Document Recommendations Based on Knowledge Flows: A Hybrid of Personalized and Group-Based Approaches

Duen-Ren Liu
Institute of Information Management, National Chiao Tung University, Hsinchu, Taiwan.
E-mail: dliu@iim.nctu.edu.tw

Chin-Hui Lai
Department of Information Management, Chung Yuan Christian University, Taoyuan County, Taiwan.
E-mail: chlai@cycu.edu.tw

Ya-Ting Chen
Institute of Information Management, National Chiao Tung University, Hsinchu, Taiwan.
E-mail: yavicky13@gmail.com

Recommender systems can mitigate the information overload problem and help workers retrieve knowledge based on their preferences. In a knowledge-intensive environment, knowledge workers need to access task-related codified knowledge (documents) to perform tasks. A worker's document referencing behavior can be modeled as a knowledge flow (KF) to represent the evolution of his or her information needs over time. Document recommendation methods can proactively support knowledge workers in the performance of tasks by recommending appropriate documents to meet their information needs. However, most traditional recommendation methods do not consider workers' KFs or the information needs of the majority of a group of workers with similar KFs. A group's needs may partially reflect the needs of an individual worker that cannot be inferred from his or her past referencing behavior. In other words, the group's knowledge complements that of the individual worker. Thus, we leverage the group perspective to complement the personal perspective by using hybrid approaches, which combine the KF-based group recommendation method (KFGR) with traditional personalized-recommendation methods. The proposed hybrid methods achieve a trade-off between the group-based and personalized methods by exploiting the strengths of both. The results of our experiment show that the proposed methods can enhance the quality of recommendations made by traditional methods.

Introduction

Because of the rapid development of information technologies in recent years, it is now relatively easy to access knowledge resources. In knowledge-intensive environments, knowledge workers need to access task-related codified knowledge (documents) to perform tasks. However, the huge volumes of documents that exist in various knowledge domains often lead to information overload. Thus, there is a need for document recommendation methods that support knowledge workers as they perform tasks by recommending appropriate documents to suit their information needs (i.e., task needs).

Workers may have various information needs (Cole, 2011; Wu, Liu, & Chang, 2009) when executing tasks. Because each worker's information needs may change over time, we model a worker's document-referencing behavior for a specific task as a *knowledge flow* (KF) to represent the evolution of his or her information needs (Lai & Liu, 2009). From the personal perspective, a worker's KF is derived from his or her past referencing behavior to represent his or her personal needs. The topics and documents included in the KF are related to the worker's specific personal needs. Usually, workers like to refer to other workers' codified knowledge to complement their own knowledge as they perform tasks. Codified knowledge derived from a group of workers with similar referencing behavior (KFs) also is important because it facilitates knowledge reuse and sharing in an organization. From the group perspective, the information needs of the majority of the group's members are more important than those of individual members. A group's needs may partially reflect the needs of an individual worker that cannot be inferred from his or her past referencing behavior. In other words, the group's knowledge complements that of the individual worker.

Recommender systems can alleviate the information overload problem and help workers identify and retrieve

Received September 1, 2011; revised March 1, 2012; accepted March 16, 2012

 DOI: 10.1002/asi.22705

needed documents based on their preferences or information needs (Adomavicius & Tuzhilin, 2005; Furner, 2002; Liu, Lai, & Huang, 2008). There are two widely used filtering methods: collaborative filtering (CF) (Konstan et al., 1997; Sarwar, Karypis, Konstan, & Reidl, 2001) and content-based filtering (CBF) (Balabanovic & Shoham, 1997; Mooney & Roy, 2000; Pazzani & Billsus, 2007). These methods identify items of interest (e.g., documents) that are likely to match workers' previous preferences. However, both CF and CBF make personalized recommendations based on the preferences of the individual worker. Moreover, the referencing behavior of knowledge workers may vary over time, but most recommendation methods do not consider workers' KFs. In contrast, our recommendation methods consider such flows.

Several group-based recommendation methods have been proposed (Jameson, 2004; Lorenzi, dos Santos, Ferreira, & Bazzan, 2008; McCarthy & Anagnost, 1998; O'Connor, Cosley, Konstan, & Riedl, 2001) because traditional recommendation methods focus on personalized recommendations and have certain limitations. For example, if a group of people want to choose a restaurant for dinner or decide which movie to watch, traditional methods are not appropriate since they only consider the preferences of one group member. Group recommendation solves the problem by merging members' preferences to generate a group profile (J.K. Kim, Kim, Oh, & Ryu, 2010; McCarthy & Anagnost, 1998) or by combining the recommendations of all members of the group to form a group recommendation (O'Connor et al., 2001). Existing group-recommendation schemes satisfy the information needs of most workers in a group, but they often neglect individual workers' preferences. Traditional group-based recommendation methods can be used to generate a group profile by simply merging all of the members' profiles derived from the documents they referenced in their KFs. However, from the perspective of KFs, documents and topics referenced in different time periods should have different degrees of importance. That is, more weight should be given to documents/topics referenced in the recent past because that referencing behavior is more likely to reflect the workers' current information needs. Traditional group-based recommendation methods do not consider recommendations in the context of a KF environment.

In this work, we propose hybrid recommendation methods that combine a KF-based group recommendation (KFGR) method with traditional recommendation methods, including CF and CBF. The proposed recommendation methods consist of three phases: (a) compiling individual KFs, (b) grouping knowledge workers and generating group profiles, and (c) recommending documents to workers. The first phase involves three steps—document profiling, document clustering, and KF generation—for creating users' KFs to reflect their preferences or information needs over time. Information retrieval and document-clustering techniques are used to extract users' KFs from their document access work logs. In the second phase, we cluster knowledge workers with similar KFs into groups and then derive group profiles from the KFs of the group members. In the last phase, the proposed recommendation methods (which consider both the group and individual perspectives) are used to recommend suitable documents to the knowledge workers.

Most traditional recommendation methods focus on the personal perspective rather than on the group perspective; however, the group's information needs may be important because they partially reflect an individual's needs. In other words, the group's knowledge may complement that of the individual worker. Therefore, we take the group perspective into consideration to offset the drawback of the personal perspective. The proposed KFGR method is a novel recommendation method that takes workers' KFs and their personal preferences into account to recommend documents for a group of workers with similar KFs. The drawback of the group perspective is that it may not satisfy the information needs of some individuals since it focuses on the needs of the majority of group members. To resolve the problem, we combine the KFGR method with traditional recommendation methods to enhance the quality of recommendations. The proposed hybrid method achieves a trade-off between the group-based and personalized methods by combining the merits of both methods. The results of the experiment show that the proposed model can improve the quality of recommendations provided by traditional recommendation methods.

The remainder of the article is organized as thus: The following section contains a review of related works. Next, we describe the KF model and the proposed hybrid recommendation method. Details of the experiment results and a discussion of their implications then are presented. We conclude with suggestions for future research.

Related Work

In this section, we consider existing works on KFs, information retrieval, task-based knowledge support, and clustering methods. As mentioned earlier, the two most popular approaches for recommending items are CBF (Balabanovic & Shoham, 1997; Basu, Hirsh, & Cohen, 1998; Lang, 1995; Mooney & Roy, 2000; Pazzani & Billsus, 2007) and CF (Glance, Arregui, & Dardenne, 1998; Konstan et al., 1997; Rucker & Polanco, 1997; Shardanand & Maes, 1995) (discussed later). In addition, we also will review group-based recommender systems.

Knowledge Flows

Knowledge flows (KFs) among people and processes facilitate knowledge sharing and reuse. The concept of KFs has been applied in various domains such as scientific research, communities of practice, teamwork environments, industry, and organizations (S. Kim, Hwang, & Suh, 2003; Zhuge, 2006a). In a scientific research domain, scholarly articles represent the major medium for disseminating

DOI: 10.1002/asi

knowledge among scientists to generate new ideas (Zhuge, 2006a). A citation implies that there is KF between the citing article and the cited article. Such citations form a KF network that enables knowledge to flow between different scientific projects and thereby promote interdisciplinary research and scientific development.

Knowledge management enhances the effectiveness of teamwork by accumulating and disseminating knowledge among team members to facilitate peer-to-peer knowledge sharing (Zhuge, 2002). To improve the efficiency of teamwork, Zhuge (2006b) proposed a pattern-based approach that combines codification and personalization strategies to form an effective KF network while S. Kim et al. (2003) combined a KF model with a process-oriented approach to capture, store, and transfer knowledge. Workers involved in a business process need different types of knowledge to support their tasks. Zhang and Xi (2009) developed a model that combines the work flow and KF to improve the efficiency of implementing tasks. Moreover, Luo, Hu, Xu, and Yu (2008) introduced the concept of textual KFs based on the management of knowledge maps.

In an organization, knowledge workers normally have various information needs over time when performing tasks. In a previous work (Lai & Liu, 2009), we defined a KF from the perspective of a worker's information needs to represent the evolution of referencing behavior and the knowledge accumulated for a specific task; we also proposed KF-based methods for recommending task-related codified knowledge. The methods, which exploit the strengths of typical CF and KFs, enhance the quality of document recommendation by considering workers' preferences for codified knowledge and their knowledge-referencing behavior (Lai & Liu, 2009).

Information Retrieval and Clustering Methods

A knowledge worker may acquire knowledge from a large number of documents. Since the documents can reveal the worker's information needs, we need to filter the documents by using information retrieval techniques that enable us to access specific items of information (Baeza-Yates & Ribeiro-Neto, 1999; Feldman & Sanger, 2007). The vector space model (Salton & Buckley, 1988) is normally used to represent documents as vectors of index terms. The weights of the terms are measured by the *tf-idf* approach, where *tf* denotes the occurrence frequency of a particular term in the document (Salton & Buckley, 1988), and *idf* denotes the inverse document frequency of the term (Jones, 1972; Salton, Wong, & Yang, 1975). Terms with higher *tf-idf* weights are used as discriminating terms to filter out common terms. The weight of a term i in a document j, denoted by $w_{i,j}$, is expressed as follows:

$$w_{i,j} = \left(0.5 + \frac{0.5 \times tf_{i,j}}{\max tf_{i,j}}\right) \times \left(\log \frac{N}{df_{i,j}} + 1\right), \qquad (1)$$

where $tf_{i,j}$ is the frequency of term i in document j, $df_{i,j}$ denotes the number of documents that contain the specific term i, and N is the total number of documents in the set. We apply the best weighted probabilistic method with a 0.5 weight (Salton & Buckley, 1988) as our *tf* method. In addition, we apply the *idf* weighting method; that is, $\log(N/df_{i,j}) + 1$ (Jones, 1972; Salton et al., 1975) to calculate the *idf* values. We only collected a small set of documents from a few specific domains. The common terms may represent the basic subject terms of the specific domains. Thus, we use "+1" in the *idf* weighting method to avoid the term weight $w_{i,j}$ being zero.

A typical information-filtering application often uses a similarity-based approach (e.g., cosine measure) to compare those documents with the user profiles and find the documents that are relevant to their profiles (Belkin & Croft, 1992). Information filtering also can be used to locate knowledge items relevant to the task at hand. The discriminating terms of a task are usually extracted from a knowledge item/task to form a task profile, which is used to model a worker's information needs. Task-based knowledge support can provide knowledge workers with task-relevant information based on the task profile (Liu & Wu, 2008; Wu et al., 2009).

