IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 9, 2019, accepted May 1, 2019, date of publication May 15, 2019, date of current version May 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916350

A Correlation-Experience-Demand Based Personalized Knowledge Recommendation Approach

XIYAN YIN, BUYUN SHENG, FEIYU ZHAO, XINGGANG WANG, ZHENG XIAO, AND HUI WANG
School of Mechanical and Electronic Engineering, Wuhan University of Technology, Wuhan 430070, China
Hubei Digital Manufacturing Key Laboratory (Wuhan University of Technology), Wuhan 430070, China

Corresponding author: Feiyu Zhao (zhaofeiyu1992@whut.edu.cn)

This work was supported in part by the National Key R&D Program of China under Grant 2016YFB1101701.

ABSTRACT Knowledge recommendation is an important means of knowledge reuse that can improve the efficiency and quality of product design. However, at present, there is no good way to fully consider the personalized demands of designers while ensuring the applicability of the recommendation results. Previous studies have usually been based on the similarity between tasks and knowledge or use collaborative filtering technology to accomplish knowledge recommendation. However, these methods do not consider the personal experience of designers and the characteristics of knowledge. This paper proposes a knowledge recommendation approach that integrates the degree of correlation between knowledge and tasks, the feedback-based personal experience, the collective experience of designers, and the degree of demand for knowledge based on the forgetting curve. A knowledge assistance score is generated based on these factors, and the knowledge recommendation list is obtained by ranking the knowledge in descending order of this score. Finally, the approach is applied to a machine shop layout design task and a computer numerical control (CNC) machine tool's spindle design and bearings selection task. The experimental results on two tasks demonstrate that the proposed approach outperforms three baselines on three ranking oriented evaluation metrics. This approach can effectively shorten the time for designers to acquire knowledge by recommending applicable knowledge to assist designers in completing design tasks with high quality and efficiency.

INDEX TERMS Collaborative filtering, degree of assistance, degree of correlation, degree of demand, knowledge recommendation, ontology model.

I. INTRODUCTION

At present, the integration and complexity of industrial products are constantly improving, and the lifecycle is gradually shortening, which puts greater requirements on the designer's product design abilities [1]. Knowledge reuse is a common means for assisting designers in accomplishing design tasks with high efficiency and quality in a short time [2]. Knowledge retrieval and knowledge recommendation are two main approaches to knowledge reuse [3]. The advantage of knowledge retrieval is that it fully meets the user's retrieval needs and can quickly and accurately find the best matching results from the retrieval keywords [4], [5]. However, the efficiency of knowledge retrieval decreases as the capacity of the knowledge base grows. It is also difficult for inexperienced designers to use accurate keywords to find the most appropriate knowledge. Therefore, knowledge recommendation is gradually replacing knowledge retrieval as the key technology in knowledge reuse [6]. Knowledge recommendation technologies adopt algorithms to search the knowledge base for knowledge that best meets the current needs of the designer and then actively recommend it to the designer. This characteristic requires less experience and is more acceptable to designers [7].

To find knowledge that can solve design tasks, the similarity between the task and knowledge needs to be computed. The knowledge that has the highest similarity value to the task is generally considered to be the best result. There are

The associate editor coordinating the review of this manuscript and approving it for publication was Hamid Mohammad-Sedighi.

three main types of similarity computations [8]. The first type is based on keyword matching, e.g., the N-gram method [9] or the Jaccard method [10]. The second type is based on vector spaces and is the most widely used method at present. It extracts the keywords from the knowledge and uses them to build a vector to represent the knowledge. Therefore, the similarity between two knowledge results can be compared by computing the distance between these vectors. The common methods use the Minkowski distance [11], the cosine similarity method [12]–[14], the Pearson correlation coefficient method [15], the Hamming distance method [16], etc. The third type is based on ever-evolving artificial intelligence algorithms [17]–[19]. The accuracies of the above similarity computation methods are gradually improving, and the scope of application is also increasing. However, when these algorithms are used to compute the similarity between tasks and knowledge, the result is the objective similarity between them without considering the subjective experience and demands of designers. Therefore, these methods do not apply to all designers in practice.

Recommendation systems that consider users' needs are widely used in many fields such as movies recommendation, advertisements recommendation on some websites, and especially the e-commerce field [20]–[23]. The most classic algorithm is collaborative filtering (CF) [24]. CF is mainly divided into user-based CF and item-based CF. The main idea of the former is that for two users with similar preferences, if one of them likes an item, the other one will like it as well. The main idea of the latter is that if a user likes one of two similar items, he/she will like the other one. CF combines the feedback information of a large number of users with the objective similarity between users or items and enables users to obtain filtered information from other users according to their needs. It is a typical way to use collective wisdom. However, recommendation systems in the engineering design field are different from other fields. For instance, CF algorithms in the e-commerce field mainly consider the ratings or preferences of a user when recommending new items to him/her. This characteristic restrains the types of recommendation results, which is not appropriate for the engineering design field. In the engineering design field, one of the functions of the recommendation systems is to assist the designers in improving their design ability. Therefore, the pieces of knowledge that useful but not frequently used by the designers should be recommended to the designers. The current recommendation algorithms in other fields do not possess this characteristic.

Researchers have done much work to apply the CF algorithm to the engineering field to build personalized knowledge recommendation systems. Liu *et al.* [25] use an ontology matching algorithm to explore the relationship between the context in the workflow system to find accomplished tasks that are similar to the current task, and the knowledge used by them is recommended to the current designer to realize knowledge reuse. Xu *et al.* [26] proposed an intelligent and personalized multiperson cooperative intention capture technology for product concept design. They built a cooperative intention capture model and then built a knowledge recommendation system that met the requirements of collaboration through a text content matching algorithm. Similarly, Zhang *et al.* [27] constructed a knowledge component model, including knowledge ontology, an inference engine and an evaluation method based on entropy weights to optimize matching results. The model was successfully applied to a plane parts design task. In addition, some personalized recommendation technologies that only consider individual knowledge and experience also exist. Li *et al.* [28] proposed a double recommendation strategy based on a complex network to capture the designer's intention and match the corresponding knowledge according to the intention. Gao *et al.* [29] proposed a cognitive information gain model to measure the assistance that knowledge gave to designers. The model classifies designers into four categories: assistant engineer, junior engineer, intermediate engineer, and senior engineer according to the number of completed tasks, and the model computes the assistance of the knowledge to each type of designer by assigning different weights. Wang *et al.* [30] explored a knowledge recommendation method that combined design intention and user interest, and the knowledge was classified according to user interest to better meet the user's personalized demands. Li *et al.* [31] established the ontology model of knowledge and constructed an inference engine that meets manufacturing constraints by using semantic web rule language. Then, knowledge can be reused through the inference engine.

However, the essence of CF is still to use the individual and collective experience to search the target through similarity computation, which is not fully applicable to knowledge recommendation in the engineering field. In engineering design, compared with knowledge retrieval, the purpose of knowledge recommendation is to minimize the time spent acquiring applicable knowledge. A good knowledge recommendation approach should recommend the right knowledge (what to recommend) to the right person (who to recommend) at the right time (when to recommend) in the right way (how to recommend). Most researches have been focused on the "what to recommend" problem using similarity computation techniques and the "when to recommend" problem using context matching techniques, but few studies considered all the four problems simultaneously. The "who to recommend" and "how to recommend" problems are also important. These two problems determine if the designers willing to use the recommendation system. Both the problems relate to the designers' prior knowledge and experience. In general, designers can quickly find the knowledge they are familiar with, but they need much more time to find unfamiliar knowledge and judge its applicability. Therefore, the designer's familiarity with knowledge is an important indicator of the designer's personalized demands. This paper proposes a correlation-experience-demand (CED) integrated knowledge recommendation approach to solve the above four problems. The CED approach uses the workflow engine of the product data management (PDM) system to establish the relationship between the design process and tasks, which solves the "when to recommend" problem. The term frequency-inverse document frequency algorithm (TF-IDF) and cosine similarity algorithm are adopted in each workflow node of the design process to compute the similarity between tasks and knowledge to find the knowledge that matches the task, which solves the "what to recommend" problem. Then, for a specific designer, the tasks completed by that individual and the knowledge usage information by all designers are combined to compute the subjective judgment score of the correlation between the task and knowledge. Moreover, according to that individual's access to knowledge information, that individual's degree of demand model for knowledge is constructed based on the forgetting curve, which solves the "who to recommend" problem. Finally, the assistance score of the knowledge for the current designer to complete the task in the current context is computed combining the above three parts. The recommendation list is obtained by ranking the knowledge in assistance score descending order to build personalized and accurate knowledge recommendations, which solves the "how to recommend" problem.

The paper is structured as follows. Section 2 presents the framework of the CED approach. The ontology models of knowledge and designer are constructed in section 3. Section 4 describes the mathematical model of the knowledge recommendation algorithm. Section 5 provides a case study using the CED approach and includes a discussion of the results. The conclusions of the study and planned future work are summarized in section 6.

II. THE FRAMEWORK OF THE CED APPROACH

Knowledge recommendation involves five parts: design process, task, knowledge, designer and knowledge recommendation algorithm. To enable the computer to process the five parts, they are constructed into four models, and the CED approach consists of these four models. The four models are as follows: design process-based task model, knowledge ontology model, designer ontology model and the mathematical model for the knowledge recommendation algorithm. The design process-based task model uses a PDM system to manage the design process. Tasks can be divided into multiple levels or multiple subtasks according to the actual situation. Each task has one and only one corresponding design process, and its subtasks are included in the nodes of the design process. The knowledge ontology model and the designer ontology model use ontology technology [27] to integrate the information from both knowledge and designer. The mathematical model of the knowledge recommendation algorithm consists of three submodules, namely, the degree of correlation, the degree of demand and the degree of assistance, and the model is used to evaluate the knowledge and build the knowledge recommendation.

Definition 1: Degree of correlation (DoC). DoC is the evaluation of the correlation between a task and a piece of knowledge. The DoC consists of objective correlation and subjective correlation.

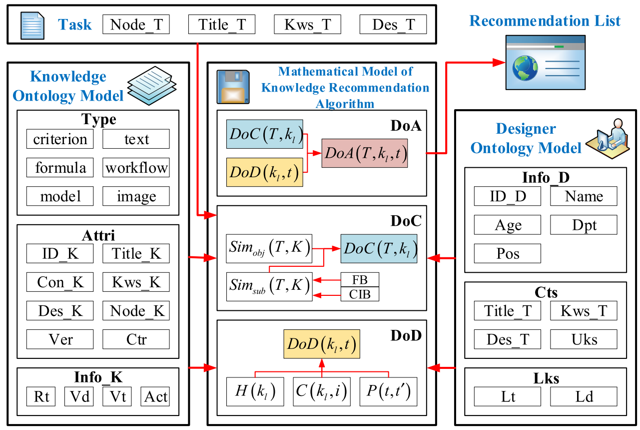


FIGURE 1. The framework of the CED approach.

Definition 1.1: Objective correlation. Objective correlation is the objective evaluation of the correlation between a task and a piece of knowledge. Objective correlation is obtained by a similarity computation between the task and the knowledge.

Definition 1.2: Subjective correlation. Subjective correlation is the subjective evaluation of the correlation between a task and a piece of knowledge given by a designer based on his/her experience. If the knowledge has been used to complete a task by the designer in the node, the subjective correlation is computed through the feedback of the designer. If the knowledge has not been used to complete a task by the designer in the node, the subjective correlation is computed by using the usage information of the knowledge in the node by all the other designers.

Definition 2: Degree of demand (DoD). DoD is the evaluation of the degree of demand of a designer for a piece of knowledge at a particular moment. The DoD is computed by using the designer's access to knowledge information based on the forgetting curve function.

Definition 3: Degree of assistance (DoA). The DoA is the evaluation of the degree of assistance that a piece of knowledge gives to a designer in completing a task at a given moment. DoA is computed using the DoC and the DoD.

A. RELATIONSHIP AMONG MODELS

The design process-based task is assigned by the PDM and includes four parts: the corresponding design process node (Node_T), the task title (Title_T), the keywords (Kws_T) and the description (Des_T). The knowledge ontology model and the designer ontology model store knowledge and designer information, which will participate in the computation of the mathematical model in combination with the task information. The DoA score of knowledge is computed by the mathematical model, and the recommendation list (RL) is obtained according to the DoA score. The relationship among the models is shown in Fig. 1.

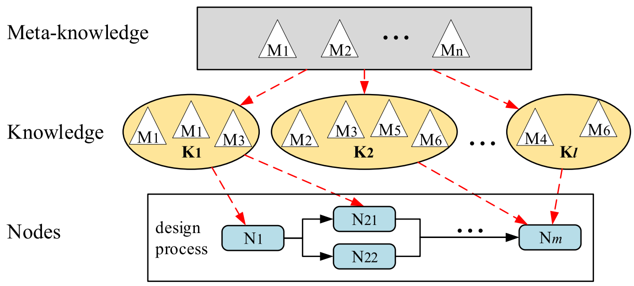


FIGURE 2. The granularity and structure of knowledge.

