https://doi.org/10.1016/j.knosys.2021.106739

A Context-aware Diversity-oriented Knowledge Recommendation Approach for Smart Engineering Solution Design

Xinyu Li[1,2,3], Chun-Hsien Chen[2], Pai Zheng[4], Zuhua Jiang[3], Linke Wang[3]

[1] *Delta-NTU Corporate Laboratory for Cyber-Physical System, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798*

[2] *School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore, 639798*

[3] *Department of Industrial Engineering and Management, Shanghai Jiao Tong University, Shanghai, China, 200240*

[4] *Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong, China, 999077*

Abstract: To proactively fulfill multiple stakeholders' needs in the engineering solution design process, the knowledge recommendation approach is adopted as a key element in the knowledge management system. Nevertheless, most existing knowledge recommendation approaches cannot simultaneously meet the higher standard of in-context accuracy and diversity. To address the issue, this paper proposes a context-aware diversity-oriented knowledge recommendation approach, thereby assisting stakeholders to accomplish engineering solution design in a smarter manner. Three diversity concerns, namely item-diversity, context-diversity, and user-diversity are addressed by semantic-based content analysis, context definition and awareness, and user profile modelling, respectively. Hence, the proposed approach not only maximizes the diversity of the recommended knowledge but also guarantees its accuracy under multiple problem-solving contexts. Moreover, a practical engineering solution design case on a Smart 3D printer platform is conducted, to validate the efficacy of the proposed approach in providing usable and diverse knowledge items. It is anticipated this work can provide useful insights to practitioners in their knowledge-based engineering solution design process.

Keywords: knowledge recommendation; diversity; context-aware; engineering solution design; engineering knowledge

Notations and Abbreviations

$\mathbf{U}$	Set of users, $\mathbf{U} = \{U_1, \ldots, U_M\}$	$BRH_L(U)$	List of recent L records in the browsing history
$\mathbf{I}$	Set of knowledge items, $\mathbf{I} = \{I_1, \ldots, I_N\}$	$CRH(U)$	List of records in the creating history
C	Problem-solving context	$REH(U)$	List of records in the revising history
O	Operation, $O \in \{Create, Revise, Browse\}$	$CxtSim$	Similarity between two contexts
ΔT	Duration of an operation	$WordSim$	Similarity between two keywords
R	Behavior record, $R = \{U, I, C, \Delta T, O\}$	$ItemSim$	Similarity between two knowledge items
L	Length of the user's recent preference	$ProSim$	Similarity between two user professions
v	Value of a context feature	$UserSim$	Similarity between two users
w	Keyword in a knowledge item	$REL(I_1, I_2)$	Item relevance in solution design
$\mathbf{TP}$	Set of topics, $\mathbf{TP} = \{TP_1, \ldots, TP_t\}$	$Pref(U, I)$	User's preference on a knowledge item
$f(w)$	TF-IDF weight of a keyword	$Pref(\mathbf{U}, \mathbf{I})$	Preference matrix
$S(I)$	Sequence number of browsing an item	$Pro(U, TP)$	User's overall profession to a topic
$q(I)$	Quality of an item	$Usa(I, C)$	Item's usability under a context

$\mathbf{U}(I)$	Users who have browsed I	$\mathbf{I}(U)$	Items that have filled/prefilled preference
$Pf(U)$	User's profession on all keywords	$Div(RL)$	Diversity of the recommendation list
K	Number of neighbors	N	Number of recommended items
RL	List of recommended knowledge items	N_D	Number of deleted items in diversification
KMS	Knowledge management system	UCF	User-based collaborative filtering
NDCG	Normalized discounted cumulative gain	BPR	Bayesian Personalized Ranking

1. Introduction

To meet the rising user requirements from technical, social, and environmental aspects, engineering R&D is no longer restrained in an individual enterprise but becomes an open innovation process. For example, Smart Product-Service Systems that bundle e-services to customize the products [1], appear as a novel trend in the industry. It allows various stakeholders (e.g. manufacturers, service providers, customers) co-designing the functionalities, co-implementing the process, and co-evaluating the performance, which makes the engineering solution design process a more multidisciplinary knowledge-intensive design activity. To support the stakeholders and fulfill their possible knowledge need with high efficiency and efficacy, a knowledge management system (KMS) is essential. When stakeholders encounter technical problems in product design and service innovation, KMS is required to proactively supply a few but most-related knowledge items [2], thus assisting them to accomplish engineering solution design in a smarter way.