Document clustering or unsupervised document classification is used in many applications. Most methods apply preprocessing steps to the document set and represent each document as a vector of index terms. To cluster similar documents, the similarity between documents is usually measured by the cosine measure (Baeza-Yates & Ribeiro-Neto, 1999; Van RijsBergen, 1979), which computes the cosine of the angle between the documents' corresponding feature vectors. Two documents are regarded as similar if the cosine similarity value is high. The cosine similarity of two documents, X and Y, is $simcos(X,Y) = \frac{\vec{X} \cdot \vec{Y}}{\|\vec{X}\|\|\vec{Y}\|}$, where $\vec{X}$ and $\vec{Y}$ are the respective feature vectors of X and Y.

Hierarchical clustering solutions have been obtained using agglomerative and partitioning algorithms (Aggarwal, Gates, & Yu, 1999; Steinbach, Karypis, & Kumar, 2000). In agglomerative algorithms, each item is initially a single cluster, and then pairs of clusters are merged repeatedly until the whole hierarchical tree is formed. However, partitioning algorithms are less effective than are agglomerative algorithms. For this reason, many hierarchical document-clustering methods mainly focus on agglomerative methods. Hierarchical agglomerative clustering (Johnson, 1967) is a popular document-clustering method that uses various merging criteria—single-link, average-link, complete-link, and so on—in the clustering process. In this work, we use the average-link agglomerative method to cluster codified knowledge (documents) into topic domains based on the average similarity of the pairwise documents in each cluster. We also utilize the CLIQUE clustering method (Agrawal, Gehrke, Gunopulos, & Raghavan, 1998; Jain, Murty, & Flynn, 1999) to derive worker groups.

DOI: 10.1002/asi

Content-Based Filtering

Personalized news content programming (PENG; Pasi & Villa, 2005) is a flexible CBF model for estimating the relevance of incoming documents based on five properties: document similarity, coverage, novelty of contents, trust, and timeliness in the information sources. Moreover, a flexible, multicriterion-based information-filtering model (Bordogna & Pasi, 2010) is proposed, which allows users to specify and define their personal filtering criteria and aggregation strategies.

In addition, CBF has been utilized in several fields to provide users with appropriate information, such as movie recommendations (Basu et al., 1998), book recommendations (Mooney & Roy, 2000), and news items of interest (Lang, 1995). The CBF approach analyzes a user's preferences for particular features of an item to compile a personal feature profile and then predicts items that may be of interest to the user. In other words, this approach recommends items with similar features to those in the customer's profile that he or she referenced previously. For book recommendations, Mooney and Roy (2000) proposed a book recommender system called LIBRA, which uses a Bayesian learning algorithm to learn the user's profile and generate a recommendation list. Subsequently, Huang, Chung, and Chen (2004) proposed a graph model to represent user-product information for recommending books.

Collaborative Filtering

Collaborative filtering (CF) is widely used in recommender systems such as GroupLens (Konstan et al., 1997), Ringo (Shardanand & Maes, 1995), Siteseer (Rucker & Polanco, 1997), and Knowledge Pump (Glance et al., 1998). CF recommends various items, such as products, movies, and documents, based on the preferences of people who have the same or similar interests to those of the target user (Breese, Heckerman, & Kadie, 1998). The approach involves two steps: neighborhood selection and prediction. The neighborhood of a target user is selected according to his or her similarity to other users, and the score is computed by Pearson's correlation coefficient or the cosine similarity measure. Either the *k*-NN (nearest neighbor) approach or a threshold-based approach is used to choose *n* users that are most similar to the target user. We use a threshold-based approach in this article. In the prediction step, the predicted rating is calculated from the aggregated weights of the selected *n* nearest neighbors' ratings, as shown in Equation 2:

$$P_{u,j} = \bar{r}_u + \frac{\sum_{i=1}^{n} w(u,i)\left(r_{i,j} - \bar{r}_i\right)}{\sum_{i=1}^{n} |w(u,i)|}, \quad (2)$$

where $P_{u,j}$ denotes the predicted rating of item j for the target user u; $\bar{r}_u$ and $\bar{r}_i$ are the average ratings of user u and user i, respectively; $w(u, i)$ is the similarity between target user u and user i; $r_{i,j}$ is the rating of user i for item j; and n is the number of users in the neighborhood. There are several metrics, such as the Pearson correlation coefficient, cosine similarity, adjusted cosine similarity, Spearman's rank correlation coefficient, the mean squared difference, or entropy measure, used in recommender systems to determine the similarity between users. However, empirical analyses have shown that the Pearson coefficient outperforms other user-similarity measures for user-based recommendation systems (Herlocker, Konstan, Borchers, & Riedl, 1999). We therefore use the Pearson correlation metric as the similarity measure for the user-based CF (UCF) method.

Similar to the UCF method, the item-based CF (ICF) algorithm (Linden, Smith, & York, 2003; Sarwar et al., 2001) first analyzes the relationships between items (e.g., documents), rather than the relationships between users. Then the relationships are used to indirectly compute recommendations for workers by finding items that are similar to other items that the worker accessed previously. Thus, the prediction for an item j for a user u is calculated by using the weighted sum of the ratings given by the user for items similar to j, as shown in Equation 3:

$$p_{u,j} = \frac{\sum_{m=1}^{n} w(j,m) \times r_{u,m}}{\sum_{m=1}^{n} |w(j,m)|}, \quad (3)$$

where $p_{u,j}$ represents the predicted rating of item j for user u; $w(j, m)$ is the similarity between two items j and m; and $r_{u,m}$ denotes the rating of user u for item m. A number of methods can be used to determine the similarity between items, such as the cosine-based similarity, correlation-based similarity, and adjusted cosine similarity methods. Since the adjusted cosine similarity method yields the best performance (Sarwar et al., 2001), we use it as the similarity measure for the ICF method. The adjusted cosine similarity between two items i and j is given by Equation 4:

$$sim(i,j) = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_u)(r_{u,j} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_u)^2}}, \quad (4)$$

where $r_{u,i}/r_{u,j}$ is the rating of item i/j given by user u; and $\bar{r}_u$ is the average item rating given by u.

Group-Based Recommendation Methods

Typical group recommendation methods merge the preferences of all group members to form a group preference. Group recommender systems are used for music, movies, television programs, and tourist attractions.

MusicFX (McCarthy & Anagnost, 1998) selects music stations for the members of a fitness center and attempts to maximize the satisfaction of the group. PolyLens (O'Connor et al., 2001) is a movie recommender system that suggests movies for a small group of people who watch movies together. It recommends movies for the least satisfied group member and attempts to satisfy all users to some degree. Group recommender systems used in the tourism domain

DOI: 10.1002/asi

include Intrigue (Ardissono, Goy, Petrone, Segnan, & Torasso, 2003) and Travel Decision Forum (Jameson, 2004). Intrigue helps a group of users organize a trip and recommends sightseeing locations by considering the preferences and differences of a heterogeneous group of users. Travel Decision Forum helps group members to collaboratively specify their preferences and agree on arrangements for their trip.

In summary, group recommender systems can be classified as (a) systems that aggregate the profiles/preferences of various users to form a group profile/preference (Garcia, Sebastia, Onaindia, & Guzman, 2009; J.K. Kim et al., 2010; Masthoff, 2008; McCarthy & Anagnost, 1998; Shin & Woo, 2009; Yu, Zhou, Hao, & Gu, 2006) and (b) systems that merge individual recommendation lists to form a group recommendation list (O'Connor et al., 2001). Under the first approach, there is a high probability of discovering valuable recommendations that will satisfy the majority of the group's members. The second approach gives users more information when they need to make decisions, and the recommendation results are relatively easy to explain. However, it is not easy to identify unexpected items, and it is very time consuming if the group is large. Therefore, we follow the first approach and aggregate workers' topic domains based on their KFs to generate a group profile.

Hybrid Personalized and Group-Based Methods

Overview

In a knowledge-intensive environment, a high degree of knowledge sharing can have a significant effect on workers' efficiency. Each worker accumulates knowledge when he or she executes a task, and that knowledge can be shared with, and reused by, other team members with similar information needs. In this article, we propose personalized group-based recommendation methods to facilitate knowledge sharing among a group of workers. The method combines the KFGR method and personalized methods to enhance the quality of document recommendation. The rationale behind the proposed model is that a group's information needs may partially reflect an individual member's information needs that cannot be inferred from his or her past document-referencing behavior. In other words, the group's knowledge can be used to satisfy the individual member's needs. Thus, the group-based method complements the personalized method. However, the group perspective may neglect the specific information needs of an individual because it focuses on the needs of the majority of the group's members. To resolve this problem, the proposed hybrid recommendation methods exploit the strengths of the two approaches to improve the quality of recommendations. The group-based method recommends documents from the perspective of the majority's information needs while the personalized methods recommend documents according to the specific needs of an individual.

The proposed recommendation methods are implemented in three phases: (a) compiling individual KFs (codified-level KFs and topic-level KFs), (b) grouping knowledge workers and generating group profiles, and (c) recommending documents to workers.

The first phase involves three steps: document profiling, document clustering, and KF generation. To accomplish tasks, knowledge workers may need to access various documents, and those documents can reflect the workers' preferences or requirements in different periods. We align the documents in a sequence, called a *codified-level KF*. Each document in the sequence is represented by an n-dimensional vector, which comprises key terms in the document and their weights. Next, we cluster the documents into several topics based on their cosine similarity scores. To observe the evolution of information needs, we generate a *topic-level KF* as a topic sequence by mapping the documents in the codified-level KF to the corresponding clusters (topics). Details of the process will be described later.

In the second phase, we gather similar knowledge workers into groups by using a KF similarity measure derived from the alignment similarity and aggregate profile similarity measures. The KF similarity score indicates whether the referencing behavior of two workers is similar. After grouping the workers, each group's important codified knowledge can be elicited from the topics accessed by the group members. To represent each group's important knowledge, we compile a group profile by the process described earlier.

In the last phase, we use the proposed personalized group-based recommendation methods, which consider both the group and individual perspectives, to recommend suitable documents to the knowledge workers. The group-based approach derives a group-based score (preference) of a group, k, for a target document based on the topic-level KFs of the group's members. As similar documents are grouped into clusters (topics), topic-level KFs should provide a larger number of related documents to satisfy workers' task needs than should codified-level KFs. Thus, the group-based approach utilizes the topic-level KF to predict a group's ratings for documents. In this work, we propose three recommendation methods, a hybrid of KF-based group recommendation and user-based CF (KFGR-UCF), a hybrid of KF-based group recommendation and item-based CF (KFGR-ICF), and a hybrid of KF-based group recommendation and CBF (KFGR-CB). Further details are given later.

Knowledge Flow Model

A worker's KF represents the evolution of his or her information needs and preferences during a task's execution (Lai & Liu, 2009; Liu & Lai, 2011). Workers' KFs are identified by analyzing their knowledge referencing behavior based on their historical work logs, which contain information about previously executed tasks, task-related documents, and when the documents were accessed.

