行政院國家科學委員會專題研究計畫　成果報告

以工作相關知識評估機制建構工作為基礎之知識推薦系統之研究

計畫類別：個別型計畫
計畫編號：NSC93-2416-H-009-011-
執行期間：93年08月01日至94年07月31日
執行單位：國立交通大學資訊管理研究所

計畫主持人：劉敦仁

計畫參與人員：吳怡瑾、楊昆學、陳韋孝

報告類型：精簡報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 94年10月7日

行政院國家科學委員會專題研究計畫成果報告

以工作相關知識評估機制建構工作為基礎之
知識推薦系統之研究

Research on Deploying Task-based Knowledge Support Systems by Task-Relevant Knowledge Assessment

計劃編號：NSC 93-2416-H-009-011
執行期間：93 年 8 月 1 日至 94 年 7 月 31 日
主持人： 劉敦仁 交通大學 資訊管理研究所
計劃參與人員：吳怡瑾、楊昆學、陳韋孝

摘要

如何提供知識工作者工作相關之知識，為建構知識管理系統之重要議題。本研究主要為探討建構以工作為基礎之知識支援系統相關研究議題。主要成果包括：(1)本研究提出系統化的工作相關知識評估機制，透過工作者間之協同合作以支援其資訊需求，並整合工作相關知識評估機制於知識支援系統中，以協助組織人員透過工作特徵檔擷取工作所需的知識；(2)提出適性化的工作特徵模式，藉由工作相關回饋機制修正工作特徵檔，以描述知識工作者之動態性工作資訊需求；(3)提出模糊分析法，依據工作者特徵檔分析知識工作者資訊需求之相似性，並建立工作社群網路；(4)實作以工作為基礎之知識支援系統，建構協同合作之工作環境，以提供有效的工作相關知識遞送與分享。

關鍵詞：知識支援；工作相關知識評估；相關回饋；工作特徵檔；知識遞送與分享

Abstract

A pertinent issue in deploying knowledge management systems is providing task-relevant information to fulfill the information needs of knowledge workers during task execution. This project mainly investigates the issues related to deploying task-based knowledge support systems. The research results include the following. (1) A task-relevance assessment mechanism is proposed to support organizational workers' task-needs (profiles) by collaborations among workers. The mechanism is integrated into the system to deliver task-relevant knowledge to workers; (2) An adaptive task-based profiling approach is proposed to model workers' dynamic task-needs based on relevance feedback; (3) A fuzzy analytical method is proposed to identify task-based peer-groups with similar task-needs based on workers' profiles; (4) A knowledge support system is developed to provide a collaborative task-based workplace facilitating knowledge retrieval and sharing among peer-groups.

Keywords: knowledge support; task-relevance assessment; task profile; knowledge delivery; knowledge sharing

1. Background and research objective

Deploying knowledge management systems (KMSs) is an important strategy for organizations to gain sustainable advantages. A pertinent issue in deploying KMS is providing task-relevant information (codified knowledge) to fulfill the information needs of knowledge workers [1][7][8][9]. Generally, KMS employs Information Technologies (ITs), such as document management and workflow management to facilitate the access, reuse and sharing of knowledge assets within and across organizations [5][13][17][18].

Contemporary ITs focus on explicit and tacit dimensions in knowledge management activities [13][18]. The former is achieved by a codified approach, whereas the latter is putting emphasis on dialoging via social networks to facilitate knowledge sharing. Notably, empirical findings indicate that codifying intellectual content into a knowledge repository makes workers highly exploit existing organizational resources [10][23]. Accordingly, knowledge (information) retrieval is considered a core component to retrieve codified knowledge in KMS.

Recently, the information retrieval (IR) technique coupled with workflow management systems (WfMS) was employed to support proactive delivery of task-specific knowledge according to the context of tasks within a process [1][8]. The KnowMore system maintains task specifications (profiles) to specify the process-context of tasks and associated knowledge items [1]. Context-aware delivery of task-specific knowledge thus can 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, the above works focus on specifying the process-context of tasks to support context-aware or process-aware knowledge retrieval, rather than on a systematic approach to construct task profiles. Also, the adaptation of profiles to track workers' dynamic information needs is not addressed.