Directly impacting the performance of using/reusing knowledge resources in engineering solution design, the knowledge recommendation approach serves as the key element in KMS. Derived from universal recommendation approaches, previous studies of the knowledge recommendation mainly rely on knowledge item similarities and user similarities, and aim to improve the accuracy of the recommendation results [3]. However, under the engineering solution design scenario, these accuracy-centric recommendation approaches will bring two fatal issues, namely, *monotony topics* and *long-tail issues*. For the first issue, since similar items share similar accuracy in the recommendation, these approaches are inclined to provide a list of items that are most accurate but narrowly ranged in one or a few topics, which cannot fulfill the needs for multidisciplinary knowledge in product/service solution design [4]. For example, in co-developing an online maintenance service for a 3D printer to improve its sustainability, stakeholders need to thoroughly determine the service logic in the maintenance workflow, where all possible working conditions and corresponding cost/benefit analyses are necessary to concern. In this situation, knowledge items in multiple disciplines are simultaneously demanded, like printing constraints under different working conditions, failure modes in mechanical and electrical structure, material properties of the selected filaments, conflicts between the printing schedule and maintenance schedule, and the corresponding impacts to the environment and business [5]. If the recommended knowledge just covers a single or only a few aspects, stakeholders are hard to proceed with their solution design process and come up with a comprehensive solution. In this case, the knowledge recommendation approach will lose the usability and efficacy in supporting engineering solution design. Therefore, the diversity of the recommendation list should be a concern.

For the second issue, among all pieces of knowledge stored in KMS, only a small portion of frequently-rated items are more likely to be recommended, and in turn, more follow-up browsing records/user ratings are generated on these popular items. The rest and majority, named *long-tail items*, are hard to participate in the recommendation process and then gradually become discarded. As a long-term result, the suggested items are highly homogeneous to what stakeholders have received in the past [6], thus contributing to the unexpectedly poor performance in engineering solution design. Actually, different problem-solving contexts occur in the engineering solution design, where stakeholders will probably raise context-dependent questions and only generate specific knowledge needs beyond his/her expertise [7]. Popular knowledge items may fit for the majority of contexts and stakeholders, but reversely lose their pertinence in a specific case. Therefore, it is necessary to regard stakeholders' expertise and their facing contexts in the recommendation process, thus maximizing the pertinency of the provided knowledge and minimizing the occurrence of the already-known pieces in the recommendation list.

Aiming to fill the abovementioned gaps and supply higher satisfying results for engineering solution design, this paper proposes a context-aware diversity-oriented knowledge recommendation approach (CDKR), which fully concerns the diversities derived from knowledge items, problem-solving contexts, and user professions. The main contributions of this paper have threefold: (1) identifying three diversity requirements for the knowledge recommendation approach used in engineering solution design; (2) introducing two effective information sources to support engineering knowledge recommendation; and (3) proposing a user profession-based prefilling process and a context-based diversification strategy.

The remainder of this paper is organized as follows. Section 2 briefly introduces previous studies related to recommendation approaches and diversity strategies, proposes concerns in recommending knowledge for engineering solution design, and summarizes issues and requirements for CDKR. Section 3 proposes the framework of CDKR and articulates all its modules. To evaluate the performance of the proposed approach, Section 4 provides a case study using real data collected in the solution design of a Smart 3D printer platform, and further compared the proposed methodologies with former studies. At last, the conclusion and future work are highlighted in Section 5.