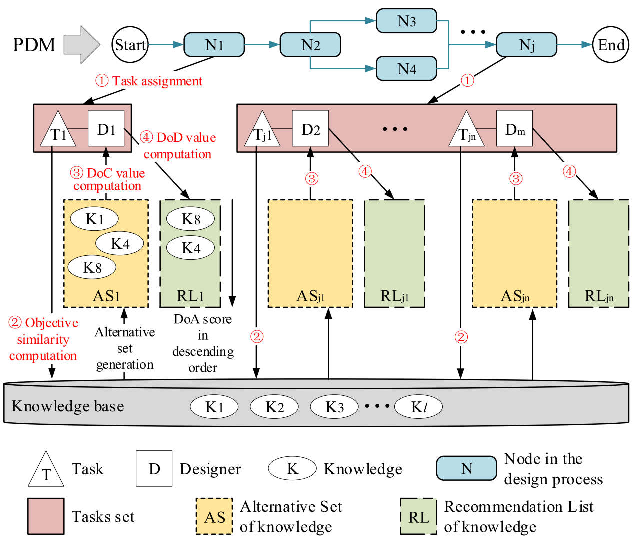


FIGURE 3. The process of the CED knowledge recommendation approach and relationship among PDM workflow, tasks, designers and knowledge.

When completing tasks, knowledge is the basic unit used by designers and is the recommended object. However, two different pieces of knowledge have different amounts of information (AoI) and may share some of the same content. Therefore, the concept of meta-knowledge is introduced and is considered to be the basic unit of knowledge in this paper. A piece of knowledge contains multiple meta-knowledge pieces, and a piece of meta-knowledge can be contained in multiple knowledge pieces. A piece of knowledge belongs to one or more nodes in the design process and is used to solve the tasks corresponding to the node. The relationship between meta-knowledge, knowledge and design process nodes is shown in Fig. 2.

B. KNOWLEDGE RECOMMENDATION PROCESS

The knowledge recommendation process is completed by the above four models together, as shown in Fig. 3.

In the first step, the chief designer assigns task T corresponding to node N in the design process to designer D through PDM. If a task requires the collaboration of more than one designer, it will be split into multiple subtasks with each designer undertaking one or more of them. In the second step, after D receives T, the knowledge recommendation system will automatically extract the relevant information from T and compute the objective correlation value between T and all the knowledge corresponding to node N in the knowledge base. If the objective correlation value of a piece of knowledge is higher than a predefined threshold value, the knowledge will be added into the alternative set (AS) of knowledge. In the third step, the subjective correlation value and the DoC value of the knowledge in the AS are computed according to the relevant information in the designer and knowledge ontology models. In the fourth step, the DoD value of the knowledge in the AS is computed. Finally, the DoA score of the knowledge in the AS is computed. The knowledge in the AS is ranked in DoA score descending order to generate the RL for D.

Since each task corresponds to a workflow node in the PDM system, if a designer is assigned multiple tasks, the relationship between arbitrary two tasks can be one of the following three situations: (1) the two tasks are in the same node of the same workflow; (2) the two tasks are in two different nodes of the same workflow; (3) the two tasks are in two different workflows. When processing multiple tasks, the precedence is different according to the relationship between tasks. For situation (1), tasks are closely related to each other and designers need to process them simultaneously. For situation (2), the sequential order of two tasks is determined by the corresponding workflow, and designers need to process them successively according to the workflow. For situation (3), tasks have no relation to each other and designers can process them according to his/her preference. Tasks in the situation (2) and (3) do not need to be processed simultaneously.

For illustrative purposes, some symbols are defined in the paper, as shown in Table 1.

TABLE 1. Definition of symbols in the paper.

Symbols	Meaning
M, m_i	M represents meta-knowledge, and m_i is ith meta-knowledge.
K, k_l	K represents knowledge, and k_l is lth knowledge.
C_j	C_j is jth knowledge category.
T, T_i	T represents tasks, and T_i is the ith subtask.
$Num(X)$	The number of X.
$Num_Y(X)$	The number of X in set Y, where X∈Y or X⊆Y.

III. ONTOLOGY MODEL CONSTRUCTION

A. KNOWLEDGE ONTOLOGY MODEL

The knowledge ontology model is defined as follows.

$$Knowledge = \{Type, Attri, Info\_K\} \tag{1}$$

a) *Type* is the type of knowledge belonging to one of the following six classes: *criterion*, *text*, *formula*, *workflow*, *model*, and *image*, denoted as

$$Type = \{criterion, text, formula, workflow, model, image\} \tag{2}$$

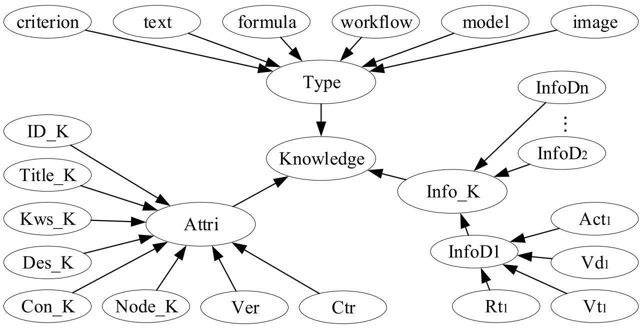


FIGURE 4. **The knowledge ontology model.**

b) *Attri* contains the basic attributes of knowledge and is defined as

$$Attri = \{ID\_K, Title\_K, Kws\_K, Des\_K, Con\_K, Node\_K, Ver, Ctr\} \quad (3)$$

where *ID_K*, *Title_K*, *Kws_K*, *Des_K*, *Con_K*, *Node_K*, *Ver*, and *Ctr* are the ID, title, keywords, description, content, corresponding node in the design process, version, and creator of the knowledge, respectively.

c) *Info_K* is the usage information of the knowledge and consists of the usage information of users. *Info_K* is defined as

$$Info\_K = (Info_{D1}, Info_{D2}, \cdots, Info_{Dn})^T \quad (4)$$

where $Info_{Dn}$ is the usage information of user Dn. The ith time usage information of Dn is recorded from the following four aspects: relevant task Rt_{ni}, access time (unit: minutes) Vt_{ni}, access date Vd_{ni}, and the action Act_{ni}. $Info_{Dn}$ is denoted as

$$Info_{Dn} = \begin{pmatrix} Rt_{n1} & Vt_{n1} & Vd_{n1} & Act_{n1} \\ Rt_{n2} & Vt_{n2} & Vd_{n2} & Act_{n2} \\ \vdots & \vdots & \vdots & \vdots \\ Rt_{ni} & Vt_{ni} & Vd_{ni} & Act_{ni} \end{pmatrix} \quad (5)$$

The action belongs to one of the four types, namely, *Use*, *Read*, *Ignore* and *Decline* and are denoted as

$$Act_{ni} \in \{Use, Read, Ignore, Decline\} \quad (6)$$

The definition and meaning of the actions are illustrated in the subsection *feedback-based subjective correlation*. The structure of the knowledge ontology model is shown in Fig. 4.

B. DESIGNER ONTOLOGY MODEL

One designer's experience and personalized demands can be reflected by their completed tasks, access to knowledge and feedback on the recommended results. Therefore, the designer ontology model is defined as follows.

$$Designer = \{Info\_D, Cts, Lks\} \quad (7)$$

a) *Info_D* is the basic information of the designer, defined as

$$Info\_D = \{ID\_D, Name, Age, Dpt, Pos\} \quad (8)$$

where *ID_D*, *Name*, *Age*, *Dpt,* and *Pos* are the ID, name, age, department, and post of the designer, respectively.

b) *Cts* is the set of tasks completed by the designer, denoted as

$$Cts = (Task_1, Task_2, \cdots, Task_m)^T \quad (9)$$

where $Task_m$ is the mth completed task of the designer. $Task_m$ is recorded from the following five aspects: title $Title\_T_m$, keywords Kws_T_m, description Des_T_m, corresponding node $Node\_T_m$ in the design process, and used knowledge Uks_m. $Task_m$ is denoted as

$$Task_m = \{Title\_Tm, Kws\_Tm, Des\_Tm, Node\_Tm, Uks\_m\} \quad (10)$$

Since designers access more than one piece of knowledge when completing tasks, Uks_m is the set that consists of all the accessed knowledge and the corresponding actions, defined as

$$Uks_m = \{(ID\_K_{m1}, Act_{m1}), (ID\_K_{m2}, Act_{m2}), \cdots, (ID\_K_{mn}, Act_{mn})\} \quad (11)$$

c) *Lks* is the set of all the knowledge the designer has learned, denoted as

$$Lks = (Lks_{K1}, Lks_{K2}, \cdots, Lks_{Kn})^T \quad (12)$$

Designers can also use knowledge recommendation systems to learn knowledge when not undertaking tasks. This knowledge is recorded in *Lks*. Lks_{Kn} is the learning information of knowledge Kn and is recorded from the following two aspects: the ith learning time Lt_{ni} (unit: minutes) and the ith learning date Ld_{ni}. Lks_{Kn} is denoted as

$$Lks_{Kn} = \begin{pmatrix} Lt_{n1} & Ld_{n1} \\ Lt_{n2} & Ld_{n2} \\ \vdots & \vdots \\ Lt_{ni} & Ld_{ni} \end{pmatrix} \quad (13)$$

The knowledge in *Cts* and *Lks* is different. Knowledge in *Cts* is task-relevant and can reflect the subjective evaluation of the designer, which will be considered in subsequent knowledge recommendation processes for the designer. Knowledge in *Lks* is task-irrelevant. Its information will only be used to compute the DoD value of the designer.

The structure of the designer ontology model is shown in Fig. 5.

IV. THE MATHEMATICAL MODEL OF KNOWLEDGE RECOMMENDATION ALGORITHM

A. DoC MODULE

The DoC consists of the objective correlation and the subjective correlation between task and knowledge. The objective correlation is only related to the task and knowledge

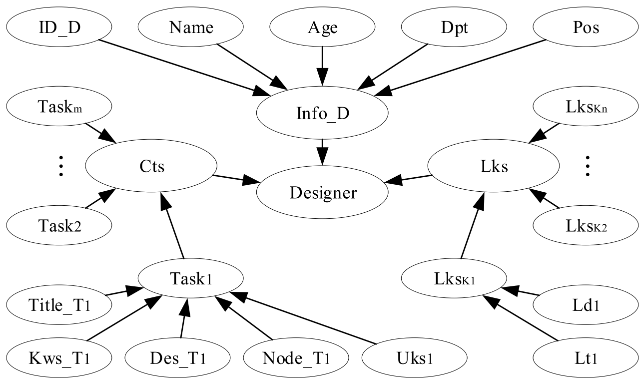


FIGURE 5. **The designer ontology model.**

and is evaluated by the similarity between them. The objective correlation between task and knowledge is proportional to their similarity. The subjective correlation is related to the task, knowledge and subjective evaluation of the designer.

1) OBJECTIVE CORRELATION

A task T consists of Node_T, Title_T, Kws_T, and Des_T. Node_T is used to rapidly filter the knowledge base to select the knowledge that is used to complete the corresponding tasks of Node_T. Title_T, Kws_T, and Des_T are specific requirements of the task. Since the TF-IDF algorithm is easy to implement and runs quickly [6], this paper uses the TF-IDF algorithm and the cosine similarity to compute the objective similarity between tasks and knowledge to generate the AS.

In this paper, the objective correlation value is computed through the corresponding similarity between the pairs Title_K and Title_T, Kws_K and Kws_T, and Des_K and Des_T. Therefore, the m_i in the six parts is extracted and used to construct a vector to represent the six parts. The form of the vector is shown in (14).

$$vec = \{[m_1, w_{vec}(m_1)], [m_2, w_{vec}(m_2)], \cdots, [m_n, w_{vec}(m_n)]\} \quad (14)$$

where $w_{vec}(m_i)$ is the weight of m_i in vec, and vec can be Title_K, Title_T, Kws_K, Kws_T, Des_K, or Des_T. If T can be split into multiple subtasks, then we write $T_1, T_2, \ldots, T_n, T_{vec} = \bigcup_{i=1}^{n} T_{i-vec}$, where T_{vec} is the vector of T and T_{i-vec} is the vector of T_i.

If multiple tasks that assigned to a designer do not need to be processed simultaneously, the vector that represents each task is used to compute the relevant parameters in this paper. If the tasks need to be processed simultaneously, the meta-knowledge in these tasks is retrieved to construct the vector that represents all of them. Then, the vector is used to compute the relevant parameters in this paper of these tasks.

$w_{vec}(m_i)$ is computed and normalized by the term frequency $TF_{vec}(m_i)$ of m_i in vec position and inverse document frequency $IDF(m_i)$ of m_i through (15).

$$\begin{cases} w_{vec}(m_i) = \dfrac{TF_{vec}(m_i) \cdot IDF(m_i)}{\sqrt{\sum_{i=1}^{n} [TF_{vec}(m_i) \cdot IDF(m_i)]^2}} \\ TF_{vec}(m_i) = Num_{vec}(m_i) / Num_{vec}(M) \\ IDF(m_i) = \lg \dfrac{Num(K)}{Num(K_{m_i}) + 1} \end{cases} \quad (15)$$

where K_{mi} is the knowledge that contains m_i.

As the global weight of m_i, $IDF(m_i)$ is significant to both knowledge and task. High $IDF(m_i)$ means that m_i appears in few knowledge pieces. Therefore, an m_i with high $IDF(m_i)$ has a good ability to distinguish the knowledge that contains it. For tasks, an m_i with high $IDF(m_i)$ highlights the focus of the task and enables the knowledge that is most applicable to the task to be selected.

