\boldsymbol{T}}^{\alpha=0.64}=\begin{bmatrix}1&1&1&0\\1&1&1&0\\1&1&1&0\\0&0&0&1\end{bmatrix},$$

where $\alpha=0.64$

5.3. Task-based knowledge sharing

Effectively codifying tacit knowledge is difficult. The $\mathcal{K}$-Support system uses work profiles to identify task-based peer-groups. Two kinds of task-based peer-groups are located. The first is formal task-related members who join the same projects. The other is informal peer-groups with similar task-needs identified by the system. The system not only provides a knowledge support platform for gathering and exchanging task-relevant knowledge among workers, but also presents the peer-group member's personalized ontology on the target task for knowledge sharing.

6. Task-based $\mathcal{K}$-Support portal

The task-based $\mathcal{K}$-Support portal is a Web-based application, allowing workers to retrieve, organize and share task-relevant knowledge. Accordingly, there are two main applications provided in the proposed portal. One is *K-Delivery* application, which delivers task relevant knowledge proactively to support task execution. The other is *K-Sharing* application, which stimulates knowledge sharing by locating possible task-based peer-groups. The peer-group members' perspectives on the task needs, namely personalized ontology on the target task, are displayed in the form of a sharing tree structure. Both applications will be demonstrated.

6.1. K-Delivery: Delivering codified knowledge proactively

As described in Section 4, weighted discriminating terms are kept in the task profile for retrieving task-relevant knowledge. The system can proactively deliver task-relevant information based on the worker's task profiles. Fig. 6 shows the top-5 relevant tasks, top-30 relevant documents and 10 task-associated terms provided by the system. A tree-like structure is employed to organize task-relevant information. Once the worker selects a document or a task to read, the detailed information will be displayed, as shown in the right frame of Fig. 6. Meanwhile, the worker can view the description of any task-relevant document, as denoted in circle 1. If the worker gave a positive rating on the knowledge item (document or task sets), the system will preserve the item in the worker's MyFavorite folder. Notably, the *user behavior tracker* in the profile modeling server will track the worker's feedback or access behavior to adjust the task and work profiles.

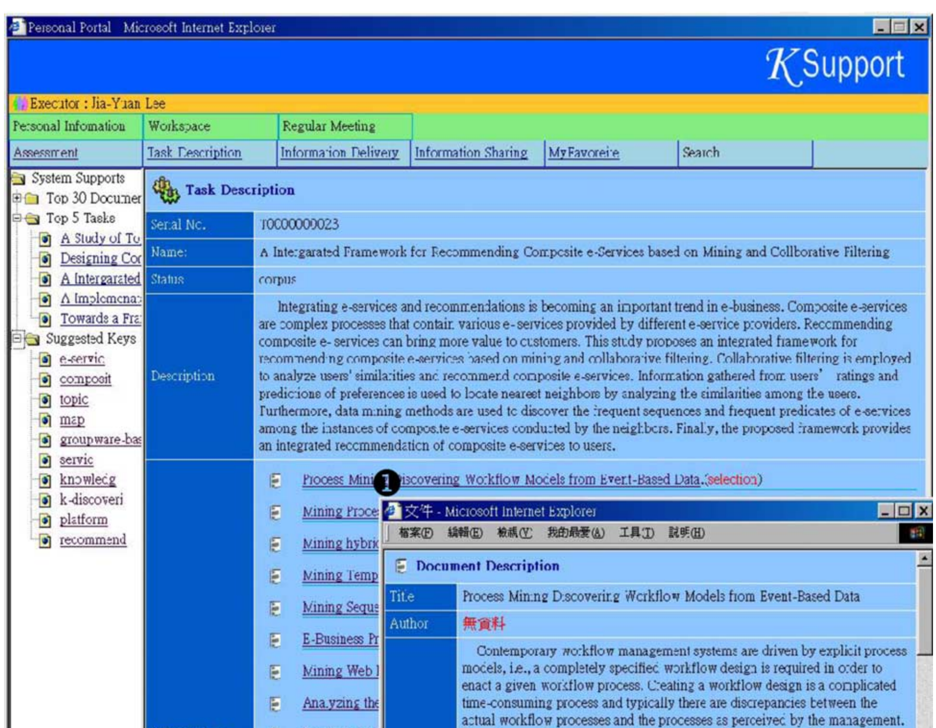


Fig. 6. Knowledge delivery.