A KF comprises a codified level and a topic level (Lai & Liu, 2009; Liu & Lai, 2011). The knowledge in the codified-level indicates the KF between documents based on the times that they were accessed. In most situations, the

DOI: 10.1002/asi

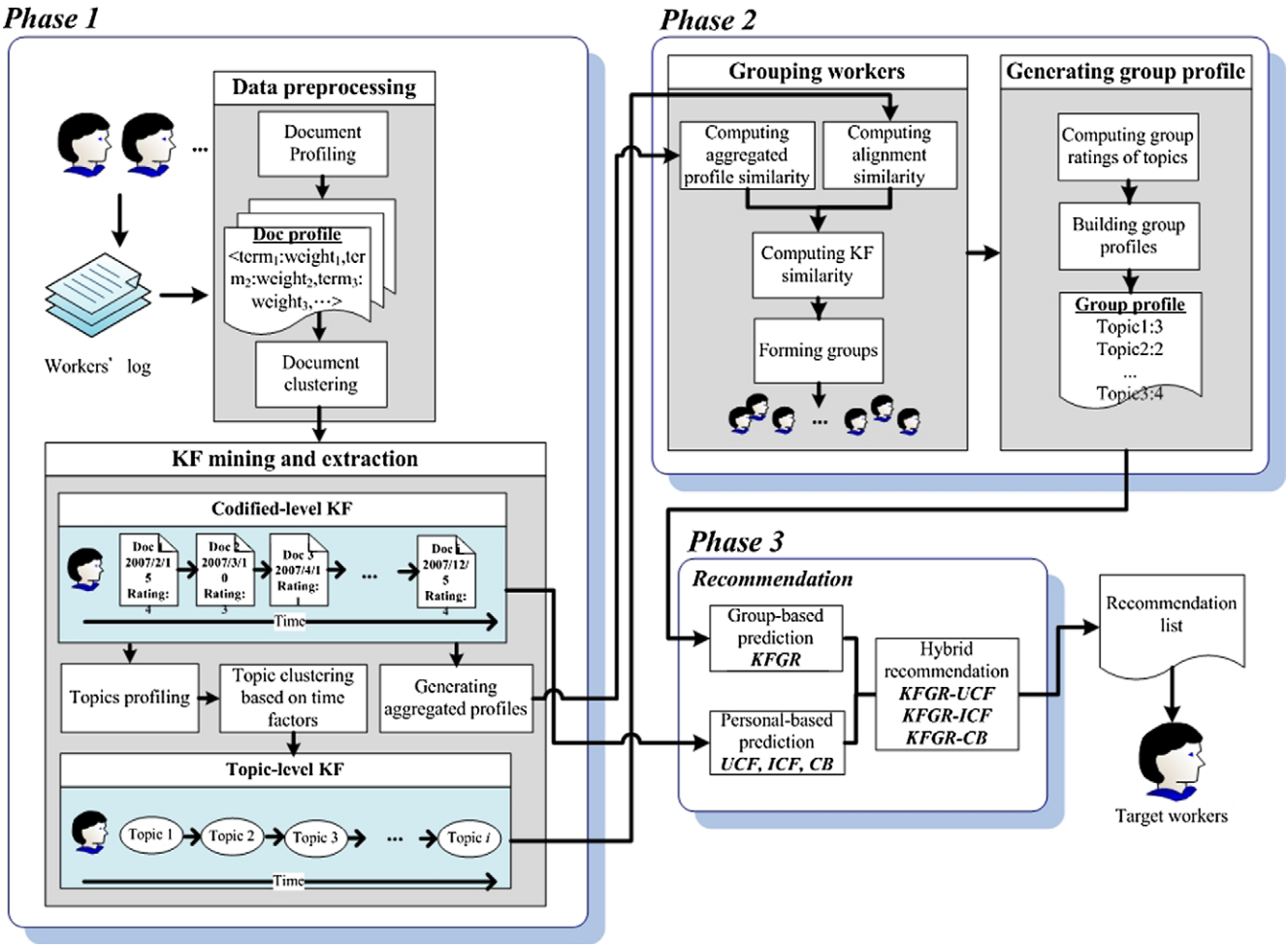

FIG. 1. Overview of the proposed recommendation methods. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

knowledge obtained from one document prompts a knowledge worker to access the next relevant document (codified knowledge). Hence, the task-related documents are sorted in order of the times that they were accessed to obtain a document sequence as the codified-level KF.

Documents with similar concepts and access times are automatically grouped together to form a topic-level abstraction of the task knowledge. Note that each topic may contain several task-related documents. The codified-level KF is abstracted to form a topic-level KF, which represents the transitions between various topics. Since the task knowledge in the topic level may flow between topics, it could prompt the worker(s) to retrieve knowledge from the next related topic. Formally, the KF is defined by Lai and Liu (2009; Liu & Lai, 2011) as follows:

Definition 1. Knowledge flow (KF): The KF of a worker, w, for a specific task is formulated as $KFlow_w = \{TKF_w, CKF_w\}$, where TKF_w is the worker's topic-level KF and CKF_w is his or her codified-level KF.

Definition 2. Codified-level KF (CKF): CKF is a sequence of documents arranged according to the times that the documents were accessed. Formally, it is defined as

$$CKF_w \leq d_w^{t_1}, d_w^{t_2}, \cdots, d_w^{t_f} > \text{and } t_1 < t_2 < \cdots < t_f,$$

where $d_w^{t_j}$ denotes the document that a worker w accessed at time t_j for a specific task. Each document can be represented by a document profile, which is an n-dimensional vector containing weighted terms that indicate the key content of the document.

Definition 3. Topic-level KF (TKF): A TKF is a topic sequence derived by clustering similar documents based on the times that the documents were accessed in the CKF. Formally, it is defined as

$$TKF_w \leq TP_w^{t_1}, TP_w^{t_2}, \cdots, TP_w^{t_f} >, t_1 < t_2 < \cdots < t_f,$$

where $TP_w^{t_j}$ denotes the corresponding topic of the document that worker w accessed at time t_j for a specific task. Each topic is represented by a topic profile, which is an n-dimensional vector containing weighted terms that indicate the key content of the topic.

Document profile generation. Two profiles, a document profile and a topic profile, are used to represent a worker's

DOI: 10.1002/asi

KF. A document profile can be represented as an n-dimensional vector, which comprises the key terms in the document and their respective weights derived by the normalized *tf-idf* approach according to Equation 1. Based on the term weights, terms with higher values are selected as discriminative terms to describe the characteristics of the document. The document profile d_j is composed of these discriminative terms. Let the document profile be $DP_j = < dt_{1j}: dtw_{1j}, dt_{2j}: dtw_{2j}, \ldots, dt_{nj}: dtw_{nj} >$, where dt_{ij} is a term i in document d_j and dtw_{ij} is the degree of importance of the term i to the document d_j, which is derived by the normalized *tf-idf* approach. For a specific task, the worker's document profiles are used to represent the documents in the CKF and describe his or her accumulated knowledge at the codified level.

Topic profile generation. The TKF, which is derived by clustering documents with similar content and access times in the CKF, is represented by a topic sequence. Based on the order of documents in each worker's CKF, documents with similar content are grouped into clusters by using a hierarchical agglomerative clustering method with a time variant (HACT) algorithm. When clustering a series of time-ordered documents, the algorithm considers the documents' contents as well as the times that the documents were accessed. The steps of the HACT algorithm are detailed in the Appendix.

Initially, each document in the CKF is regarded as a single topic. The HACT algorithm then iteratively merges topics until the number of topics is less than a prespecified minimum number of topics. A time window, which defines the merging scope of the candidate topics, is moved from the first to the last topic in the TKF to determine the number of merged candidates. In an iteration of the merging process, the two candidates with the maximum similarity are merged if neither of them has been merged with another candidate in the current iteration. We use the cosine measure to calculate the similarity between the profiles of two candidate documents and employ the average linkage method (Jain & Dubes, 1988; Jain et al., 1999) to calculate the similarity between two clusters. The similarity measure between two clusters c_i and c_j is defined as the average of all pairwise similarities among the documents in c_i and c_j. The HACT algorithm groups similar documents or similar topics that are in the same time window into clusters. The algorithm performs hierarchical clustering, and thus various levels of clustering result can be derived. The quality measure, the CQ value, is derived to indicate the quality of each clustering result. We select the clustering result that leads to the best quality value.

Documents in the same cluster contain similar content and form a topic set. The key features of the cluster are described by a topic profile derived from the profiles of documents in the cluster. Let $TPf_x = < tt_{1x}: ttw_{1x}, tt_{2x}: ttw_{2x}, \ldots, tt_{nx}: dtw_{nx} >$ be the profile of a topic (cluster) x, where tt_{ix} is a topic term and ttw_{ix} is the weight of the topic term. In addition, let D_x be the set of documents in cluster x. The weight of a topic term is determined by Equation 5 as follows:

$$ttw_{ix} = \frac{\sum_{j \in D_x} dtw_{ij}}{|D_x|}, \tag{5}$$

where dtw_{ij} is the weight of term i in document j, and $|D_x|$ is the number of documents in cluster x. The weight of a topic term is obtained from the average weight of the terms in the document set. According to Equation 5, if a specific term in most documents of the topic has high weights, the topic weight of that term will be high. On the contrary, if a specific term in most documents of the topic has low weights, the topic weight of that term might be low. Terms with low weights in a topic are not good enough to represent the topic profile.

Grouping Knowledge Workers and Generating Group Profiles

To find a target worker's neighbors, we compare his or her TKF with those of other workers to compute the similarity of their KFs. The resulting similarity score indicates whether the KF referencing behavior of two workers is similar. Since each KF is a sequence, the sequence alignment method (Charter, Schaeffer, & Szafron, 2000; Oguducu & Ozsu, 2006), which computes the cost of aligning two sequences, can be used to measure the similarity of two KF sequences. Based on this concept, we use a hybrid similarity measure, which comprises the KF alignment similarity and the aggregated profile similarity, to evaluate the similarity of two workers' KFs, as shown in Equation 6:

$$sim(KF_i, KF_j) = \gamma \times sim_a(TKF_i, TKF_j) + (1-\gamma) \times sim_P(AP_i, AP_j), \tag{6}$$

where $sim_a(TKF_i, TKF_j)$ represents the KF alignment similarity, $sim_p(AP_i, AP_j)$ represents the aggregated profile similarity, and γ is a parameter used to adjust the relative importance of the two types of similarity. The KF alignment similarity is derived based on the topic sequence and topic coverage while the aggregated profile similarity is derived based on the aggregated profiles derived from the profiles of referenced documents in the KFs. Note that the KF alignment similarity considers the topic sequence in the KF without considering the content of workers' profiles, whereas the aggregated profile similarity considers the content of profiles without considering the topic sequence in the KF. By linearly combining these two similarities, we can balance the trade-off between KF alignment similarity and the aggregated profile similarity. The resulting similarity score indicates whether the KF referencing behavior of two workers is similar. The similarity of two workers is generally high if both their referenced topic sequences and aggregated content profiles are similar. Here, we only give a brief explanation of the measure (for further details, refer to Lai & Liu, 2009).

DOI: 10.1002/asi

In this work, we adopt the CLIQUE clustering method (Agrawal et al., 1998; Jain et al., 1999) to group knowledge workers based on a similarity matrix of their KFs. Each entry in the matrix represents the degree of KF similarity between two workers, derived by Equation 6. Workers in the same cluster are closely connected because they have similar referencing behavior and information needs. Thus, a cluster represents a team comprised of several knowledge workers with similar task knowledge.

The members of a group have similar KFs because their information needs are similar, and they usually need to refer to related documents for a specific topic. Thus, the group-based approach generates the group-based score (preference) of a group k for a target document based on the TKFs of the group's members. Since similar documents are grouped into clusters (topics), a larger number of related documents that may satisfy workers' task needs can be recommended by considering TKFs rather than CKFs. We identify the important topics that the members accessed and compute their weights based on each member's KF (Equation 7). Let T_u be the set of topics in the TKF of user u, and let U_k be the set of users in group k. $GTS_k = \bigcup_{u \in U_k} T_u$ is the set of topics accessed by members of group k. For a topic x in GTS_k, $GTR_{k,x}$ is group k's accumulated rating for topic x. The rating also indicates the importance of topic x to group k, as defined in Equation 7. This topic rating of a group is obtained by averaging the personal ratings of group k's members for topic x.

$$GTR_{k,x} = \frac{\sum_{u \in U_k} PTR_{u,x}}{|U_k|}, \quad (7)$$

where $|U_k|$ is the number of workers in the group; and $PTR_{u,x}$ is the personal rating of worker u for topic x, indicating the importance of topic x to worker u. Note that the worker u belongs to group k. $GTR_{k,x}$ is high if topic x is important to most of group k's members.