The similarity between two objects in vector form can be computed by the cosine similarity algorithm. However, in general, the dimension of the two vectors is different, so preprocessing is required to equalize the dimension. A task may require more than one piece of knowledge, and each one of these may support a part of the task. To obtain the part of a task supported by a piece of knowledge, the vectors of task and knowledge are intersected. Then, the intersection is compared with the vector of knowledge, and the elements that the intersection does not contain are added into it; the weights are set to zero to obtain another vector, B. B represents the part of the task that is supported by the knowledge and has the same dimension as the vector of knowledge. Mathematically, the above operation projects the task vector into the vector space that contains the knowledge vector, and this operation is performed correspondingly in the title, keywords and description parts of the knowledge and B vectors. The similarity between each part is computed by using (16), as shown at the bottom of the next page.

Where $Sim_t(T,K)$, $Sim_k(T,K)$ and $Sim_d(T,K)$ are the similarity between the title, keywords and description of the task and knowledge, respectively; $w_{Title\_B}(m_i)$, $w_{Kws\_B}(m_i)$ and $w_{Des\_B}(m_i)$ are the weights of m_i in the title, keywords and description of the B vector, respectively; $w_{Title\_K}(m_i)$, $w_{Kws\_K}(m_i)$ and $w_{Des\_K}(m_i)$ are the weights of m_i in the title, keywords and description of knowledge vector, respectively.

The objective correlation value, $Sim_{obj}(T,K)$, between a task and a piece of knowledge is computed by using (17).

$$Sim_{obj}(T, K) = w_t Sim_t(T, K) + w_k Sim_k(T, K) + w_d Sim_d(T, K) \quad (17)$$

where, w_t, w_k and w_d are the weights of $Sim_t(T,K)$, $Sim_k(T,K)$ and $Sim_d(T,K)$, respectively, and can be computed by the analytic hierarchy process (AHP) algorithm [32].

2) FEEDBACK-BASED SUBJECTIVE CORRELATION

A designer gives feedback actions to the recommended knowledge when using the knowledge recommendation

TABLE 2. Designer's actions to the recommended knowledge.

Action	Definition	Correlation
Use	The designer opens a piece of recommended knowledge and lasts more than 1min.	Perfect
Read	The designer opens a piece of recommended knowledge, but lasts less than 1min, and then closes it.	High
Decline	The designer removes a piece of recommended knowledge from the recommendation list.	Low
Ignore	The designer neither opens nor declines a piece of recommended knowledge.	Unknown

system. If a designer once gave a valid feedback to a piece of knowledge, the subjective correlation value is computed based on the feedback to retain his/her preference. Feedback actions can be defined differently according to the specific requirements of the knowledge recommendation system. The definitions of feedback actions in this paper are shown in Table 2. *Use*, *read* and *decline* are valid feedback actions, and *ignore* is invalid feedback. The correlation column in Table 2 is the designer's subjective evaluation of the correlation between tasks and the recommended knowledge.

Since the task is design-process based, the tasks that share the same Node_T are similar to each other. Similarly, knowledge pieces that belong to the same design process node are similar to each other. When recommending knowledge to a designer, the knowledge that the designer deems to be highly correlated with the task has priority in recommendation. To quantify feedback actions, subjective weights need to be assigned to feedback actions. Based on the previous analysis, the weight of feedback actions should be *use*> *read*> *decline*, and *ignore* should not be assigned a weight since it does not provide useful information. To best

TABLE 3. The weight of designer's feedback actions.

	Use	Read	Decline	Ignore
w_r	0.9	0.5	0.1	N/A

retain the preference of designers and distinguish different feedback actions while keeping the possibility of variation of the recommendation results, the weight of *use* should be lower than 1 and the weight of *decline* should be higher than 0. Combined with the above analysis and the idea of the AHP algorithm, which divides the importance of features into 9 grades of 1-9, the weights of the feedback actions are shown in Table 3.

When computing the subjective correlation value between a piece of knowledge K in the AS and current task T for a designer, the knowledge recommendation system obtains his/her previously completed tasks that correspond to K in the Node_T. Then, the similarity between T and each of these tasks is computed to obtain T_p that has the highest similarity. Since T is similar to T_p, the designer is assumed to have the same feedback action in response to K for both T and T_p.

Similar to the objective similarity computation between tasks and knowledge, the similarity, $Sim(T,T_p)$, between T and T_p is also computed from the title, keywords and description parts using (16) and (17). The difference is that the preprocessing is simpler. The union of the vector form of title, keywords and description parts of T and T_p is obtained. Then, each vector form is compared with the corresponding union, and the elements that the vector does not contain are added into it, and the weights are set to zero to equalize the dimension of the two vectors. After this preprocessing, the feedback-based subjective correlation is computed by using (18).

$$Sim_{sub}\left(T,K\right)=w_r Sim\left(T,T_p\right) \tag{18}$$

$$\begin{cases} Sim_t\left(T,K\right)=\dfrac{\sum_{i=1}^{n} w_{Title\_B}\left(m_i\right)\cdot w_{Title\_K}\left(m_i\right)}{\sqrt{\left(\sum_{i=1}^{n}\left(w_{Title\_B}\left(m_i\right)\right)^2\right)\cdot\left(\sum_{i=1}^{n}\left(w_{Title\_K}\left(m_i\right)\right)^2\right)}} \\ Sim_k\left(T,K\right)=\dfrac{\sum_{i=1}^{n} w_{Kws\_B}\left(m_i\right)\cdot w_{Kws\_K}\left(m_i\right)}{\sqrt{\left(\sum_{i=1}^{n}\left(w_{Kws\_B}\left(m_i\right)\right)^2\right)\cdot\left(\sum_{i=1}^{n}\left(w_{Kws\_K}\left(m_i\right)\right)^2\right)}} \\ Sim_d\left(T,K\right)=\dfrac{\sum_{i=1}^{n} w_{Des\_B}\left(m_i\right)\cdot w_{Des\_K}\left(m_i\right)}{\sqrt{\left(\sum_{i=1}^{n}\left(w_{Des\_B}\left(m_i\right)\right)^2\right)\cdot\left(\sum_{i=1}^{n}\left(w_{Des\_K}\left(m_i\right)\right)^2\right)}} \end{cases} \tag{16}$$

3) COLLECTIVE INTELLIGENCE-BASED SUBJECTIVE CORRELATION

If a designer has not given a valid feedback to a piece of knowledge, the subjective correlation value is computed based on all the other designers' feedback actions to fully take advantage of the collective intelligence. An improved information gain algorithm (IIG) is proposed in this paper to predict the subjective correlation value of a piece of knowledge.

Information gain is a commonly used measurement approach for the importance of features. It has a high precision since it considers two situations: the presence and absence of features. In addition, the precision can be further improved if the distribution of features is considered. The basic assumption of information gain is that if a feature brings more information to a classification system, the feature is more important to the system. The AoI brought to the system by a feature is reflected in the difference in the system's AoIs in the presence or absence of the feature.

To compute the information gain value of a meta-knowledge m_i in a design process node, all the recommended knowledge of the node is required. According to the feedback of designers, each piece of recommended knowledge is divided into two categories: used knowledge (category C) and unused knowledge (category $\overline{C}$). N pieces of knowledge from the node and their corresponding feedback are randomly selected to create a training set, Q. The information gain value of m_i in Q is an unbiased estimation of the information gain value of m_i in the node and is computed by using (19).

$$\begin{aligned} IG(m_i) = &-\sum_{\substack{C_j=C,\\ C_j=\overline{C}}} P(C_j)\cdot \log_2 P(C_j) \\ &+P(m_i)\cdot \sum_{\substack{C_j=C,\\ C_j=\overline{C}}} P(C_j|m_i)\cdot \log_2 P(C_j|m_i) \\ &+P(\overline{m_i})\cdot \sum_{\substack{C_j=C,\\ C_j=\overline{C}}} P(C_j|\overline{m_i})\cdot \log_2 P(C_j|\overline{m_i}) \end{aligned} \tag{19}$$

where m_i is the meta-knowledge in Q and is contained in knowledge k_l in Q; $P(C_j)$ is the probability that k_l belongs to category C_j; $P(m_i)$ is the probability that the knowledge contains m_i; $P\overline{m_i})$ is the probability that the knowledge does not contain m_i; $P(C_j|m_i)$ is the conditional probability that k_l belongs to category C_j if m_i is contained; $P(C_j|\overline{m}_i)$ is the conditional probability that k_l belongs to category C_j if m_i is not contained.

Furthermore, to avoid the drawback that the distribution of the features is not considered in the information gain algorithm, two parameters inspired by [29] are introduced in IIG. The two parameters are the distribution of m_i in a category and the distribution of m_i in all categories. In this paper, because only the impact of m_i on category C is of concern, the two parameters are denoted as $DI(C,m_i)$ and $D(m_i)$, respectively. The two parameters are computed by the sample variance of m_i. $DI(C,m_i)$ reflects the importance of m_i to C. The higher the sample variance of m_i in a category is, the less of the knowledge that contains m_i in the category and the less important m_i is to the category. $D(m_i)$ reflects the ability of m_i to distinguish categories. The fewer categories m_i appears in, the higher the concentration of m_i in these categories and the more important m_i is to these categories. $DI(C,m_i)$ and $D(m_i)$ are defined in (20) and (21).

$$\begin{cases} DI\left(C, m_i\right) = 1-\sqrt{S_C^2\left(X\left(m_i\right)\right)}\Big/\sqrt{\sum\limits_{i=1}^{n} S_C^2\left(X\left(m_i\right)\right)} \\ S_C^2\left(X\left(m_i\right)\right) = \dfrac{\sum\limits_{r=1}^{N}\left(X_r\left(m_i\right)-\overline{X_C\left(m_i\right)}\right)^2}{Num_C\left(k_l\right)-1} \\ \overline{X_C\left(m_i\right)} = Num_C\left(m_i\right)\big/Num_C\left(k_l\right) \end{cases} \tag{20}$$

where $X_r(m_i)$ is the number of m_i in the rth piece of knowledge in Q.

$$\begin{cases} D\left(m_i\right) = \sqrt{S^2\left(Y\left(m_i\right)\right)}\Big/\sqrt{\sum\limits_{i=1}^{n} S^2\left(Y\left(m_i\right)\right)} \\ S^2\left(Y\left(m_i\right)\right) = \dfrac{\sum\limits_{\substack{C_j=C,\\ C_j=\overline{C}}}\left(Num_{C_j}(m_i)-\overline{Y(m_i)}\right)^2}{m-1} \\ \overline{Y\left(m_i\right)} = Num_Q\left(m_i\right)/m \end{cases} \tag{21}$$

Since the knowledge is only divided into two categories, (21) can be simplified to (22).

$$\begin{cases} D\left(m_i\right) = \sqrt{S^2\left(Y\left(m_i\right)\right)}\Big/\sqrt{\sum\limits_{i=1}^{n} S^2\left(Y\left(m_i\right)\right)} \\ S^2\left(Y\left(m_i\right)\right) = \left(Num_C\left(m_i\right)-\overline{Y\left(m_i\right)}\right)^2 \\ \qquad\qquad + \left(Num_{\overline{C}}\left(m_i\right)-\overline{Y\left(m_i\right)}\right)^2 \\ \overline{Y\left(m_i\right)} = Num_Q\left(m_i\right)\big/2 \end{cases} \tag{22}$$

In (19), (20) and (22), the parameters relevant to m_i should also be computed through the vector form of title, keywords, and description of knowledge. Then, the IIG value that takes the distribution of m_i into account is computed by using (23).

$$IIG_{vec}\left(C, m_i\right) = IG_{vec}\left(m_i\right)\cdot\left(\frac{DI_{vec}\left(C, m_i\right)+D_{vec}\left(m_i\right)}{2}\right) \tag{23}$$

where $IG_{vec}(m_i)$, $DI_{vec}(C,m_i)$ and $D_{vec}(m_i)$ are the values of $IG(m_i)$, $DI(C,m_i)$ and $D(m_i)$, respectively, after considering the position of m_i.

$IIG_{vec}(C,m_i)$ is obtained by the feedback of all designers and reflects the importance of m_i to the knowledge that contains it. Therefore, when computing collective intelligence-based subjective correlation, the IIG value is used as the weight of the meta-knowledge. The collective intelligence-based subjective correlation can also be computed using (16)

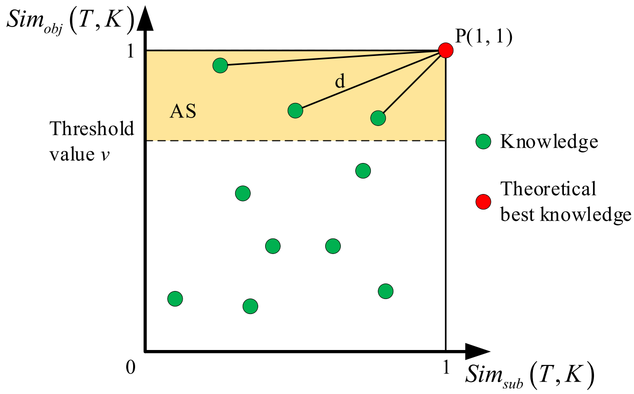


FIGURE 6. The coordinate system for DoC value computation.

and (17), with $IIG_{vec}(C, m_i)$ as the weight of m_i in the vector form of the knowledge. Note that the preprocessing is also required in this case.

4) DoC VALUE COMPUTATION

The DoC value is the synthesis of the objective and subjective correlation values. To combine the applicability of the knowledge and experience of designers, the objective and subjective correlation values are deemed to be of equal importance. To combine them, a coordinate system is established with subjective correlation value as the abscissa and objective correlation value as the ordinate, as shown in Fig. 6. The DoC value is computed by using the Euclidean distance method [11].