6.2. K-Sharing: knowledge support from peer-group

The system expands the personalized ontology of a worker with the peer-group member's personalized ontology for knowledge sharing. Notably, a WPO represents a worker's perspective of task-needs on the target task. The personalized ontology is derived from the work profile to record tasks or fields that are relevant to the target task. The system identifies peer-groups with similar task needs based on work profiles. The system facilitates knowledge sharing by displaying the shared information such as relevant tasks and documents retrieved from peer-group members. The left frame of Fig. 7 shows the sharing tree of 'Jia-Yuan Lee', as denoted in circle 1. A sharing tree is a tree-like structure, which represents the personalized ontology of a worker. Meanwhile, the shared information from task-based peer-groups is also presented in the sharing tree.

In the given example, the ontology {H3.3 Information Retrieval and K4.3 Organization Impact, Mining Association Rule for Recommendation in Enterprises} is shared from 'Mike Lee', as denoted in circle 1 of Fig. 7. Another tree-like structure below the sharing tree is used to organize the shared document sets from the task-based peer-group (as denoted in circle 3). All information is calculated timely and automatically according to the feedback results. The *peer group analyzer* in the profile modeling server will be activated to identify task-based peer-group once the work profiles have been updated. Notably, a threshold, α-cut level, which is shown in the top left frame, can be adjusted by the workers to find more peer-group members by decreasing the α value.

7. Experimental evaluation

Various experiments are conducted to evaluate the effectiveness of the proposed $\mathcal{K}$-Support system. Evaluation is conducted with respect to the information needs of users participating in knowledge-intensive tasks such as conducting thesis work or research projects. Two evaluation metrics were considered to examine the effectiveness of the system-the novelty and the quality of the knowledge items suggested by the system. The $\mathcal{K}$-Support system consists mainly of two applications, *K-Delivery* and *K-Sharing*, as described in Section 6. The *K-Delivery* application delivers task relevant knowledge to workers based on the adaptation of task profiles. Notably, task profiles are adapted according to worker's dynamic information needs, namely access behaviors or explicit feedback. The *K-Sharing* application provides peer-group members' task relevant knowledge for knowledge sharing. Accordingly, we evaluate the effectiveness of *K-Delivery* application by comparing the *K-Delivery* based on initial task profiles (without adaptation) with the *K-Delivery* based on adapted task profiles. Moreover, we evaluate the quality and novelty of shared knowledge items provided by the *K-Sharing* application.

Section 7.1 describes the experimental data and evaluation metrics. Section 7.2 presents the evaluation result. The implications from the proposed system are also discussed.

