Knowledge-Based Systems 57 (2014) 146–160

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

QA document recommendations for communities of question–answering websites

Duen-Ren Liu *, Yu-Hsuan Chen, Chun-Kai Huang

Institute of Information Management, National Chiao Tung University, Hsinchu 300, Taiwan

ARTICLE INFO

Article history:
Received 1 December 2012
Received in revised form 13 December 2013
Accepted 15 December 2013
Available online 21 December 2013

Keywords:
Knowledge community
Group recommendation
Knowledge complementation
Question–answering websites
Link analysis
Knowledge reputation

ABSTRACT

With the rapid development of Internet and Web 2.0 technology, Question & Answering (Q&A) websites have become an essential knowledge-sharing platform. This platform provides knowledge community services where users with common interests or expertise can form a knowledge community. Community members can collect and share QA knowledge (documents) regarding their interests. However, due to the massive amount of QAs created every day, information overload can become a major problem. Consequently, a recommendation mechanism is needed to recommend QA documents for communities of Q&A websites. Existing studies did not investigate the recommendation mechanisms for knowledge collections in communities of Q&A Websites. Traditional recommendation methods use member importance as weight to consolidate individual profiles and generate group profiles, which in turn are used to filter out items of recommendation. However, they do not consider certain factors of the recommended items, such as the reputation of the community members and the complementary relationships between documents.

In this work, we propose a novel method to recommend related QA documents for knowledge communities of Q&A websites. The proposed method recommends QA documents by considering factors such as the community members' reputation in collecting and answering QAs, the push scores and collection time of QAs, the complementary relationships between QAs and their relevance to the communities. This research evaluates and compares the proposed methods using an experimental dataset collected from Yahoo! Answers Taiwan website. Experimental results show that the proposed method outperforms other conventional methods, providing a more effective manner to recommend Q&A documents to knowledge communities.

1. Introduction

With the rapid development of Internet and Web 2.0 technology, individuals or organizations increasingly gain needed information through the Internet. The Web 2.0 boom enabled Question & Answering (QA) websites to become an important knowledge-sharing platform, where knowledge is acquired through the mechanism of question posting and answering. The Yahoo! Answers Taiwan website [48], also named Yahoo! Knowledge Plus, is a community-driven knowledge website; each user can share experiences and exchange knowledge by asking and answering questions. On the website, users can browse the questions that other users have asked, search for answers to particular questions or post questions and then wait for answers. Every solved question has a "best answer".

QA websites provide a knowledge community service whereby users with common interests or expertise can form a knowledge community to collect and share question–answering knowledge regarding their interests. The community knowledge is collected by community members through browsing and searching the QA documents. As the number of posting questions and answers increases rapidly through time, the massive amount of question–answering knowledge creates the problem of information overload, which in turn affects the effectiveness of knowledge sharing. Consequently, a recommendation mechanism is needed to recommend QA documents to communities of QA websites, in order to enhance the effectiveness of knowledge sharing in knowledge communities.

Currently, related research on Question Answering Websites focuses on finding appropriate experts for answering target questions [21], or retrieving high quality answers in community question–answering websites [45]. Previous researchers did not investigate the recommendation mechanisms for knowledge collection in question–answering websites. Moreover, previous studies on recommender systems focused on personalized recommendations of items of interest to individual users via collaborative filtering or content-based approaches [7,13,18,42].

* Corresponding author. Tel.: +886 35131245.
E-mail address: dliu@mail.nctu.edu.tw (D.-R. Liu).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.12.017

The recommendation of QA documents for knowledge communities differs from personalized recommendations, since a group-based recommendation approach is required to recommend items related to a community's interests.

Traditional group-based recommendation methods mainly include two kinds of approaches [17]. The first one aggregates interest profiles for each member in a group to form the group's interest profile. The group's interest profile is then used to filter recommended items. The second kind of approach generates a group recommendation list via aggregating the recommendation list of each member derived from personal recommendations. Traditional group-based recommendation mechanisms use the importance of a group member as a weight to derive weighted aggregation. To the best of our knowledge, there is no study on the recommendation mechanisms for knowledge collections in communities of question–answering websites. Traditional group-based recommendation mechanisms have not considered certain factors of recommended items, such as knowledge complementation and the reputation of the community member in terms of his/her collected QA documents.

In this work, a novel group recommendation method is proposed to recommend QA documents to communities of QA websites. The proposed recommendation method generates community profiles from previously collected QA documents by considering community members' reputations in collecting and answering QAs, the push scores and collection time of QAs. Moreover, users are usually interested in browsing relevant QAs of related questions to get more complete and complementary information. Traditional QA systems mainly use keywords to find relevant QAs, without considering the issue of redundancy or complementation. Users may need to find more details by using related keywords in a QA's answer to search more complementary information. In order to provide and collect more complementary knowledge for members in knowledge communities, the complementary relationship between QA documents should be considered in designing a recommendation method for QA websites. The proposed approach generates recommendations of QA documents by considering the complementary knowledge of the documents and the relevance degree between the QA document and the community profile. Finally, we use the data collected from Yahoo! Answers Taiwan to conduct our experimental evaluation. Experimental results show that the proposed method outperforms other conventional methods, providing a more effective manner of recommending QA documents to knowledge communities.

The rest of this paper is organized as follows. Section 2 describes related works, including the service of the Yahoo! Answers' knowledge community, group-based recommendations, expert recommendation systems and complementary information retrieval. Section 3 describes the details of the proposed methods for recommendation. Section 4 presents experiments and evaluation results. Finally, the conclusion and possible directions for future work are presented in Section 5.

2. Related work

In this section, we introduce literature related to this study, which include the virtual community, Yahoo knowledge community service, recommendation systems, expert findings and complementary information retrieval.

2.1. Virtual community and Yahoo! answers

A community is formed by people with common interests, experience, needs and goals, where topics of interest are shared and discussed within the group. Knowledge sharing or information circulation among the users can be effectively facilitated through communities [3,13]. There are several forms of virtual community, including: the BBS, discussion boards, emergent blogs, Facebook, Wretch and Yahoo! Answers. Yahoo! Answers Taiwan is a knowledge sharing platform based on posting and answering questions [48]. It also provides knowledge community services, where users with common interests or expertise can gather together to form a virtual community and actively share knowledge. In each knowledge community, community members can conduct several activities, such as collecting QA knowledge, posting articles, initiating a discussion topic or answering questions on behalf of the community. When community members browse or search QA documents on the QA website, they can collect QA documents related to their interests for their community. Other community members who think that the collected QA documents are helpful can push those QA documents in order to express their recommendation. Community members can answer questions posted by Yahoo users on behalf of their knowledge community; those QA documents whose answers are chosen as best answers are recorded in the community platform.

2.2. Recommendation approaches

To solve the problem of information overload, several recommender systems have been developed to assist people in information filtering, based on the data analysis technique [2,37,41]. The recommender systems have been widely used in different domains, such as recommending movies [36], books [20], user-generated videos [46], products [24], documents [23], and information on the Web [27].

Various techniques have been proposed for generating recommendations, including: content-based filtering, collaborative filtering and hybrid recommendation systems [2,5,9].

The Content-based filtering (CBF) approach analyzes users' preferences according to the attributes of an item, in order to build up a personal feature profile and then predict which items the user will like [9,34]. Content-based filtering (CBF) has been mainly used in the context of recommending items such as web pages, documents and news articles; it operates by analyzing given content descriptions. The content is parsed and item features are extracted to establish a characteristic profile. Items previously liked by a user are used to generate a user profile. Instead of analyzing the content information of target items, the collaborative filtering (CF) approaches focus on identifying the relationship between items or users. Generally, CF approaches can be divided into two types: user-based CF and item-based CF [2]. The user-based CF clusters users into different groups or finds the neighborhood of target users first, and then observes the behavior of neighbors to make recommendations [18,36]. The item-based CF approach analyzes similarities between items on the basis of a user's ratings among items. These item similarities are then used to generate recommendations for a user by finding items that are similar to those the user has previously liked [41]. Hybrid approaches have been proposed to overcome drawbacks of the CF and CBF methods [6,9]. They combine content-based filtering and collaborative filtering to improve recommendation accuracy. Moreover, matrix factorization-based methods have been proposed to discover latent factors of items and users for enabling effective recommendation [19,25]. Some studies have addressed the diversity of recommendations [1,11].

2.3. Group recommendation systems

Due to the popularity of community websites on the Internet, related group-based recommendation researches have drawn more attention recently. However, while traditional recommendation methods may be suitable for recommending items to individuals, they may not satisfy a group because the behavior of a community cannot be represented by an individual's behavior [17]. Existing group-based recommendation researches were divided into two aspects: the first kind of method aggregates the interest profile of each member in a group to form the group's interest profile; the group's interest profile is then used to filter recommended items. The second kind of approach generates a group recommendation list via aggregating the recommendation list of each member derived from personalized recommendations [17]. However, the second method does not take into account the importance of each member and the interactions between members. To improve upon the first manner of existing group-based recommendation researches, this study not only considers the preference of group members but also the interactive relationships between group members, in order to find highly influential members.

Saha and Getoor [39] think that a group profile composed of all members' preferences is not suitable, since the relative importance of each member differs in that they observe the friendship relationship among group members and regard members with more connections as the core members with highest influence. Through the interactions between core members and other members, those with similar interests to the core member are found and used to construct the basis of group-based recommendations. The social trust between members is, in addition, taken into consideration to generate the group profile accurately [35].

The current group-based recommendation systems are widely utilized in different fields, especially in the life and entertainment field. For example, in MusicFx [29], each member can give a rating to the music based on his or her preference. Then, each song will be given a different playing probability according to the level of popularity derived from members' preferences. Group-based recommendations are also used for movies or TV programs such as PolyLens [31], TV4M [50] and The Adaptive Vehicular Multimedia System [51]. These recommendation systems combine individual preferences of movies or programs and then generate a common recommendation list for the group to satisfy all the members' demands. Some recommendation systems for TV programs [44] generate group recommendation lists by considering the group's and individual's preferences in a linear combination method, in the hope of enhancing the recommendation quality by considering both factors.