Because not every topic has an explicit rating given by a user, the personal rating for a topic is derived by Equation 8 based on worker u's TKF, assuming that topic y_t is the topic accessed by worker u at time t. $PTR_{u,x}$ is the weighted average of worker u's ratings for topics in u's TKF. It is derived by using the time factor and the similarity measures of the topics to topic x as the weights. $PTR_{u,x}$ is high if topic x is similar to worker u's recently accessed and high-rating topics.

$$PTR_{u,x} = \frac{\sum_{t=1}^{t_{now}} \overline{TR}_{u,y_t} \times tw_{t,t_{now}}^{u,y_t} \times csim(TPf_x, TPf_{y_t})}{\sum_{t=1}^{t_{now}} tw_{t,t_{now}}^{u,y_t} \times csim(TPf_x, TPf_{y_t})}, \quad (8)$$

where $\overline{TR}_{u,y_t}$ is the average rating of worker u for topic y_t. $\overline{TR}_{u,y_t}$ is derived by averaging the ratings of worker u for documents belonging to topic y_t. TPf_x/TPf_y is the topic profile of topic x/topic y_t described earlier, and $csim(TPf_x, TPf_{yt})$ is the profile similarity between topic x and topic y_t measured by the cosine formula. In addition, $tw_{t,t_{now}}^{u,y_t}$ is the time weight of topic y_t accessed by worker u at time t. It is formulated as $tw_{t,t_{now}}^{u,y_t} = (t - St)/(t_{now} - St)$, where St is the start time of the worker's KF and t_{now} is the time that the worker accessed the most recent topic in his or her KF.

Based on Equation 7, the personal ratings of a topic given by a group's workers can be aggregated as a group rating for the topic. This group rating can be used to measure the importance of the topic to the group. The higher the $GTR_{k,x}$ score, the more important topic x is to group k.

Recommendation Phase

This phase combines the KF-based group recommendation (KFGR) method with the personalized methods to generate recommendation lists for workers. In the following subsections, we discuss KFGR and three hybrid methods: the KFGR-UCF method, the KFGR-ICF method, and the KFGR-CB method.

The KFGR method. The KFGR method combines the group rating derived from the document ratings of group members and the group rating based on the TKF. Group members can access and rate the target document, so we consider the members' personal ratings when calculating the predicted group rating for the target document. In addition, some topics may be of interest or important to the majority of the group's members. Since documents related to those topics will probably satisfy the workers' information needs, the proposed group-based approach considers the importance of the topics accessed by group members. To combine the two types of group ratings, we use an activity weighting as an adjustment parameter. The weighting is based on the ratio of group members who have rated the target document.

More specifically, let $GDR_{k,i}$ be the predicted group rating of group k for a target document i, as shown in Equation 9. The value of $GDR_{k,i}$ is derived by combining the two types of group ratings; that is, $\overline{Gr}_{k,i}$, the group rating derived from the document ratings of the group members (the first part of Equation 9), and the group rating based on the TKFs (the second part of Equation 9). The group rating $\overline{Gr}_{k,i}$ is derived by using the group members' personal ratings for target document i. The group rating based on the TKFs is the weighted average of group k's ratings for topics. It is derived by using the similarity measures of the topics to the target document as the weights.

$$GDR_{k,i} = Aw_{k,i} \times \overline{Gr}_{k,i} + (1 - Aw_{k,i}) \times \frac{\sum_{x \in GTS_k} csim(TPf_x, DPf_i) \times GTR_{k,x}}{\sum_{x \in GTS_k} csim(TPf_x, DPf_i)}, \quad (9)$$

where $GTR_{k,x}$ is the predicted group rating of group k for topic x measured by Equation 7, TPf_x is the profile (term vector) of topic x, DPf_i is the profile (term vector) of document i, and GTS_k is the topic set of group k. $Aw_{k,i}$, the activity weighting of group k for document i, is a parameter that is

DOI: 10.1002/asi

used to adjust the relative importance of the two types of group ratings.

The value of $Aw_{k,i}$ varies according to the rating ratio of group k's members who have rated target document i. The predicted group rating takes into account both the average document ratings given by the group k's members and the weighted average of group k's ratings on topics. $GDR_{k,i}$ is high if both the average document ratings and the weighted average of group k's ratings on topics are high. The predicted group rating can be used to measure the importance of a document to a group.

Next, we discuss $\overline{Gr}_{k,i}$ and $Aw_{k,i}$ in detail. $\overline{Gr}_{k,i}$ is the weighted average of the members' document ratings for document i in the KFs of the group members, and is derived by using the time of the rating as the weight, as shown in Equation 10.

$$\overline{Gr}_{k,i} = \frac{\sum_{u \in U_k} (r_{u,i} \times tw_{t,t_{now}}^{u,i})}{\sum_{u \in U_k} tw_{t,t_{now}}^{u,i}}, \quad (10)$$

where U_k is the set of users in group k, $r_{u,i}$ is worker u's rating for document i, and $tw_{t,t_{now}}^{u,i}$ is the time weight of the rating ($r_{u,i}$) that worker u gave i at time t. The value of $\overline{Gr}_{k,i}$ is the weighted average of the personal ratings of group k's members for document i, and is derived by considering the time factors of the ratings in members' KFs. $\overline{Gr}_{k,i}$ is generally high if most personal ratings of group k's members for document i are high. In Equation 9, the activity weighting of group k for document i (i.e., $Aw_{k,i}$) is a parameter used to adjust the relative importance of the two types of group ratings. The value of $Aw_{k,i}$ varies between 0 and 1. Let $M_{k,i}$ denote the set of group k's members who have rated target document i. U_k is the set of users in group k. We determine $Aw_{k,i}$ by considering group k's rating ratio for document i (i.e., $|M_{k,i}|/|U_k|$), as defined in Equation 11:

$$Aw_{k,i} = \begin{cases} \beta + (1-\beta) \times min\left(\frac{2 \times |M_{k,i}|}{|U_k|}, 1\right), if\ |M_{k,i}| > 0 \\ 0, if\ |M_{k,i}| = 0 \end{cases}, \quad (11)$$

where β is a parameter used to adjust the importance of group k's rating ratio for document i in deriving $Aw_{k,i}$. The value of β varies between 0 and 1, and is determined by experimental analysis.

If $|M_{k,i}| = 0$ (i.e., document i is not rated by any group member), the value of the activity weighting will be 0. If document i has been rated by one or more of group k's members ($|M_{k,i}| > 0$), the activity weighting is assigned a basic value β, and is added with $(1 - \beta)$ multiplied by the $min(2 \times |M_{k,i}|/|U_k|, 1)$. The larger group k's rating ratio for document i, the higher the value of the group's activity weighting for the document. If at least half the members of group k rate document i, the group's activity weighting will be 1. When β is 0, the activity weighting is derived entirely from group k's rating ratio for document i. However, if β is 1 and $|M_{k,i}| > 0$, the activity weighting will be 1 regardless of group k's rating ratio for document i. In the experiments, we vary the value of β from 0 to 1 in increments of 0.1 to evaluate the effect of the activity weighting with and without group k's rating ratio for document i.

The larger group k's rating ratio for document i, the higher the value of the group's activity weighting for the document. For the KFGR method (Equation 9), a high value for the activity weighting $Aw_{k,i}$ implies that $\overline{Gr}_{k,i}$ is reliable for representing group k's rating of document i. That is, the group rating based on the document ratings of group members (i.e., $\overline{Gr}_{k,i}$) will contribute more to the predicted group rating, $GDR_{k,i}$. If very few group members have rated document i, the value of $Aw_{k,i}$ will be smaller (the weight value $1 - Aw_{k,i}$ will be larger); therefore, the group rating based on the TKFs will contribute more to the predicted group rating.

To improve the quality of recommendations, we combine the KFGR method with three traditional recommendation methods—UCF, ICF, and CBF—to form hybrid methods, which we discuss in the following subsections.

The hybrid KFGR-UCF method. We combine the KFGR method with UCF to recommend documents to a target worker, as shown in Figure 2. The recommendation list is generated by combining the predicted ratings of KFGR and UCF. Recall that, to make recommendations, KFGR uses the group's information needs based on the members' KFs. It recommends a group's preferred documents to a target worker by considering the group members' preferences (i.e., ratings for the target documents) as well as the group's accumulated ratings for topics. Meanwhile, the UCF method recommends documents to the target worker based on the ratings of workers with similar information needs. The similarity between workers' information needs is determined by calculating Pearson's correlation coefficient based on the workers' ratings for documents. Thus, the predicted rating of a document is obtained from neighbors who have similar preferences to the target worker and whose similarity scores are higher than a threshold θ, as shown in Equation 2.

To integrate the merits of the KFGR and UCF methods, we combine them as the hybrid KFGR-UCF method to improve recommendation performance. Based on the hybrid method, the predicted rating of worker a for document i, $PDR_{a,i}$, is derived by Equation 12:

$$PDR_{a,i} = \alpha_{KFGR-UCF} \times GDR_{k,i} + (1 - \alpha_{KFGR-UCF}) \times \left(\overline{r_a} + \frac{\sum_{u \in Neighbor(a)} Psim(R_a, R_u) \times (r_{u,i} - \overline{r_u})}{\sum_{u \in Neighbor(a)} Psim(R_a, R_u)}\right), \quad (12)$$

where $GDR_{k,i}$ is the predicted rating of group k for document i based on Equation 9; $Psim(R_a, R_u)$ is Pearson's correlation coefficient between user a and user u measured by their rating vectors R_a and R_u; $\overline{r_a}$ and $\overline{r_u}$ are the average ratings of worker a and worker u, respectively; $r_{u,i}$ is the rating given by worker u for document i; and $\alpha_{KFGR-UCF}$ is a parameter

DOI: 10.1002/asi

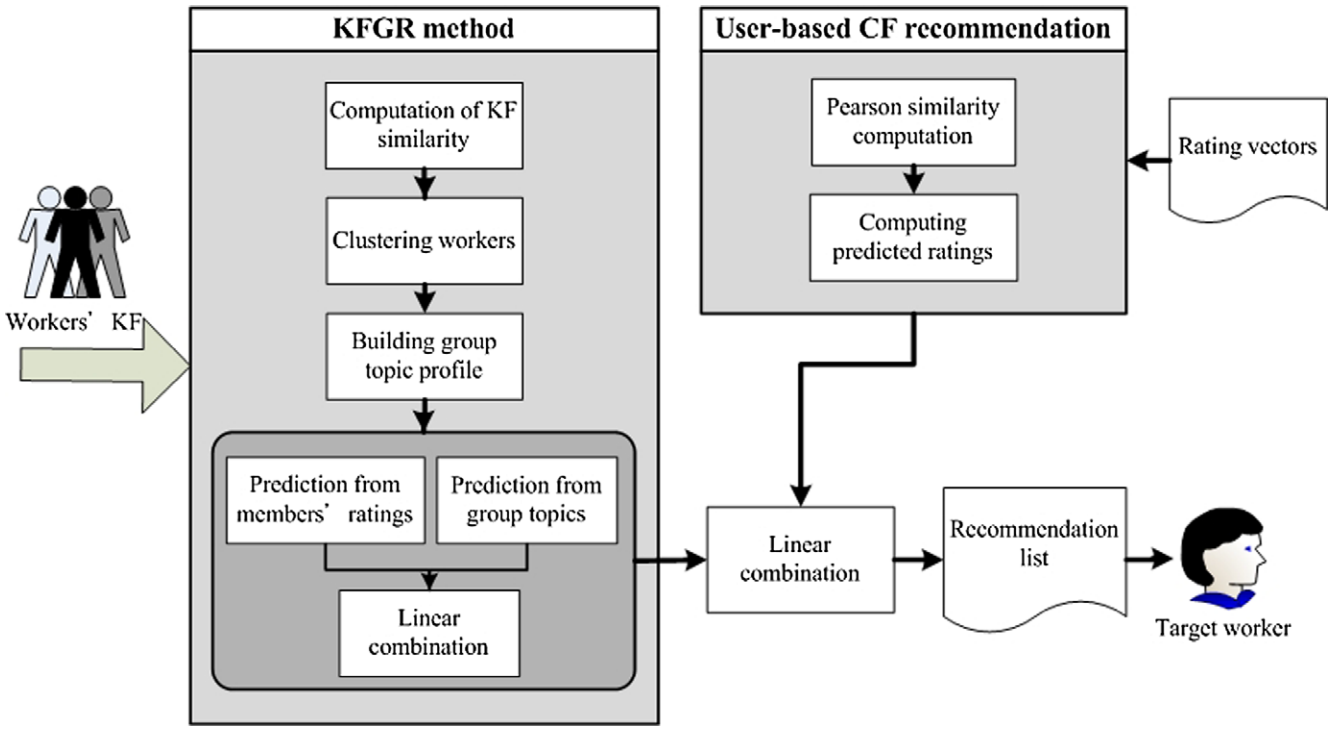

FIG. 2. The recommendation process of the hybrid KFGR-UCF method. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

used to adjust the weight between group-based prediction and UCF prediction.