2. Related works

2.1 Recommendation approaches and diversity strategies

Learning and predicting users' interest according to their profiles and historical rating behaviors, the personalized recommender system is an essential tool to solve the 'information overload' issue [8]. The universal recommendation approaches leveraged in the recommender system can be mainly categorized into five types [2-3, 9]: *demographic/social-network-based recommendation (DB/SNB)*, *content-based/keyword-based recommendation (CB/KB)*, *collaborative filtering (CF), ontology/knowledge-based recommendation (OB/KB)*, and *hybrid filtering (HF)*, are briefly summarized in Table 1. Although multiple recommendation manners rely on different processing of user's profiles and historical rating behaviors, what they share is that they based on similarity computation, either of users, items, or both. Besides, the performance evaluation of these manners focuses overwhelmingly on only one important factor, namely, the accuracy in reproducing known user opinions that have been removed from a test dataset [10].

Table 1. Summary of universal recommendation approaches

Manners	Representatives	Principles	Pros	Cons
DB/SNB	[11-12]	Users with similar personal profiles / social relationships will have similar interests	• Easy to conduct • Low computing cost	• New item problem • Data collection issue due to privacy concerns
CB/KB	[13-15]	Recommend items that have similar content to the user's past liked items	• Transparency • User independence	• New user problem • Overspecialization • Heterogeneous data problem
CF	[16-19]	Find similar users and recommend their favorite items (user-based CF); or find a similarly rated item and recommend it to the same user (item-based CF)	• No overspecialization • Few input data needed	• Both new user and new item problems (cold start) • Rating sparsity problem • Long-tail issue
OB/KB	[9, 20-21]	Leverage domain ontology/knowledge graph to match items and user interests	• Explainable • No cold-start problem • No rating sparsity • No long-tail problem	• High cost in acquiring expert knowledge and constructing domain ontology • High difficulty in reasoning
HF	[22-25]	A combination of two or more abovementioned recommendation manners; or containing some novel deep learning/graph-based models.	• Overcome the most of limitations	• High cost and difficulties in the combination

However, for practical concern, a highly accurate recommendation may not be necessarily the targeted one. For instance, some popular but generic items may cater to massive users, which are more likely to be recommended [26]. Given this, introducing a diversity strategy will better fulfill users' needs. It will bring more recommendation opportunities to some less-accurate items. The diversity in the personalized recommendation is formally defined as *the average dissimilarity between all pairs of items in the recommendation list (RL)*, as calculated in Eq. 1 [27]. $Sim(I_i, I_j)$ indicates the similarity of two items in the list, and $|RL|$ indicates the length of the list.

$$Div(RL) = \frac{\sum_{I_i \in RL} \sum_{I_j \in RL, I_j \neq I_i} \left(1 - Sim(I_i, I_j)\right)}{\frac{1}{2}|RL|\left(|RL| - 1\right)} \qquad \text{(Eq. 1)}$$

Based on this fundamental equation, studies in diversity-oriented recommendation mainly focus on two folds, namely, proposing more appropriate dissimilarity/similarity computing manners for the diversity evaluation, and developing better diversification strategies in recommendation algorithms [3]. For diversity evaluation, more delicate calculations are performed by evolving Eq. 1, which further clarifies the item dissimilarity/similarity using the Gini coefficient, normalized discounted cumulative gain (NDCG), or other metrics [28-29]. The evaluation process is also refined to concentrate more on the user's experience and includes additional factors like choice probability, preferred genres/topics, and user satisfaction [4, 6, 23, 26, 30]. As for diversification strategies, its nature is a trade-off between diversity and accuracy. One common and easy-to-conduct strategy is *post-filtering*, which recommends top $N+N_D$ accurate items first, and then removes N_D items to achieve the highest diversity [31], or just select top N representative items after the clustering [32]. Another strategy regards this trade-off as a *multi-objective optimization*, which leverages swarm intelligence, simulated annealing, genetic algorithms, or other heuristic algorithms to re-rank the recommendation list and achieves a Pareto Optimality on all the metrics (e.g. precision, recall, diversity, and serendipity) [33-35].