In addition, group recommendation is generally used to recommend tourism schedules or scenic spots due to the fact that the participants are usually in a group. CATS [30], Intrigue [4], Travel Decision Forum [14,15] and GRSK [12] all generate group recommendations. CATS is a collaborative recommendation system based on comments which help a member of a group to plan a skiing schedule to satisfy other members; it utilizes the individual preference model and then constructs a group preference model to generate a recommendation list for the group. Intrigue provides services for mobile devices; it considers the conflict and preference relationship between the sub-group's preference and the group's preference when recommending appropriate tourism sites to the group. The Travel Decision Forum system designs a group-oriented interface where members can easily observe each other's preferences and behavior. The system will analyze and recommend a suitable scenic spot according to the social relationships between members when conflict happens. GRSK computes a group recommendation list by aggregating or incremental intersecting individual recommendations based on personal preference.

2.4. Expert recommendation systems

There are two related research topics in QA websites. The first topic is concerned with how to help users find high quality answers in order to solve their problem. The second one discusses how to find a capable expert to answer the question proposed by the asker, where social network analysis is usually used to find a suitable expert. It calculates the authority of each user in each field by constructing an asking and answering relationship between users, and then by using a connection analysis method. Jurczyk and Agichtein [16] constructed a QA social network between those asking and those answering, subsequently calculating the authority value of each user by an HITS algorithm which is based on a graphical connection relationship. Shen et al. [43] further utilized an HITS algorithm with different weights according to various roles in the QA network to evaluate the authority values of users. Zhang et al. [52] also constructed a social network graph in the same way and used a PageRank-like method to calculate the authority values of users. Liu et al. [26] proposed an approach whereby expert recommendation is based on the similarity analysis of target knowledge and expert profiles, with expert profiles formed by historical QA documents. Zhang et al. [53] considered not only the similarity between target knowledge and expert profiles, but also factors such as user evaluation, voting and question-proposing time, in order to measure the expertise level of answers. Aside from link analysis, Liu et al. [21] also considered previous answering records for analyzing the expertise level and reputation of the answerer in a particular field.

2.5. Complementary information retrieval

The identification of knowledge complementation is unclear due to the definition of complementary knowledge depending on the users themselves. Ma and Tanaka [28] use the concept of topic-structure to measure the complementary degree between two documents. However, the complementation relationship between QA knowledge documents may not belong to the same topic, and the corpus is hard to build due to the massive data at the QA website. Liu et al. [22] define two types of knowledge complementation in a QA website: partial complementation and extended complementation, and propose a method to predict complementation relationships between QA documents by building a classification model based on three measures: question similarity, answer novelty and answer correlation. The existing studies do not consider the recommendation of complementary QAs to communities of QA Websites.

3. Proposed QA recommendation approach

This section mainly introduces the proposed recommendation method, which recommends QA documents to knowledge communities. The first part gives a brief overview and framework of the group recommendation process. The second part describes data pre-processing of QA documents. The third part illustrates the proposed recommendation approach for recommending QA documents to knowledge communities by considering community preferences and complementary knowledge.

3.1. Overview of recommendation for community knowledge collection

Traditional group-based recommendation methods utilize the importance of a group member as a weight to aggregate the interest profile of each group member, in order to form the group's interest profile, which is then used to filter recommended items. However, there is no research on recommending QA documents to knowledge communities of QA websites that takes into consideration the reputation of a community member and complementary relationships between QA documents.

In this study, we propose a recommendation mechanism for community knowledge collection so as to recommend QA documents that are of interest and related to the knowledge community. A knowledge community may collect QA documents on several topics. Thus, the proposed recommendation method will build several community-topic profiles by considering factors such as community members' reputations in collecting/answering QA documents, push (recommendation) scores of QA documents and the collection time of QA documents from historical collected QA documents.

Sometimes the answer to a question may only provide partial information, so community members may wish to collect relevant QAs with partial complementation to get complete information. However, some relevant QAs may be redundant because QA websites, such as Yahoo! Answers, are open platforms that allow users to ask or answer questions freely. On the other hand, if community members wish to learn more about the information provided in an answer, they may use keywords in the answer to search for and collect extended complementary information. Target QA documents, which provide different perspectives of answers or further information of collected QA documents, should be recommended to knowledge communities. The proposed method recommends complementary QA documents to a knowledge community by considering the complementary relationships between QA documents that are measured based on question similarity, answer novelty and answer correlation.

Fig. 1 illustrates the framework of our proposed group recommendation method for a knowledge community. The framework can be partitioned into three main stages: pre-processing of QA documents, analysis of community preferences and analysis of complementary knowledge.

In the first stage, the QA documents are collected by a knowledge community, and the target QA documents posted in the knowledge sharing platform are obtained from the QA website. Then, the content of each QA document is preprocessed into a document profile vector such that the important terms are extracted via information retrieval techniques.

In the second stage, the collected QA documents are grouped into several topics according to their tags, which are labeled by knowledge collectors. A community topic profile representing the topic interests of a community is generated in order to derive the community's preference score on the target QA document by measuring the content similarity between each topic profile and target QA document (Section 3.3.1). A community topic profile is derived from a weighted aggregate of document profiles of a topic's collected documents by considering members' reputations in collecting QA knowledge and answering questions on behalf of communities (Section 3.3.2), push scores of QA documents (Section 3.3.3) as well as the time factors (Section 3.3.4) of collected QA documents. In each knowledge community, community members can conduct several activities, such as: collecting QA knowledge, posting articles, initiating a discussion topic or answering questions on behalf of the community. When community members browse or search QA documents on the QA website, they can collect QA documents on their interests for their community. Other community members, who think the collected QA documents helpful, can push those QA documents in order to express their recommendation. Members' reputations in collecting QA knowledge are obtained from analyzing the knowledge collection and recommendation (push) interactions between community members. Moreover, QA documents with higher push scores more clearly represent the community's interests. In addition, the most recent QA documents collected by a knowledge community can better reflect the current interest of the knowledge community. Hence, the time factor is considered in analyzing the interests of a knowledge community.

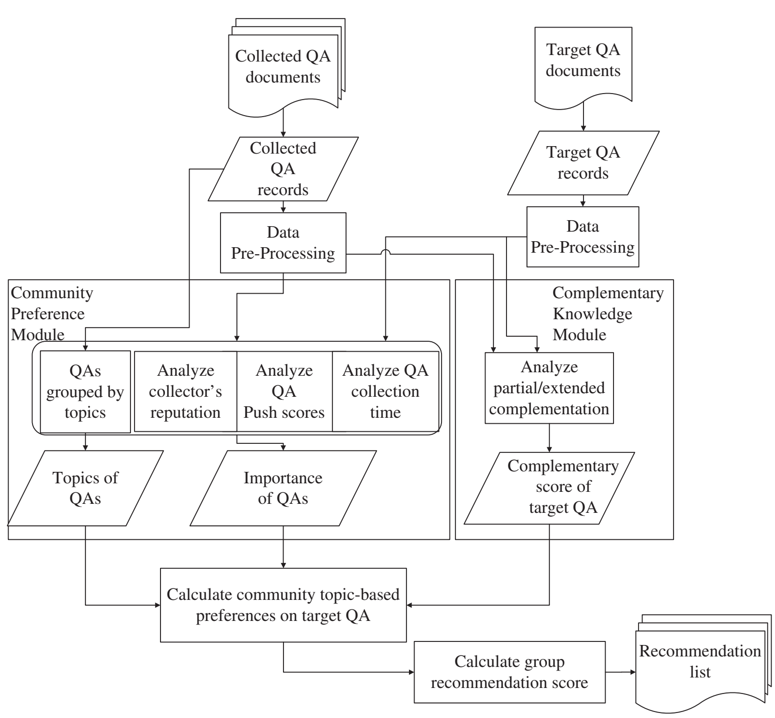

Fig. 1. The process of recommendation for knowledge community.

In the third stage, each target QA document is compared with each collected QA document of the knowledge community to determine a complementary relationship and complementary score. The complementary score is estimated through complementary knowledge analysis (Section 3.4) based on question similarities, answer novelty and answer correlation. Finally, the approach of group recommendations combines the community preference score and complementary score of each target knowledge document to generate a recommendation list for knowledge communities.

3.2. Data preprocessing

In this step, we use a document profile, i.e. term vector, to represent the knowledge subjects of a knowledge document that consists of the question title, question description and answer fields. The term vector of a QA d is denoted as KP_d. The content (the question title, question description and answer) in each QA is analyzed using the TF-IDF approach [40] to extract the important terms which can represent the knowledge subjects of the QA. The data preprocessing steps include: Word Segmentation of Yahoo API, removing stop words and calculating the TF-IDF for each term. The Word Segmentation of Yahoo API [49] is a system developed by the Yahoo Developer Network (YDN). It provides an automatic mechanism to segment the article into meaningful terms. However, there are some terms from the original article that are not representative; therefore, we remove these terms by building a stop word list which contains about 2629 words.

TF-IDF [40] is used to calculate $w_{i,d}$, the weight of term i in a profile of QA document d, KP_d. It can be calculated for a given category or a whole data set, as defined in Eq. (1):

$$w_{i,d} = \frac{tf_{i,d}}{Max_j tf_{j,d}} \times \left(\log \frac{N}{df_i}\right) \tag{1}$$

where $tf_{i,d}$ indicates the frequency of term i in document d, df_i indicates the number of documents which contain the term i and N is the total number of documents.

3.3. Preference analysis of knowledge community

A knowledge community may collect QA documents on several topics. The community preference analysis considers the topics of knowledge collected in a community; the importance of community members, which includes members' reputations in collecting and answering QA documents; push scores of QA documents, as well as the collection time for the QA documents. These factors are integrated to generate community topic profiles representing the topic interests of a community. A community topic profile is derived from a weighted aggregate of document profiles of a topic's collected documents by considering those factors as weights. Community topic profiles are used to derive the community's preference score on the target QA document by measuring the content similarity between each topic profile and the document profile of a target QA.