The value of $\alpha_{KFGR\text{-}UCF}$ is between 0 and 1. It is derived through experiments by systematically making adjustments in increments of 0.1. When the value of $\alpha_{KFGR\text{-}UCF}$ is 1, $PDR_{a,i}$ is derived by the KFGR method (i.e., $GDR_{k,i}$). That is, the recommendations are dominated by the group preferences. In contrast, when the value of $\alpha_{KFGR\text{-}UCF}$ is 0, $PDR_{a,i}$ is derived by the UCF method, which means that the recommendation is dominated by personal preferences. $PDR_{a,i}$ is generally high if the group-based predicted rating and the UCF predicted rating are high. From our experimental analysis, the best setting of $\alpha_{KFGR\text{-}UCF}$ is determined by the recommendation result with the lowest mean absolute error (MAE) value. Based on the predicted ratings derived by Equation 12, documents with high ratings are used to compile a recommendation list. Then, the top-N documents are recommended to the target worker.

The hybrid KFGR-ICF method. The hybrid KFGR-ICF method linearly combines the KFGR method with the ICF method to recommend documents to a target worker. The recommendation list is generated by combining the predicted ratings of the two methods: KFGR and ICF. The ICF method (Sarwar et al., 2001), described earlier, recommends documents by identifying documents that are similar to a target document. The similar documents are selected based on their adjusted cosine similarity scores, derived by Equation 4. Then, the predicted rating is obtained by taking the weighted average of the target worker's ratings for the similar documents, as shown in Equation 3. As the ICF method does not consider a group's information needs, it may neglect some important documents needed by both the group and the target worker. To resolve the problem, we propose the hybrid KFGR-ICF method, which combines the KFGR method and the ICF method to recommend suitable documents to the target worker. This method takes both personal and group perspectives into account to provide recommendations. $PDR_{a,i}$, the predicted rating of worker a for document i, is derived by Equation 13:

$$PDR_{a,i} = \alpha_{KFGR-ICF} \times GDR_{k,i} + (1-\alpha_{KFGR-ICF}) \times \left(\frac{\sum_{j \in ASDS(i)} ADsim(D_i, D_j) \times r_{a,j}}{\sum_{j \in ASDS(i)} ADsim(D_i, D_j)} \right), \quad (13)$$

where $ADsim(D_i, D_j)$ is the adjusted cosine similarity (Equation 4) between document i and document j measured by their respective rating vectors D_i and D_j, $r_{a,j}$ is the rating of document j given by worker a, $ASDS(i)$ is the similar document set of document i based on the adjusted cosine similarities of the documents, and $\alpha_{KFGR\text{-}ICF}$ is a parameter used to adjust the weights of the KFGR method and the ICF method. $PDR_{a,i}$ is generally high if the group-based predicted rating and the ICF predicted rating are high. The value of $\alpha_{KFGR\text{-}ICF}$, which ranges from 0 to 1, is determined in the same way as the value of $\alpha_{KFGR\text{-}UCF}$ in the hybrid KFGR-UCF method. Documents with high ratings are used to compile a recommendation list and are recommended to the target worker.

DOI: 10.1002/asi

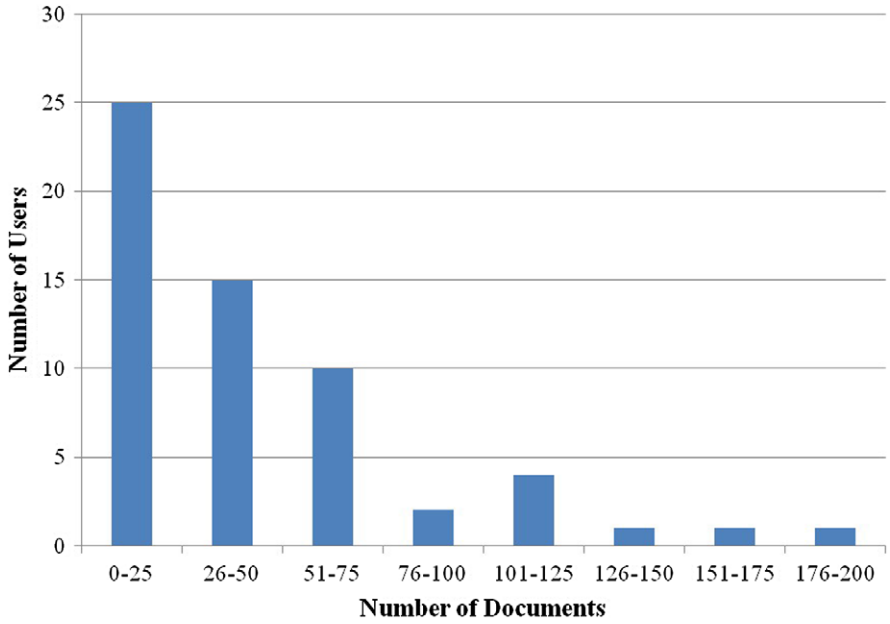

FIG. 3. The number of documents versus the number of users in the data set. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

The hybrid KFGR-CB method. The KFGR-CB method recommends documents to a target worker by linearly combining two predicted ratings, obtained by using CBF and the KFGR methods, respectively. The CB method recommends documents by considering the content (term vectors) of each document and identifies similar documents by comparing them with documents previously referenced by the target worker. It then predicts a document's rating based on the ratings that the worker gave the previously referenced documents. Because the CB method does not consider a group's information needs, it may ignore important knowledge required by the group. To take advantage of these two methods, the proposed hybrid KFGR-CB method recommends documents to a target worker by integrating the traditional CBF method with the KFGR method, as shown in Equation 14:

$$PDR_{a,i} = \alpha_{KFGR-CB} \times GDR_{k,i} + (1-\alpha_{KFGR-CB}) \times \left(\frac{\sum_{j \in SDS(i)} csim(DPf_i, DPf_j) \times r_{a,j}}{\sum_{j \in SDS(i)} csim(DPf_i, DPf_j)} \right), \quad (14)$$

where $PDR_{a,i}$ is the predicted rating of worker a for document i, $csim(DPf_i, DPf_j)$ is the cosine similarity between document profile DPf_i and document profile DPf_i, $r_{a,j}$ is the rating of document j given by worker a, $SDS(i)$ is the similar document set of document i based on the cosine similarity scores of the documents, and $\alpha_{KFGR-CB}$ is a parameter used to adjust the combined weight of the group-based method and the CBF method. $PDR_{a,i}$ is generally high if the group-based predicted rating and the CB predicted rating are high. The value of $\alpha_{KFGR-CB}$, which ranges from 0 to 1, is determined in the same way as the value of $\alpha_{KFGR-UCF}$ in the hybrid KFGR-UCF method. Documents with high ratings are used to compile a recommendation list and are recommended to the target worker.

Experiments and Evaluations

We conducted a number of experiments to evaluate the proposed hybrid methods. In the following subsections, we describe the experiment setup and discuss the results.

Experiment Setup

To demonstrate that KFs can support the recommendations of task-relevant knowledge (documents) for knowledge workers, we conducted the experiments on a data set from a real application domain; namely, research tasks in the laboratory of a research institute. In our experiments, the collected documents are related to the research work of a research lab. We build a knowledge management system (KMS) to collect such task-related documents from knowledge workers' tasks. The KMS records every knowledge worker's access behaviors on documents. Since it is difficult to obtain such a data set, using the real application domain restricts the sample size of the data and the number of participants in the experiments.

In our experiments, the data set collected from our KMS system was comprised of over 600 documents that had been accessed by about 60 workers. Figure 3 shows a graph about the number of documents versus the number of users having that number of documents in the data set. Our data

DOI: 10.1002/asi

set included usage logs, which provided information about the workers' access behavior (i.e., browsing, rating, downloading, and uploading documents). The ratings given to documents on a scale of 1 to 5 indicate their relevance and usefulness to the worker's task. A high rating (i.e., 4 or 5) indicates that the document is perceived as relevant or useful; a low rating (i.e., 1 or 2) indicates that the document is deemed not relevant. In addition, browsing behavior and uploading/downloading behavior are given default ratings (3 and 4, respectively) to indicate a user's preference for a document.

We used the log data to determine each worker's preferences. We can discover the workers' KFs from their usage logs. Based on the order of documents in workers' CKF, we use the HACT algorithm (described earlier) to group documents with similar content into clusters (topics) and derive workers' TKFs. There are 53 topics in our data set. Each topic contains several documents with similar content. On average, there are approximately 12 documents in each topic. Note that the topic profiles rather than the topic IDs are used to compute the topic ratings or the similarity measures of topics and documents.

In our experiment, the data set is divided into a training set and a testing set. In our proposed methods, the access time of documents is an important factor. In general, more recently accessed documents of knowledge workers were used for testing data; however, some recently accessed documents may become new documents to the system if all the users are used as the target users for testing evaluation. To avoid the new-item problem of CF methods, we basically used a stratified sampling approach to select the target workers from each group of workers. Some users who access very few documents and give very few ratings are not suitable for testing evaluation. Thus, we may need to reselect the target workers. The data of nontarget workers are included in the training set. Data of target workers are further divided as 70% for training and 30% for testing based on the access time of documents in those workers' document sequences. The training set is used to generate recommendation lists; the test set is used to verify the quality of the recommendations. The training data are separated from the testing data. Accordingly, we evaluate the performances of our proposed methods and compare them with the traditional methods.

To measure the recommendation quality of the methods, we use the MAE and the mean square error (*MSE*). The MAE compares the average absolute deviation of the predicted rating and the true rating while the *MSE* measures the average of the squares of errors, which are the differences between the predicted ratings and the real ratings. Similar to the MAE, the *MSE* is another predictive accuracy metric for evaluating the accuracy of recommender systems (Herlocker, Konstan, Terveen, & Riedl,, 2004). *MSE* squares the error before summing it, and its result emphasizes large errors. The lower the MAE/*MSE* score, the better the accuracy of the recommendation method. The MAE and *MSE* are derived by Equations 15 and 16, respectively:

$$MAE = \frac{\sum_{i}^{N} \left| \hat{P}_i - r_i \right|}{N}, \tag{15}$$

$$MSE = \frac{\sum_{i}^{N} \left(\hat{P}_i - r_i \right)^2}{N}, \tag{16}$$

where N is the number of user-document rating pairs in the testing data set, $\hat{P}_i$ is the predicted rating of document i, and r_i is the real rating of document i given by the user. Even though the *MSE* differs slightly from the MAE, it accounts for recommendation errors in the same way.