2.2 Recommend knowledge for engineering solution design

In nowadays engineering solution design, advanced IoT techniques facilitate sensing data collection from ubiquitous-connected machines [36]. Meanwhile, massive stakeholders are also enabled to contribute their information through various channels (e.g. mobile crowdsensing networks [37]). A large volume of valuable knowledge is hence available and accessible to all the stakeholders in product/service operations and innovations. Under this novel situation in engineering solution design, although recommendation approaches mentioned in Section 2.1 can be directly transplanting to recommend knowledge items, two perspectives are further concerned to achieve an ideal knowledge recommendation performance.

First, unlike recommending music/movies, which has simple item representations, shallow logical inferences, and domain-independent applications, recommending knowledge largely relies on some prerequisites, namely, knowledge representation and reasoning. Knowledge representation processes the structured (e.g. numerical sensing records), semi-structured (e.g. formatted annotations) or unstructured (e.g. natural language discussions) datasets collected from engineering solution design into a unified format, and knowledge reasoning establishes the semantical and logical relationship among the formalized knowledge items [7, 38]. As briefly summarized in Table 2, multiple pre-processing techniques are leveraged in representing and reasoning knowledge items that have various sources and types. To fulfill the prerequisites, ontology techniques are widely adopted to define the concepts, instances, and relations embedded in the unstructured and semi-structured data, and natural language processing (NLP) techniques and graph-based techniques can further extract and infer the complex semantic relations.

Table 2. Prerequisites in recommending knowledge for engineering solution design

Studies	Recommended knowledge	Data source	Data type	Key pre-processing techniques
[26]	Documents for design tasks	Industrial knowledge base	Unstructured	Ontology; NLP
[39-40]	Component information for solving requirement	Online user discussion	Unstructured	Ontology; Graph embedding
[41]	Novel functionalities and solutions	Online webpages	Semi-structured	Ontology; NLP; Graph-based computing
[42]	Maintenance plans	Industrial cases	Semi-structured	Ontology; Case-based reasoning
[43]	Empirical rules for product/service reconfiguration	Standard surveys	Structured	Rough set; Cluster analysis
[44]	Design principles for service innovation	Operational logs	Structured	Co-relation analysis

Besides, since knowledge items are generated and leveraged under certain engineering scenarios, the context-dependency will largely impact the effectiveness of knowledge items in engineering solution design [45-46]. Therefore, the second concern in recommending knowledge items for engineering solution design is problem-solving contexts. These contexts are problematic situations characterized by the corresponding objectives and constraints, which desire specific knowledge to solve [2]. Considering the sorts and contents that can be cost-effectively collected from sensing devices and stakeholders' profiles, problem-solving contexts in engineering solution design can be modelled with four domain-independent types of features [41], namely, (1) *Physical context* (information about the surrounding environment, like time, speed, temperature); (2) *Operational context* (information about the running status, like error code, software version, maintenance history); (3) *Social context* (information about the nearby products and services, like peer product family, third-party service provider, spare part supplier); and (4) *User context* (information about the stakeholders and their usage, like demographics, usage experience). Obviously, when different problem-solving contexts occurred, stakeholders' unuttered knowledge needs will be altered, and the knowledge recommendation list needs to be diversified accordingly.

2.3 Summary

2.3.1 Issues in current knowledge recommendation approaches for engineering solution design

To assist stakeholders in engineering solution design in a smarter manner, higher standards of in-context accuracy and diversity are proposed in realizing a proactive knowledge supply. Summarized from the recent literature about knowledge recommendation and diversification strategies, two main issues are identified in the application scenario of engineering solution design.

Firstly, a comprehensive evaluation of the diversity of knowledge items is missing. In the conventional scenario of recommending music/movies with diversification strategies, the diversity calculation of Eq. 1 is simply evaluated by the similarity/dissimilarity of tags or genres of items. However, in the scenario of engineering solution design, knowledge items are transdisciplinary and context-dependent. To define and evaluate the diversity among knowledge items, a delicate inspection on the content of each item is necessary. Semantic relations among keywords, logical relations among sentences, and re-occurrence relations among problem-solving contexts should be concerned [2, 45].