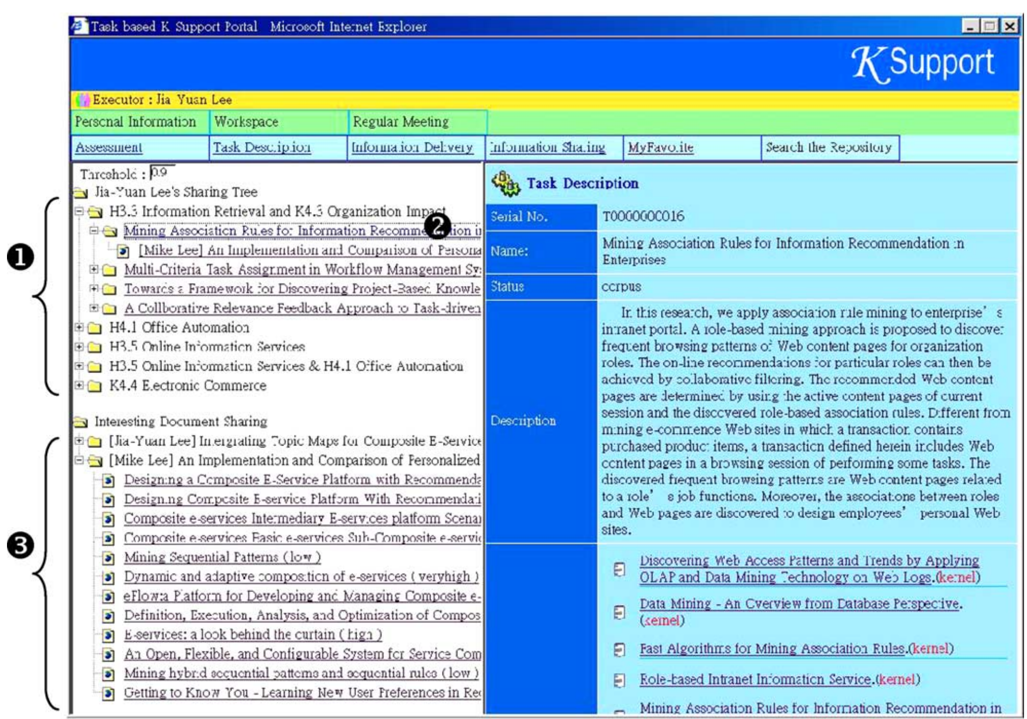


Fig. 7. Knowledge sharing ($\alpha=0.9$).

7.1. Experimental setup

7.1.1. Data and participants

Experiments were conducted using data collected from a research institute. Forty-eight research tasks were collected, with thirty-two existing-tasks and sixteen executing-tasks (target tasks that workers conduct at hand). Six executing-tasks were chosen as the testing set for evaluations. Over 500 task-related documents were collected. Tasks are classified into five categories and then grouped into thirteen fields. The smallest meaningful components of document information elements, such as title, abstract, journal and author, were extracted from documents. Each document contained an average of ninety distinct terms after information extraction, and document pre-processing (e.g. case folding, stemming, and stop word removal).

Twelve users were selected to participate in the evaluation. Two kinds of user group were selected to conduct the experiments. One group consisted of experienced users who were familiar with the executing task and the other group consisted of novices who were unfamiliar with the executing task.

7.1.2. Three phases of $\mathcal{K}$-Support

The $\mathcal{K}$-Support system contains three phases, including *phase I*: *K-Delivery* based on initial task profiles, *phase II*: *K-Delivery* based on adapted task profiles, and *phase III*: *K-Sharing* from task-based peer-groups. The $\mathcal{K}$-Support system performs phase I, then phase II, and finally phase III. We evaluate the effectiveness of profile adaptation based on the experimental result of phase II. The evaluation will demonstrate whether the proposed adaptive task-based profiling method can model worker's dynamic information needs properly. We also evaluate the effectiveness of knowledge sharing based on the experimental result of phase III. The evaluation will demonstrate whether the proposed peer-group analytical model can effectively identify task-based peer-groups based on work profiles.

7.1.3. Evaluation metrics

In general, evaluating the retrieval performance by considering all retrieved items is difficult, since users may not give feedback values (relevance ratings) on all retrieved items. User-oriented metrics derived from users' perceptions on retrieved items were usually used to evaluate the retrieval performance (Baeza-Yates & Ribeiro-Neto, 1999). Two user-oriented metrics, novelty and quality, were adopted to evaluate the effectiveness of the proposed system from users' perceptions on knowledge items, namely users' feedback values (relevance ratings) on knowledge items. The retrieved knowledge items which have been rated by users were used to derive the evaluation metrics. Moreover, the evaluation considered three phases of the system.

The *novelty* metric measures the ratio of relevant knowledge items retrieved that are unknown to the user (worker) E_i, as defined in Eq. (10). The relevant knowledge items are those items retrieved with feedback value above 'Normal' from worker's perception.

$$\text{Novelty} = \frac{|R_u|}{|R_u| + |R_k|} \tag{10}$$

where $|R_k|$ denotes the number of relevant knowledge items retrieved (in current and previous phases) which are known to worker E_i, whereas $|R_u|$ denotes the number of relevant knowledge items retrieved (in current phase) which are unknown to worker E_i. Notably, an item is known (unknown) to worker E_i, if that item had (not) been rated by E_i in previous phases. The *novelty* metric is used to measure the effectiveness of the system in discovering new (previously unknown) knowledge items that suit user needs.

The *quality* metric measures the fraction of aggregated ratings of retrieved knowledge items to the aggregated maximum ratings of retrieved knowledge items, as defined in Eq. (11).

$$\text{Quality} = \frac{\sum_{j \in R} a_j^{E_i}}{CV(\tilde{P})^{E_i} \times |R|} \tag{11}$$

$CV(\tilde{P})^{Ei}$ denotes the corresponding crisp value of maximum relevance rating 'Perfect ' given by worker E_i. R denotes the set of knowledge items retrieved and rated by worker E_i in current phase. a_j^{Ei} is the crisp feedback value on retrieved knowledge item j given by worker E_i. The *quality* metric is used to measure the worker's satisfaction degree on the retrieved knowledge items (e.g. tasks and documents).