In the following subsections, we explain how we determine the parameters used in the experiments and then compare the performance of the proposed methods and the traditional methods.

Evaluation of the Effects of the Time Factor and the Activity Weighting

In this experiment, we evaluate the effects of the time factor and the activity weighting of the proposed KFGR method. We compare the KFGR method with the KFGR-NT method, which derives the predicted ratings based on the KFGR method without considering the time factor.

First, we explain how to determine the value of parameter β, which is used to derive the activity weighting (Equation 11) of the KFGR and KFGR-NT methods. The KFGR method (Equation 9) is a hybrid method that uses an activity weighting to combine two types of group ratings. One type, derived by Equation 10, is the group rating based on the group members' ratings for a target document; the other type is the group rating based on the TKFs. Because the group members' information needs may change over time, both types of ratings take the time factor into account. To combine the ratings, the activity weighting (Equation 11), which is based on the group's rating ratio for the target document, is used to adjust the relative importance of the two ratings. The KFGR-NT method uses a similar process to obtain the predicted ratings for documents. The major difference is that KFGR-NT does not consider the time factor when calculating the two types of group ratings. That is, it uses Equations 7–10 without considering the time weights to derive the predicted group rating for the target document.

For the KFGR and the KFGR-NT methods, the value of parameter β is a decimal in the range 0 to 1. It is used as the weight $(1 - \beta)$ to adjust the activity weighting according to the rating ratio of group members that have rated the target document. To obtain the best predicted rating, we conducted an experiment in which we systematically adjusted the value of β in increments of 0.1, and chose the optimal value (i.e., the lowest MAE value) as the best setting for the KFGR and KFGR-NT methods.

Figure 4 shows the MAE value under different settings of β for activity weightings in the KFGR method. We observe

DOI: 10.1002/asi

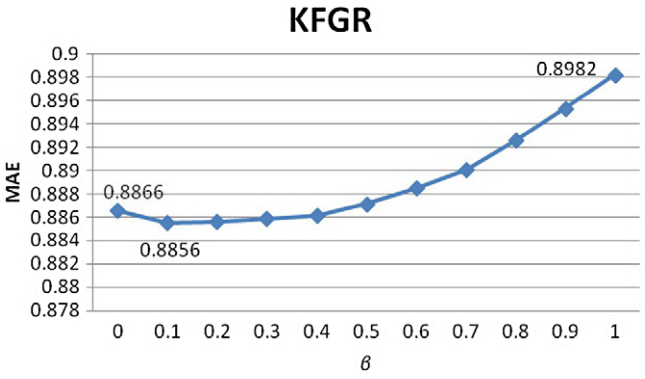

FIG. 4. The mean absolute error (MAE) values for different β settings under KFGR. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

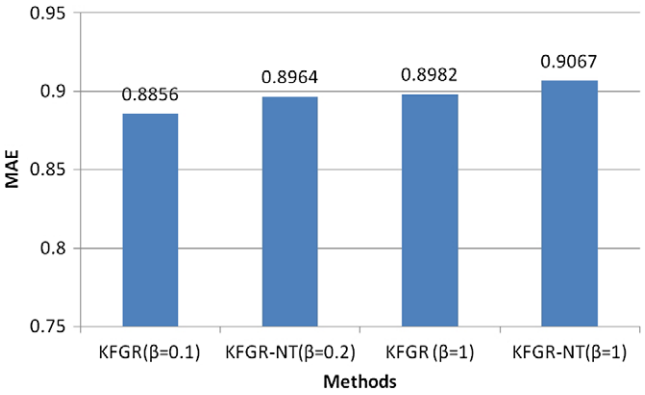

FIG. 5. Comparison of KGFR and KGFR-NT. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

that the lowest MAE occurs when β is 0.1. The setting of the parameter $(1-\beta)$ is 0.9 for the importance of the group's rating ratio for the target document. In the rest of the experiments, we set $\beta = 0.1$ for the activity weighting of the KFGR method to predict document ratings. When $\beta = 0$, the activity weighting is derived entirely from the group's rating ratio for the target document. However, when $\beta = 1$ and $|M_{k,i}| > 0$, the activity weighting is set to 1 without dynamically adjusting the value based on the group's rating ratio for the target document. The predicted group rating of KFGR is derived from the group members' ratings for the target document (Equation 10). Under the KFGR-NT method, the best setting of β for activity weighting is 0.2. The MAE value of the KFGR-NT method with $\beta = 0.2$ is larger than that of KFGR method with $\beta = 0.1$.

Figure 5 shows the MAE values of four variations of KFGR: KFGR($\beta = 0.1$), KFGR($\beta = 1$), KFGR-NT($\beta = 0.2$), and KFGR-NT($\beta = 1$). In KFGR($\beta = 1$) and KFGR-NT($\beta = 1$), the predicted group rating is derived from the group members' ratings for the target document, and the value of the activity weighting is set to 1 without using the group's rating ratio for the target document to dynamically adjust the activity weighting. Both KFGR($\beta = 0.1$) and KFGR($\beta = 1$) methods take the time factor into account to obtain the group ratings, but the KFGR-NT($\beta = 0.2$) and KFGR-NT($\beta = 1$) methods do not consider it.

Clearly, KFGR($\beta = 0.1$), which considers the time factor, outperforms KFGR-NT($\beta = 0.2$), and the performance of KFGR($\beta = 1$) is better than that of KFGR-NT($\beta = 1$). In our methods, the documents that are accessed recently are more important to users than are older documents, so they are assigned a higher time weight. Because the KFGR method considers the time factor, it is more capable of satisfying users' information needs. In addition, KFGR($\beta = 0.1$) outperforms KFGR($\beta = 1$) while KFGR-NT($\beta = 0.2$) outperforms KFGR-NT($\beta = 1$). Adjusting the activity weighting dynamically based on the group's rating ratio for the target document helps improve the recommendation quality. If at least half of a group's members rate a document, the value of the activity weighting derived from the group's rating ratio for the document would be 1; otherwise, the value would be less than 1. The larger the group's rating ratio, the higher the value of the activity weighting. For the KFGR method (Equation 9), the high value of activity weighting $Aw_{k,i}$ implies that $\overline{Gr}_{k,i}$ is reliable for representing group k's rating of document i. That is, the group rating based on the group members' ratings for the target document (i.e., $\overline{Gr}_{k,i}$) contributes more to the predicted group rating, $GDR_{k,i}$. If very few group members rate document i, the value of $Aw_{k,i}$ will be smaller (The weight value $1 - Aw_{k,i}$ will be larger.); thus, the group rating based on TKFs will contribute more to the predicted group rating.

In the following experiments, KFGR($\beta = 0.1$) is used as the KFGR method to assess the performance of the proposed hybrid methods.

Evaluation of the Recommendation Quality Under Different Numbers of Groups

Users are clustered into groups based on their similarity. Since the number of groups may affect the recommendation quality, in this experiment, we evaluate the effect of different numbers of groups. The recommendation results for KFGR with six groups and two groups are shown in Figure 6. The MAE values of KFGR for six groups and two groups are 0.8856 and 1.0661, respectively. The results show that KFGR yields more accurate predictions under six groups. The average similarity between members in a group is 0.0758 for six groups and 0.0614 for two groups. In other words, the members of a group are more similar when there are six groups. This finding implies that the preferences of the members of the six groups are more consistent than are those of the members of the two groups. Thus, the group preferences derived under six user groups are more capable of reflecting the preferences of individual members.

Although KFGR yields more accurate predictions than do the three traditional methods (i.e., UCF, ICF, CB) when there are six groups, the traditional methods outperform KFGR when there are only two groups. In the six groups, the members are quite similar and share some preferences that can be predicted successfully based on the group's

DOI: 10.1002/asi

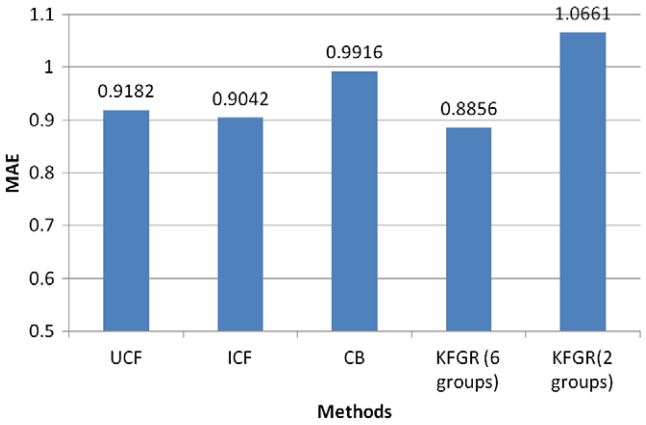

FIG. 6. Comparison of KGFR and traditional methods on different numbers of groups. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

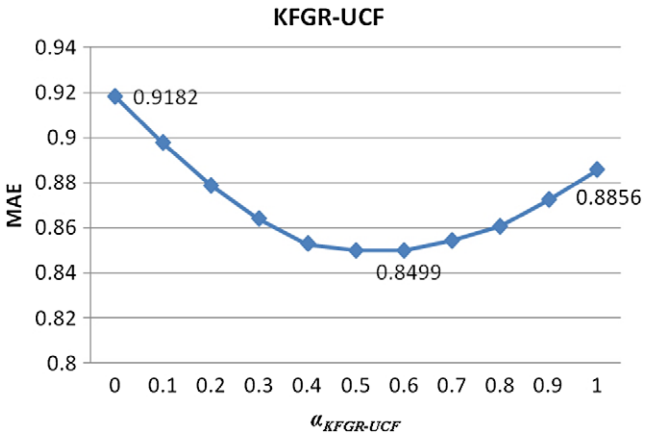

FIG. 7. Mean absolute error (MAE) under different $\alpha_{KFGR\text{-}UCF}$ settings. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

preferences. Thus, KFGR yields more accurate predictions than do the three traditional methods in this situation. In the two user groups, the members may be dissimilar and their preferences may be inconsistent, so the group preferences may not reflect the preferences of the individual members. As a result, the three traditional methods yield more accurate predictions than does KFGR on two groups.

The results demonstrate that clustering users into different numbers of groups affects the recommendation performance. The group preferences derived from user groups with appropriate clustering can reflect some common preferences of group members; therefore, they can be used to effectively predict individual members' preferences. However, the group preferences may not reflect individual members' needs if the group members' preferences vary due to the inclusion of dissimilar users in the group. Based on this result, we cluster knowledge workers into six groups in the rest of the experiments.

In an organization, worker clustering can be performed to form task-based groups such that workers in a task-based group have similar information needs. Generally, new workers who do not have much historical data can join these groups. The group recommendation method uses the information from the group and other workers in the same group to make recommendations for new users. The *k*-NN method does not apply the group concept in finding similar neighbors. In addition, the *k*-NN method has difficulties in making recommendations for new workers because they do not have enough historical data for analysis. Grouping workers is important for task-based organizations and yields some advantages while the clustering step is useful for improving the recommendation quality.

Evaluation of the Effect of the Group Perspective and the Hybrid Methods

In this subsection, we evaluate the effect of the group's perspective on predicting the information needs of individual users under KFGR. We explain how to determine the parameters used in the hybrid methods and compare the performance of the proposed hybrid methods (KFGR-UCF, KFGR-ICF, KFGR-CB) with that of the traditional methods (UCF, ICF, CB).