Secondly, although a better diversity is achieved using current diversification strategies, regardless of users' facing contexts and actual knowledge needs, there is some randomness in filtering or re-ranking the initial recommendation list [33]. In fact, in engineering solution design, it is meaningless to recommend an irrelevant engineering knowledge item under a particular problem-solving context, though including this item in the recommendation list may result in a higher diversity. Meanwhile, if the user has strong backgrounds in specific fields and has already gained enough knowledge in solving some sorts of engineering solution design tasks, he/she may not generate the needs for the corresponding knowledge items. In designing the context-based diversification strategy for engineering solution design, the already-known items should be perceived first, and then re-ranked to latter positions, or filtered out with higher priorities.

2.3.2 Requirements for a novel knowledge recommendation approach

To meet with the abovementioned gaps, a novel knowledge recommendation approach for engineering solution design should be developed. Concerning the perspectives of diversity and context-awareness, three requirements are proposed for this novel approach:

1) *Item-diversity: Analyze semantic meanings in knowledge items*. Semantic-level content analysis is required to evaluate the dissimilarity/similarity of knowledge items in Eq.1, thus diversifying items using their containing concepts and relevance relations.

2) *Context-diversity: Provide context-based recommendations.* To consider the context-dependency in knowledge items, the knowledge recommendation approach needs to adjust its recommendation list, with the concern of problem-solving context information in the current task of engineering solution design.

3) *User-diversity: Self-adjust for the backgrounds of stakeholders*. As a further specified concern to the *User context*, stakeholders' levels of mastering an engineering domain are essential. It enables a more targeted prediction on their actual knowledge need in the knowledge recommendation for engineering solution design.

3. A context-aware diversity-oriented knowledge recommendation approach

3.1 The overall framework

Aiming to achieve high in-context accuracy and diversity simultaneously, this paper proposes a context-aware diversity-oriented knowledge recommendation approach (CDKR) for smart engineering solution design. Figure 1 depicts the overall flowchart of this approach, which contains four modules.