7.2. Experimental result

Table 1 shows the novelty of *K-Delivery* and *K-Sharing*, respectively. Notably, the novelty for Initial *K-Delivery* in phase-I is not filled out since the novelty is 1.000, i.e., all retrieved knowledge items are unknown to

Table 1
Users' perceptions of information novelty

Phases of $\mathcal{K}$-Support	Conditions		Experienced		Novices	
			Task	Document	Task	Document
Phase I	Initial K-delivery	Average novelty	–	–	–	–
Phase II	Adapted K-delivery	Average novelty	0.283	0.540	0.373	0.520
Phase III	K-Sharing	Average novelty	0.612	0.570	0.650	0.613

Table 2
Users' perceptions of information quality

Phases of $\mathcal{K}$-Support	Conditions		Experienced		Novices	
			Task	Document	Task	Document
Phase I	Initialized K-delivery	Average quality	0.703	0.639	0.702	0.621
Phase II	Adapted K-delivery	Average quality	0.657	0.773	0.784	0.774
Phase III	K-sharing	Average quality	0.569	0.725	0.689	0.767

users in *initial K-Delivery*. The result shows that *K-Delivery* in phase-II can discover new (unknown) and relevant (feedback value above 'Normal') items based on adapted task profiles. Thus, the adaptation of task profiles to model workers' dynamic information needs is important to provide necessary knowledge support. Moreover, Table 3 shows that the novelty of *K-Sharing* in phase-III is higher than that of *K-Delivery* in phase-II. The result reveals that *K-Sharing* in phase-III can help workers find more new and relevant knowledge items from peer-group members.

The novelty of task-items under *K-Delivery* in phase-II is below 0.5. The result implies that the relevant task set is stable from phase-I to phase-II. Furthermore, for experienced workers, the novelty of task-items under *K-Delivery* in phase-II is lower than that for novices. The result implies the task profiles of experienced workers are more stable than those of novices. Novices are usually uncertain about their information needs in the beginning, and thus often adjust their information needs during task performance. The *adapted K-Delivery* can find more proper relevant tasks for novices based on the adaptation of task profiles.

Table 2 shows the quality of *K-Delivery* and *K-Sharing*, respectively. All three phases can provide workers' knowledge items that suit their needs. In general, the quality (satisfaction degree) of *K-Delivery* in phase-II is higher than that of the other two phases. *K-Delivery* in phases-II shows good adaptation capability to satisfy workers' needs based on adapted task profiles. The result indicates that the $\mathcal{K}$-Support system can provide workers appropriate and needed knowledge items based on the adaptive task-based profiling approach.

Interestingly, the quality for novices is better than experienced workers, especially in phase-II and phase-III. We observed that experienced workers are more knowledgeable on the executing-tasks, thus are more certain on the relevance of knowledge items; most novices are not knowledgeable on the executing tasks, thus are uncertain on the relevance of knowledge items, and tend to give relevant ratings.

The experimental results conclude that the adaptive task-based profiling method and the fuzzy peer-group analytical model are effective to stimulate knowledge retrieval and knowledge sharing.

8. Conclusions and future works

A task-based $\mathcal{K}$-Support system is developed to acquire, model and disseminate codified knowledge among workers in task-based environments. The proposed architecture can be tailored to manage codified knowledge and human resources to support task execution. A $\mathcal{K}$-Support portal is built upon the system to facilitate task-based knowledge retrieval and sharing among task-based peer-groups. The knowledge support is realized by the proposed profile modeling approach. Therefore, the problem of accessing needed knowledge items from vast amounts of codified knowledge can be alleviated. In addition, this system identifies task-based peer-groups based on the proposed fuzzy analytical method. Knowledge sharing is achieved by enabling workers to share their task-relevant knowledge among peer-groups.

The proposed system can be applied to knowledge-intensive and task-based business environments, such as R&D department, Intellectual Property department, school laboratory, and the like. The proposed system can provide an effective portal to assist knowledge workers to fully reuse knowledge assets and to further achieve the goal of business tasks. Several issues need further investigations. First, this work does not consider the process-aspect and context awareness, as discussed in (Kwan & Balasubramanian, 2003). Future studies could extend the proposed approach to support context-aware or process-aware delivery of task-relevant knowledge. Moreover, the information needs of knowledge workers are associated with their roles in undertaken tasks; however, this work does not consider the role/job perspective (Alvarado et al., 2004) to acquire and disseminate task-relevant knowledge. Future studies could extend the proposed profiling approach by considering role/task to acquire and reuse corporate memory effectively.