Moreover, KMSs rely on an effective approach to construct a community of practice to promote knowledge sharing. OntoShare, an ontology-based KMS, models the interests of users and provides automatic knowledge sharing in communities of practice with the aid of profiles [6]. 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.

The research objectives of this work are the following. (1) Investigating the characteristics of knowledge management activities in organizations, and further deploying a task-based knowledge support system to deliver task relevant knowledge; (2) Proposing a task-relevance assessment approach to analyzing the relevance of tasks and codified knowledge; (3) Proposing an adaptive task-based profiling approach to model workers' dynamic task-needs (profiles); (4) Delivering task-relevant knowledge and promoting knowledge sharing among peer-groups based on task profiles; (5) Developing a *K*-Support system in a research institute to evaluate the effectiveness of the proposed system.

2. Research result

This project develops a task-based knowledge support (*K*-Support) system to acquire, organize, and disseminate an organization's task-relevant codified knowledge from the aspect of business task.

A task-oriented repository is constructed with support from domain ontology (category schema) to provide workers meaningful access to organization intelligent content. A collaborative task-relevance assessment approach is proposed to identify the worker's tasks-needs via the collaboration of knowledge workers to analyze the relevance of tasks and codified knowledge. The task profile is constructed based on the assessment result with the aid of relevance feedback (RF) technique. The task profile describes the key features of a task is a kernel for discovering and disseminating task-relevant knowledge. Furthermore, an adaptive task-based profiling approach is proposed to model workers' dynamic information needs via considering feedback behaviors. An analytical method is proposed to identify task-based peer-groups based on workers' profiles, namely task-needs. With the aid of task-based profiles and peer-groups, the *K*-Support system effectively provides task-relevant knowledge and knowledge sharing among task-based peer-groups.

3. Related work

3.1. Task-based Knowledge retrieval

The repository of structured, explicit knowledge, especially document form, is a codified strategy to manage knowledge [5][11]. However, with the growing amount of information in organizational memories, knowledge retrieval is considered a core component to access knowledge items in knowledge repository [7]. The technique of Information Filtering (IF) with a profiling approach to model users' information needs is an effective approach to proactively delivering relevant information to users. The profiling approach has been addressed by some KMSs to enhance knowledge retrieval and further promote knowledge sharing among project-based or interesting groups [1][2][6].

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 [1][8]. Furthermore, a process meta-model specifying the knowledge-in-context is integrated with workflow systems to capture and retrieve knowledge within a process context [16]. 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.

3.2. Knowledge sharing in community of practices

Koh and Kim [15] investigated knowledge sharing in virtual communities from an e-business perspective. For complex and knowledge-intensive tasks, the collaboration among knowledge workers may arise around common goals, problems and interests. The ultimate goal of KM is to enable innovative activities by promoting collaboration or communication among knowledge workers in organizations [9][22]. 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 [2][4][6][15].

4. System framework of task-based *K*-Support

This section presents the system architecture of the proposed task-based knowledge support system.

4.1. Overview of K-Support

The framework comprises three main modules, namely task-oriented repository, profile handler and task-oriented retrieval router.

Task-oriented repository: The task-oriented repository is designed for organizing and managing task relevant information. Such a repository stores codified knowledge corresponding to task execution, and contains three main databases, including the *document-indexing database, task corpus, and task categorization database*. Task-oriented repository is constructed with support from domain ontology to effectively organize codified knowledge.

Profile handler: The *profile handler* provides mechanisms such as profile creation, adjustment, integration and profile adaptation to conduct profile management. Two kinds of profile, task profile and work profile are maintained to model workers' information needs. 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 the task–interests of a worker. Workers' information needs may change. The *profile handler* uses an adaptive task-based profiling approach to capture workers' dynamic behaviors, and further adjust workers' profiles. The peer-group analyzer employs a fuzzy analytical method described in Section 6.2 to identify peer-groups with similar task-interests based on work profiles.