3.3.1. Deriving community preferences based on community-topic profiles

The content-based recommendation method needs to generate an individual or group interest profile to represent its behavior and interest. Therefore, a group interest profile of a knowledge community must be set up in order to recommend suitable QA documents to the knowledge community. After the document pre-processing step, each QA document is represented as a document profile, i.e. a term vector composed of weighted terms. A knowledge community's interest profile can be constructed through the aggregation of the document profiles of the collected knowledge documents within the knowledge community.

In Yahoo! Answers Taiwan, a knowledge community may collect various QA document topics due to the open knowledge sharing platform. Accordingly, a community profile, which is derived by aggregating all the document profiles of the collected QA within the community, may not be appropriate to represent the community's interests. For this reason, we group the collected QA documents of a knowledge community into several topics based on their tags, which are labeled by knowledge collectors. In a knowledge community, each topic is denoted by several tags which only belong to that specific topic; a QA document can be assigned to more than one topic according to its different tags. The process of grouping collected QA documents into topics is shown in Fig. 2.

Community members have different levels of importance in a community. QA documents collected by members with higher importance should be generally more significant in representing a community's interest. Members' reputations in collecting QA knowledge are determined from analyzing the knowledge collection and recommendation (push) interactions between community members. Moreover, community members can answer questions posted by Yahoo users on behalf of their knowledge community; those QA documents whose answers are chosen as best answers are recorded in the community platform. A community member with a greater number of best answers generally has a better reputation.

Community members who consider the collected QA documents to be helpful can push those QA documents in order to express their recommendations. QA documents with higher push scores are generally more important in the representation of the community's interests. In addition, the most recent QA documents collected by a knowledge community can better reflect the current interest of the knowledge community. Hence, the time factor is considered in analyzing the interests of a knowledge community. Accordingly, a community topic profile is derived from a weighted aggregate of document profiles on a topic's collected documents by considering member reputations in collecting QA knowledge and answering questions on behalf of communities, push scores of QA documents as well as the time factors of collected QA documents.

3.3.1.1. Deriving community preferences. The topic relevance score of target QA document q to a topic of community G can be derived by calculating the cosine similarity between q's profile and the community topic profile. A community G's preference score on the target QA q can then be derived as the maximal topic relevance score over all topics of community G. A community topic profile is more appropriate than a community profile (without considering topics) in deriving a community's interests because a knowledge community may collect various QA document topics. However, the diversity of QAs still exists in each topic of a knowledge community due to the open knowledge sharing platform; thus, the recommendation effect may decrease while determining community preferences by considering all QAs of each topic. Accordingly, rather than considering all QAs of each topic, we derive a community G's preference score on a target QA q by considering the top-k QAs in each topic collected by community G that have the highest weighted relevance scores to the target QA q, as shown in Eq. (2). More specifically, let $D_{G,q}^{z,topk}$ be the set of top-k QAs in topic z collected by community G that has the highest weighted relevance scores to the target QA q. The weighted relevance score of a QA d to the target QA q is derived from their cosine similarity multiplied with the QA d's collection weights, including the collection member's reputation, push score of QA d, and the collection time of QA d. The community G's topic-based preference score on target

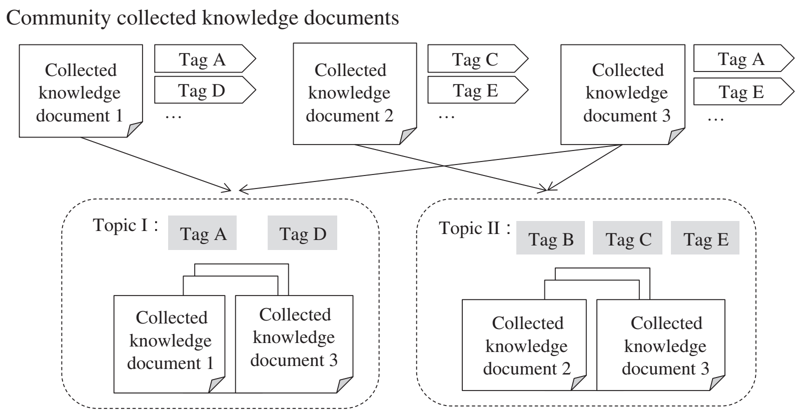

Fig. 2. The process of grouping QA documents into topics.

QA q in topic z, denoted as $TPR^{z}_{G,q}$, is an aggregation of the weighted relevance scores between the target QA and the QAs in $D^{z,topk}_{G,q}$:

$$TPR^{z}_{G,q} = \frac{\sum_{d \in D^{z,topk}_{G,q}} sim(KP_d, KP_q) \times MI^{z:d}_{u_c,G} \times WRec^{z}_{d,G} \times WT_{d,G}}{|D^{z,topk}_{G,q}|}, \quad (2)$$
$$GTPR_{G,q} = Max_z(TPR^{z}_{G,q})$$

where KP_d is the document profile of QA d; $MI^{z:d}_{u_c,G}$ is the importance of member u_c that collected QA d for topic z; $WRec^{z}_{d,G}$ is the push score of d within topic z; and $WT_{d,G}$ is the weight of d's collection time. The details of collection weights, including $MI^{z:d}_{u_c,G}$, $WRec^{z}_{d,G}$ and $WT_{d,G,}$ are described in Section 3.3.2–3.3.4, respectively. The collected QAs with higher collection weights are more important in representing a community's preference. $TPR^{z}_{G,q}$ is high if the collected QAs in $D^{z,topk}_{G,q}$ have high relevance scores related to target QA q and high collection weights. Finally, the community's preference score on the target QA document q, denoted as $GTPR_{G,q}$, is the maximal topic-based preference score over all topics of community G.

3.3.2. Analyzing community members' reputations

The importance of each member in a knowledge community differs due to the diversity of their contributions and participation. Important community members are usually regarded as the leaders of community; they play an important role in collecting QA knowledge and leading the interests of the knowledge community. Therefore, greater weighting will be given to QA documents collected by members with greater importance for deriving community topic profiles. A community member u's importance in topic z, $MI^{z}_{u,G}$ consists of two parts: the reputation of member u for collecting/pushing QA documents in community G, $MCR_{u,G}$, and the reputation of member u for answering questions on topic z on behalf of community G, $MAR^{z}_{u,G}$. The importance of community members is defined, as shown in Eq. (3), which adjusts the relative importance between the member's reputation for collecting QA ($MCR_{u,G}$) and for answering questions ($MAR^{z}_{u,G}$) by parameter α:

$$MI^{z}_{u,G} = \alpha \times MCR_{u,G} + (1-\alpha)MAR^{z}_{u,G} \quad (3)$$

$MCR_{u,G}$ is derived from the link analysis of the knowledge collection and push (recommendation) interactions between community members, while $MAR^{z}_{u,G}$ is derived based on the number of best answers obtained by member u.

Community members can collect QA documents related to their interests for their community. Other community members can push those QA documents in order to express their recommendations. We adopt a link analysis algorithm, PageRank, to calculate members' reputations according to the collect/push relationships among community members. PageRank [33] is used to calculate the importance of each webpage through the link relationships among webpages. On the Internet, if Web Page X is linked to Web Page Y, it means that Web Page X gives Web Page Y one vote. A PageRank algorithm establishes the importance of web pages by computing the PageRank score of a webpage, which is derived from the PageRank scores of those web pages pointing to the web page.

A similar concept can be applied to the push interactions between QA document collectors and recommenders to construct a QA document collection-recommendation network within a knowledge community. Fig. 3(a) illustrates the collection-recommendation interactions within a knowledge community. Fig. 3(b) illustrates the relationships between collectors and recommenders, which are converted from Fig. 3(a). The starting point of an arrow denotes the member who recommends (pushes) the collected QA document, and the ending point of the arrow denotes the member who collects the QA document. For instance, Member 3 recommends the QA document which is collected by Member 1 in Fig. 3(b).

The reputation (importance) of a community member u_c is derived based on the reputations (importance) of members who recommend QA documents collected by u_c. According to the QA document collection-recommendation network constructed above, a member u_c's reputation on collecting QAs, $MCR'_{u_c,G}$, is calculated through the modified PageRank algorithm, as shown in Eq. (4):

$$MCR'_{u_c,G} = \beta \times \sum_{u_r : u_r \to u_c} \frac{MCR'_{u_r,G}}{Outlink(u_r)} + (1-\beta) \times \frac{1}{N} \quad (4)$$

where $u_r \to u_c$ indicates that member u_r recommends QA documents collected by member u_c; $Outlink(u_r)$ is the number of community members whose collected QAs are recommended by u_r; N is the number of community members within a knowledge community and the parameter β is a damping factor, generally set to be 0.85.

After obtaining each member's reputation on collecting QAs, the reputation value is normalized into a value between 0 and 1, as shown in Eq. (5):

$$MCR_{u_c,G} = \frac{MCR'_{u_c,G}}{Max_{u_r} MCR'_{u_r,G}} \quad (5)$$

We also consider the reputation of member u for answering questions on topic z on behalf of community G, $MAR^{z}_{u,G}$, which is a normalized $GBestAns^{z}_{u,G}$, the number of best answers obtained by member u on topic z for knowledge community G. The higher the $GBestAns^{z}_{u,G}$, the higher the answering reputation of member u, as shown in Eq. (6):

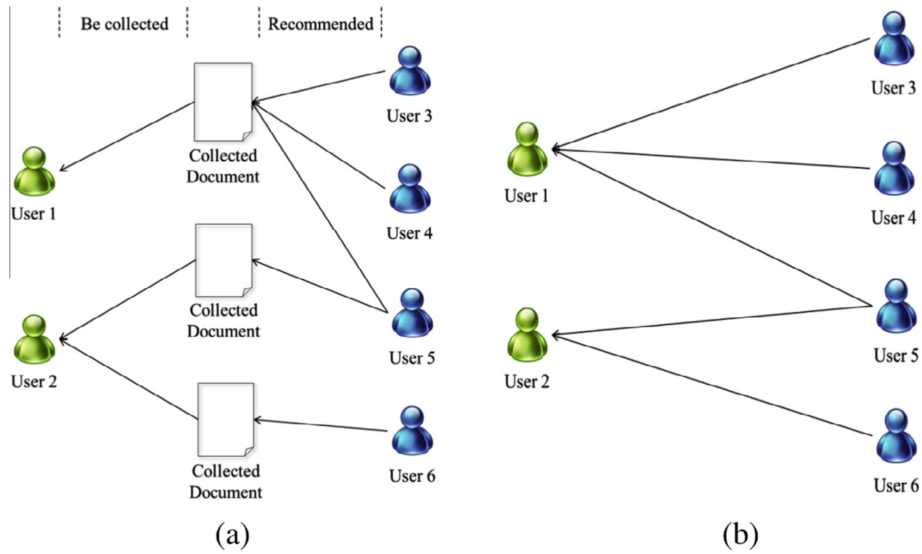

Fig. 3. The collection and recommendation behavior within a knowledge community.