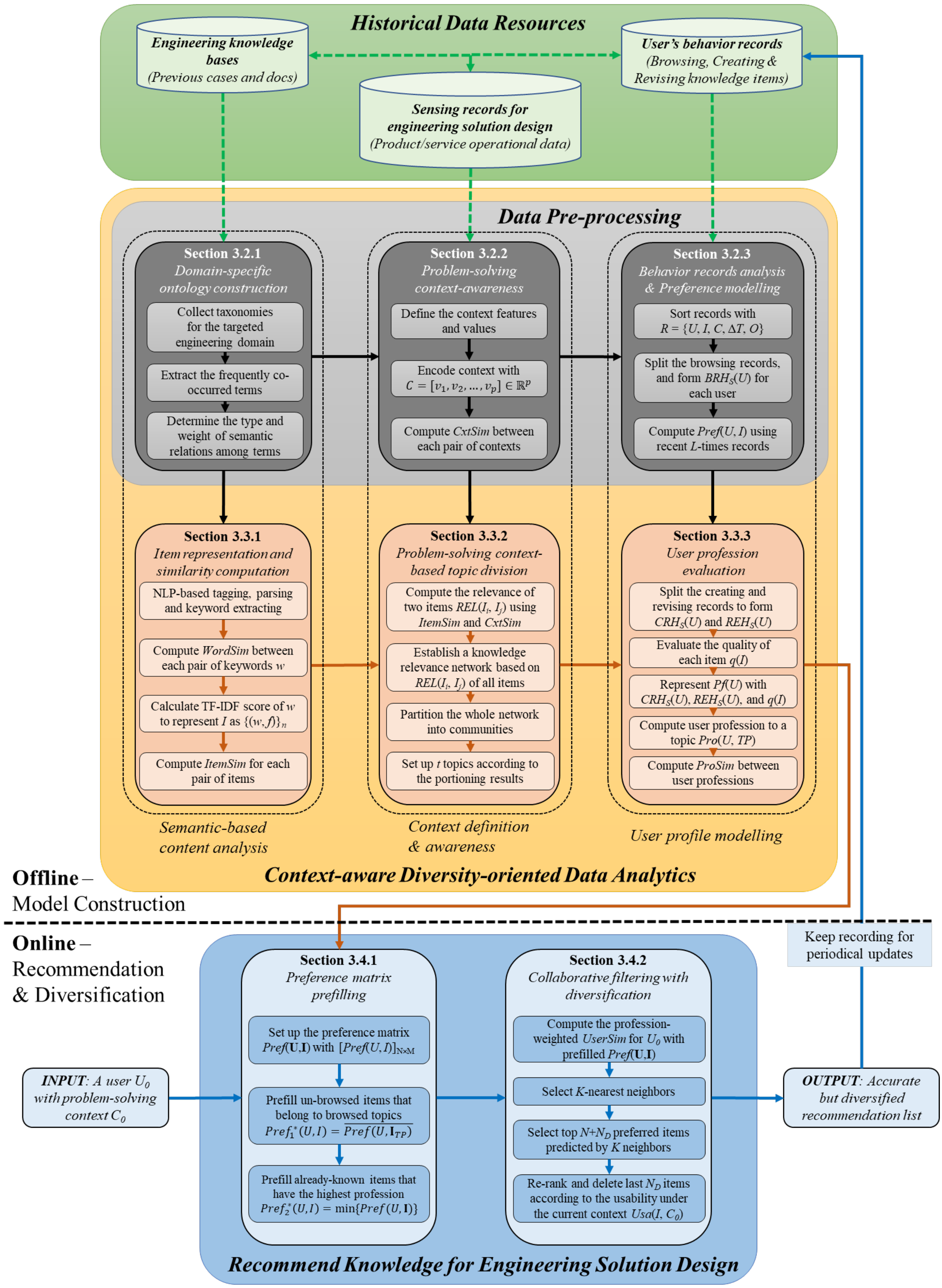

Figure 1. The overall flowchart of the proposed approach

The first module is constituted of three inter-related historical data resources. Among them, the *engineering knowledge base* stores knowledge items, and these items include accumulated cases and lesson-learned documents in the previous engineering solution design. As a typical data resource for the recommendation approach, *user's behavior records* on knowledge items (e.g. browsing, creating, or revising the items) are also collected and leveraged. Besides, a novel data resource, *sensing records for engineering solution design* (e.g. operational logs for product/service components), is involved in the recommendation. It contains abundant problem-solving context information and can be cost-effectively collected with IoT-enabled smart sensors.

Aiming to address the three requirements proposed in Section 2.3.2, the second and third modules, *data pre-processing* and *context-aware diversity-oriented data analytics*, derive three diversity-oriented data analytics models, namely, *semantic-based content analysis* aimed at *item-diversity*, *context definition & awareness* aimed at *context-diversity*, and *user profile modelling* aimed at *user-diversity*. As several former studies and research outcomes provide some mature methodologies for *constructing domain-specific ontology* [26, 47-49], *perceiving context information* [39-41, 50], and *modelling user preference with behavior records* [16, 23, 26], the process for data pre-processing will be briefly introduced. The subsequent processes, including knowledge items representation and similarity calculation, context-based topic division, and user profile modelling, will be illustrated with details in this paper.

The last module interacts with the user, who is a stakeholder encountering a particular problem-solving context in engineering solution design. A user profession-based collaborative filtering manner with context-based diversification strategies is conducted to recommend appropriate knowledge items, so as to assist him/her in accomplishing engineering solution design tasks. After the recommendation, the system will continue to record his/her following browsing/creating/revising behaviors, preparing for the next time recommendation and periodical updates of computing models.

3.2 Data pre-processing

3.2.1 Construct domain-specific ontology for evaluating semantic relations

Defining all the concepts and their relations in a particular engineering domain, domain-specific ontology lays the foundation for a semantic-based similarity calculation in the proposed manner. A typical process for constructing ontology is based on the taxonomies in engineering domains [47]. Taxonomy reveals the superior-subordinate relations among terms, and it usually serves as an index for the knowledge base in the related domain. An example of taxonomy is shown in Figure 2.