In Fig. 6, the red point P is the knowledge that theoretically best supports the current task, T, and the green points are the knowledge that belongs to Node_T. For knowledge in the AS, d is the distance between the knowledge and P. The DoC value is inversely proportional to d. If the coordinate of knowledge k_l is ($Sim_{sub}(T, k_l)$, $Sim_{obj}(T, k_l)$), the DoC value is defined and computed by using IV-B.

$$\begin{cases} DoC\,(T, k_l) = e^{-d} \\ d = \sqrt{[Sim_{sub}\,(T, k_l) - 1]^2 + [Sim_{obj}\,(T, k_l) - 1]^2} \end{cases} \tag{24}$$

B. DoD MODULE

Technically speaking, the degree of demand of designers for knowledge is actually the degree of demand of designers for the knowledge recommendation service. DoD is generally decided by the experience of designers. Experienced designers are familiar with knowledge content or can rapidly find the target knowledge in the knowledge base. Therefore, the DoD of experienced designers is relatively low. DoD is commonly measured through the modeling of designers' design ability [20], [29]. Designers are graded according to the number of their completed tasks. However, the precision of this approach is limited, and the difference between subjects at the same level cannot be reflected. Therefore, the DoD module in this paper quantizes the designer's demands for knowledge based on the forgetting curve from four aspects of the AoI of knowledge, the logic of the knowledge, the memory ability of the designers, and the designers' access to knowledge information. In this section, the AoI of knowledge is computed based on Shannon's information theory, and the rest of the three parts are constructed into a designers' memory model.

1) AoI OF KNOWLEDGE

The AoI of each piece of knowledge is different because different pieces of knowledge contain a different number of meta-knowledge. According to Shannon's information theory, the AoI of knowledge can be represented by the information entropy in (25).

$$H = -\sum_{i=1}^{n} P\,(m_i) \cdot \log_2 P\,(m_i) \tag{25}$$

where $P(m_i)$ is the probability of m_i in the knowledge that contains it and is computed by the term frequency of m_i.

In this paper, the AoI of a piece of knowledge is computed by the summation of the AoI of its title, keywords, and description parts, as shown in (26).

$$\begin{cases} H\,(k_l) = \sum H_{vec}\,(k_l) \\ H_{vec}\,(k_l) = -\sum\limits_{i=1}^{n} P_{vec}\,(m_i) \cdot \log_2 P_{vec}\,(m_i) \end{cases} \tag{26}$$

where $H(k_l)$ is the AoI of k_l; $H_{vec}(k_l)$ is the AoI of vec position of k_l, and vec can be Title_K, Kws_K or Des_K; $P_{vec}(m_i)$ is the probability of m_i in vec position of k_l.

According to (26), if k_l only contains one meta-knowledge, its AoI is zero. However, one meta-knowledge does not possess enough semantic information to make k_l a piece of knowledge, which means that the AoI of an arbitrary knowledge is greater than zero. Thus, the range of AoI for any pieces of knowledge is $(0, +\infty)$. For a specific designer, the required time to master a piece of knowledge is proportional to the AoI of the knowledge. The access time of a designer to a piece of knowledge can be obtained from the *Cts* and *Lks* of the designer ontology model.

2) MEMORY MODEL OF DESIGNERS

Psychologists have found that humans forget knowledge according to the Ebbinghaus forgetting curve [33]. To better simulate designers' memories and precisely quantize the DoD values, the memory model of designers is constructed based on the forgetting curve and from the following four aspects: the age of the designers, the memory ability of the designers, the logic of the knowledge and the information regarding the designers' access to each piece of knowledge. The age and the access information of designers can be obtained from the knowledge and designer ontology models. The logic of knowledge is decided by the type of knowledge and reflects how difficult it is for designers to memorize the knowledge. The memory ability of designers is calculated as the longest time interval between two times of correctly recalling the same piece of knowledge. The time interval can be obtained from *Uks* in the designer ontology model.

TABLE 4. The weight of logicality for different types of knowledge.

	workflow	formula	model	image	text	criterion
ε_t	0.90	0.75	0.60	0.45	0.30	0.15

The forgetting curve has multiple forms, e.g., the exponential function, the power function or the Pareto function [33]. Each form of the forgetting curve can reach the same precision if the coefficients are properly selected. Considering multiple factors such as function complexity, the exponential function is selected to create the forgetting curve in this paper, as shown in (27).

$$\begin{cases} P\left(t, t'\right) = e^{-\frac{t-t'}{b}} \\ b = \left(e^{\frac{\Delta t}{10}} + c - a + a_r\right)\varepsilon_t \end{cases} \tag{27}$$

where $P(t, t')$ is the proportionality coefficient of the correctly memorized information of a piece of knowledge for a designer to the total information of the knowledge at moment t; t' is the previous moment that the designer accesses the knowledge; Δt is a coefficient relevant to the memory ability of the designer; c is the time that the designer has access to the knowledge; a is the age of the designer; a_r is the maximum age of designers and can be taken as the retirement age; ε_t is the weight of logic for different types of knowledge. The higher the ε_t is, the more unforgettable the knowledge type. According to our experience, ε_t values are assigned in Table 4.

3) DoD VALUE COMPUTATION

The DoD value is a significant indicator to determine if a piece of knowledge should be recommended to a designer. If a piece of knowledge has a low DoD value to a designer, the new information that the designer can obtain from the knowledge is limited, which means that the designer can complete the task without the assistance of the knowledge. Therefore, the knowledge should not be recommended to the designer. Specifically, the DoD value of a piece of knowledge to its creator is deemed to be zero in this paper.

The learning speed of a designer is a constant value; thus, the percentage of a designer's mastered information is proportional to the learning time. As the learning time increases, the newly added learning time brings declining useful information from a piece of knowledge. Therefore, after the ith time accessing knowledge k_l, the index of mastery for a designer to k_l can be defined as in (28).

$$C\left(k_l, i\right) = \log_2 Time_i\left(k_l\right)/H\left(k_l\right) \tag{28}$$

where $H(k_l)$ is the AoI of k_l and is computed by (26); $Time_i(k_l)$ is the valid access time to k_l for a designer for the ith time. The unit of $Time_i(k_l)$ is minutes, and $Time_i(k_l) \geq 1$. $Time_i(k_l)$ is obtained from Vt_{ni} in the knowledge ontology model and Lt_{ni} in the designer ontology model.

Combining (27) with (28), the index of mastery for a designer to k_l at moment t is defined in (29).

$$F\left(k_l, t\right) = \sum_{i=1}^{c} P\left(t, t_i\right) \cdot C\left(k_l, i\right) \tag{29}$$

where t_i is the moment that the designer accesses k_l for the ith time. If a designer never accesses a piece of knowledge, $F(k_l, t) = 0$ for that individual at any moment.

Since the DoD value is inversely proportional to the index of mastery, it is computed by IV-B to limit its range to [0, 100].

$$DoD\left(k_l, t\right) = \frac{100}{F\left(k_l, t\right) + 1} \tag{30}$$

C. DoA SCORE AND RL GENERATION

The DoA score is a comprehensive evaluation of the ability of a piece of knowledge to determine if the knowledge can assist a designer in completing a task. The DoA score is relevant to the DoC and DoD values and is proportional to both of them. Therefore, the DoA score is defined and computed in (31).

$$DoA\left(T, k_l, t\right) = DoC\left(T, k_l\right) \cdot DoD\left(k_l, t\right) \tag{31}$$

Since knowledge with a higher DoA score has a better ability to assist the designer, the RL is generated by ranking the knowledge in the AS in descending order of their DoA scores. The process used in the CED approach to generate the RL is shown in Fig. 7.

Step 1. Task assignment. PDM assigns a task, T, to a designer, D, and the information of T and the meta-knowledge in T are extracted.

Step 2. Objective correlation computation and AS generation. Compute the objective correlation between knowledge belongs to Node_T and T. If the result is higher than the predefined threshold value, v, the knowledge is added to the AS.

Step 3. Knowledge removal. Remove all the knowledge created by D from the AS.

Step 4. Subjective correlation computation. Compute the subjective correlation between knowledge in the AS and T according to the feedback information of the knowledge.

Step 5. DoC value computation. Compute the DoC value based on the objective and subjective correlation values.

Step 6. DoD value computation. Compute the DoD value of a piece of knowledge in the AS to D at moment t.

Step 7. DoA score computation and RL generation. Compute the DoA value based on the DoC and DoD values. Generate the RL for D by ranking the DoA scores in descending order.

V. CASE STUDY

In this section, two different tasks are adopted to validate the proposed CED approach. First, the tasks' information, the relevant parameters, the experimental subjects, and the

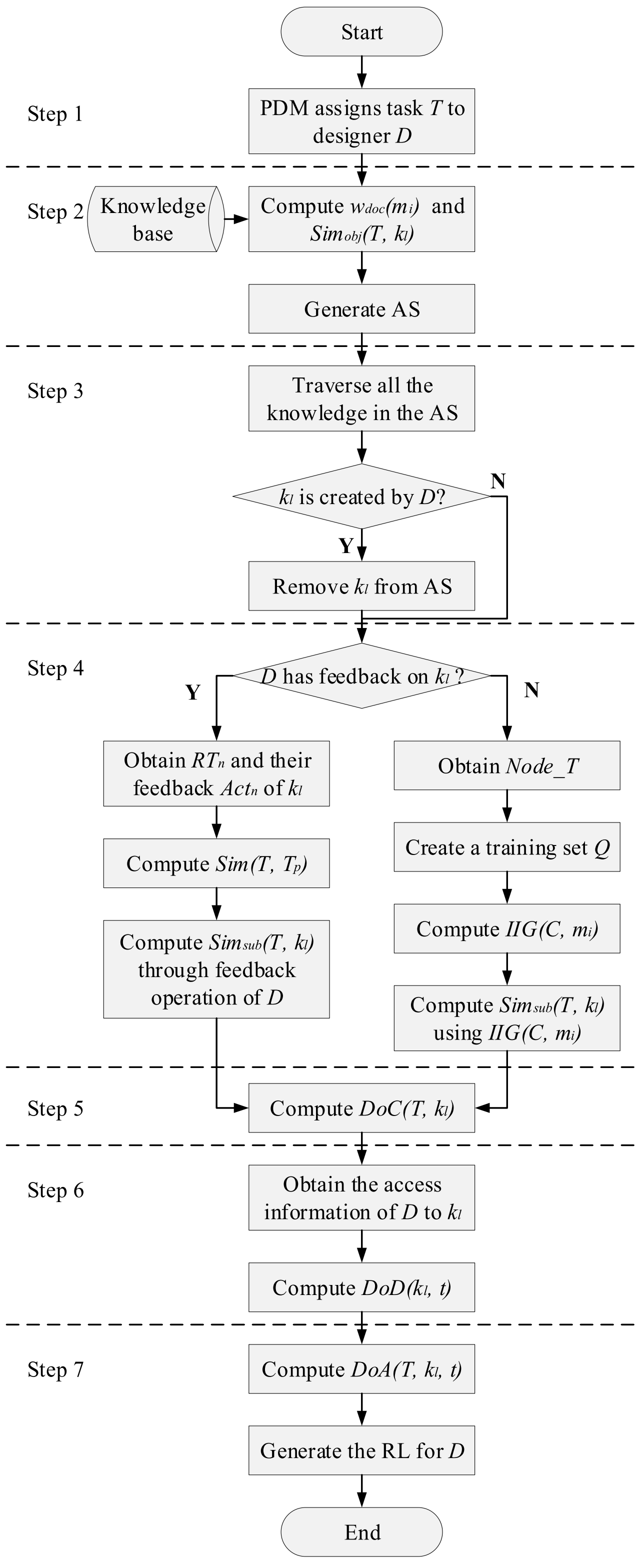


FIGURE 7. **The process of the CED approach.**

experimental environment are listed. Second, the computation results of the CED approach are shown and analyzed in detail. Third, to further evaluate the CED approach, three evaluation metrics are adopted for comparisons between the CED approach and three knowledge recommendation algorithms.

A. DATASETS AND PARAMETERS SETTINGS

1) PARAMETERS SETTINGS

The computation of the CED approach involves many designer-related parameters, and these parameters are obtained from the designer ontology model. The a_r in (27) is set to 60.

The predefined threshold value, v, for generating the AS should be neither too high nor too low. If v is too low, more knowledge with a relatively low DoA score will be recommended to the designers, which will cause the designers to spend more time filtering them. If v is too high, some knowledge that might be used will not be recommended to the designers, which will cause the designers to actively retrieve this part of knowledge in the knowledge base. Both situations will reduce design efficiency. In this paper, v is set to 0.7 according to our experience. In practice, v is set to 0.7 initially and can be changed by designers based on their experience and preference.