Acknowledgements

This research was supported by the National Science Council of Taiwan (Republic of China) under the grant NSC 93-2416-H-009-011.

Appendix A. Relevance feedback techniques

Relevance feedback effectively improves search effectiveness through query reformulation. Various studies have demonstrated that relevance feedback applied in the vector model is an effective technique for information retrieval (Rocchio, 1971; Salton & Buckley, 1990). Eqs. (A1) and (A2) illustrate two classical relevance feedback methods designed by Rocchio (1971); Ide (1971), respectively. A modified query vector $\vec{q}_{\mathrm{m}}$ is derived using the relevance of documents (as feedback) to adjust the query vector $\vec{q}$ (Baeza-Yates & Ribeiro-Neto, 1999).

Standard_Rocchio :

$$\vec{q}_{\mathrm{m}} = \alpha\vec{q} + \beta\frac{1}{|D_{\mathrm{r}}|}\sum_{\forall d_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_n|}\sum_{\forall d_j \in D_n}\vec{d}_j \tag{A1}$$

$$\text{Ide\_Dec\_Hi}: \quad \vec{q}_{\mathrm{m}} = \alpha\vec{q} + \beta\sum_{\forall d_j \in D_r}\vec{d}_j - \gamma \max\nolimits_{\mathrm{irrelevant}}(\vec{d}_j) \tag{A2}$$

where D_r denotes the set of relevant documents and D_n represents the set of irrelevant documents according to user judgment. $|D_r|$ and $|D_n|$ represent the number of documents in the sets D_r and D_n. Meanwhile, α,β,γ are tuning constants. The function of $\max_{\mathrm{irrelevant}}$ returns the most irrelevant document. The two methods produce similar results (Baeza-Yates & Ribeiro-Neto, 1999). Most studies suggest that the information of relevant documents is more important than that of irrelevant documents (Herman, 1992; Salton & Buckley, 1990).

References

Abecker, A., Bernardi, A., Maus, H., Sintek, M., & Wenzel, C. (2000). Information supply for business processes: coupling workflow with document analysis and information retrieval. *Knowledge Based Systems*, *13*(1), 271–284.

Agostini, A., Albolino, S., De Michelis, G., De Paoli, F., & Dondi, R. (2003). Stimulating knowledge discovery and sharing. *Proceedings of the International ACM Conference on Supporting Group Work, Sanibel Island, Florida, USA, November* , 248–257.

Alvarado, M., Romero-Salcedo, M., & Sheremetov, L. (2004). A corporative memory based on the user profiles. *Expert Systems with Applications*, *26*(1), 87–94.

Andrade, J., Ares, J., Garcia, R., Rodriguez, S., & Suárez, S. (2003). *Lessons learned for the knowledge management systems development Proceedings of the 2003 IEEE International Conference on Information Reuse and Integration* pp. 471–477.

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval*. New York: The ACM Press.

Brown, J. S., & Duguid, P. (1991). Organization learning and communities of practice. *Organization Science*, *2*(1), 40–57.

Celentano, A., Fugini, M. G., & Pozzi, S. (1995). Knowledge-based document retrieval in office environment: The Kabiria system. *ACM Transactions on Information Systems*, *13*(3), 237–268.

Chen, S. M., & Horng, Y. J. (1999). Fuzzy query processing for document retrieval based on extended fuzzy concept networks. *IEEE Transaction on System, Man, and Cabernetics-Part B:Cybernetics*, *29*(1), 96–104.

Davenport, T. H., & Prusak, L. (1998). *Working knowledge: How organizations manage what they know*. Boston MA: Harvard Business School Press.

Davies, N. J., Duke, A., & Stonkus, A. (2003). OntoShare: Evolving ontologies in a knowledge sharing system. In N. J. Davies, D. Fensel, & F. van Harmelen (Eds.), *Towards the semantic web—ontology-based knowledge management* (pp. 161–176). London UK: Wiley, 161–176.

Dubis, D., & Prade, H. (1978). Operations on fuzzy numbers. *International Journal of Systems Science*, *9*(3), 613–626.