First, we determine the value of the parameter $\alpha_{KFGR\text{-}UCF}$ for the hybrid KFGR-UCF method (Equation 12). The parameter is used to adjust the relative importance of both KFGR and UCF, whose values range from 0 to 1. To obtain the best MAE, we systematically adjust the value of $\alpha_{KFGR\text{-}UCF}$ in increments of 0.1, as shown in Figure 7. According to our experiments, the lowest MAE value is chosen as the best $\alpha_{KFGR\text{-}UCF}$ setting. The best MAE value is generated by setting $\alpha_{KFGR\text{-}UCF}$ to 0.6. The importance weight of KFGR is 0.6 while that of UCF is 0.4. Note that when $\alpha_{KFGR\text{-}UCF}$ is set to 0, the predicted rating is derived entirely by the UCF method; however, when $\alpha_{KFGR\text{-}UCF}$ is set to 1, the predicted rating is derived entirely by the KFGR method.

Figure 8 compares the performance of UCF and KFGR-UCF. The results indicate that KFGR-UCF outperforms UCF. The KFGR-UCF method improves the recommendation quality because it considers both the group perspective (KFGR) and the individual perspective (UCF). More specifically, KFGR is capable of predicting (complementing) individual users' information needs from the group's perspective.

Next, we determine the value of $\alpha_{KFGR\text{-}ICF}$ for the hybrid KFGR-ICF method (Equation 13). The value, which ranges from 0 to 1, represents the relative importance of both KFGR and ICF. When the value of $\alpha_{KFGR\text{-}ICF}$ is 1, the recommendations are dominated by the group preferences (KFGR method). By contrast, when the value of $\alpha_{KFGR\text{-}ICF}$ is 0, $PDR_{a,i}$ is derived by the ICF method. The results indicate that the smallest value of MAE (0.8668) occurs when $\alpha_{KFGR\text{-}ICF}$ is 0.6, which means the importance weight of KFGR is 0.6 and that of ICF is 0.4. Thus, we set $\alpha_{KFGR\text{-}ICF}$ to 0.6 to derive the predicted rating of a document in the KFGR-ICF method. The comparison of the performance of KFGR-ICF and ICF

DOI: 10.1002/asi

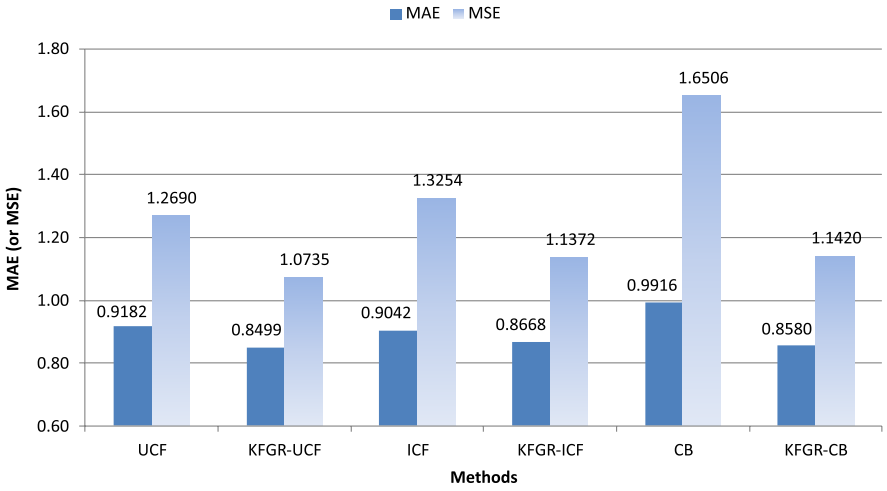

FIG. 8. Comparison of the hybrid methods and traditional methods. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

also is shown in Figure 8. Clearly, the KFGR-ICF method outperforms the ICF method. The group perspective (KFGR) improves the recommendation quality by considering the preferences of the majority of the group's members; thus, it complements the individual member's information needs.

We also evaluate the performance of CB and KFGR-CB. As with the evaluation of KFGR-UCF, we first determine the value of $\alpha_{KFGR\text{-}CB}$ for the hybrid KFGR-CB method (Equation 14). The value of $\alpha_{KFGR\text{-}CB}$, which is in the range 0 to 1, represents the relative importance of both the KFGR and CB methods. When the value of $\alpha_{KFGR\text{-}CB}$ is 1, the recommendations are dominated by the group preferences. By contrast, when the value of $\alpha_{KFGR\text{-}CB}$ is 0, $PDR_{a,i}$ is derived by the traditional CBF method. Once again, we adjust the value of $\alpha_{KFGR\text{-}CB}$ by increasing it in increments of 0.1. According to our experiments, the lowest MAE value is chosen as the best setting of $\alpha_{KFGR\text{-}CB}$. The lowest value of MAE is 0.8580, when $\alpha_{KFGR\text{-}CB}$ is 0.7. To derive the predicted rating of a document, we set $\alpha_{KFGR\text{-}CB}$ to 0 and 0.7 for the CB and KFGR-CB methods, respectively, and compare the methods. The bar chart in Figure 8 shows that the KFGR-CB outperforms the CB method. In other words, the KFGR method (group perspective) helps improve the recommendation quality. The reason is the same as under the KFGR-UCF and KFGR-ICF methods; that is, the KFGR-CB method considers both the group's preferences and the individual preferences. Hence, the resulting recommendations are more likely to match the information needs of users than are those derived by traditional methods.

Finally, we compare the hybrid KFGR-UCF, KFGR-ICF, and KFGR-CB methods with the traditional UCF, ICF, and CB methods based on MAE values, as shown in Figure 8. Of the hybrid methods, KFGR-UCF achieves the best recommendation performance, and the KFGR method clearly improves the performance of all three hybrid methods. With regard to the traditional methods, ICF outperforms UCF and CB. Overall, the KFGR-UCF method with the lowest MAE value is the best recommendation method in our experiments.

We also use the *MSE* to evaluate the recommendation performances of our proposed methods. The average *MSE* values for each of our methods are compared in Figure 8. Because of the nature of *MSE*, the *MSE* value of each method is higher than that of the MAE value. Hybrid methods (i.e., KFGR-UCF, KFGR-ICF, and KFGR-CB) have better accuracies than do traditional recommendation methods. However, the trend of recommendation accuracy of the *MSE* results differs from that of the MAE results for the UCF, ICF, KFGR-ICF, and KFGR-CB methods. In the MAE results, the trend of recommendation accuracy is KFGR-UCF < KFGR-CB < KFGR-ICF < ICF < UCF < CB. Note that the notation "<" means "is better than." In the *MSE* results, the trend of recommendation accuracy is KFGR-UCF < KFGR-ICF < KFGR-CB < UCF < ICF < CB. Because *MSE* penalizes larger errors more severely than does MAE, there is a varied result for the recommendation accuracy of these methods.

Our results demonstrate that the three hybrid methods improve the recommendation quality. The KFGR-UCF method yields the best quality recommendations, even though UCF is not the best traditional recommendation method according to the results. The KF-based group recommendation approach (KFGR method) considers group members' information needs based on their TKFs. It also considers the relative importance of group members' information needs for documents and topics over time. Unlike the traditional methods, the KFGR method can predict users' information needs from the group perspective, and thereby complements the individual member's knowledge. Moreover, combining the KF-based group recommendation approach with a traditional personalized recommendation

DOI: 10.1002/asi

method can yield a lower MAE value and enhance the quality of recommendations. Since the proposed hybrid methods consider both the group's preferences and the individual preferences, they are more likely to satisfy users' information needs than are those derived by traditional personalized methods.

Conclusion and Future Work

We have proposed three hybrid methods—the hybrid KFGR-UCF, the hybrid KFGR-ICF, and the hybrid KFGR-CB—to enhance the quality of recommendations. The methods recommend documents from two perspectives: a group perspective and a personal perspective. From the personal perspective, some documents are only relevant to a worker's specific information needs (i.e., they are not related to the group's information needs). A member's personal information needs are derived from his or her previous referencing behavior. From the group perspective, there are some documents that most group members consider relevant. The group's information needs may partially reflect an individual member's information needs that cannot be inferred from his or her past referencing behavior; hence, the group's knowledge can complement the individual member's knowledge. In this work, we consider the group perspective to compensate for the limitation of the personal perspective. However, the group perspective may neglect the information needs of individual members because it focuses on the needs of the majority of the group's members. Since the group-based method and the personalized method have distinct advantages, we combine them to exploit their respective strengths. In addition, the proposed group-based approach is based on KFs. Our results demonstrate that combining the KF-based group recommendation approach with a traditional method yields a lower MAE value and enhances the quality of recommendations.

The current study does have a limitation. Our experiments were conducted using a real application domain: research tasks in a research institute's laboratory. The domain restricted the sample size of the data and the number of participants in the experiments since it is difficult to obtain a data set that contains information that can be used for KF mining. Because of this limitation, in our future work, we will evaluate the proposed approach on other application domains involving larger numbers of workers, tasks, and documents.

In our future work, we will consider the relative importance of the opinions of the workers in a group. The workers may emphasize different aspects of the group's task-needs; however, the opinions of experienced workers should be considered more important and trustworthy than should those of new workers. We will also build a group KF to represent the evolution of a group's information needs, and recommend documents based on this group KF. In addition, the task-specific recommendation model is an interesting research topic. In this work, the clustering method is used to cluster workers with similar information needs for identifying task-based groups. We have applied only a few basic task concepts and do not explore more task-related information in our proposed method. In future work, we will extend our current work and apply more task-related information (e.g., task relations and relevance) in the recommendation model to provide the relevant documents based on personal tasks or teamwork tasks. Moreover, considering all previously accessed documents to analyze a worker's behavior is expensive and inefficient in an online environment. To address this problem, we will investigate how best to model a user's current preferences or task needs in a given period of time and then conduct relevant experiments.

Moreover, we will extend our current research in future work to explore when and how much group preferences complement personal preferences. An individual worker's opinions may be influenced by a group's preferences, and follow the information of interest provided by the group. For example, a worker or a new worker in a group who does not have enough knowledge about the group's task may need the group to provide more knowledge to support his or her work. Thus, the group preference may have more influence on the new worker than would their personal preferences. That is, in such instances, the group preference should be given a higher weight than the personal preference for complementing the group and personal preferences. Conversely, an expert worker usually has more task-related knowledge or more specific domain knowledge than have general workers. Because the expert worker has explicit domain knowledge and personal preferences, the group preference may not influence such an expert worker much. In such situations, the personal preferences of the expert worker should be given a higher weight than the group preference. In addition, it is possible to assign different weights for users by their level of expertise, roles, or amount of knowledge/preferences. The structure of a group may be loose or dense, and this may affect the importance of a group and the relation of workers to a group. Groups with different features also have various influences on different workers. Thus, it is possible to allow different weights for different groups. We will discuss this issue in our future work.

Acknowledgment

This research was supported by the National Science Council of the Taiwan under Grants NSC 96-2416-H-009-007-MY3 and NSC 99-2410-H-009-034-MY3.

References

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734–749.

Aggarwal, C.C., Gates, S.C., & Yu, P.S. (1999). On the merits of building categorization systems by supervised clustering. In Proceedings of the Fifth Association for Computing Machinery's (ACM) Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) International Conference on Knowledge Discovery and Data Mining (pp. 352–356). New York: ACM Press.

DOI: 10.1002/asi

Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications. In Proceedings of International Conference of the Association for Computing Machinery Special Interest Group on Management of Data (ACM SIGMOD) (pp. 94–105). New York: ACM Press.