The w_t, w_k, and w_d in (17) are computed by using the AHP algorithm. Firstly, 10 experts are asked to give their subjective evaluations to the weights of $\{w_t, w_k, w_d\}$. The weight evaluation matrixes are listed as follows.

$$M_1 = \begin{bmatrix} 1 & 1 & 3 \\ 1 & 1 & 3 \\ 1/3 & 1/3 & 1 \end{bmatrix} \quad M_2 = \begin{bmatrix} 1 & 2 & 6 \\ 1/2 & 1 & 6 \\ 1/6 & 1/6 & 1 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 1 & 1/2 & 1 \\ 2 & 1 & 2 \\ 1 & 1/2 & 1 \end{bmatrix} \quad M_4 = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 1 & 2 \\ 1/2 & 1/2 & 1 \end{bmatrix}$$

$$M_5 = \begin{bmatrix} 1 & 1/2 & 2 \\ 2 & 1 & 3 \\ 1/2 & 1/3 & 1 \end{bmatrix} \quad M_6 = \begin{bmatrix} 1 & 3 & 8 \\ 1/3 & 1 & 4 \\ 1/8 & 1/4 & 1 \end{bmatrix}$$

$$M_7 = \begin{bmatrix} 1 & 1/3 & 1/2 \\ 3 & 1 & 2 \\ 2 & 1/2 & 1 \end{bmatrix} \quad M_8 = \begin{bmatrix} 1 & 6 & 4 \\ 1/6 & 1 & 1/3 \\ 1/4 & 3 & 1 \end{bmatrix}$$

$$M_9 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad M_{10} = \begin{bmatrix} 1 & 1/3 & 1 \\ 3 & 1 & 2 \\ 1 & 1/2 & 1 \end{bmatrix}$$

Second, obtain the eigenvectors and the max characteristic roots of the 10 matrixes. Eigenvectors should be normalized. The results are as follows.

Normalized eigenvectors:

$w_1 = (0.4286, 0.4286, 0.1429)$, $w_2 = (0.5644, 0.3591, 0.0765)$, $w_3 = (0.25, 0.5, 0.25)$, $w_4 = (0.4, 0.4, 0.2)$, $w_5 = (0.2973, 0.5390, 0.1638)$, $w_6 = (0.6690, 0.2572, 0.0738)$, $w_7 = (0.1638, 0.5390, 0.2973)$, $w_8 = (0.6853, 0.0934, 0.2213)$, $w_9 = (0.3333, 0.3333, 0.3333)$, $w_{10} = (0.2106, 0.5485, 0.2409)$.

Max characteristic roots:

$\lambda_1 = 3$, $\lambda_2 = 3.0539$, $\lambda_3 = 3$, $\lambda_4 = 3$, $\lambda_5 = 3.0592$, $\lambda_6 = 3.0184$, $\lambda_7 = 3.0092$, $\lambda_8 = 3.0541$, $\lambda_9 = 3$, $\lambda_{10} = 3.0183$.

TABLE 5. Information of the tasks.

Task	Title_T	Kws_T	Des_T
T_{11}	Takt-time computation of the machine shop production line	takt-time computation; line balancing;	Compute the takt-time of the production line in the premise of meeting the requirement of production capacity, and achieve the line balancing.
T_{12}	Equipment selection of the machine shop	machine shop; equipment selection; production line design;	Finish the equipment selection and the production line design in the machine shop.
T_{13}	Layout design of the machine shop equipment	machine shop; layout design;	Finish the layout design of the selected equipment in the given field.
T_2	The supporting structure design and calculation of the CNC machine tool's spindle and bearings selection	CNC; spindle; supporting structure; design; bearing; selection; mathematical model; precision; rigidity; reliability; dynamic properties; analysis; span; lifetime; load; calculation; structure drawing	Construct the mathematical model for CNC machine tool's spindle properties analysis, including precision, rigidity, reliability and dynamic properties. Design the supporting structure of the spindle to achieve the precision, rigidity, reliability and dynamic properties requirements of the spindle. Select the proper type and combination of bearings, and determine the number and position of each bearing according to the supporting structure and span. Check the load, precision and lifetime of each bearing according to the actual parameters. Draw the assembly drawing of the spindle.

Third, test the consistency of the results. The consistency ratio (CR) should be less than 0.1 to pass the consistency test. The CR is computed by using $CR = CI/RI$, where CI is the consistency index and is computed by using $CI = (\lambda - 1)/(n - 1)$; the RI is the random index; n is the rank of the evaluation matrix. When $n = 3$, RI = 0.58. The test results are as follows.

Consistency indexes:

$CI_1 = 0$, $CI_2 = 0.027$, $CI_3 = 0$, $CI_4 = 0$, $CI_5 = 0.0046$, $CI_6 = 0.0092$, $CI_7 = 0.0046$, $CI_8 = 0.0271$, $CI_9 = 0$, $CI_{10} = 0.0092$.

Consistency ratios:

$CR_1 = 0$, $CR_2 = 0.0465$, $CR_3 = 0$, $CR_4 = 0$, $CR_5 = 0.0079$, $CR_6 = 0.0158$, $CR_7 = 0.0079$, $CR_8 = 0.0467$, $CR_9 = 0$, $CR_{10} = 0.0158$.

All the evaluation matrixes pass the consistency test. Thus, the w_t, w_k, and w_d can be obtained by computing the mean values of the normalized eigenvectors. The results are $w_t = 0.4002$, $w_k = 0.3998$ and $w_d = 0.2$.

2) DATASETS

A machine shop layout design task, T_1, and a computer numerical control (CNC) machine tool's spindle design and bearings selection task, T_2, are selected to validate the CED approach. T_1 consists of three subtasks, T_{11}, T_{12}, and T_{13}. The tasks' information is shown in Table 5.

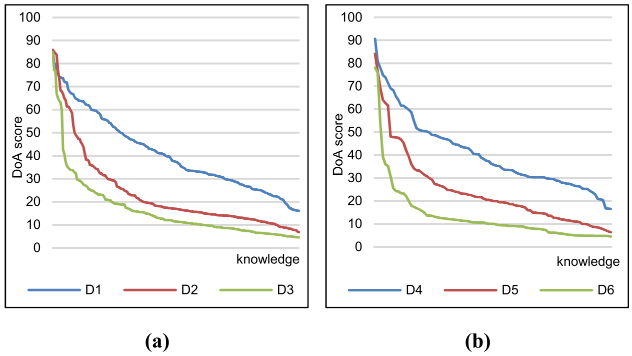


FIGURE 8. DoA scores of knowledge in the AS in descending order of both experiments. (a) DoA scores of knowledge in experiment 1. (b) DoA scores of knowledge in experiment 2.

We randomly selected 500 pieces of knowledge belonging to Node_T$_1$ and all 117 pieces of knowledge belonging to Node_T$_2$ to construct the original knowledge bases for T_1 and T_2. Recommending knowledge for T_1 and T_2 using the CED approach is named experiment 1 and 2 respectively.

Then, by using (17) on the original knowledge bases of the two experiments and according to the values of w_t, w_k and w_d, 193 pieces of knowledge and 92 pieces of knowledge that meet the requirement of $Sim_{obj}(T, K) > v$ are selected to create the alternative sets for T_1 and T_2, respectively. In each AS, the knowledge is successively numbered. One hundred pieces of recommended knowledge in Node_T$_1$ and Node_T$_2$ are randomly selected to create the training sets for T_1 and T_2, respectively. The IIG value of each meta-knowledge is computed by using (19)-(23). The first 20 pieces of knowledge in each AS are selected to create the test sets for each experiment. The information of pieces of knowledge in the test sets is listed in Table 6. In the table, the jth piece of knowledge in the test set i is denoted as k_{i-j}. Due to space constraints, Kws_K and Des_K are not listed.

B. EXPERIMENTAL SUBJECTS AND ENVIRONMENT

To better demonstrate the characteristics of the proposed CED approach and compare with other knowledge recommendation approaches, both experiments involve 45 designers with different levels of experience. In each experiment, 45 designers are consisting of 15 junior engineers (JEs), 15 intermediate engineers (IEs), and 15 senior engineers (SEs).

The SolidWorks Enterprise PDM system (2016 edition) is selected to establish the design workflow. The relevant information of the workflow is retrieved through secondary development using the application programming interfaces and then written into the knowledge and designer ontology models.

C. ANALYSIS OF THE CED APPROACH

Six designers, *D1* to *D6*, are randomly selected from the experimental subjects to show the computation results of the CED approach. *D1*, *D2*, and *D3* are selected from the JEs, IEs, and SEs from experiment 1, respectively. *D4*, *D5*, and *D6* are selected from the JEs, IEs, and SEs from

TABLE 6. Information of pieces of knowledge in the test sets.

ID_K	Title_K	Type	$H(k_i)$
$k_{1\text{-}1}$	The process of shop layout design	*workflow*	5.5850
$k_{1\text{-}2}$	Shop layout design examples	*image*	5.9219
$k_{1\text{-}3}$	The shop equipment selection method and line balancing calculation model	*model*	8.4773
$k_{1\text{-}4}$	The takt-time calculation method	*formula*	6.2288
$k_{1\text{-}5}$	Shop layout design method	*criterion*	8.2204
$k_{1\text{-}6}$	The principles of shop equipment selection	*criterion*	6.3219
$k_{1\text{-}7}$	The process of the production line design	*workflow*	5.4069
$k_{1\text{-}8}$	Machine shop and machine line introduction	*text*	6.9693
$k_{1\text{-}9}$	The calculation method of line production capacity	*formula*	7.4919
$k_{1\text{-}10}$	The introduction of machine process and equipment	*text*	4.6699
$k_{1\text{-}11}$	The design method and mathematical model of machine shop production line	*model*	9.5013
$k_{1\text{-}12}$	The principle of shop design and planning	*criterion*	7.3906
$k_{1\text{-}13}$	The calculation process of production planning	*workflow*	7.5471
$k_{1\text{-}14}$	The calculation method and formula of shop production capacity	*formula*	8.3685
$k_{1\text{-}15}$	The selection method of production line equipment parameters	*criterion*	8.9338
$k_{1\text{-}16}$	The graphic method of shop layout design	*image*	8.0288
$k_{1\text{-}17}$	The goal and principle of shop layout design	*criterion*	5.9069
$k_{1\text{-}18}$	The goal and principle of line balancing	*criterion*	7.9698
$k_{1\text{-}19}$	The method and principle of machine shop design	*criterion*	7.8326
$k_{1\text{-}20}$	The selection method of machine process parameters	*text*	8.6452
$k_{2\text{-}1}$	The assembly drawing of the CNC machine tool's typical spindles box structure	*image*	6.8885
$k_{2\text{-}2}$	The mathematical model for property analysis of CNC machine tool's spindles supporting structure	*model*	9.3265
$k_{2\text{-}3}$	The typical forms of CNC machine tool's spindles supporting structure	*image*	7.1655
$k_{2\text{-}4}$	The types and selection methods of the commonly used rolling bearings of CNC machine tool's spindles	*criterion*	7.0738
$k_{2\text{-}5}$	The optimal layout design method of CNC machine tool's spindles bearings	*text*	8.6805
$k_{2\text{-}6}$	The rigidity calculation formulas of CNC machine tool's spindles components	*formula*	6.7390
$k_{2\text{-}7}$	The calculation formula of CNC machine tool's spindles bearings load	*formula*	6.0033
$k_{2\text{-}8}$	The precision grade table of CNC machine tool's spindles bearings	*criterion*	6.5850
$k_{2\text{-}9}$	The calculation formula of CNC machine tool's spindles bearings lifetime	*formula*	6.0033
$k_{2\text{-}10}$	The optimal span calculation diagram of CNC machine tool's spindles	*image*	7.3939
$k_{2\text{-}11}$	The analysis process of the CNC machine tool's spindles components dynamic properties	*workflow*	7.5433
$k_{2\text{-}12}$	The supporting structure of radial-thrust ball bearings	*image*	4.1505
$k_{2\text{-}13}$	The selection method of CNC machine tool's spindles bearings	*text*	6.8955
$k_{2\text{-}14}$	The application situation and method of CNC machine tool's spindles bearings	*text*	7.3582
$k_{2\text{-}15}$	The design method of CNC machine tool's spindles span	*text*	7.3939
$k_{2\text{-}16}$	GB/T 9160.2-2006 Locknut and locking device: the attachment of rolling bearings	*criterion*	1.5488
$k_{2\text{-}17}$	GB/T 275-1993 The mate between rolling bearings and the spindles and the shells: the attachment of rolling bearings	*criterion*	4.8968
$k_{2\text{-}18}$	The typical mate forms of rolling bearings	*image*	4.7893
$k_{2\text{-}19}$	The form and the early warning of the bearings' failures	*text*	3.6822
$k_{2\text{-}20}$	The typical structures of rolling bearings combination	*image*	5.0425

experiment 2, respectively. The DoA scores of pieces of knowledge and the distribution of correlations between the task and knowledge in the AS for the six designers are shown in Fig. 8 and Fig. 9 to show the overall characteristics of the CED approach.

Fig. 8 clearly shows that the DoA scores overall decrease as the designers' experience increase. For an arbitrary designer, the DoA scores sharply decrease at first, and then the downward trend gradually slows down. The characteristic indicates that only a few pieces of knowledge are of great assistance, and most pieces of knowledge are of limited assistance, which should not be recommended to the designer.

In Fig. 9, each point represents a piece of knowledge in the AS. The pieces of knowledge are divided into two types of feedback-based knowledge (FBK) and collective intelligence-based knowledge (CIBK), and the top 5 pieces of knowledge with the highest DoA scores for each designer are also highlighted. Differences can be seen from the comparisons between Fig. 9(a) to Fig. 9(c) and Fig. 9(d) to Fig. 9(f). First, designers are obviously less supported by the IIG algorithm as their experience increases, which means the CED approach can well retain the designer's autonomy. Second, in the region with high DoC value, the density of knowledge for experienced designers is lower than for inexperienced designers, indicating that the distinction of knowledge for designers is determined by the experience of the designers. For experienced designers, the number of pieces of knowledge that are actually assistive is low because tasks can be completed on his/her own. Third, the top 5 pieces of knowledge for all the six designers are not the pieces of knowledge with the highest DoC values, which reflects the comprehensive consideration of DoC and DoD in the CED approach.