$$MAR^z_{u,G} = \frac{GBestAns^z_{u,G}}{Max_{u_r} GBestAns^z_{u_r,G}} \tag{6}$$

3.3.3. Analyzing the push scores of QA documents

Community members who deem the collected QA documents to be helpful can push those QA documents in order to express their recommendations. QA documents with higher push scores are generally more vital in representing the community's interests. QA documents pushed by community members with greater importance hold more importance and should generally have higher push scores for the knowledge community. In addition, the number of community members who recommend the QA document is another factor that should be considered, because the more members who recommend the QA, the greater the interest of the community in the QA. Hence, a QA document with a higher number of recommendations will be given a higher push score. In this study, the push score of a QA document d in topic z of community G, $WRec^z_{d,G}$ is shown in Eq. (7). $MI^z_{u_r,G}$ signifies the importance of recommender u_r in topic z of knowledge community G; $UR^z_{d,G}$ is the set of members who recommend the collected QA document d in topic z of knowledge community G. In order to avoid a zero push score in the case where there are no recommendations of the collected QA document, the push score has a value of 1 added:

$$WRec^z_{d,G} = 1 + \left[\frac{\sum_{u_r \in UR_{d,G}} MI^z_{u_r,G}}{|UR^z_{d,G}|} \times \left(1 - \frac{1}{|UR^z_{d,G}| + 1}\right)\right] \tag{7}$$

3.3.4. Analyzing the time factors of collected QA documents

The interests of a knowledge community may change as time goes on. The more recent QA documents collected by a knowledge community can better reflect the current interest of the knowledge community. Hence, the time factor is considered in analyzing the interests of a knowledge community. Each QA document is assigned a time weight according to the time of collection, with higher time weights being given to QAs collected most recently. Conversely, older QAs are given lower time weights. The time weight of a QA document d collected by community G, $WT_{d,G}$ is defined in Eq. (8). We adopt the formula given in [52] to compute our time factor:

$$WT_{d,G} = \frac{1}{e^{\tau(t_{now} - t_{d,G})}} \tag{8}$$

where t_{now} is the current time (present date); $t_{d,G}$ is the collection time (date) of QA document d by community G; and τ is a tunable parameter which is set to be 1/365 (a period of one year) to avoid the time-lag from dropping too fast.

3.4. Complementary analysis and recommendations of complementary QAs

In this section, we present our proposed approach for the recommendation of QAs to communities by considering complementary QAs. In a previous work [22], we have proposed a complementary QA analysis approach for the retrieval of complementary QAs without considering the recommendation of QAs to communities of QA Websites. In Sections 3.4.1 and 3.4.2, we first describe the approach for determining the complementary QAs based on question similarity, answer novelty and answer correlation. Then, we describe the proposed approach for recommending complementary QAs to communities of QA Websites in Section 3.4.3.

3.4.1. Notions of partial and extended complementation

The complementary relationships among QA documents include partial complementation and extended complementation. Community members normally use the "keyword search" function in the search engines of QA systems to browse and find questions and answers (QAs) of interest. As some questions in a system are related, community members often wish to collect the QAs of related questions. The information provided in the answer part of a collected QA may be partial and incomplete, so the community may wish to search for related QAs to get complete information. However, the information in some related QAs may be redundant to the collected QA and of no interest to the community. QAs that provide related information that is not redundant are called *partially complementary QAs* of a collected QA. Moreover, some information in the collected QA's answer may not be clear, so the community may wish to conduct an extended search by using keywords in the original answer to search for related QAs that contain extended complementary information. Such QAs are called *extended complementary QAs* of a collected QA.

More specifically, a partial complementary QA provides a different perspective on the answer part of a collected QA; thus, it supplements the original answer by making up for insufficient information in the collected QA. On the other hand, an extended complementary QA provides further information that clarifies some aspect of the original answer. It contains extra information that extends and improves the original answer; thus, it is an extended complement of the collected QA. Community members often wish to collect QA documents that are complementary to the collected QA documents of the community.

Given two QAs, suppose one is called the Collected QA and the other is called a Target QA. In view of the rationale for partial complementation described above, the complementary degree between a collected QA and a target QA is measured through the cosine similarity of a question and answer between the two QAs. We use the cosine similarity measure to determine the degree of similarity between the question of a collected QA and the question of a target QA. If the question similarity is high, the questions of the two QAs are related, so we analyze their answers to derive each answer's novelty.

Let A_d and A_k denote the answers of the Collected QA d and the Target QA q, respectively. We measure the novelty of the two answers, A_d and A_q by Eq. (9), which refers to [22]. We use the term vectors generated by TF-IDF to measure the cosine similarity between the answers of the two QAs. If the similarity is high, this means that the answers contain a lot of common information, so their novelty is low:

$$Nov(A_q, A_d) = 1 - sim(A_q, A_d) \tag{9}$$

If the question similarity score is high, this implies that the two questions are related; and if the answers are not redundant, i.e. the answer novelty score is high, partial complementation is inferred.

On the other hand, if the question similarity is low, the two questions are different; thus, we have to check to see if any term appears in both the answer of the collected QA and the question of the target QA. If such a term exists, we consider that the target QA may contain some information that can explain the unknown subject (term) in the collected QA's answer. However, the answers of the two QAs may be redundant or unrelated, so we have to check the answer novelty and correlation between the collected QA and the target QA. The answer correlation is measured by the correlation of terms in the answers of the two QAs. Extended complementation generally can be inferred if the answer novelty and answer correlation are high.

We use the all-confidence metric [32], which measures the mutual dependence of two variables, to derive the answer correlation, as shown in Eq. (10). The higher the value of all-confidence (x, y), the closer the association of x and y. The correlation between the two answers, A_d and A_q, denoted by $AC(A_d, A_q)$, is derived by summing the all-confidence (x, y) scores for $x \in S_d^A$ and $y \in S_q^A$. Note, that S_d^A/S_q^A is the term set of A_d/A_q:

$$AC(A_d, A_q) = \sum_{x \in S_d^A} \sum_{y \in S_q^A} \frac{P(x \wedge y)}{MAX(P(x), P(y))} \tag{10}$$

where x/y is the term contained in the answer for document q/d; $P(x)$ is the probability of documents containing term x, and $P(x \wedge y)$ is the probability of documents containing both terms x and y. In this study, the dependence of two terms is measured by the number of documents which contain the two terms returned by the Google search engine. Thus, the number of hits of term x returned by the search engine is denoted by $hits(x)$. In a similar way, the number of hits containing both term x and term y is denoted by $hits(x \wedge y)$. Eq. (10) can be rewritten as below:

$$AC(A_d, A_q) = \sum_{x \in S_d^A} \sum_{y \in S_q^A} \frac{hits(x \wedge y)}{MAX(hits(x), hits(y))} \tag{11}$$

3.4.2. Determination of the complementary relationship and score

The complementary relationships among QA documents is hard to judge due to the complex situations in a QA website, even though the measures of complementary relationships, such as question similarity, answer novelty and answer correlation, have been defined above. In addition, the thresholds of these measures are difficult to decide. Accordingly, we use a decision tree classification approach to build a classification model and predict the complementary relationships among QAs [22]. Decision tree learning is widely used in the data mining field because it is easy to understand and interpret, and it can handle numerical and categorical data. We use the decision tree classification approach to build a model that can predict the complementary relationship between two QAs based on three input variables: question similarity, answer novelty and answer correlation. Specifically, we use Weka's Classification and Regression Tree (CART) model [47] to build a classification model. CART can handle numerical input variables, and it builds decision trees based on the Gini Index. We use CART because our input variables are numerical, and the predicted complementary type is categorical.

We train partial and extended complementary classification models separately, as their input variables differ. The complementary relationship of a target QA and a collected QA can be determined by the classification models. The process used to identify the complementary relationship based on the classification models is shown in Fig. 4(a) and (b), respectively.

Besides predicting the complementary relationship between collected and target QAs, the complementary score, i.e. the probability of the complementary relationship, is also measured via the decision tree. For a target QA predicted as a partial or extended complementation to a collected QA, we further calculate the complementary score of the target QA to the collected QA. In the prediction of a target QA through the classification tree models, the decision process reaches a leaf node of the classification tree based on question similarity, answer novelty and answer correlation between two QAs. The complementary score of target QA, q, to the collected QA, d, $CPS_{q,d}$, is the partial or extended probability which can be calculated as the ratio of the number of training cases in the leaf node with a positive label to the total number of training cases in the leaf node.

3.4.3. Recommendations considering complementary QAs

The proposed method recommends complementary QA documents to a knowledge community by considering the complementary scores of target QA documents that are measured based on question similarity, answer novelty and answer correlation. Two approaches are discussed, a QA-based complementary approach and topic-based complementary approach for recommending complementary QAs. The QA-based complementary approach derives community preference by using the individual relevance score and complementary score of the target QA to each collected QA, while the topic-based complementary approach derives community preference by considering the relevance scores of target QAs to collected QAs and the complementary scores of target QAs to topics.

3.4.3.1. QA-based complementary approach. As described in Section 3.3, we group the collected QA documents into several topics and then derive community-topic preferences from top-k QAs collected under topics by considering the factors, such as members' reputations in collecting and answering QAs, push scores of QAs and the collection time of QAs, as defined in Eq. (2). The community preference analysis needs to be adjusted by considering the complementary scores of the target QA to some QAs collected for a topic.