Ardissono, L., Goy, A., Petrone, G., Segnan, M., & Torasso, P. (2003). Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. Applied Artificial Intelligence, 17(8), 687–714.

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). Modern information retrieval. Boston: Addison-Wesley.

Balabanovic, M., & Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. Communications of the ACM, 40(3), 66–72.

Basu, C., Hirsh, H., & Cohen, W. (1998). Recommendation as classification: Using social and content-based information in recommendation. In Proceedings of the 15th National Conference on Artificial Intelligence (pp. 714–720). Menlo Park, CA: American Association for Artificial Intelligence.

Belkin, N.J., & Croft, W.B. (1992). Information filtering and information retrieval: Two sides of the same coin? Communications of the ACM, 35(12), 29–38.

Bordogna, G., & Pasi, G. (2010). A flexible multi criteria information filtering model. Soft Computing—A Fusion of Foundations, Methodologies and Applications, 14(8), 799–809.

Breese, J.S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (pp. 43–52). San Francisco: Morgan Kaufmann.

Charter, K., Schaeffer, J., & Szafron, D. (2000). Sequence alignment using FastLSA. In Proceedings of the International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences (METMBS 2000) (pp. 239–245).

Cole, C. (2011). A theory of information need for information retrieval that connects information to knowledge. Journal of the American Society for Information Science and Technology, 62(7), 1216–1231.

Feldman, R., & Sanger, J. (2007). The text mining handbook: Advanced approaches to analyzing unstructured data (Vol. 34). New York: Cambridge University Press.

Furner, J. (2002). On recommending. Journal of the American Society for Information Science and Technology, 53(9), 747–763.

Garcia, I., Sebastia, L., Onaindia, E., & Guzman, C. (2009). A group recommender system for tourist activities. In Proceedings of the 10th International Conference on E-Commerce and Web Technologies (EC-Web) (pp. 26–37). Berlin: Springer-Verlag.

Glance, N., Arregui, D., & Dardenne, M. (1998). Knowledge pump: Community-centered collaborative filtering. In Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering (pp. 83–88).

Herlocker, J.L., Konstan, J.A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In Proceedings of the 22nd annual Association for Computing Machinery's Special Interest Group on Information Retrieval (ACM SIGIR) International Conference on Research and Development in Information Retrieval (pp. 230–237), Berkeley, CA. New York: ACM Press.

Herlocker, J.L., Konstan, J.A., Terveen, L.G., & Riedl, J.T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems, 22(1), 5–53.

Huang, Z., Chung, W., & Chen, H. (2004). A graph model for E-commerce recommender systems. Journal of the American Society for Information Science and Technology, 55(3), 259–274.

Jain, A.K., & Dubes, R.C. (1988). Algorithms for clustering data. Englewood Cliffs, NJ: Prentice-Hall.

Jain, A.K., Murty, M.N., & Flynn, P.J. (1999). Data clustering: A review. ACM Computing Surveys, 31(3), 264–323.

Jameson, A. (2004). More than the sum of its members: Challenges for group recommender systems. In Proceedings of the Working Conference on Advanced Visual Interfaces (pp. 48–54). New York: ACM Press.

Johnson, S.C. (1967). Hierarchical clustering schemes. Psychometrika, 32(3), 241–254.

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

Kim, J.K., Kim, H.K., Oh, H.Y., & Ryu, Y.U. (2010). A group recommendation system for online communities. International Journal of Information Management, 30(3), 212–219.

Kim, S., Hwang, H., & Suh, E. (2003). A process-based approach to knowledge-flow analysis: A case study of a manufacturing firm. Knowledge and Process Managment, 10(4), 260–276.

Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., & Riedl, J. (1997). GroupLens: Applying collaborative filtering to Usenet news. Communications of the ACM, 40(3), 77–87.

Lai, C.H., & Liu, D.R. (2009). Integrating knowledge flow mining and collaborative filtering to support document recommendation. Journal of Systems and Software, 82(12), 2023–2037.

Lang, K. (1995). Newsweeder: Learning to filter netnews. In Proceedings of the 12th International Conference on Machine Learning (pp. 331–339). Tahoe City, CA.

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7(1), 76–80.

Liu, D.-R., & Lai, C.-H. (2011). Mining group-based knowledge flows for sharing task knowledge. Decision Support Systems, 50(2), 370–386.

Liu, D.R., Lai, C.H., & Huang, C.W. (2008). Document recommendation for knowledge sharing in personal folder environments. Journal of Systems and Software, 81(8), 1377–1388.

Liu, D.-R., & Wu, I.-C. (2008). Collaborative relevance assessment for task-based knowledge support. Decision Support Systems, 44(2), 524–543.

Lorenzi, F., dos Santos, F., Ferreira, P., & Bazzan, A. (2008). Optimizing preferences within groups: A case study on travel recommendation. In G. Zaverucha & A. da. Casta (Eds.) Lecture Notes in Computer Science, Vol. 5249 (pp. 103–112). Berlin/Heidelberg: Springer-Verlag.

Luo, X., Hu, Q., Xu, W., & Yu, Z. (2008). Discovery of textual knowledge flow based on the management of knowledge maps. Concurrency and Computation: Practice and Experience, 20(15), 1791–1806.

Masthoff, J. (2008). Group adaptation and group modelling. In M. Virvou & L. Jain (Eds.), Intelligent interactive systems in knowledge-based environments. Studies in Computational Intelligence, Vol. 104 (pp. 157–173). Berlin, Germany: Springer.

McCarthy, J.F., & Anagnost, T.D. (1998). MusicFX: An arbiter of group preferences for computer supported collaborative workouts. In Proceedings of the Association of Computing Machinery Conference on Computer Supported Cooperative Work (ACM CSCW) (pp. 363–372). New York: ACM Press.

Mooney, R.J., & Roy, L. (2000). Content-based book recommending using learning for text categorization. In Proceedings of the Fifth Association for Computing Machinery (ACM) Conference on Digital libraries (pp. 195–204). New York: ACM Press.

O'Connor, M., Cosley, D., Konstan, J. A., & Riedl, J. (2001). PolyLens: A recommender system for groups of users. In Proceedings of the Seventh European Conference on Computer Supported Cooperative Work (pp. 199–218). Dordrecht, The Netherlands: Kluwer.

Oguducu, S.G., & Ozsu, M.T. (2006). Incremental click-stream tree model: Learning from new users for web page prediction. Distributed and Parallel Databases, 19(1), 5–27.

Pasi, G., & Villa, R. (2005). Personalized news content programming (PENG): A system architecture. Proceedings of the 16th International Workshop on Database and Expert Systems Applications (pp. 1008–1012). IEEE.

Pazzani, M., & Billsus, D. (2007). Content-based recommendation systems. In P. Brusilovsky A. Kobsa, & W. Nejdl (Eds.), The adaptive web—Methods and strategies of web personalization. Lecture Notes in Computer Science, Vol. 4321 (pp. 325–341). Berlin, Germany: Springer.

Rucker, J., & Polanco, M.J. (1997). Siteseer: Personalized navigation for the web. Communications of the ACM, 40(3), 73–76.

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

Salton, G., Wong, A., & Yang, C.S. (1975). A vector space model for automatic indexing. Communications of ACM, 18(11), 613–620.

Sarwar, B., Karypis, G., Konstan, J., & Reidl, J. (2001). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on the World Wide Web (pp. 285–295). New York: ACM Press.

Shardanand, U., & Maes, P. (1995). Social information filtering: Algorithms for automating "word of mouth." In Proceedings of the Special Interest Group on Computer–Human Interaction (SIGCHI) Conference on Human Factors in Computing Systems (CHI '95) (pp. 210–217). Denver, CO: ACM Press/Addison-Wesley.

Shin, C., & Woo, W. (2009). Socially aware TV program recommender for multiple viewers. IEEE Transactions on Consumer Electronics, 55(2), 928.

Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. Knowledge Discovery and Data Mining (KDD) Workshops on Text Mining (Vol. 400, pp. 525–526). Boston.

Van RijsBergen, C.J. (1979). Information retrieval. London: Butterworths.

Wu, I.-C., Liu, D.-R., & Chang, P.-C. (2009). Learning dynamic information needs: A collaborative topic variation inspection approach. Journal of the American Society for Information Science and Technology, 60(12), 2430–2451.

Yu, Z., Zhou, X., Hao, Y., & Gu, J. (2006). TV program recommendation for multiple viewers based on user profile merging. User Modeling and User-Adapted Interaction, 16(1), 63–82.

Zhang, C., & Xi, J. (2009). An integration model of workflow and knowledge flow. In Proceedigns of the International Conference on Management and Service Science (MASS '09) (pp. 1–4). doi:10.1109/ICMSS.2009.5302121

Zhuge, H. (2002). A knowledge flow model for peer-to-peer team knowledge sharing and management. Expert Systems With Applications, 23(1), 23–30.

Zhuge, H. (2006a). Discovery of knowledge flow in science. Communications of the ACM, 49, 101–107.

Zhuge, H. (2006b). Knowledge flow network planning and simulation. Decision Support Systems, 42(2), 571–592.

Appendix

Hierarchical Agglomerative Clustering Method with a Time Variant (HACT) Algorithm

Input: A codified-level knowledge flow of user u_w,
$CKF_w = \langle d_w^{t_1}, d_w^{t_2}, \cdots, d_w^{t_f} \rangle$, where $t_1 < t_2 < \ldots < t_f$
Output: A topic-level knowledge flow of user u_w,
$TKF_w = \langle TP_w^{t_1}, TP_w^{t_2}, \ldots, TP_w^{t_h} \rangle$, where $t_1 < t_2 < \ldots < t_h$

1 **function** HACT(CKF_w) {
2 Set a constant *offset* be 0.01; Set a time window size to be *s*;
3 $TS = CKF_w$; // each document d_i in CKF_w is a single cluster c_i in TS
4 **do while** (number of clusters in $TS > MinNumTopics$) **{**
5 **for each** cluster c_i **in** TS {
6 $c_{temp} = \{c_{i-s}, c_{i-s+1}, \ldots, c_{i-1}, c_i, c_{i+1}, \ldots, c_{i+s-1}, c_{i+s}\}$;
7 // According to time window size to determine clusters
8 **For each** cluster c_j **in** c_{temp} **where** $c_i \neq c_j$ {
9 Calculate the similarity of c_i and c_j as $sim(c_i,c_j)$;
10 **if** ($sim(c_i,c_j) \geq Threshold$) **then**
11 add {c_i, c_j, $sim(c_i,c_j)$ } to the *candidateMergeList*;
12 }
13 }
14 **do while** (*candidateMergeList* is not empty) {
15 Select and remove the pair (c_i, c_j) with maximum *similarity* from *candidateMergeList*;
16 **if** (c_i and c_j had not been merged with another cluster in current phase) **then {**
17 Merge c_i and c_j in *TS;*
18 **if** (number of clusters in $TS \geq MinNumTopics$) **then** {
19 Calculate the clustering quality value *CQ* of *TS*;
20 $CQ(TS) = 1 \Big/ \left(\frac{1}{|TS|} \sum_{c_i \in TS} \frac{sim(c_i, \bar{c}_i)}{sim(c_i, c_i)} \right)$, where $\bar{C}_i = \cup_{k \neq i} c_k$
21 Add {*TS*, *CQ*} to the list *result*;
22 }
23 }
24 }
25 **if** (there's no cluster merged) **then** decrease the *Threshold by offset*;
26 }
27 Let *TopicList* be the clustering result with highest *CQ* from the list *result*
28 // *TopicList* is the clustering result that leads to the best clustering quality value
29 **return** *TopicList*;

DOI: 10.1002/asi