Study on Learning Resources Recommendation Based on Tasks in Team Collaboration

Lin Gong, Jian Xie, Yang Liu
School of Mechanical Engineering
Beijing Institute of Technology
Beijing, China
e-mail: gonglin@bit.edu.cn

Xiaodan Zhang
Computer Network Information Center
Chinese Academy of Sciences
Beijing, China
e-mail: zhangxiaodan@cstnet.cn

Abstract—Nowadays, learning team has become an important foundation for modern work. In a team, how to recommend learning resources to the appropriate team member according to the task requirements is a key factor of success. This paper firstly reviewed related methods and concepts in knowledge modeling and learning recommendation. Then, it constructed a model for teamwork tasks, knowledge and team members. Based on knowledge background and experience of team members in the project, it recommended learning resources to the members according to the assigned tasks. Finally, the prototype system was built for practical validation.

I. INTRODUCTION

In modern society, lots of research projects need to be completed by teams, which have become the basic form for working. Due to the limitation of knowledge, team members should learn some knowledge to according to task specification in the process of completing. Traditional learning resources are recommended by experienced team members. However, because of multidisciplinary knowledge of teamwork, complexity of tasks and the diversity of learning resources, it is difficult for team members to get the right resources, thereby affecting the completion of task in the team.

Based on research of knowledge engineering, this study constructed models of the task, knowledge, learning resource, team member. Furthermore, the knowledge ability and task ability of team members were calculated and necessary learning resources were recommended to team members, which laid the foundation for completion of team project.

II. RESEARCH STATUS

Many foreign and domestic scholars in this research area have done some researches including various areas. M. Vozalis[1] analyzed resources interesting to the user using content filtering technology, formulated user profiles, compared the similarity between user profiles and resource files and recommended resources with high similarity to the user. K. Wang[2] used Naive Bayesian Method to study Collaborative Filtering Recommendation Approach. AP De.Vries[3] proposed Item-based Collaborative Filtering Approaches, calculated similarity between projects to provide recommendation services to users and effectively raised the recommendation quality. CE Kahn Jr[4] used Collaborative Filtering Approach to study recommendation of Radiology Knowledge Resources. M. Aiken[5] studied task knowledge and task techniques in virtual team and found that task-technology fit mediates between knowledge sharing and team satisfaction. HP Andres[6] studied knowledge sharing in global virtual teams through collaborative engineering analysis; X. Zheng[7] built an Instructional and Collaborative Learning System using content recommendation. Zhang Z[8] studied the recommendation problem in statistical learning based on Ontology-Based collaborative filtering recommendation algorithm.


III. FRAMEWORK OF RESEARCH

Before learning resource recommendation, we should first analyze task and team members to determine knowledge required by tasks, knowledge background and task background of team members. Then learning resources recommendation can be made. Therefore, this study firstly models basic information such as knowledge, team member, task, learning resource. The knowledge ability and task ability are analyzed based on the previous model. Then, learning resource recommendation is finish after task assignment. Overall, the research framework is shown in Fig. 1. Basic information modeling, analysis methods and recommendation algorithm will be introduced later.

IV. BASIC INFORMATION MODELING

A. Knowledge Modeling

Knowledge is the basis of the task allocation. In this paper, knowledge is described as the following:
Knowledge modeling

Team member modeling

Task modeling

Learning resource modeling

Analysis of knowledge ability of team members

Analysis of knowledge ability of team members

Task allocating

Learning resource recommendation

Figure 1 Framework of Research

K = (KID, KName, KDescription, KFather, KSon, KEntire, KPart)

KID: the knowledge identification, which is used to identify the concept of knowledge uniquely.
KName: the name of knowledge concept.
KDescription: the definition and interpretation of the knowledge.
KFather: the collection of father concept of the knowledge.
KSon: the collection of son concept of the knowledge.
KEntire: the collection of upper concept of the knowledge.
KPart: the collection of lower concept of the knowledge.

The relationship between different knowledge will be used for knowledge recommendation, which includes two kinds as follows:

There are two kinds of relationship between knowledge: parent-child and entirety-part relationship. If knowledge \( K_i \) has all characteristics of knowledge \( K_j \), knowledge \( K_j \) does not have all knowledge of knowledge \( K_i \), and instances of knowledge \( K_j \) contains all instances of knowledge \( K_i \), it can be said that knowledge \( K_i \) is the parent of knowledge \( K_j \), and knowledge \( K_j \) is the child of knowledge \( K_i \). The relationship of knowledge \( K_i \) and \( K_j \) is the parent-child relationship. If the union of attributes set and instances set of knowledge \( K_i, K_j, \ldots, K_r \) are equal to attributes set and instances set of knowledge \( K_n \), it means that knowledge \( K_i, K_j, \ldots, K_r \) are parts of knowledge \( K_n \), and the knowledge \( K_n \) is the entirety of knowledge \( K_i, K_j, \ldots, K_r \).