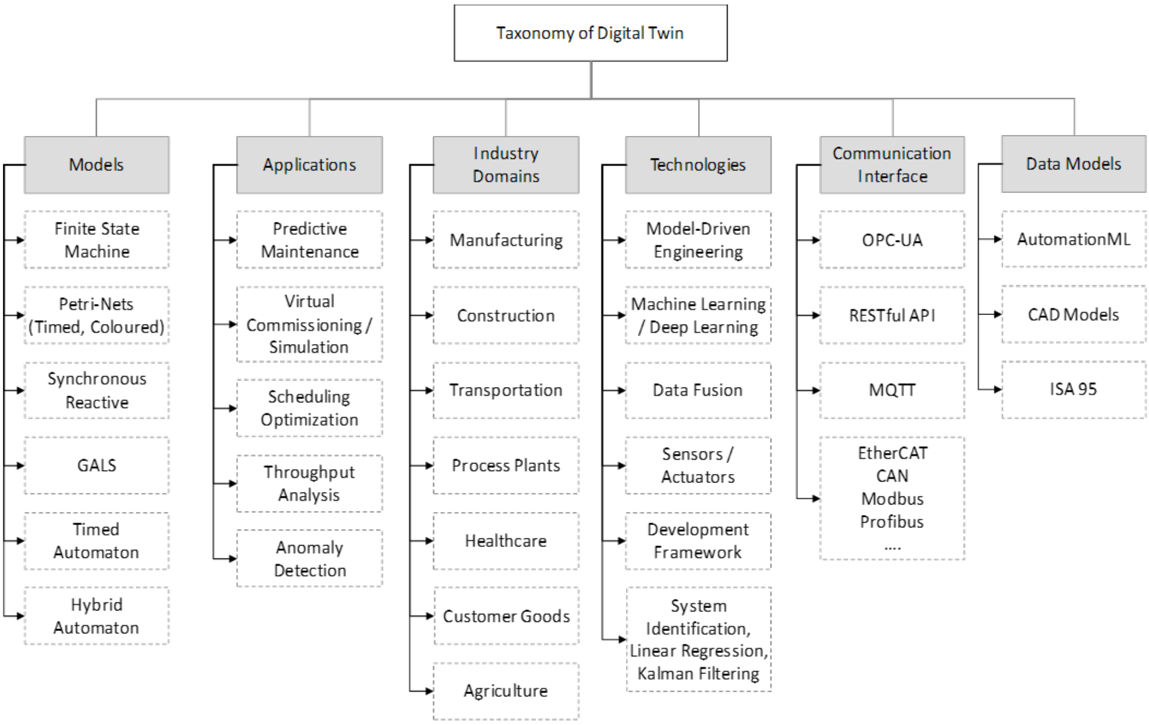

Figure 2. An illustrative taxonomy of *Digital Twin* used for engineering solution design

To merge the taxonomies and establish a complete ontology, a co-occurrence-based analysis on the contents of knowledge items stored in the knowledge base will largely alleviate the burden of domain experts. Specifically, if two terms frequently co-occur in the knowledge items, an additional semantic relation is established to link two terms in the ontology. Then, the domain experts will determine the type and weight of this relation, as referenced in Table 3. When all pairs of terms have been traversed, a complete domain-specific ontology is established.

Table 3. Type and weight of semantic relations in the ontology, referred from [48-49]

Type of semantic relation	**Weight**	**Instance**
Same / Synonym / Abbreviation	1.0	Engine *Synonym* Motor; IoT *Abbreviation* Internet of things
Similar	0.95	Transportation *Similar* Logistics
Is_a (superior-subordinate)	0.7	CAD model *Is_a* Data model
Cause / Effected_by	0.6	Anomaly detection *Cause* Fault analysis
Part_of	0.5	RFID *Part_of* IoT
Before / After	0.3	Predictive maintenance *Before* Production scheduling
(NULL)	0.0	(No relation established)
Antonym	-1.0	Negative sample *Antonym* Positive sample

3.2.2 Perceive problem-solving context in smart engineering solution design

In engineering solution design, stakeholders encountering different problem-solving contexts will raise different knowledge need. To diversify the recommendation list, massive product-sensed data and stakeholder-contributed data collected from IoT-enabled smart sensors can be leveraged to perceiving the context information [50].