Fensel, D., Staab, S., Studer, R., van Harmelen, F., & Davies, N. J. (2003). A future perspective: Exploiting peer-to-peer and the semantic web for knowledge management. In N. J. Davies, D. Fensel, & F. van Harmelen (Eds.), *Towards the semantic web-ontology—based knowledge management* (pp. 245–265). London, UK: Wiley, 245–265.

Fenstermacher, Kurt D. (1999). *Putting context in knowledge management*. Thesis (Ph.D.), The University of Chicago.

Fenstermacher, Kurt D. (2002). *Process-Aware knowledge retrieval Proceedings of the 35th Hawaii international conference on system sciences, big island, Hawaii, USA* pp. 209–217.

Fischer, G., & Ostwald, J. (2001). Knowledge management: Problems, promises, realities, and challenges. *IEEE Intelligent Systems*, *16*(1), 60–73.

Gray, P. H. (2001a). Problem-solving perspective on knowledge management practices. *Decision Support Systems*, *31*(1), 87–102.

Gray, P. H. (2001b). The impact of knowledge repositories on power and control in the workplace. *Information Technology and People*, *14*(4), 368–384.

Harman, D. (1992). Relevance feedback revisited. *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Copenhagen, Denmark* , 1–10.

Herlocker, J. L., & Konstan, J. A. (2001). Content-independent, task-focused recommendation. *IEEE Internet Computing*, *5*(6), 40–47.

Ide, E. (1971). New experiments in relevance feedback. In G. Salton (Ed.), *The SMART retrieval system: Experiments in automatic document processing* (pp. 337–354). Englewood Cliffs, NJ: Prentice Hall, 337–354.

Kankanhalli, A., Tanudidjaja, F., Sutanto, J., & Tan (Bernard), B. Y. (2003). The role of IT in successful knowledge management initiatives. *Communications of ACM*, *46*(9), 69–73.

Klir, G. J., & Yuan, B. (1995). *Fuzzy sets fuzzy logic theory and applications.*. Englewood Cliffs, NJ: Prentice Hall.

Koh, J., & Kim, Y. G. (2004). Knowledge sharing in virtual communities: an e-business perspective. *Expert Systems with Applications*, *26*, 155–166.

Kwan, M. M., & Balasubramanian, P. (2003). KnowledgeScope: Managing knowledge in context. *Decision Support Systems*, *35*, 467–486.

Lee, E. S., & Li, R. L. (1988). Comparison of fuzzy number based on the probability measure of fuzzy events. *Computer and Mathematics with Applications*, *15*, 887–896.

Liebowitz, J. (1999). *The knowledge management handbook*. Boca Raton, FL: CRC Press.

McDonald, D. W., & Ackerman, M. S. (2000). Expertise recommender: A flexible recommendation system and architecture. *Proceeding of the ACM 2000 Conference on Computer Supported Cooperative Work, Philadelphia, PA* , 231–240.

Middleton, S. E., Shadbolt, N. R., & De Roure, D. C. (2004). Ontological user profiling in recommender system. *ACM Transactions on Information Systems*, *22*(1), 54–88.

Pazzani, M., & Billsus, D. (1997). Learning and revising user profiles: The identification of interesting Web sites. *Machine Learning*, *27*, 313–331.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, *14*(3), 130–137.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. In G. Salton (Ed.), *The SMART retrieval system: Experiments in automatic document processing* (pp. 313–323). Englewood Cliffs, NJ: Prentice Hall, 313–323.

Salton, G., & Buckley, C. (1988). Term weighting approaches in automatic text retrieval. *Information Processing and Management*, *24*(5), 513–523.

Salton, G., & Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, *41*(4), 288–297.

Shen, M., & Liu, D.-R. (2004). Discovering role-relevant process-views for disseminating process knowledge. *Expert Systems with Applications*, *26*(3), 301–310.

Wolverton, M. (1999). Task-based information filtering. *ACM Computing Surveys*, *31*(2), 10.

Wu, I.-C., & Liu, D.-R. (2003). A collaborative relevance feedback approach to task-driven recommendation. *Proceedings of 7th Pacific Asia Conference on Information Systems, Adelaide, South Australia* , 52–67.

Zack, M. H. (1999). Managing codified knowledge. *Sloan Management Review*, *40*(4), 45–58.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353.

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. Parts 1, 2, and 3. *Information Sciences*, *8*(2), 199–249 (see also 8(3), 301–357; 9(1), 43–80).