The complementary relationship and score are determined for each pair containing a target QA and a collected QA, as presented

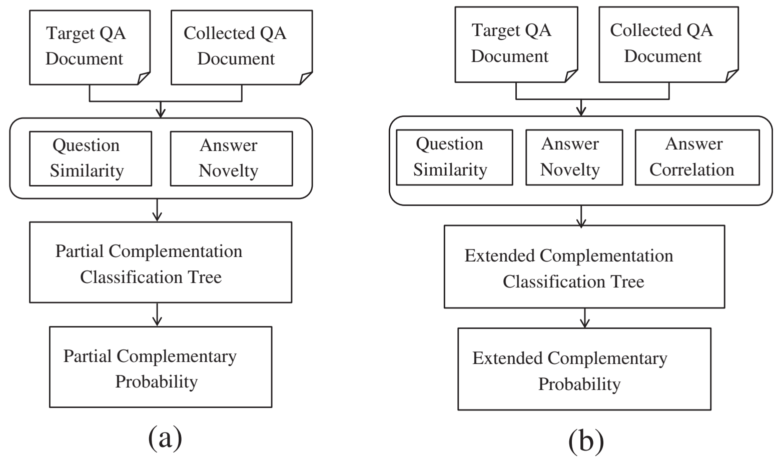

Fig. 4. The model for determining complementary QAs and complementary scores.

in previous Sections 3.4.1 and 3.4.2. For each QA collected for a specific community topic, we determine whether the target QA is a partial or extended complementary to the collected QA by using the classification models described in Section 3.4.2. For each pair containing a target QA and a collected QA predicted as a partial or extended complement, we further calculate the complementary score of the target QA to the collected QA. A target QA may be complementary to zero or some QAs collected under a specific community topic.

The complementary score of a target QA related to a collected QA should be regarded as a complementary weight when the relevance scores of target QAs and collected QAs are aggregated. This determines the preference scores of the community on target QAs, as defined in Eq. (12), which is modified from Eq. (2). More specifically, the relevance scores are weighted by considering the collection weights, including members' reputations, push scores of QAs, collection time of QAs and the complementary scores of target QAs related to collected QAs for topic z. The weighted relevance score of a collected QA d to the target QA q is derived from their cosine similarity multiplied with the QA d's collection weights and the complementary score of target QA q to QA d. The community G's preference score on target QA, q, in topic z, denoted as $TPRC^z_{G,q}$, is an aggregation of the weighted relevance scores between the target QA and the top-k collected QAs with highest weighted relevance scores:

$$TPRC^{z,QA}_{G,q} = \frac{\sum_{d\in D^{z,topk}_{G,q}} sim(KP_d,KP_q)\times MI^{z:d}_{u_c,G}\times WRec^z_{d,G}\times WT_{d,G}\times(1+CPS_{q,d})}{|D^{z,topk}_{G,q}|},$$
$$GTPRC^{QA}_{G,q} = Max_z(TPRC^{z,QA}_{G,q}) \qquad (12)$$

where $D^{z,topk}_{G,q}$ is the set of top-k QAs in topic z collected by community G that has the highest weighted relevance scores related to the target QA q; KP_d is the document profile of QA d; $MI^{z:d}_{u_c,G}$ is the importance of member u_c that collected QA d for topic z; $WRec^z_{d,G}$ is the push score of d within topic z; $WT_{d,G}$ is the weight of d's collection time and $CPS_{q,d}$ is the complementary score of target QA q to collected QA d. We note that target QA may not have a complementary relationship with a collected QA; thus, we use $(1 + CPS_{q,d})$ as the complementary weight in Eq. (12).

Finally, the community's preference score for the target QA document q by the QA-based complementary approach, denoted as $GPRC^{QA}_{G,q}$, is the maximal topic-based preference score over all topics of community G, as shown in Eq. (12).

3.4.3.2. Topic-based complementary approach. Eq. (12) derives the community G's preference score on the target QA by considering the individual complementary score of a target QA to a collected QA. Generally, a target QA may be complementary to very few QAs collected in a community. Thus, there may be very little or no effect when considering individual complementary scores in deriving the preference of the community on a target QA through averaging the weighted relevance scores between a target QA and every collected QA by Eq. (12) when the target QA is only complementary to one or very few collected QAs. Therefore, in order to enhance the effect of complementary QA recommendations, we derive the complementary score of target QA q to a topic z of a community G, $CPS^z_{G,q}$ by aggregating the complementary scores of a target QA to the QAs collected in z, as shown in Eq. (13):

$$CPS^z_{G,q} = f_{d\in D^z_G}(CPS_{q,d}) \qquad (13)$$

where D^z_G is the set of QAs in topic z collected by community G; $CPS_{q,d}$ is the complementary score of target QA q to collected QA d; and $f()$ is an aggregation function, such as the average, max or sum of the complementary scores of target QA to the QAs collected in z, that can be used to determine the complementary score of target QA to a topic. In the experiment, the max function is applied for measuring the complementary score.

Once the complementary score of target QA q to a topic z $CPS^z_{G,q}$ is derived, we modify Eq. (12) by considering $CPS^z_{G,q}$, instead of $CPS_{q,d}$, to enhance the effect of recommending complementary QAs in deriving community G's preference score on target QA q, $GPRC^{topic}_{G,q}$, by the topic-based complementary approach, as shown in Eq. (14):

$$TPRC^{z,topic}_{G,q} = \frac{\sum_{d\in D^{z,topk}_{G,q}} sim(KP_d,KP_q)\times MI^{z:d}_{u_c,G}\times WRec^z_{d,G}\times WT_{d,G}\times(1+CPS^z_{G,q})}{|D^{z,topk}_{G,q}|},$$
$$GTPRC^{topic}_{G,q} = Max_z(TPRC^{z,topic}_{G,q}) \qquad (14)$$

where $CPS^z_{G,q}$ is the complementary score of target QA q to a topic z collected by community G (derived by Eq. (13)), and KP_q is the document profile of QA document q.

Community G's preference score on target QA q of topic z is obtained by multiplying the two factors: the weighted relevance scores of target QA q to top-k QAs in topic z and the complementary score of target QA to topic z. Finally, community G's preference score on the target QA document q, $GTPRC^{topic}_{G,q}$, is the maximal topic-based preference score over all topics of community G. We enhance the effect of the complementary QA recommendation by using the *Max* function to derive complementary scores of target QA to topics. The topic-based complementary approach is expected to improve the recommendation performance for recommending complementary QA documents. Target QA documents, with higher preference scores, generally are more relevant and complementary to the collected QAs of a knowledge community. QA documents with high preference scores are used to compile a recommendation list, from which the top-N QAs are chosen and recommended to the target knowledge community.

4. Experimental evaluations

In this section, we evaluate the performance of the proposed approach by using the QA documents collected in knowledge communities at Yahoo! Answers Taiwan. We evaluate the performance of factors mentioned in Section 3 in four parts. The first part mainly analyzes the effect of recommending documents to a community via utilizing community topic profiles. Next we assess the effect of recommending documents with QA collection factors. Then, we observe the effect of recommending complementary QA documents to knowledge communities. Details on the experiments are described in the following sections.

4.1. Data collection

Experimental data are collected from Yahoo! Answers Taiwan, a knowledge sharing platform which provides knowledge community services. The members of a knowledge community can collect the useful and relevant QA documents. In this work, we choose 15 knowledge communities from a platform which consists of three domains: computer, medicine and finance. The data set contains four parts, including 14,924 collected QA documents, 6349 community members, 50,741 best answers obtained by community members and 36,047 push records of community members.

4.2. Evaluation metrics

The performance metrics, precision, recall, and F1 [8,10,38], are used to evaluate the performance of the proposed recommendation approach. These metrics have been widely used in information retrieval. Precision is determined by the fraction of the recommended articles that are actually found to be interesting for knowledge communities. Recall involves the fraction of the interesting articles that are correctly identified by the recommendation process. The equations of precision and recall are presented in the following formulas:

$$\text{Precision} = \frac{\text{number of correctly recommended QA documents}}{\text{number of recommended QA documents}}$$

$$\text{Recall} = \frac{\text{number of correctly recommended QA documents}}{\text{number of collected QA documents}}$$

F-measure considers both precision and recall to compute the score for balancing the trade-off between precision and recall. More specifically, the F1-measure involves the harmonic means of precision and recall as described in the following formula:

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.3. Experiment setup

4.3.1. Experiment design

We divide the data set into training data and testing data. The data from each community is separated into two parts: 80% for training data and 20% for testing data. In the experimental evaluation, top N QA documents will be recommended to each knowledge community. The precision, recall and F1 measures are used to evaluate the recommendation performances according to the different numbers of recommended QA documents. The testing data will be utilized to evaluate the effectiveness of our proposed approach by comparison with traditional methods.

Our proposed approach recommends complementary QAs based on three measures: question similarity, answer novelty and answer correlation between QAs. However, for deriving answer correlations between QA documents, a great deal of effort is needed to acquire the hits of terms by a search engine. Hence, at most, 300 pairs of QA documents in the training set of each knowledge community are chosen for constructing the partial and extended complementation classification model, respectively. Moreover, the top 5 terms with the highest TF-IDF values for each chosen QA document are used to calculate answer correlations.

4.3.2. Methods compared in the experiment

Our proposed methods (*GTPR*, *GTPRC-QA* and *GTPRC-T*) are compared with the traditional content-based group recommendation methods (*GP-CB* and *GPT*). The content-based group recommendation method consolidates individual profiles to generate group profiles, which in turn are used to filter out items of recommendation, without considering several factors from the recommended items. Based on the content-based group recommendation method, our proposed methods further consider community topics, the reputations of community members, push scores of QAs, collection times of QAs and the complementary scores of QAs. For various methods, QA documents with high recommendation scores are used to compile a recommendation list, from which the top-N QAs are chosen and recommended to the target community. The methods compared in the experiments are briefly introduced as follows.

The ***traditional content-based group profiling method (GP-CB)*** mainly considers the content similarity between the recommended document and the community profile in order to recommend related QA documents to the community without considering community topics and QA collection weights.

The ***group topic preference profiling method (GPT)*** constructs the community topic profiles from the profiles of QAs in topics. The *GPT* recommends QA documents to the community based on the relevance of target QA to the community-topic profiles without considering QA collection weights (e.g. reputations, push scores and collection time).