B. Learning Resource Modeling

In this study, we can measure pros and cons of learning resources from two aspects: professional level and difficulty level of learning resources. Mathematical description of learning resources is as follow:

\[
R = (RID, RName, RDescription, RType, RLearnScore, RKnowledgeScore, RK)
\]

RID: the identification of a learning resource
RName: the name of a learning resource.
RDescription: the description of a learning resource.
RType: the type of a learning resource.
RLearnScore: the difficulty level of a learning resource.
RLearnScore is a triple:

\[ RLearnScore = (MID, MLScore) \]

MID: the identification of a team member who have learned this resource.
MLScore: the score of difficulty level from the MID team member.
RKKnowledgeScore: the professional level of a learning resource, which is used to evaluate the accuracy and richness of professional information of this resource.
RKKnowledge: it can record the knowledge points of a resource. RKKnowledge is a triple:

\[ RKKnowledge = (KID, KWeight) \]

KID: the knowledge identification.
KWeight: the weight of knowledge KID, which descripts the importance of a resource.

C. Task Modeling

The following modeling principles should be complied with during tasks modeling: 1). Tasks can’t be split; 2). Tasks can be combined to a project.

Task model is described as following:

\[ T = (TID, TName, TKnowledge) \]

TID: task identification, which is used to identify a task uniquely.
TName: the name of a task.
TKnowledge: knowledge set needed for the task, TKnowledge is a 2-tuple:

\[ TKnowledge = (KID, TImportance) \]

TImportance : The importance of the knowledge to the task.

D. Team Member Modeling

In addition to the basic information of a team member, we need to know the knowledge background and task background of each member. Knowledge background refers to the knowledge which a team member masters and the
level of knowledge mastering. Task background refers to
tasks which a team member has actually participated and
completed, and how well he or she has done. The team
member model is proposed as the following:

\[ M = (MID, MName, MAttribute, MKnowledge, MTask) \]

- **MID**: the identification of a team member, which is used
to identify a team member uniquely.
- **MName**: the name of a team member.
- **MAttribute**: the attributes of a team member, which
records some team members’ personal information. It is an
n-tuple.
- **MKnowledge**: the knowledge background of a team
member, which record the identification of knowledge point
which a team member mastered. It is described as the
following:

\[ MKnowledge = (KID, KExperience, KScore) \]

- **KID**: the knowledge identification.
- **KExperience**: the knowledge experience, which
represents experiences of a team member during using the
knowledge of KID, including published papers and
completed tasks which contain the knowledge of KID.
- **KScore**: the level which represents the team member
masters the knowledge of KID during KExperience.
- **MTask**: the task background of a team member, which
records the tasks which a team member participated in. It is
described as the following:

\[ MTask = (TID, TExperience, TScore) \]

- **TID**: the identification of a task which the team member
participated in.
- **TExperience**: task experience, which records tasks
including task of TID that the team member participated in.
- **TScore**: the level which represents the team member
completes the task of TID during TExperience.

V. ANALYSIS AND ALGORITHM FOR LEARNING RESOURCES RECOMMENDATION

Before resource recommendation, we need to assign
tasks to team members. This process can be done by manual
assignment method or automatic assignment method based
on team members’ experience and tasks. In the tasks
assignment process, the general principle is as follows: the
better \( M_a \) master the knowledge required by task \( T_s \), the
better \( M_a \) can be capable of task \( T_s \); the better \( M_a \) master
task \( T_s \) during task experience, the better \( M_a \) can be capable
of task \( T_s \). However, it can be greatly beneficial for the
development of teams and members that let some team
members undertake some tasks without much experience.

After the completion of assignment, we need to analyze
from the views of tasks and knowledge, and recommend for
team members who undertake the tasks. The main steps are
as follows:

A. Knowledge selecting based on tasks

Firstly, the task ability of team member is calculated.
After team member \( M_a \) completes task \( T_s \), the No. \( u \)
experience is formed. We can score \( TScore_{au} \) for it. The
ability of team member \( M_a \) mastering task \( T_s \):

\[ TS_{au} = \sum_{TID} TScore_{au} \quad (1) \]

If the team member has not completed the assigned task
\( TG \), namely, \( TS_{auTG} = 0 \), all the knowledge of \( TG \) will be
added into the alternative set. If the team member has ever
completed the similar task of assigned \( TG \), we can set a
certain threshold \( \nu \). When \( TS_{auTG} < \nu \), all knowledge of \( TG \)
will be added into the alternative set. When \( TS_{auTG} \geq \nu \), we
don’t need to make recommendations to team members.