Fig. 8 and Fig. 9 show the overall situations of all the knowledge in the AS. The same characteristics can also be observed in the test sets. The detailed information of the top 5 pieces of knowledge in terms of the DoA score in the test

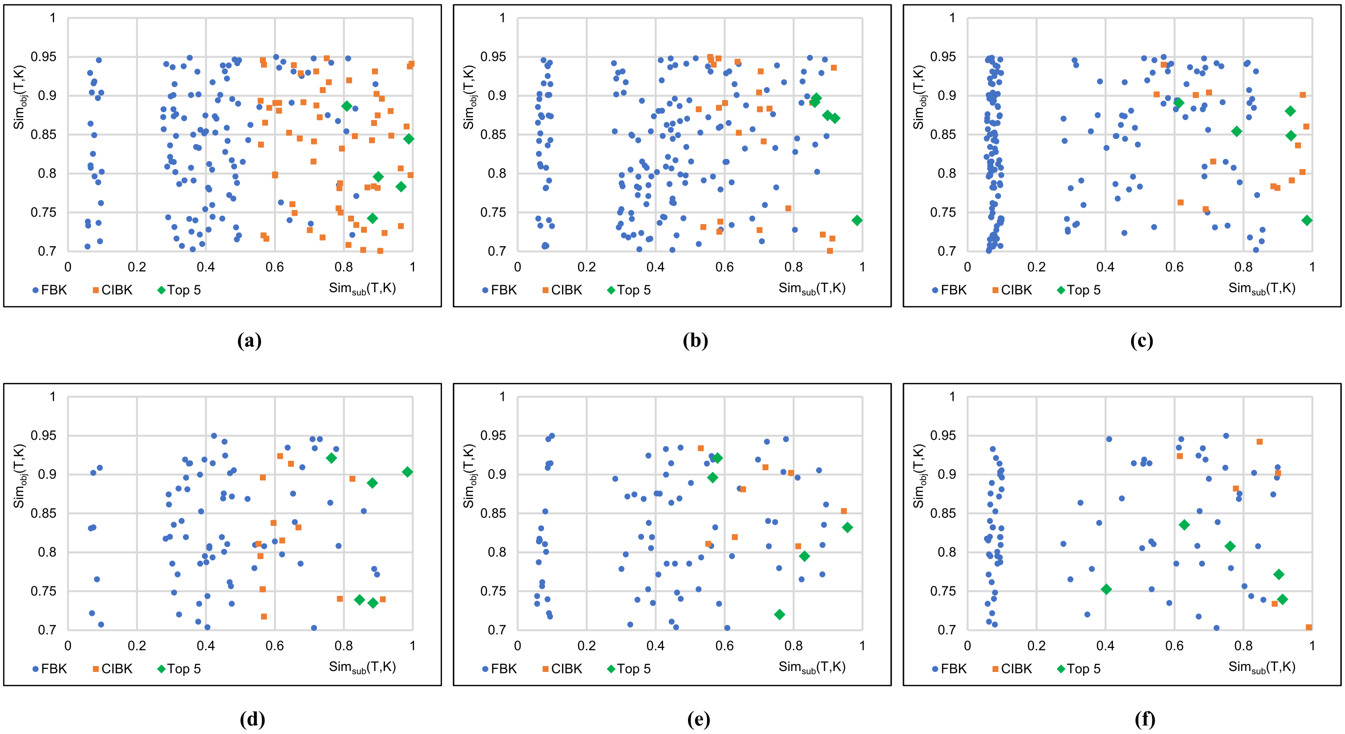


FIGURE 9. Distribution of correlations between task and knowledge in the AS for the six designers. (a) D1. (b) D2. (c) D3. (d) D4. (e) D5. (f) D6.

sets for the six designers are listed in Table 7. The results are actually the RL for the six designers when $n = 5$.

(1) The recommendation results for different designers are quite different. The recommendation results for most designers are different from either the DoC or the DoD values in descending order, which reflects the comprehensive consideration of these factors.

(2) The DoC value and the DoD value have a different impact on the DoA score for different designers. For JEs, the DoC value is the more significant factor. This is because the DoD values of most pieces of knowledge for JEs are high, so the difference between DoA scores is mainly reflected in the DoC values. For SEs, the situation is quite the opposite. For IEs, the DoD value has a relatively greater impact on the DoA score for the DoC value. This characteristic seems to ignore the applicability of the recommendation results. However, in fact, it can greatly reduce the design time. When completing tasks, the most time-consuming process is searching the applicable knowledge. For most designers, searching for unfamiliar knowledge requires more time than searching for familiar knowledge. The DoD module in the CED approach can greatly shorten the time to search for unfamiliar knowledge and consequently improve design efficiency. Meanwhile, designers are assumed to have less access to knowledge with a low DoA score. According to the forgetting curve, the demand for such knowledge will increase gradually, and the DoA score will increase accordingly. Therefore, the DoA score of each knowledge will appear with some kind of periodicity. This characteristic can assist designers in becoming more familiar with the knowledge in the knowledge base and improve their ability to acquire and properly use this part of knowledge in the shortest time.

(3) The CED approach can well retain a designer's autonomy and assist him/her precisely. For example, as a senior engineer, D6 is assumed to have rich experience. However, the DoD value for D6 on k_{2-2} indicates that he/she has never accessed this piece of knowledge. Thus, compared with other knowledge in the RL, the DoA value of k_{2-2} is high enough to rank it at the top of the RL, which increases the possibility for D6 to see it.

The CED approach can appropriately guide inexperienced designers by using collective intelligence and can well retain the personalized demands of experienced designers by using their experience. The DoC value neither completely relies on personalized experience nor collective intelligence but rather uses the flexible adjustment of their combination according to the designers' access to knowledge information.

Compared with most knowledge recommendation algorithms [6], [21]–[23], [29], another advantage of the CED approach is the cold start problem can be well disposed of. If a designer has not accessed any knowledge, the subjective correlation value to all knowledge is 0, and the DoD values are all 100. Therefore, the DoA score is completely dependent on the objective correlation value. The knowl-

TABLE 7. Top five pieces of knowledge in the test sets for the six selected designers using CED approach.

	Order in RL	ID_K	Sim_{obj}	Sim_{sub} FB[a]	Sim_{sub} CIB[b]	DoC	DoD	DoA
D1	1	$k_{1\text{-}18}$	0.8867	N/A	0.8090	0.8007	96.7021	77.4451
	2	$k_{1\text{-}3}$	0.7018	N/A	0.8566	0.7183	100	71.8267
	3	$k_{1\text{-}17}$	0.7988	N/A	0.6012	0.6397	100	63.9736
	4	$k_{1\text{-}9}$	0.8605	N/A	0.9827	0.8689	71.4901	62.1176
	5	$k_{1\text{-}7}$	0.9175	N/A	0.7571	0.7737	72.0557	55.7521
D2	1	$k_{1\text{-}19}$	0.8971	N/A	0.8663	0.8447	100	87.4743
	2	$k_{1\text{-}14}$	0.7253	N/A	0.5849	0.6079	100	60.7900
	3	$k_{1\text{-}10}$	0.9221	0.2857	N/A	0.4875	94.0210	45.8331
	4	$k_{1\text{-}4}$	0.9150	0.0950	N/A	0.4029	89.0335	35.8740
	5	$k_{1\text{-}15}$	0.8727	0.0712	N/A	0.3916	91.0929	35.6744
D3	1	$k_{1\text{-}5}$	0.8746	0.4481	N/A	0.5678	61.2998	34.8071
	2	$k_{1\text{-}20}$	0.9018	N/A	0.5487	0.6301	53.6531	33.8057
	3	$k_{1\text{-}17}$	0.7988	0.0640	N/A	0.3839	75.7964	29.0985
	4	$k_{1\text{-}15}$	0.8727	0.6321	N/A	0.6775	41.6358	28.2080
	5	$k_{1\text{-}11}$	0.7835	N/A	0.8870	0.7833	32.3427	25.3355
D4	1	$k_{2\text{-}13}$	0.8893	0.8830	N/A	0.8512	95.0441	80.9026
	2	$k_{2\text{-}12}$	0.8946	N/A	0.8253	0.8154	87.5692	71.4031
	3	$k_{2\text{-}10}$	0.7796	0.5409	N/A	0.6010	86.0343	51.7027
	4	$k_{2\text{-}1}$	0.7037	0.4094	N/A	0.5144	92.8199	47.7472
	5	$k_{2\text{-}6}$	0.8693	0.4502	N/A	0.5683	82.3615	46.8067
D5	1	$k_{2\text{-}17}$	0.7484	0.4625	N/A	0.5524	86.3469	47.6970
	2	$k_{2\text{-}3}$	0.8638	0.4457	N/A	0.5651	74.4630	42.0776
	3	$k_{2\text{-}7}$	0.8755	0.4123	N/A	0.5484	59.0107	32.3632
	4	$k_{2\text{-}1}$	0.7037	0.4596	N/A	0.5400	50.6867	27.3685
	5	$k_{2\text{-}9}$	0.8614	0.8947	N/A	0.8402	32.3030	27.1408
D6	1	$k_{2\text{-}2}$	0.7397	N/A	0.9136	0.7601	100	76.0127
	2	$k_{2\text{-}11}$	0.9093	0.8994	N/A	0.8733	27.8258	24.2998
	3	$k_{2\text{-}20}$	0.8744	0.8866	N/A	0.8444	21.1702	17.8752
	4	$k_{2\text{-}17}$	0.7484	0.0797	N/A	0.3852	33.4407	12.8801
	5	$k_{2\text{-}1}$	0.7037	N/A	0.9899	0.7434	16.6075	12.3467

[a]FB: The subjective correlation based on feedback action
[b]CIB: The subjective correlation based on collective intelligence

edge with the highest objective correlation value will be ranked at the top of the RL. However, the CED approach also has a deficiency. At the beginning of the recommendation system, if the feedback actions given by the initial designers are not good enough, the subsequent designers will be misled, thereby slowing the convergence of the approach.

The following comparisons will demonstrate the advantages of the CED approach.

D. BASELINES FOR COMPARISON

Three knowledge recommendation algorithms are adopted as baselines for comparisons with the CED approach.

1) CIG

The cognitive information gain (CIG) algorithm is proposed by Gao *et al.* [29] for personalized recommendation. Since the ''collective intelligence-based subjective correlation'' part of the CED approach is inspired by CIG, it would be straightforward to compare the CED approach with CIG.

2) NB

The Naïve Bayes (NB) algorithm [34] is a classic recommendation algorithm, which classifies knowledge into different categories by the historical data.

3) ITEM-CF

The item-based collaborative filtering (Item-CF) algorithm [23] is a typical recommendation algorithm that uses the collective intelligence. In engineering design field, inexperienced designers should improve their abilities by learning from experienced designers, so the Item-CF is more applicable than the user-based CF.

E. EVALUATION METRICS

Since the CED approach is a personalized recommendation approach and the ''what to recommend'' problem is solved by using the cosine similarity algorithm, the ''how to recommend'' problem (i.e. the rank of pieces of knowledge in the RL) is of more concern. Therefore, three ranking-oriented evaluation metrics, namely, the normalized discounted cumulative gain (NDCG@n) [35], the half-life utility (HLU@n) [36], and the mean average precision (MAP@n) [36] are adopted for evaluations.

For illustrative purposes, some symbols are defined in this section. For all the metrics, n is the length of the RL and u is the test subject. Because each recommendation approach has its own way to evaluate knowledge, $Scr_m(u, i)$ is defined as the evaluation score of the ith piece of knowledge in the RL for user u using method m, and $m \in$ {CED, CIG, NB, Item-CF}.

1) NDCG@n

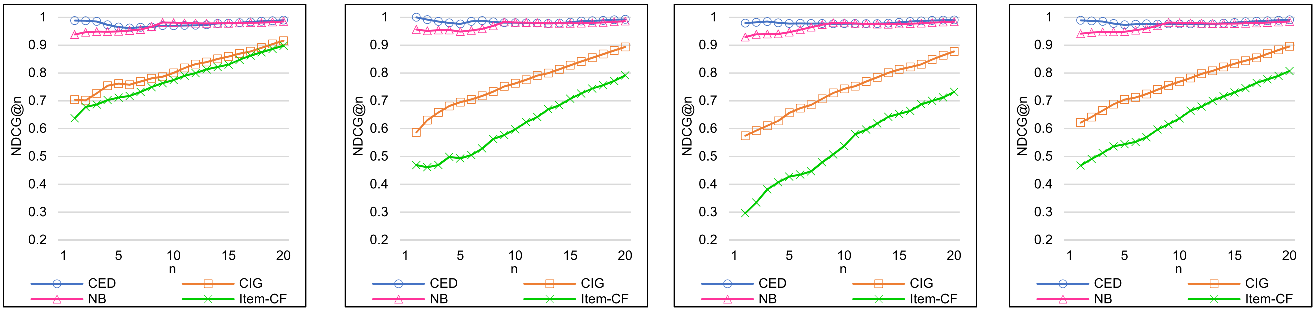
The NDCG@n is used to evaluate the cumulative gain value of an RL in a specific rank. The idea of the NDCG@n is that if a highly relevant result ranks low in the RL, its influence should be discounted. The NDCG@n is computed using (32).

$$\begin{cases} NDCG@n = \dfrac{1}{Num\,(u)} \displaystyle\sum_{u=1}^{Num(u)} NDCG_u@n \\ NDCG_u@n = \dfrac{DCG_u@n}{Max\,(DCG_u@n)} \\ DCG_u@n = \displaystyle\sum_{i=1}^{n} \dfrac{Scr_m\,(u,i)}{\log_2\,(i+1)} \end{cases} \tag{32}$$

where $Max(DCG_u@n)$ is the maximum value of $DCG_u@n$ that can be obtained.