As described in Section 3.3, the diversity of QAs still exists in each topic of a knowledge community; thus, rather than considering all QAs of each topic, a community G's preference score on a target QA q is derived by considering the top-k QAs in each topic collected by community G that have highest relevance scores regarding to the target QA q. In a community topic profile $TP^z_{G,q} = \sum_{d \in D^{z,topk}_{G,q}} KP_d \Big/ |D^{z,topk}_{G,q}|$, $D^{z,topk}_{G,q}$ is the set of top-k QAs in topic z collected by community G that is most similar to the target QA q; KP_d is the document profile of QA document d. The community's preference score on the target QA document q is the maximal topic relevance score over all topics of community G, i.e. $Max_z(sim(TP^z_{G,q}, KP_q))$. Note that if top-$k$ is large enough, then all collected QAs in each topic are used to generate community-topic profiles and derive community preferences.

The ***group topic-based preference by weighted relevance analysis (GTPR)*** method constructs the community topic profiles from the profiles of QAs in topics by considering QA collection weights, including: the reputations of community members, push scores and the collection time of the QAs. *GTPR* uses QA collection weights to derive weighted relevance scores and derive a community G's preference score from top-k QAs in each topic by using Eq. (2), as described in Section 3.3.

The ***group topic-based preference by weighted relevance and complementation analysis (GTPRC)*** method recommends QA documents to the community by not only considering the relevance of QAs and QA collection weights, but also the complementary scores of QAs as described in Section 3.4. Two methods: the QA-based complementary method (*GTPRC-QA*) and the topic-based complementary method (*GTPRC-T*) are proposed for recommending complementary QAs. The *GTPRC-QA* uses Eq. (12) to derive community preferences by using the individual weighted relevance score and complementary score of the target QA to each collected QA. The *GTPRC-T* method uses Eq. (14) to derive community preferences by considering the relevance scores of target QAs to collected QAs and the complementary scores of target QAs to topics.

4.4. Experimental results

4.4.1. Effects of selecting top-k QAs for deriving community preferences

As described in Section 3.3, the diversity of QAs may exist in each topic of a knowledge community; thus, a community *G*'s preference is derived by considering the top-*k* QAs, rather than all QAs of each topic. In this experiment, we determine the value of *k*, which varies from 5 to 100, for selecting the top-*k* collected QAs to derive community preferences. The average F1 value, calculated over various top-*N* (top 30 ∼ top 120) recommendations, is used to measure the recommendation quality. The performance of *GPT-topk* method is used to evaluate the effect of selecting top-*k* QAs and community topics by comparison with the *GP-CB* method. The experiment result is shown in Fig. 5, where the "*GPT-All*" indicates the result of *GPT* that considers all collected QAs in each topic.

Fig. 5 shows that the recommendation quality of *GPT-All* is better than that of the traditional *GP-CB* method. The result implies that a community topic profile is indeed more appropriate than a community profile without considering topics, in deriving a community's interests because a knowledge community may collect various QA document topics. Moreover, the recommendation quality is the best when *k* equals 10. The recommendation quality will drop when the value of *k* is more than 10. It implies that considering top-*k* QAs performs better than considering all QAs in deriving community preferences since the diversity of QAs may still exist in each topic of a knowledge community. Based on the result, we choose top-10 QAs of each topic to derive community preferences for the *GPT* method and our proposed methods, including *GTPR* and *GTPRC* in the rest of our experiments.

4.4.2. Parameter settings for deriving the importance of community members

The importance of community members is defined by Eq. (3) which, by parameter α, adjusts the relative importance between the member's reputation of collecting QAs (MCR) and the member's reputation for answering questions (MAR). In this experiment, we determine the parameter setting for α. To determine which α value will result in the highest recommendation quality, we use the *GTPR* method (Eq. (2)) to make recommendations by varying the value of α from 0 to 1 with increments of 0.1. The average F1 value, calculated over various top-N (top 30 ∼ top 120) recommendations, is used to measure the recommendation quality.

Fig. 6 shows that the recommendation quality is the best when the value of parameter α equals 0.2. Note, that member importance solely depends on the MCR when $\alpha = 1$, while member importance solely depends on the MAR when $\alpha = 0$. The results suggest that the members' reputations for answering QAs are more important than the members' reputations for collecting QAs. In the rest of our experiments, the value of parameter α is set at 0.2, as it produces better recommendation results.

4.4.3. Effects of community-topic preferences

This experiment evaluates the effect of deriving community preferences based on community-topics by comparing the *GPT* method, *GTPR* method and the traditional content-based group profiling method (*GP-CB*). The *GPT* recommends QAs based on top-*k* (top-10) QAs in topics that are most relevant to target QA without considering QA collection weights. The *GTPR* uses QA collection weights, including the reputations of community members, push scores and collection time of QAs, to derive weighted relevance scores and community *G*'s preference score from top-*k* QAs in each topic. The *GP-CB* method does not consider the topics and the three QA collection factors. In this experiment, 80% of the data set is used to analyze the factors and derive the community preference profiles, and 20% of the data set (testing set) is used to evaluate the recommendation's performance.

Fig. 7 shows the F1 values of three methods under various top-*N* recommendations. The result shows that the recommendation quality of *GPT* is better than that of the traditional *GP-CB* method. The *GTPR* method performs better than the *GPT* and the *GP-CB* methods. Considering the topic profiles, the QA collection factors can achieve better recommendation performance than the traditional content-based group profiling method. The result implies that the interests of a community can be more precisely represented by generating difference topic profiles and considering QA collection factors, including the reputations of community members, push scores and collection time of QAs.

4.4.4. Effects of recommending complementary QAs

In this section, we compare the performance of our proposed approaches for group recommendation in knowledge community including *GTPR*, *GTPRC-QA*, and *GTPRC-T*. We evaluate the effects of recommendations by considering partial and extended complementary QAs. Our proposed approach recommends complementary

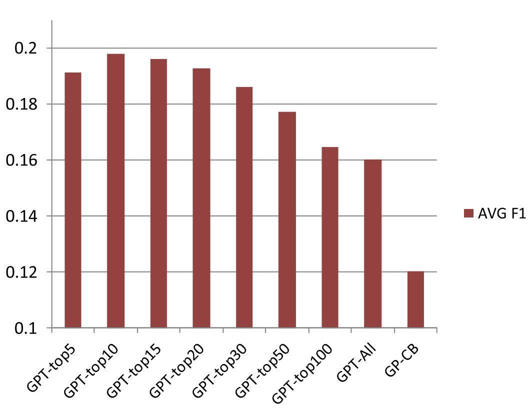

Fig. 5. F1 measures of selecting top-k QAs for deriving community preferences.

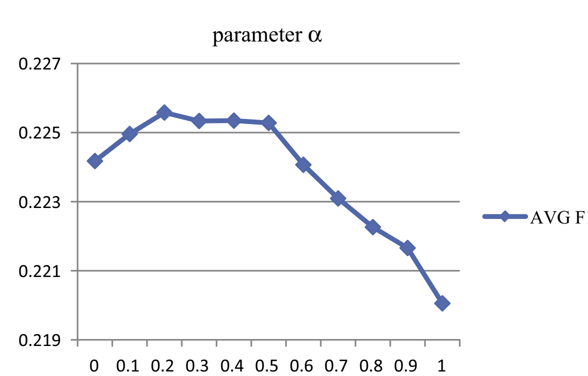

Fig. 6. Determining the parameter for measuring member importance.

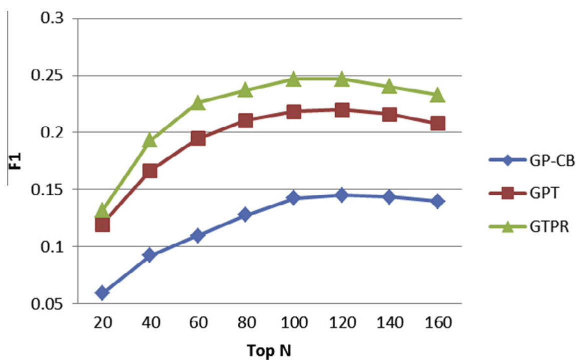

Fig. 7. F1 measures of *GTPR*, *GPT* and *GP-CB* methods under various top-N recommendations.

QAs based on three measures: question similarity, answer novelty, and answer correlation between the QAs. At most, 300 pairs of QA documents in the training set from each knowledge community are chosen for constructing the partial and extended complementation classification model, respectively. The models are then used to predict the complementary scores of target QA to QAs collected in the community.

The proposed ***GTPR (group topic-based preference by weighted relevance analysis)*** method constructs the community topic profiles from the profiles of QAs in topics by considering the relevance of QAs and QA collection weights, including the reputations of community members, push scores and the collection time of the QAs. The proposed ***group topic-based preference by weighted relevance and complementation analysis (GTPRC)*** method recommends QA documents to the community by not only considering the relevance of QAs and QA collection weights, but also the complementary scores of QAs as described in Section 3.4. In this section, we evaluate the QA-based complementary method (*GTPRC-QA*), which uses Eq. (12) to derive community preferences by using an individual weighted relevance score and complementary score of the target QA to each collected QA. We also evaluate the topic-based complementary method (*GTPRC-T*), which uses Eq. (14) to derive community preferences by considering the weighted relevance scores of target QAs to collected QAs and the complementary scores of target QAs to topics.

Table 1 shows the F1 measures of the recommendation results using the *GTPR*, *GTPRC-QA* and *GTPRC-T* methods. Generally, the *GTRPC-QA* performs slightly better than the *GTPR*. The *GTRPC-T* performs better than the *GTPR*. Moreover, the topic-based complementary approach (*GTPRC-T*) performs better than the QA-based complementary approach (*GTPRC-QA*). The results imply that considering complementary QAs helps to improve the recommendation quality. Our methods boost the community's preference scores on a target QA if the QA is a partial or extended complementary one to those QAs collected in the community. Moreover, some relevant QAs may be redundant to the QAs collected in the community since Yahoo! Answers is an open platform that allows users to ask or answer questions freely. Using the classification model, those relevant but redundant target QAs will be predicted as non-partial complementation or having low complementary scores; their preference scores can thus be decreased with the chance of being filtered out of the recommendation list.