Task-oriented retrieval router: The task-oriented retrieval 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, that is derived based on his/her profiles 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.

4.2. Domain ontology

The domain ontology is structured into four levels, including categories, fields, tasks and knowledge items. Categories representing the main subjects of organizations are pre-defined to organize tasks and codified knowledge items. Tasks with similar subjects are grouped into fields.

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 [3]. Task corpus describing the key profile of task can be constructed by extracting knowledge from textual data gathered during task execution. 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. Tasks with similar subjects are grouped into fields.

4.3. System architecture

Figure 1 depicts the system architecture comprising four implementation layers.

Knowledge resource collection layer: 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 sharing task-relevant knowledge among workers.

Knowledge acquisition layer: This layer extracts explicit (codified) knowledge and tacit (human resource) knowledge within the organization. The *data-processing module* handles textual data represented in different formats using the information extraction engine and text pre-processing engine. The *task-processing module* comprises three offline batch-processing units, including log-parsing, task corpus selection, and task categorization. It further employs information retrieval, text mining, and database techniques to process and organize task-relevant information.

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 *profile modeling module* provides mechanisms to conduct profile creation, adaptation and integration.

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. An executor may use the *personal information editor* to organize his/her own knowledge. In addition, an executor can enter the *task-based workspace* to browse, access, and organize task-relevant knowledge. This system also identifies peer-groups with similar task-interests to promote knowledge sharing.

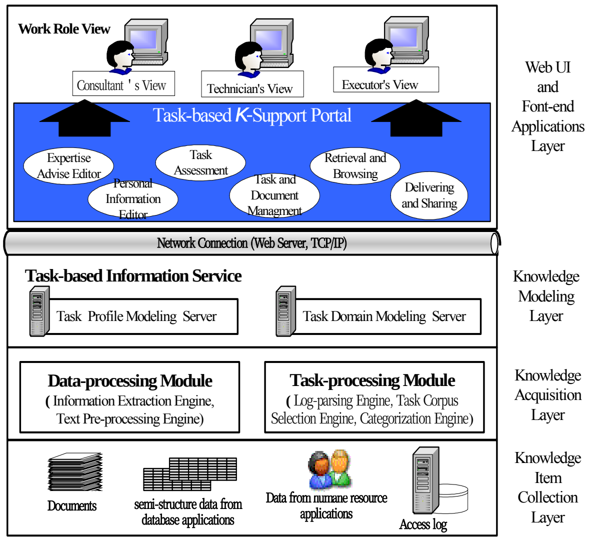

Figure 1. System architecture

5. Collaborative task-relevance assessment

This section introduces the procedure of conducting two-phase relevance-assessment by fuzzy linguistic approach for generating task profiles.

5.1. Collaborative task-relevance assessment process

Generating a task profile to support the proactive delivery of task-relevant knowledge is demanded. In our proposed system, the worker conducts category assessment (phase-1 assessment) to derive the relevance degrees of the executing task (target task at hand) to categories. And then, the knowledge worker can conduct further task assessment (phase-2 assessment) without reviewing all existing tasks. Note that the assessment is obtained by using the fuzzy linguistic approach, which is a technique for approximating human perception, and provides easier assess to qualitative problems [25].

5.2. Two-phase relevance assessment

The proposed mechanism generates the task profile based on the task corpora of existing tasks and their relevance to the executing task evaluated by knowledge workers. The details are given below.

5.2.1. Phase 1: Identifying referring tasks based on category assessment

Phase 1 of the assessment determines the relevance degrees of the executing task to categories. The referring tasks are then identified by calculating the similarity measures based on the relevance degrees of tasks to categories. Four steps are preceded in this phase, which are listed in the following.