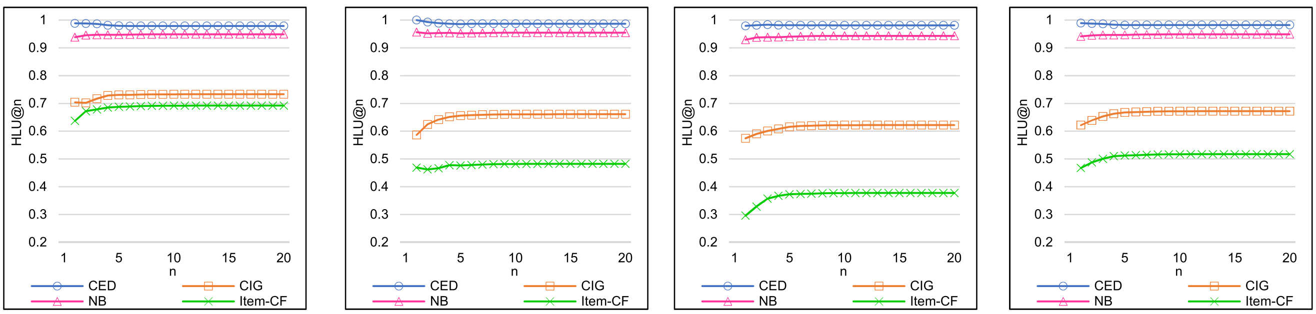
2) HLU@n

The HLU@n is used to evaluate the probability of a user clicking a recommended result. The idea of the HLU@n is that it assumes the user will successively click every recommended result in the RL with an exponential



(a) (b) (c) (d)

FIGURE 10. **The recommendation performance comparisons of experiment 1 using the NDCG@n evaluation metric. (a) The average NDCG@n of JEs. (b) The average NDCG@n of IEs. (c) The average NDCG@n of SEs. (d) The average NDCG@n of all subjects.**



(a) (b) (c) (d)

FIGURE 11. **The recommendation performance comparisons of experiment 1 using the HLU@n evaluation metric. (a) The average HLU@n of JEs. (b) The average HLU@n of IEs. (c) The average HLU@n of SEs. (d) The average HLU@n of all subjects.**

decay probability. The HLU@n is computed using (33).

$$\begin{cases} HLU@n = \dfrac{\sum_{u=1}^{Num(u)} HLU_u@n}{\sum_{u=1}^{Num(u)} Max\,(HLU_u@n)} \\ HLU_u@n = \sum_{i=1}^{n} \dfrac{Scr_m\,(u,i)}{2^{i-1}} \end{cases} \tag{33}$$

where $Max(HLU_u@n)$ is the maximum value of $HLU_u@n$ that can be obtained.

3) MAP@n

The MAP@n is used to evaluate the overall precision of a recommendation approach. Because the precision computation of MAP@n is relevant to the ranks of the recommended items, MAP@n is a ranking-oriented metric. The MAP@n is computed using (34).

$$\begin{cases} MAP@n = \dfrac{1}{Num\,(u)} \sum_{u=1}^{Num(u)} AP_u@n \\ AP_u@n = \dfrac{1}{n} \sum_{j=1}^{n} \dfrac{\sum_{i=1}^{j} Scr_m\,(u,i)}{j} \end{cases} \tag{34}$$

F. RESULTS OF THE EXPERIMENTS

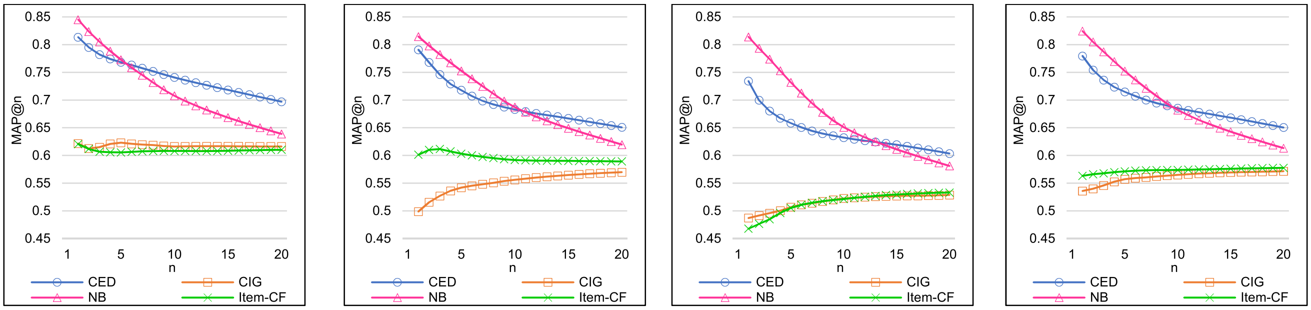
The recommendation performance comparisons between the CED approach and the three baselines using the three evaluation metrics of NDCG@n, HLU@n, and MAP@n in experiment 1 are shown in Fig. 10, Fig. 11, and Fig. 12, respectively. The same comparisons of experiment 2 are shown in Fig. 13, Fig. 14, and Fig. 15.

The following observations can be obtained from Fig. 10 to Fig. 15.

From the evaluation metric's standpoint,

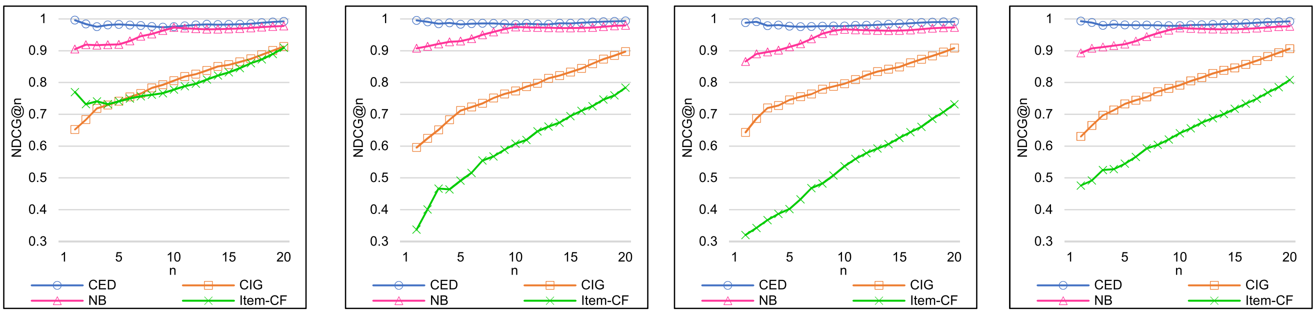
(1) Fig. 11 and Fig. 14 show that, in both experiments, the CED approach performs best against all the baselines using the HLU@n evaluation metric. In HLU@n, the importance of a piece of knowledge in the RL dramatically decays with its rank. Thus, the HLU@n only values the first several pieces of knowledge in the RL. The comparison results show that the CED approach has a good ranking ability for the first several pieces of knowledge in the RL.

(2) In terms of the NDCG@n evaluation metric, Fig. 10 and Fig. 13 indicate that the CED approach performs best against all the baselines when $n < 10$ and slightly better than the NB when $n > 10$ in both experiments. Compared with HLU@n, the downward trend of the knowledge's importance in the RL is gradually decreased. Thus, the NDCG@n reflects the overall



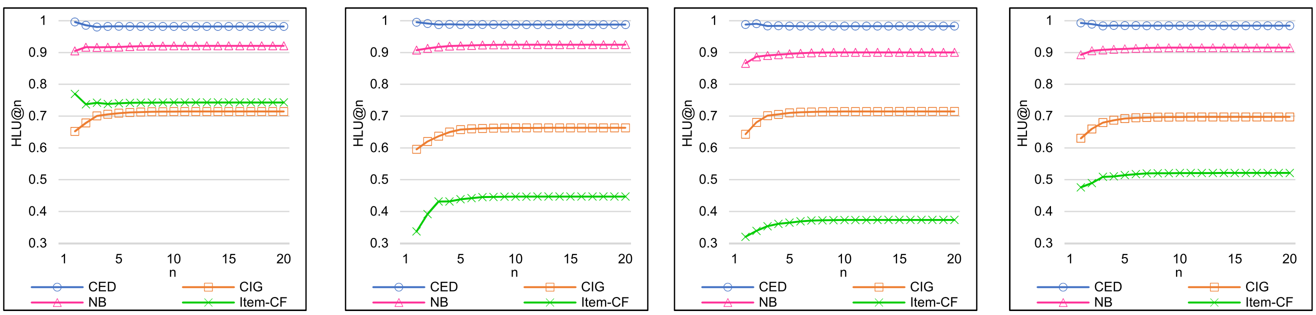
(a) (b) (c) (d)

FIGURE 12. **The recommendation performance comparisons of experiment 1 using the MAP@n evaluation metric. (a) The average MAP@n of JEs. (b) The average MAP@n of IEs. (c) The average MAP@n of SEs. (d) The average MAP@n of all subjects.**



(a) (b) (c) (d)

FIGURE 13. **The recommendation performance comparisons of experiment 2 using the NDCG@n evaluation metric. (a) The average NDCG@n of JEs. (b) The average NDCG@n of IEs. (c) The average NDCG@n of SEs. (d) The average NDCG@n of all subjects.**



(a) (b) (c) (d)

FIGURE 14. **The recommendation performance comparisons of experiment 2 using the HLU@n evaluation metric. (a) The average HLU@n of JEs. (b) The average HLU@n of IEs. (c) The average HLU@n of SEs. (d) The average HLU@n of all subjects.**

ranking ability of a recommendation approach. The comparison results show that the CED approach also has a good ranking ability for all the knowledge in the RL.

(3) When it comes to the MAP@n, the ranking ability of the CED approach is somewhat deficient. In Fig. 12 and Fig. 15, the performances of the CED approach are better than the CIG and Item-CF algorithm, but worse than the NB algorithm when n is small. The performances of the CED approach are better than all the baselines only when n is large. This is because the MAP@n evaluates performances by the precision, and the CED approach appropriately sacrifices precision for personalized demands.

From the engineers' standpoint,

(4) In both experiments, the performances of the CED approach do not significantly change with the experience of the designers using the three evaluation metrics. This characteristic enables the CED approach to recommend the appropriate knowledge to designers with any level of experience. The NB algorithm has the same characteristic. However, the NB algorithm achieves this by not considering the designers'

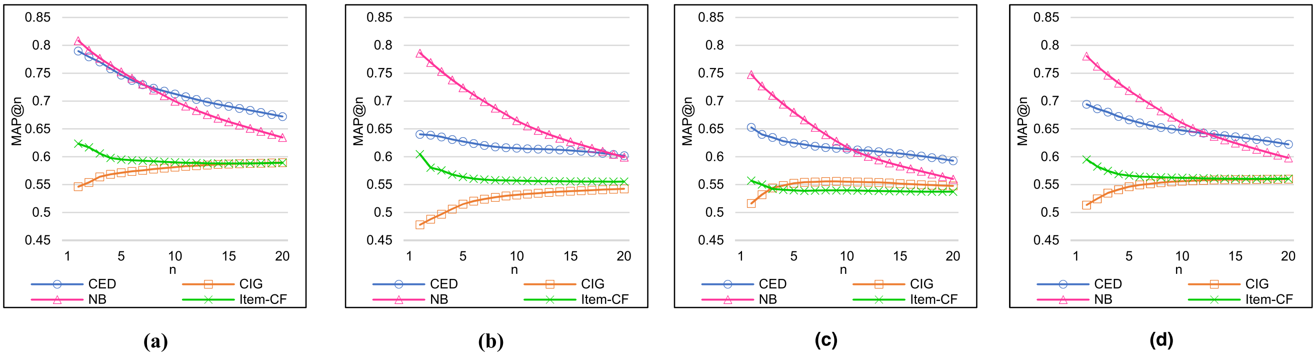


FIGURE 15. **The recommendation performance comparisons of experiment 2 using the MAP@n evaluation metric. (a) The average MAP@n of JEs. (b) The average MAP@n of IEs. (c) The average MAP@n of SEs. (d) The average MAP@n of all subjects.**

experience when recommendation, which will not fully meet the personalized demands of designers. The performances of the CIG and Item-CF algorithms decline as the designers' experience increase. The reason is these two algorithms use the collective intelligence to assist individuals, so the assistance will reduce as the designers' experience grows.

From the baselines' standpoint,

(5) The CED approach has an overall more stable performance than the other three baselines. Since the NB, CIG, and Item-CF algorithms are based on collective intelligence, their performances will be affected by the number and quality of the completed tasks. However, the CED approach is partially based on collective intelligence because the feedback of a designer is prior to collective intelligence. Meanwhile, all the sub-figures in Fig. 10 to Fig. 15 show that the classic Item-CF algorithm does not perform well in both experiments. The main reason for this is the Item-CF is based on the similarity between items, but the samples' sizes in experiments are not big enough for Item-CF to consider the applicability of the recommendation results and personalized demands of the designers simultaneously. Similarly, performances of the CIG algorithm are also not good enough when the sample size is small. In contrast, the CED approach considers only two parameters, the DoC and DoD values, its performance will not be affected by sample size.