Although the recommendation performance does improve after taking into account the complementation factor, the improvement is only slight for the QA-based complementary approach (*GTPRC-QA*). Generally, a target QA is complementary to very few QAs collected in a community. Thus, there may be very little or no effect when considering individual complementary scores in deriving the preference of the community on a target QA through averaging the weighted relevance scores between the target QA and collected QAs using Eq. (12) when the target QA is only complementary to one or very few collected QAs. The topic-based complementary approach (*GTPRC-T*) enhances the effect of a complementary QA recommendation based on the comparison of complementary scores of target QAs to topics, rather than the complementary score of an individual QA. The complementary effect of a target QA is boosted by using the *max* function to derive the complementary score of a target QA to a topic as the maximum of the complementary scores for a target QA to the QAs collected for that topic. Accordingly, the topic-based complementary approach (*GTPRC-T*) performs better than the QA-based complementary approach (*GTPRC-QA*).

4.4.5. Comparison of various recommendation methods

We have proposed three novel approaches for group recommendation in a knowledge community: *GTPR*, *GTPRC-QA* and *GTPRC-T*. The proposed *GTPR (group topic-based preference by weighted relevance analysis)* method constructs the community topic profiles from the profiles of QAs in topics by considering the relevance of QAs and QA collection weights. The *GTPRC (group topic-based preference by weighted relevance and complementation analysis)* method recommends QA documents to the community by not only considering the relevance of QAs and QA collection weights, but also the complementary scores of QAs. The QA-based complementary method (*GTPRC-QA*) derives community preferences by using the individual weighted relevance score and complementary score of the target QA to each collected QA. The topic-based complementary method (*GTPRC-T*) derives community preferences by considering the relevance scores of target QAs to collected QAs and the complementary scores of target QAs to topics. Tables 2 and 3 show the performance comparison (F1, Precision and Recall measures) among various recommendation methods: the *GP-CB*, *GPT*, *GTPR*, *GTPRC-QA* and *GTPRC-T* methods. The results show that the *GTPRC-T* performs the best among all the methods. The recommendation quality is improved when we consider the complementary scores of the target QAs. The F1, Precision and

Table 1
Comparisons of recommending complementary QAs.

	GTPR	*GTPRC-QA*	*GTPRC-T*
Top 20	0.1316	0.1327	0.1447
Top 40	0.1930	0.1940	0.2090
Top 60	0.2261	0.2277	0.2376
Top 80	0.2372	0.2372	0.2474
Top 100	0.2467	0.2487	0.2566
Top 120	0.2467	0.2485	0.2551
Top 140	0.2401	0.2428	0.2472
Top 160	0.2329	0.2355	0.2385

Table 2
F1 measures of various recommendation methods.

	GP-CB	*GPT*	*GTPR*	*GTPRC-QA*	*GTPRC-T*
Top 5	0.0188	0.0475	0.0525	0.0538	0.0563
Top 7	0.0258	0.0638	0.0626	0.0638	0.0700
Top 10	0.0370	0.0777	0.0860	0.0872	0.0956
Top 20	0.0592	0.1195	0.1316	0.1327	0.1447
Top 40	0.0923	0.1667	0.1930	0.1940	0.2090
Top 60	0.1097	0.1947	0.2261	0.2277	0.2376
Top 80	0.1278	0.2107	0.2372	0.2372	0.2474
Top 100	0.1422	0.2183	0.2467	0.2487	0.2566
Top 120	0.1450	0.2196	0.2467	0.2485	0.2551
Top 140	0.1435	0.2158	0.2401	0.2428	0.2472
Top 160	0.1397	0.2080	0.2329	0.2355	0.2385

Table 3
Precision and recall measures of various recommendation methods.

	GP-CB		*GPT*		*GTPR*		*GTPRC-QA*		*GTPRC-T*	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
Top 5	0.2000	0.0098	0.5067	0.0249	0.5600	0.0276	0.5733	0.0282	0.6000	0.0295
Top 7	0.2000	0.0138	0.4952	0.0341	0.4857	0.0335	0.4952	0.0341	0.5429	0.0374
Top 10	0.2067	0.0203	0.4333	0.0427	0.4800	0.0472	0.4867	0.0479	0.5333	0.0525
Top 20	0.1800	0.0354	0.3633	0.0715	0.4000	0.0787	0.4033	0.0794	0.4400	0.0866
Top 40	0.1633	0.0643	0.2950	0.1161	0.3417	0.1345	0.3433	0.1352	0.3700	0.1457
Top 60	0.1478	0.0873	0.2622	0.1549	0.3044	0.1798	0.3067	0.1811	0.3200	0.1890
Top 80	0.1450	0.1142	0.2392	0.1883	0.2692	0.2119	0.2692	0.2119	0.2808	0.2211
Top 100	0.1433	0.1411	0.2200	0.2165	0.2487	0.2448	0.2507	0.2467	0.2587	0.2546
Top 120	0.1339	0.1581	0.2028	0.2395	0.2278	0.2690	0.2294	0.2710	0.2356	0.2782
Top 140	0.1238	0.1706	0.1862	0.2566	0.2071	0.2854	0.2095	0.2887	0.2133	0.2940
Top 160	0.1142	0.1798	0.1700	0.2677	0.1904	0.2999	0.1925	0.3031	0.1950	0.3071

Recall measures of our proposed approach, considering community topic profiles with QA collection factors and complementary scores of QAs, are higher than those of the traditional group-based recommendation methods (*GP-CB* and *GPT*).

All our proposed approaches (*GTPR*, *GTPRC-QA* and *GTPRC-T*) perform better than traditional group-based recommendation methods (*GP-CB* and *GPT*). *GP-CB* is a traditional content-based group profiling method, while *GPT* is a traditional group topic preference profiling method. In the comparison of our proposed methods, although *GTRPC-QA* performs only slightly better than *GTPR*, the *GTRPC-T* method can perform better than the *GTPR* method. The topic-based complementary approach enhances the effect of the QA recommendation based on the complementary scores of target QAs to topics, rather than the complementary score of an individual QA. Accordingly, the topic-based complementary approach performs better than the QA-based complementary approach.

Moreover, all our proposed *GTPR*, *GTPRC-QA* and *GTPRC-T* methods can effectively improve traditional group-based recommendation methods (*GP-CB* and *GPT*) in recommending QA documents to knowledge communities. The result implies that the recommendation quality is improved when we consider the topic profiles and QA collection factors, including the reputations of community members, push scores and collection time of QAs. Moreover, the recommendation quality is improved when we consider the topic-based complementary scores of the target QAs.

4.4.6. The reliability and robustness of our topic grouping method

The Web 2.0 websites usually allow members to tag their collected items according to their knowledge demands and cognition. In Yahoo! Answers Taiwan, a knowledge community may collect various QA topics due to the open knowledge sharing platform. The topics and their corresponding tags of each knowledge group (community) are defined and organized by the administrators of each knowledge group. Each topic is denoted by several user defined tags. It is natural to group the collected QA documents of a knowledge community into user-defined topics based on their tags, which are labeled by knowledge collectors. Accordingly, we implement our approach and conduct experiments based on user-defined topics and tags. In this research, the experiment results show that considering topics of documents can improve the recommendation quality.

The reliability and robustness of our topic grouping method based on user defined tags may be affected by: (1) users' tagging quality, i.e. users may not tag documents precisely; (2) administrators' grouping quality of tag classification, i.e. administrators may not appropriately classify tags into topics. It will be interesting to use other approaches such as clustering methods to group documents into topics based on the content of QA documents. However, it is worth noting that the topics generated by automatically clustering documents may differ from the user-defined topics of documents obtained from members' cognition in knowledge groups. In addition, the lengths of QA documents are usually short, and different terms with similar meanings may be used by different users; this results in the difficulty of clustering QA documents into topics. In this work, we focus on designing and evaluating our proposed approach based on the topics and tags defined by the members of each knowledge group. Further study is required to explore the other approach, which can take both user-defined topics/tags and clustering methods for grouping documents into topics. We will evaluate other methods for grouping QA documents into topics in future work.

5. Conclusion and future works

In this research, we propose a comprehensive group-based recommendation approach to solve the issue of information overloading in community question–answer websites. Previous researches did not investigate the recommendation mechanisms for knowledge collections in question–answer website communities. Traditional recommendation methods, which neglect critical factors such as the QA collection weights and the complementary relationships between QAs, may not be suitable for recommending QAs to knowledge communities on QA websites. In addition, recommending complementary QAs is important to increase the effectiveness of knowledge collections. However, there is no research on recommending QA documents to QA website knowledge communities that takes the reputations of community members and complementary relationships between QA documents into consideration.

A novel recommendation approach is proposed on recommending relevant and complementary Q&A documents to knowledge communities of Q&A websites. The novel ideas of our proposed approach are as follows: (1) Our approach generates community topic profiles by considering QA collection factors such as community members' reputations in collecting and answering QAs, push scores of QAs and the collection time of QAs from the historically collected QA documents on specific topics. (2) It predicts the complementary scores of QAs based on question similarity, answer novelty and answer correlation. (3) It proposes a QA-based complementary approach and topic-based complementary approach to recommend complementary QA documents.

Experimental results show that considering partial or extended complementary QAs helps to improve the recommendation quality. Although the recommendation performance is improved after taking into account the complementation factor, the improvement is only slight for the QA-based complementary approach, since the complementary effect is small when the target QA is complementary to only one or very few collected QAs. The topic-based complementary approach enhances the effect of QA recommendation based on the complementary scores of target QAs to topics, rather than the complementary score of an individual QA. Accordingly, the topic-based complementary approach performs better than the QA-based complementary approach. Moreover, our proposed approach, which considers community topic profiles with QA collection factors and complementary scores of QAs, performs better than traditional group-based recommendation methods. Our proposed approach is effective in recommending complementary QA documents to knowledge communities.