B. Matching and Filtering of Task and Knowledge

After the team member \( M_a \) completes a task containing
knowledge \( K_i \) or publishes a paper containing the
knowledge \( K_i \), j-th experience is formed. According to five
levels: (0.9,0.7,0.5,0.3,0.1), we can get the knowledge score
of a team member \( M_a \) to knowledge \( K_i \) as \( KEScore_{aij} \)

The overall knowledge ability of a team member \( M_a \)
mastering knowledge \( K_i \) is as follow:

\[ KS_{ai} = \sum_{KID} KScore_{aiKID} \quad (2) \]

We need to compare the alternative set \( KR \) of current
knowledge with knowledge ability of team members. Steps
are as follows:

According to a certain threshold \( \omega \), when \( KS_{aiKR} < \omega \) ( \( k \)
is the sequence number of knowledge alternative set \( KR \) ) in
\( KR \), the knowledge will be retained. When \( KS_{aiKR} \geq \omega \), the
knowledge will be deleted from the alternative set. Finally,
we get a new \( KR \).

C. Searching Learning Resources

According to alternative set of knowledge, \( KR \), learning
resources containing all knowledge of \( KR \) will be selected,
which can form alternative set \( RR \). Assume learning
resources alternative subset of one knowledge is \( RR_i \).
D. Filtering of Learning Resources Alternative Subset

When learning resources are built, we save the scores of team members to learning resources. According to comprehensive scores of every resource in resource alternative subset, we can recommend learning resources. However, team members are often concentrated in their own interested resources and specialties, which lead to high scores of some special resources. Therefore, we need to sequence learning resources in every resource alternative subset firstly. Then, we select a certain number of resources from each resource alternative set based on the importance of each knowledge to the task.

1) Firstly: difficulty score of each resource $RR_i$ is calculated to sort learing resources in alternative subset. Difficulty score is calculated as follow:

$$ML_{i} = \frac{\sum_{a} ML_{Score_{i}} \cdot KS_{a}}{x_i}$$  (3)

For each $RR_i$, difficulty score is the average of the product of difficulty level $ML_{Score_{i}}$ scored by team member $M_a$ and knowledge ability $KS_{a}$ of the team member. The algorithm takes into account both the difficulty score of learning resources and experience of the team member.

2) According to the knowledge resource model: $KWeight_{i}$ can represents professional score of each resource for the corresponding knowledge.

3) Comprehensive Evaluation:

$$E_{i} = \alpha \times ML_{i} + \beta \times KWeight_{i}$$  (4)

$\alpha, \beta$ respectively are the coefficients of difficulty score and professional score, $\alpha + \beta = 1$.

4) According to comprehensive evaluation, learning resources are sorted in each $RR_i$.

5) Based on importance of knowledge to the task, we select different numbers of resources to add in resource recommendation set $RRe_i$. Selected numbers for different subset are calculated as follows:

$$n_{i} = \left\lceil N \times \frac{\sum_{RR} TImportance_{ER}}{\sum_{RR} TImportance_{ER}} \right\rceil$$  (5)

$\lceil \rceil$ represents rounded up to an integer. $TImportance_{ER}$ represents the importance of knowledge $RR_i$ to the task.

E. Resources Filtering

We can analyze resources frequency in resource alternative set $RR_i$. Namely, we can calculate $f_{RR_i}$ which represents the number of occurrences of each resource. If the resource appears more times, it means that the more knowledge points the resource contains. We can set a threshold value $\gamma$, when $f_{RR_i} \geq \gamma$, the resources will be added into the resource recommendation set $RRe_i$.

F. Resources Recommendation

We can complete the recommendation through recommending results combining of Step 4 and Step 5 to team members.

VI. SYSTEM REALIZATION

Based on the above theories and analytical process, this research develops a prototype system for learning resources recommendation based on tasks in team collaboration. The system is divided into five major subsystems: knowledge information management subsystem, task information management subsystem, team information management subsystem, project information management subsystem, learning resources recommendation subsystem. Figure 2 and Figure 3 are interfaces of the system.

VII. CONCLUSION

The paper studied learning resources recommendation from the perspective of knowledge and task matching. In this study, the main consideration is knowledge and learning resources. It created knowledge library and learning resources library. Also, the models of team member, resources and tasks are presented, which include knowledge information. Then, tasks are allocated and learning resources
are recommended to team members by knowledge matching, which offers help to successful completion of the task. However, the present expression models of knowledge and task are simplified to some extent. We will do more work on this kind of problems later in the research.

ACKNOWLEDGMENT

This work was supported by Beijing Higher Education Young Elite Teacher Project and the Basic Research Foundation of Beijing Institute of Technology.

REFERENCES