Step1. Determine the semantic term set and corresponding fuzzy number: For modeling the workers' perceptions on *Relevance*, the system defines six linguistic terms from "*very low*" to "*perfect*" to represent different relevance degrees. *E(Relevance)* = { $\check{r}_0$ = very low, $\check{r}_1$ = low, $\check{r}_2$ = normal, $\check{r}_3$ = high, $\check{r}_4$ = very high, $\check{r}_5$ = perfect}. where $m(\check{r}_i) < m(\check{r}_j)$, for $i < j$, and all $m(\check{r}_j)$ are distributed in [0,1]. Each worker has his/her own perception of the approximate value (fuzzy scale) for each linguistic term.

Step 2. Assess the relevance of task to categories: Workers with the executing-task at hand need to rate the relevance of executing-tasks to each category by linguistic terms. Each linguistic term is transformed into the fuzzy number. Furthermore, various methods can be used to defuzzify fuzzy numbers, and this work adopts the center of area (COA) method [17] to calculate fuzzy numbers, owing to its simplicity and practicability.

Step 3. Aggregate the relevance ratings of evaluators: Evaluators' crisp ratings obtained from collaborative assessment are aggregated in this step. The aggregated relevance of the executing task to categories is expressed as a vector of relevance degree to each category. *Let* $A_{ej}(c_i)$ denote the crisp rating of evaluator e_j regarding the relevance of the executing task t_e to category c_i. Moreover, let w_{ej} denote the associated weight representing the relative importance (weight) of the rating of evaluator e_j. The aggregated relevance of the executing task to category c_i, $A_E(c_i)$, is derived as $\sum_j w_{ej}A_{ej}(c_i)$. Accordingly, the relevance degree of task t_e to categories can be modeled as,

$$\vec{t}_e^{\,C} = < A_E(c_1), A_E(c_2), \ldots, A_E(c_m) >$$

Step 4. Select referring tasks: This step identifies a subset of existing tasks as referring tasks. A similarity (cosine) measure is adopted to calculate the similarity between the executing task and an existing task based on their relevance degrees to categories. Based on the similarity measures, the top-N similar tasks are chosen as the positive (relevant) referring tasks, whereas the last-M non-similar tasks are chosen as the negative (irrelevant) referring tasks. The referring tasks are used for further task-relevance assessment of phase 2.

5.2.2. Phase2: Assessing the relevance of referring tasks

This phase conducts an assessment to determine the relevance of the referring tasks to the executing task. The evaluators assess the degree of relevance between the executing task and referring tasks without reviewing all tasks. The task assessment procedure resembles the procedure of category assessment. The evaluators conduct relevance assessment to determine the relevance degree of each referring task to the executing task. They use linguistic terms to rate the relevance of each referring task to the executing task. The aggregated relevance rating of a referring task is derived by computing the weighted average of evaluators' crisp ratings on the relevance of the referring task to the executing task. The relevance degrees of referring tasks to the executing task are then used to construct the task profile of the executing task, as described in Section 5.3.

5.3. Task profile generation

The task profile is derived based on the assessment result, is further refined based on the relevance feedback (RF) techniques [3][20].

This work modifies the standard Rocchio and Ide_Dec_Hi methods by considering the relevance degrees of referring tasks obtained from fuzzy linguistic assessment. The modification considers the relative importance of relevant (positive) and irrelevant (negative) tasks from the perspective of users. The feature vectors of referring tasks are multiplied with their relevance degrees to reflect their relative contributions in the refinement of the task profile, as expressed below.

$$\vec{S}_e = \alpha \vec{S}_{initial} + \beta \sum_{\forall t_j \in T_r} (w_{t_j})\vec{t}_j - \gamma \sum_{\forall t_j \in T_n} (1 - w_{t_j})\vec{t}_j \quad \textbf{(1)}$$

where $\vec{S}_{initial}$ represents the initial profile derived from analyzing the collected relevant documents e.g., business plan, for the executing task, if available. T_r denotes the set of relevant tasks selected from positive referring tasks according to collaborative assessment of experts and workers. T_n represents the set of the last-M irrelevant tasks that are selected by the system automatically. Furthermore, $\vec{t}_j$ is the task corpus of task t_j with an associated weight w_{tj} representing the relevance degree of t_j to the executing task. w_{tj} is set to $A_E(t_j)$, which is the aggregated relevance rating of task t_j to the executing task.