VI. CONCLUSIONS AND FUTURE WORK

In summary, the CED approach proposed in this paper aims to solve the "what to recommend", "who to recommend", "when to recommend", and "how to recommend" problems simultaneously in the engineering field, especially the "how to recommend" problem. Consequently, it takes the correlation between knowledge and task, personalized experience, collective intelligence, and the demand for knowledge of designers into account. The correlation, experience, and demands are fully considered, and a personalized RL for every single designer is generated according to his/her access to knowledge information and the feedback on completed tasks. Finally, knowledge is comprehensively evaluated combining the DoC and DoD values, and the personalized knowledge recommendation service is achieved. Experimental results on two tasks in engineering field demonstrate that the proposed CED approach outperforms three baselines on three ranking oriented evaluation metrics.

The relationships between tasks, knowledge, and designers are studied in this paper. The correlation between tasks and knowledge and the demand of designers for knowledge are quantized. These ideas and algorithms lay the foundations for future studies on the fungibility and complementarity of knowledge and on the collaboration of multiple pieces of knowledge in completing a task. Meanwhile, they also provide theoretical and data support for further research into the correlation between designers' operations and their design intentions to provide more precise and personalized knowledge recommendation services.

REFERENCES

[1] D. Mourtzis, E. Vlachou, N. Boli, L. Gravias, and C. Giannoulis, "Manufacturing networks design through smart decision making towards frugal innovation," *Procedia CIRP*, vol. 50, pp. 354–359, 2016.

[2] J. Gao and A. Y. C. Nee, "An overview of manufacturing knowledge sharing in the product development process," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 232, no. 13, pp. 2253–2263, Nov. 2018. doi: 10.1177/0954405418759375.

[3] C. Renzi, F. Leali, and L. Di Angelo, "A review on decision-making methods in engineering design for the automotive industry," *J. Eng. Design*, vol. 28, no. 2, pp. 118–143, Jan. 2017. doi: 10.1080/09544828.2016.1274720.

[4] G. J. Hahm, M. Y. Yi, J. H. Lee, and H. W. Suh, "A personalized query expansion approach for engineering document retrieval," *Adv. Eng. Informat.*, vol. 28, no. 4, pp. 344–359, Oct. 2014. doi: 10.1016/j.aei.2014.04.002.

[5] G. J. Hamh, J. H. Lee, and H. W. Suh, "Semantic relation based personalized ranking approach for engineering document retrieval," *Adv. Eng. Informat.*, vol. 29, no. 3, pp. 366–379, Aug. 2015. doi: 10.1016/j.aei.2015.01.003.

[6] S. Y. Zhang, Y. Gu, X.-J. Liu, and J.-R. Tan, "A knowledge push technology based on applicable probability matching and multidimensional context driving," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 2, pp. 235–245, Feb. 2018. doi: 10.1631/FITEE.1700763.

[7] H. Zhu, J. Gao, and Q. Cai, "A product-service system using requirement analysis and knowledge management technologies," *Kybernetes*, vol. 44, no. 5, pp. 823–842, 2015. doi: 10.1108/K-11-2014-0244.

[8] R. S. Renu and G. Mocko, "Computing similarity of text-based assembly processes for knowledge retrieval and reuse," *J. Manuf. Syst.*, vol. 39, pp. 101–110, Apr. 2016. doi: 10.1016/j.jmsy.2016.03.004.

[9] G. M. Emelyanov, D. V. Mikhailov, and A. P. Kozlov, "The TF-IDF measure and analysis of links between words within N-grams in the formation of knowledge units for open tests," *Pattern Recognit. Image Anal.*, vol. 27, no. 4, pp. 825–831, Oct. 2017. doi: 10.1134/S1054661817040058.

[10] C.-M. Hwang and M.-S. Yang, "New similarity measures between generalized trapezoidal fuzzy numbers using the Jaccard index," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 22, no. 6, pp. 831–844, Dec. 2014. doi: 10.1142/S0218488514500445.

[11] C. Gómez-Tostón, M. Barrena, and A. Cortés, "Characterizing the optimal pivots for efficient similarity searches in vector space databases with Minkowski distances," *Appl. Math. Comput.*, vol. 328, pp. 203–223, Jul. 2018. doi: 10.1016/j.amc.2018.01.028.

[12] J. Ye, "Similarity measures of intuitionistic fuzzy sets based on cosine function for the decision making of mechanical design schemes," *J. Intell. Fuzzy Syst.*, vol. 30, no. 1, pp. 151–158, 2016. doi: 10.3233/IFS-151741.

[13] G. Wei, "Some cosine similarity measures for picture fuzzy sets and their applications to strategic decision making," *Informatics*, vol. 28, no. 3, pp. 547–564, 2017. doi: 10.15388/Informatica.2017.144.

[14] M. Lee and H. J. Zhao, "An improved cosine similarity algorithm based on document similarity," in *Proc. Int. Symp. Fuzzy Syst., Knowl. Discovery Natural Comput. (FSKDNC)*, Xiamen, China, Aug. 2014, pp. 196–204.

[15] W. L. Lu *et al.*, "Selecting a semantic similarity measure for concepts in two different CAD model data ontologies," *Adv. Eng. Informat.*, vol. 30, no. 3, pp. 449–466, Aug. 2016. doi: 10.1016/j.aei.2016.06.001.

[16] H. Zhao, Z. Wang, P. Liu, and B. Wu, "The normalized distance preserving binary codes and distance table," *J. Inf. Sci. Eng.*, vol. 33, no. 2, pp. 499–515, Mar. 2017.

[17] A. Kashyap *et al.*, "Robust semantic text similarity using LSA, machine learning, and linguistic resources," *Language Resour. Eval.*, vol. 50, no. 1, pp. 125–161, Mar. 2016. doi: 10.1007/s10579-015-9319-2.

[18] P. S. Huang *et al.*, "Learning deep structured semantic models for Web search using clickthrough data," in *Proc. CIKM*, San Francisco, CA, USA, 2013, pp. 2333–2338.

[19] H. Hua, K. Gimpel, and J. Lin, "Multi-perspective sentences similarity modeling with convolutional neural networks," in *Proc. EMNLP*, Lisbon, Portugal, 2015, pp. 1576–1586.

[20] Y. H. Yu, Y. Gao, H. Wang, and R. Wang, "Joint user knowledge and matrix factorization for recommender systems," *World Wide Web*, vol. 21, no. 4, pp. 1141–1163, Jul. 2018. doi: 10.1007/s11280-017-0476-7.

[21] D. H. Lee and P. Brusilovsky, "Improving personalized recommendations using community membership information," *Inf. Process. Manage.*, vol. 53, no. 5, pp. 1201–1214, 2017. doi: 10.1016/j.ipm.2017.05.005.

[22] X. Deng, Y. Zhong, L. Lü, N. Xiong, and C. Yeung, "A general and effective diffusion-based recommendation scheme on coupled social networks," *Inf. Sci.*, vol. 417, pp. 420–434, Nov. 2017. doi: 10.1016/j.ins.2017.07.021.

[23] T. Ha and S. Lee, "Item-network-based collaborative filtering: A personalized recommendation method based on a user's item network," *Inf. Process. Manage.*, vol. 53, no. 5, pp. 1171–1184, Sep. 2017. doi: 10.1016/j.ipm.2017.05.003.

[24] M. K. Najafabadi and M. N. Mahrin, "A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback," *Artif. Intell. Rev.*, vol. 45, no. 2, pp. 167–201, Feb. 2016. doi: 10.1007/s10462-015-9443-9.

[25] T. Liu, H. Wang, and Y. He, "Intelligent knowledge recommending approach for new product development based on workflow context matching," *Concurrent Eng.-Res. Appl.*, vol. 24, no. 4, pp. 318–329, Dec. 2016. doi: 10.1177/1063293X16640319.

[26] Y. H. Xu, G. Yin, Y. Nie, Z. He, and X. Dai, "Research on an active knowledge push service based on collaborative intent capture," *J. Netw. Comput. Appl.*, vol. 36, no. 6, pp. 1418–1430, Nov. 2013. doi: 10.1016/j.jnca.2013.04.010.

[27] F. P. Zhang, D. Wu, T. H. Zhang, Y. Yan, and S. I. Butt, "Knowledge component-based intelligent method for fixture design," *Int. J. Adv. Manuf. Technol.*, vol. 94, nos. 9–12, pp. 4139–4157, Feb. 2018. doi: 10.1007/s00170-017-1075-6.

[28] X.-R. Li, S.-H. Yu, J.-J. Chu, D.-K. Chen, and L.-J. Wu, "Double push strategy of knowledge for product design based on complex network theory," *Discrete Dyn. Nature Soc.*, vol. 2017, Oct. 2017, Art. no. 2078626. doi: 10.1155/2017/2078626.

[29] Y. Gao, Y. Feng, and J. Tan, "Exploratory study on cognitive information gain modeling and optimization of personalized recommendations for knowledge reuse," *J. Manuf. Syst.*, vol. 43, pp. 400–408, Apr. 2017. doi: 10.1016/j.jmsy.2017.01.003.

[30] Z.-S. Wang, L. Tian, Y.-H. Wu, and B.-B. Liu, "Personalized knowledge push system based on design intent and user interest," *Proc. Inst. Mech. Eng. C, J. Mech. Eng. Sci.*, vol. 230, no. 11, pp. 1757–1772, Jun. 2016. doi: 10.1177/0954406215584395.

[31] Z. Li *et al.*, "An ontology-based product design framework for manufacturing verification and knowledge reuse," *Int. J. Adv. Manuf. Technol.*, vol. 99, nos. 9–12, pp. 2121–2135, Dec. 2018. doi: 10.1007/s00170-018-2099-2.

[32] S. Lee, "Determination of priority weights under multiattribute decision-making situations: AHP versus fuzzy AHP," *J. Construct. Eng. Manage.*, vol. 141, no. 2, Feb. 2015, Art. no. 05014015. doi: 10.1061/(ASCE)CO.1943-7862.0000897.

[33] L. Averell and A. Heathcote, "The form of the forgetting curve and the fate of memories," *J. Math. Psychol.*, vol. 55, no. 1, pp. 25–35, Feb. 2011. doi: 10.1016/j.jmp.2010.08.009.

[34] S. Puntheeranurak and P. Pitakpaisarnsin, "Time-aware recommender system using Naïve Bayes classifier weighting technique," in *Proc. 2nd Int. Symp. Comput., Commun., Control Automat. (3CA)*, Singapore, 2013, pp. 266–269.

[35] H. Shi, Y. Tan, X. Zhu, and S. Wu, "Measuring stability and discrimination power of metrics in information retrieval evaluation," in *Proc. IDEAL*, Hefei, China, 2013, pp. 8–15.

[36] R. Pan *et al.*, "One-class collaborative filtering," in *Proc. IDCM*, Pisa, Italy, Dec. 2008, pp. 502–511.

XIYAN YIN was born in Xiangyang, Hubei, China, in 1988. He received the B.S. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 2011, where he is currently pursuing the Ph.D. degree in mechanical engineering.

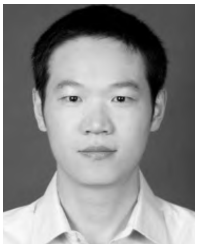
His research interests include digital manufacturing, intelligent manufacturing, and knowledge engineering.

BUYUN SHENG was born in Huarong, Hunan, China, in 1964. He received the B.S., M.S., and Ph.D. degrees in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 1985, 1988, and 2001, respectively.

He is currently a Professor with the School of Mechanical and Electronic Engineering, Wuhan University of Technology. He has authored two books and more than 100 articles. His research interests include digital and intelligent manufacturing, cloud manufacturing and service, manufacturing automation, and CAD/CAPP/PDM.

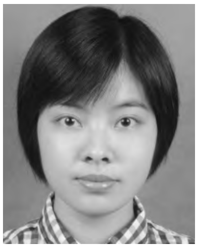
FEIYU ZHAO was born in Wuhan, Hubei, China, in 1992. He received the M.S. degree in mechanical engineering from Wuhan Polytechnic University, Wuhan, in 2016. He is currently pursuing the Ph.D. degree in mechanical engineering with the Wuhan University of Technology, Wuhan.

His research interests include the digital manufacturing, intelligent manufacturing, and CAD.



XINGGANG WANG was born in Laohekou, Hubei, China, in 1978. He received the Ph.D. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 2017.

He is currently an Associate Professor with the School of Mechanical and Electronic Engineering, Wuhan University of Technology. His research interests include the cloud manufacturing and its application, digital manufacturing, and enterprise informatization.



ZHENG XIAO was born in Wuhan, Hubei, China, in 1985. She received the B.S. degree in communication engineering and the M.S. and Ph.D. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, in 2006, 2009, and 2013, respectively.

She is currently an Assistant Professor with the School of Mechanical and Electronic Engineering, Wuhan University of Technology. Her research interests include the intelligent manufacturing, data mining, and enterprise informatization.

HUI WANG was born in Nanchang, Jiangxi, China, in 1993. He received the B.S. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 2015, where he is currently pursuing the Ph.D. degree in mechanical engineering.

His research interests include the scheduling, digital manufacturing, and project management.