A question answering website is an open knowledge platform, where the formation of question answering knowledge is relatively easy. Thus, the quality of QA knowledge varies, and the effectiveness of knowledge sharing will be affected by the collection of QA knowledge of poor quality. In our future work, we will investigate how to assess the quality of QA documents and consider document quality in the recommendation. Moreover, we developed a web crawler to download data from Yahoo! Answers Taiwan. Downloading data from Yahoo is challenging and time consuming. In addition, analyzing complementary QAs takes time to perform the pre-processing of QA documents. Accordingly, our current evaluation is limited by the size of the dataset used in our experiments. More thorough evaluations will be conducted in the future by collecting more data to measure the complementation and quality of QAs. In future work, we plan to cooperate with Yahoo to obtain a large dataset directly from their database instead of crawling data from the Website. Finally, our group recommendation approach is proposed for recommending complementary QA documents for knowledge group, especially for knowledge collection behavior. Yahoo! Answers Taiwan is a knowledge platform that provides group mechanism for users to organize and join groups, as well as to collect QA documents to their groups. To the best of our knowledge, there are very few QA knowledge sharing websites that provide group mechanism for collecting and sharing knowledge documents. In our future work, we will apply and evaluate our approach using other appropriate datasets.

Acknowledgments

This research was supported by the National Science Council of Taiwan under Grant NSC 100-2410-H-009-016.

References

[1] G. Adomavicius, Y. Kwon, Improving aggregate recommendation diversity using ranking-based techniques, IEEE Trans. Knowl. Data Eng. 24 (2012) 896–911.
[2] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (2005) 734–749.
[3] H. Alani, S. Dasmahapatra, K. O'Hara, N. Shadbolt, Identifying communities of practice through ontology network analysis, IEEE Intell. Syst. 18 (2) (2003) 18–25.
[4] L. Ardissono, A. Goy, G. Petrone, M. Segnan, P. Torasso, Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices, Appl. Artif. Intell. 17 (2003) 687–714.
[5] M. Balabanovi, Y. Shoham, Fab: content-based, collaborative recommendation, Commun. ACM 40 (1997) 66–72.
[6] A.B. Barragáns-Martínez, E. Costa-Montenegro, J.C. Burguillo, M. Rey-López, F.A. Mikic-Fonte, A. Peleteiro, A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition, Inf. Sci. 180 (2010) 4290–4311.
[7] C. Basu, H. Hirsh, W. Cohen, Recommendation as classification: using social and content-based information in recommendation, in: Proceedings of the 15th National Conference on Artificial Intelligence, 1998, pp. 714–720.
[8] D. Billsus, M.J. Pazzani, Learning collaborative information filters, in: Proceedings of the Fifteenth International Conference on Machine Learning, 1998, pp. 46–54.
[9] R. Burke, Hybrid recommender systems: survey and experiments, User Model. User-Adap. Inter. 12 (2002) 331–370.
[10] J. Cho, K. Kwon, Y. Park, Collaborative filtering using dual information sources, IEEE Intell. Syst. 22 (2007) 30–38.
[11] M. Gan, R. Jiang, Improving accuracy and diversity of personalized recommendation through power law adjustments of user similarities, Decis. Support Syst. 55 (2013) 811–821.
[12] I. Garcia, L. Sebastia, E. Onaindia, C. Guzman, A group recommender system for tourist activities, in: T. Di Noia, F. Buccafurri (Eds.), Springer, Berlin/Heidelberg, 2009, pp. 26–37.
[13] N. Glance, D. Arregui, M. Dardenne, Knowledge pump: community-centered collaborative filtering, in: Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering, 1998, pp. 83–88.
[14] A. Jameson, More than the sum of its members: challenges for group recommender systems, in: Proceedings of the Working Conference on Advanced Visual Interfaces, Gallipoli, Italy, 2004, pp. 48–54.
[15] A. Jameson, S. Baldes, T. Kleinbauer, Enhancing mutual awareness in group recommender systems, in: Proceedings of the IJCAI, 2003.
[16] P. Jurczyk, E. Agichtein, Discovering authorities in question answer communities by using link analysis, in: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, 2007, pp. 919–922.
[17] J.K. Kim, H.K. Kim, H.Y. Oh, Y.U. Ryu, A group recommendation system for online communities, Int. J. Inf. Manage. 30 (2010) 212–219.
[18] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, J. Riedl, GroupLens: applying collaborative filtering to Usenet news, Commun. ACM 40 (1997) 77–87.
[19] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Comput. 42 (2009) 30–37.
[20] G. Linden, B. Smith, J. York, Amazon.com recommendations: item-to-item collaborative filtering, IEEE Internet Comput. 7 (2003) 76–80.
[21] D.-R. Liu, Y.-H. Chen, W.-C. Kao, H.-W. Wang, Integrating expert profile, reputation and link analysis for expert finding in question–answering websites, Inf. Process. Manage. 49 (2013) 312–329.
[22] D.-R. Liu, Y.-H. Chen, M. Shen, P.-J. Lu, Complementary QA-Network Analysis for QA Retrieval in Social Question-Answering Websites. Journal of the American Society for Information Science and Technology, 2013, accepted for pubilcation.
[23] D.-R. Liu, C.-H. Lai, Y.-T. Chen, Document recommendations based on knowledge flows: a hybrid of personalized and group-based approaches, J. Am. Soc. Inform. Sci. Technol. 63 (2012) 2100–2117.
[24] D.-R. Liu, Y.-Y. Shih, Integrating AHP and data mining for product recommendation based on customer lifetime value, Inform. Manage. 42 (2005) 387–400.
[25] J. Liu, C. Wu, W. Liu, Bayesian probabilistic matrix factorization with social relations and item contents for recommendation, Decis. Support Syst. 55 (2013) 838–850.
[26] X. Liu, W. Croft, M. Koll, Finding experts in community-based question–answering services, in: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, 2005, pp. 315–316.
[27] H. Ma, I. King, M.R.-T. Lyu, Mining web graphs for recommendations, IEEE Trans. Knowl. Data Eng. 24 (2012) 1051–1064.
[28] Q. Ma, K. Tanaka, Topic-structure-based complementary information retrieval and its application, ACM Trans. Asian Lang. Inform. Process. 4 (2005) 475–503.
[29] J.F. McCarthy, T.D. Anagnost, MusicFX: an arbiter of group preferences for computer supported collaborative workouts, in: Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW), Seattle, Washington, United States, 1998, pp. 363–372.
[30] K. McCarthy, L. McGinty, B. Smyth, M. Salamo, Social interaction in the CATS group recommender, in: Workshop on the Social Navigation and Community Based Adaptation Technologies, 2006.
[31] M. O'Connor, D. Cosley, J.A. Konstan, J. Riedl, PolyLens: a recommender system for groups of users, in: Proceedings of the Seventh Conference on European Conference on Computer Supported Cooperative Work, Bonn, Germany, 2001, pp. 199–218.
[32] E.R. Omiecinski, Alternative interest measures for mining associations in databases, IEEE Trans. Knowl. Data Eng. 15 (2003) 57–69.
[33] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, Stanford Digital Library Technologies Project, 1998.
[34] M. Pazzani, D. Billsus, Content-based recommendation systems, in: The Adaptive Web, LNCS, vol. 4321, Springer, Berlin/Heidelber, 2007, pp. 325–341.
[35] L. Quijano-Sanchez, J.A. Recio-Garcia, B. Diaz-Agudo, Personality and social trust in group recommendations, in: Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence, vol. 02, 2010, pp. 121–126.
[36] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, in: Proceedings of ACM Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina, United States, 1994, pp. 175–186.
[37] P. Resnick, H.R. Varian, Recommender systems, Commun. ACM 40 (1997) 56–58.
[38] C.J.V. Rijsbergen, Information retrieval, second ed., Butterworth-Heinemann, London, 1979.
[39] B. Saha, L. Getoor, Group proximity measure for recommending groups in online social networks, in: 2nd ACM SIGKDD Workshop on Social Network Mining and Analysis, 2008.
[40] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Inf. Process. Manage. 24 (1988) 513–523.

[41] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Analysis of recommendation algorithms for e-commerce, in: Proceedings of the 2nd ACM Conference on Electronic Commerce, Minneapolis, Minnesota, United States, 2000, pp. 158–167.
[42] U. Shardanand, P. Maes, Social information filtering: algorithms for automating "word of mouth, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1995, pp. 210–217.
[43] J. Shen, W. Shen, X. Fan, Recommending experts in Q&A communities by weighted HITS algorithm, in: CORD Conference Proceedings, vol. 2, 2009, pp. 151–154.
[44] C. Shin, W. Woo, Socially aware TV program recommender for multiple viewers, IEEE Trans. Consum. Electron. 55 (2009) 927–932.
[45] H. Toba, Z.-Y. Ming, M. Adriani, T.-S. Chua, Discovering high quality answers in community question answering archives using a hierarchy of classifiers, Information Sciences, 2013, Available from: <http://dx.doi.org/10.1016/j.ins.2013.10.030>.
[46] Z. Wang, L. Sun, W. Zhu, S. Yang, H. Li, D. Wu, Joint social and content recommendation for user-generated videos in online social network, IEEE Trans. Multimedia 15 (2013) 698–709.
[47] Weka. Data Mining Software. <http://www.cs.waikato.ac.nz/ml/weka/>.
[48] Yahoo! Answer Taiwan. <http://tw.knowledge.yahoo.com/>.
[49] YahooAPI. Word Segmentation of Yahoo API. <http://tw.developer.yahoo.com/cas/>.
[50] Z. Yu, X. Zhou, Y. Hao, J. Gu, TV program recommendation for multiple viewers based on user profile merging, User Model. User-Adap. Inter. 16 (2006) 63–82.
[51] Z. Yu, X. Zhou, D. Zhang, An adaptive in-vehicle multimedia recommender for group users, in: Proceedings of IEEE 61st Vehicular Technology Conference, 2005, pp. 2800–2804.
[52] J. Zhang, M. Ackerman, L. Adamic, Expertise networks in online communities: structure and algorithms, in: Proceedings of the 16th International Conference on World Wide Web, 2007, pp. 221–230.
[53] J. Zhang, M. Ackerman, L. Adamic, K. Nam, QuME: a mechanism to support expertise finding in online help-seeking communities, in: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, 2007, pp. 111–114.