The task profile can be expressed as a feature vector of weighted terms, $\vec{S}_e = <w(k_1, t_e), w(k_2, t_e), \ldots, w(k_n, t_e)>$, where $w(k_i, t_e)$ is the weight of a term k_i in representing the main subjects of t_e; n denotes the number of discriminating terms. Meanwhile, $\vec{S}_e$ is used to retrieve relevant codified knowledge.

6. Disseminating and sharing task-relevant knowledge

An *adaptive task-based profiling approach* and a *fuzzy analytical method* are proposed to track workers' dynamic task-needs and identify workers' task-based peer-groups according to the changes of profiles.

6.1. Adaptive task-based profiling

This section describes the proposed profiling approach.

6.1.1. Profile structuring

Task profile and work profile are used to represent a worker's current information needs on the target task at hand. 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. 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.

6.1.2. Profile adaptation based on feedback analysis

Accordingly, a temporal file $\vec{T}emp_{u,p}$ is generated to represent the worker's task-needs during a specific time period. The $\vec{T}emp_{u,p}$ is derived from the feature vectors of those documents accessed by worker *u* during time period *p*, , as shown in the Eq. 2. The details are presented in our work [19].

$$\vec{T}emp_{u,p} = \frac{1}{\left|D_{u,p}^{exp}\right|} \sum_{\forall dj \in D_{u,p}^{exp}} (A_u(d_j) \times \vec{d}_j) + \frac{1}{\left|D_{u,p}^{imp}\right|} \sum_{\forall dj \in D_{u,p}^{imp}} (CV(\vec{H})^u \times \vec{d}_j) \quad \textbf{(2)}$$

The associated degree of relevance can be adjusted across time by tracking the worker's explicit or implicit feedback on recommended items. For example, if a worker is interested in an incoming document d_i, the similarity between d_i and t_j can be determined by similarity measure. If $sim(d_i,t_j)$ is above (below) a threshold, the system will increase (decrease) the associated weight of task t_j. The adjustment may change the information structure of a worker's work profile.

Furthermore, the task profile can be adjusted based on the modified relevance feedback technique, as listed below. The details are presented in [19].

$$\vec{S}_{p+1} = \alpha \vec{S}_p + \beta \vec{O} - \gamma (1 - w_{p+1}(t_j)) \sum_{\forall t_j \in T_r} \vec{t}_j \quad \textbf{(3)}$$

where $\vec{O} = \lambda \sum_{\forall t_j \in T_r} w_{p+1}(t_j)\, \vec{t}_j + (1-\lambda)\vec{T}emp_{u,p}$

6.1.3. Task-based knowledge retrieval

The adjustment of work profile across time will lead the *task profile modeling serve*r to refine the task profile and further enhance the knowledge retrieval capability. $\vec{S}_{t+1}$ 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).

6.2. Fuzzy peer-group analytical model

This section describes the fuzzy analytical method for identifying task-based peer groups.

6.2.1. Establishing 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. The task-level/field-level relevance degrees can be used to derive the task-level/field-level similarity between workers. The relevance degree in field level is derived from the task-level, namely, the value of $w_p(field_i)$ is set to the aggregation value of task-level relevance degrees, i.e, aggregation (summation) of $w_p(t_j)$ for all *task* t_j belongs to $field_i$. The cosine measure is employed to calculate the similarity among workers based on the field-level user-feedback. Finally, a reflective and symmetric similarity relationship matrix is derived.

6.2.2. Identifying task-based peer-groups

In this section, we introduced how the task-based peer-group can be automatically identified by α-cuts applied in the fuzzy equivalence matrix $\boldsymbol{S_T}$.

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 $\tilde{\varsigma}(E_i, E_j) \in [0,1]$. The method of transitive max-min closure [14] is adopted to derive a reflective, symmetric, and transitive matrix, which is a fuzzy equivalence matrix. The max-min composition and max operator for set union are used to derive the transitive max-min closure S_T.

Step2. 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 $\boldsymbol{U}$.

6.3 Task-based knowledge sharing

The *K*-Support system uses work profiles to enable workers find possible task-based peer-groups. The system can automatically adjust workers' peer-groups by analyzing the work profiles. The system not only provides a knowledge-support platform for gathering and exchanging task-relevant knowledge among workers but also present the peer-group member's personalized ontology for knowledge sharing.

7. *K*-Support

The task-based *K*-Support portal is a Web-based application, allowing workers to retrieve, organize and share task-relevant knowledge. *K-Assessment* application assists an executor to conduct task assessment to create his/her own task profile. *K-Delivery* application delivers task relevant knowledge proactively to support task execution. *K-Sharing* application stimulates knowledge sharing by locating possible task-based peer-groups.

7.1. K-Assessment: Identifying task-relevant knowledge

The worker can conduct task-assessment to generate his own task profile. If he selects the "assessment" item, the system will guide him to conduct two-phase task-relevance assessment. The results of assessment are submitted to the system's task profile modeling server to compute the initial task profile.

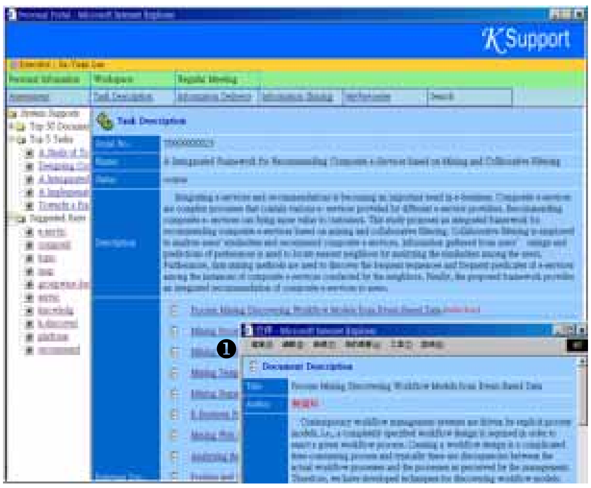

Figure 2. Knowledge delivery

7.2. K-Delivery: Delivering codified knowledge

The system can proactively deliver task-relevant information based on the worker's task profiles. Figure 2 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 Figure 2. Meanwhile, the worker can view the description of any task-relevant document, as denoted in circle 1.

7.3. K-Sharing: Knowledge support from peer-groups

The system expands the personalized ontology of a worker with the peer-group member's personalized ontology for knowledge sharing. The personalized ontology is derived from the work profile to record tasks or fields that are relevant to the target task. The left frame of Figure 3 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.

Figure 3. Knowledge sharing

8. Discussions

A task-based *K*-Support system is developed to acquire, model and disseminate codified knowledge among workers in task-based environments. The system facilitates 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. 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.

9. Project evaluation

Providing task-based knowledge supports is crucial for enterprises in effectively managing business knowledge and achieving competitive advantages. Our work not only contributes to the practice of knowledge management but also contributes to further research on task-based knowledge support. In summary, we have proposed novel idea, investigated new technology and developed a task-based knowledge support system. The research result has also been published in an international journal.

References

[1] A. Abecker, A. Bernardi, K. Hinkelmann, O. Kühn, and M. Sintek (2000), "Context-aware, proactive delivery of task-specific knowledge: The KnowMore project," *International Journal on Information Systems Frontiers (ISF)*, 2 (3/4), pp. 139-162.

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

[3] R. Baeza-Yates, and B. Ribeiro-Neto (1999), *Modern Information Retrieval*, Addison-Wesley.

[4] J.S. Brown, and P. Duguid, (1991), "Organization learning and communities of practice," *Organization Science*, 2(1), pp. 40-57.

[5] T.H. Davenport and L. Prusak (1998), *Working knowledge: How organizations manages what they know*, Boston MA: Harvard Business School Press.

[6] N.J. Davies, A. Duke, and A. Stonkus (2003), OntoShare: Evolving ontologies in a knowledge sharing system, In Davies, NJ, Fensel, D. van Harmelen, F. (Eds.): *Towards the semantic Web – Ontology-based knowledge management.* Wiley, London UK, pp. 161-176.

[7] K.D. Fenstermacher, and C. Marlow (1999), "Supporting consultants with task-specific information retrieval," *Proceedings of The American Association of Artificial Intelligence*, Orlando Florida, AAAI Press.

[8] Kurt D. Fenstermacher (2002), "Process-aware knowledge retrieval," *Proceedings of the 35th Hawaii International Conference on System Sciences*, Big Island, Hawaii, USA, pp. 209-217.

[9] G. Fischer, and J. Ostwald (2001), "Knowledge management: Problems, promises, realities, and challenges," *IEEE Intelligent Systems* 16(1) pp. 60-73.

[10] P.H. Gray (2001a), "Problem-solving perspective on knowledge management practices," *Decision Support Systems* 31(1), pp. 87-102, May.

[11] P.H. Gray (2001b), "The impact of knowledge repositories on power and control in the workplace," *Information Technology & People*, 14(4), pp.368-384.

[12] E. Ide, (1971), "New experiments in relevance feedback" in: G. Salton (Eds), *The SMART retrieval system: Experiments in automatic document processing*, Englewood Cliffs, NJ: Prentice Hall, pp. 337-354.

[13] A. Kankanhalli, F. Tanudidjaja, J. Sutanto, and C.Y. Tan (Bernard) (2003), "The role of IT in successful knowledge management initiatives," Communications of ACM 46(9), pp.69-73.

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

[15] J. Koh, and Y.G. Kim (2004), "Knowledge sharing in virtual communities: An e-Business perspective," *Expert Systems with Applications*, 26, pp.155-166.

[16] M.M. Kwan and P. Balasubramanian (2003) "KnowledgeScope: Managing knowledge in context," *Decision Support Systems*, *35*, pp.467-486.

[17] E.S. Lee, and R.L. Li (1988), "Comparison of fuzzy number based on the probability measure of fuzzy events," *Computer and Mathematics with Applications*, *15*, pp. 887-896.

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

[19] D.-R. Liu, I.-C. Wu, and K.-S. Yang (2005), "Task-based *K*-Support system: disseminating and sharing task-relevant knowledge," *Expert Systems with Applications*, 29(2), pp.408-423.

[20] J.J. Rocchio, (1971). Relevance feedback in information retrieval. in: G. Salton (Eds), *The SMART retrieval system: Experiments in automatic document processing*, Englewood Cliffs, NJ: Prentice Hall, pp. 313-323.

[21] G. Salton, and C. Buckley (1988), "Term weighting approaches in automatic text retrieval," *Information Processing & Management*, *24*(5), pp.513-523.

[22] M. Wolverton (1999). "Task-based information filtering," *ACM Computing Surveys*, 31(2), Article No. 10.

[23] M.H. Zack (1999), "Managing codified knowledge," Sloan Management Review, 40(4), pp. 45-58.

[24] L.A. Zadeh (1965), *Fuzzy sets*, Information and Control 8(3) pp.338-353.

[25] L.A. Zadeh (1975), "The concept of a linguistic variable and its application to approximate reasoning," parts 1, 2, and 3. *Information Sciences*, 8(2) pp.199-249; 8(3) pp.301-357; 9(1) pp.43-80.