Based on the context features summarized in Section 2.2, a specific problem-solving context can be encoded with key-value modeling, as illustratively shown in Figure 3. Specifically, for a problem-solving context containing p context features, a p-dimensional vector is encoded, namely, $C=\left[v_1, v_2, ..., v_p\right] \in \mathbb{R}^p$, where v_i is the value for the ith context feature. Note that the data collected from sensing devices and stakeholders' profiles are heterogeneous, Table 4 also lists out some frequently used data analysis manners for typical data sources in context value determination [39-40]. Based on the vectors, the similarity between two problem-solving contexts, $C_1=\left[v_1^1, v_2^1, ..., v_p^1\right]$ and $C_2=\left[v_1^2, v_2^2, ..., v_p^2\right]$, can be hence computed with Eq. 2.

$$CxtSim\left(C_1, C_2\right)=\frac{1}{p}\sum_{i=1}^{p} bool\left(v_i^1, v_i^2\right), bool\left(v_i^1, v_i^2\right)=\begin{cases}1 & v_i^1=v_i^2, v_i^1 \neq 0, v_i^2 \neq 0 \\ 0 & else\end{cases} \quad \text{(Eq. 2)}$$

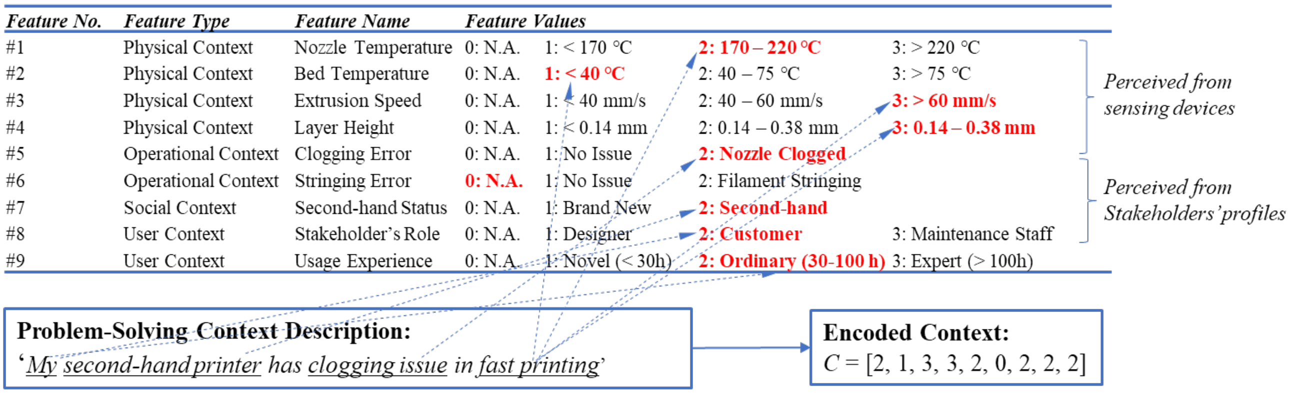

Figure 3. Encode problem-solving context with context features

Table 4. Data analysis manners in context value determination

Data sources	Stakeholder-contributed data			Product-sensed data
	Structural text or tag	Natural language	Numerical value	Numerical value
Analysis manners	Table headers & elements	Keyword extraction	Predefined rules	Predefined rules
	Formal concept analysis	Named-entity recognition	Fuzzy rules	Pattern recognition
	Schema-based annotation	Syntax analysis	Rough sets	Case-based reasoning
	Predefined template	Sentiment analysis	Classifiers	Decision tree
	……	……	……	……

3.2.3 Model user preference with behavior records

In recommending items of music/movies, user ratings on the items are leveraged for modelling user preference. However, it is sometimes hard to collect enough ratings, and also cannot guarantee the collected ratings truly reflect users' feelings. Behavior records, which are abundant and more difficult to be manipulated, are hence introduced instead. In this paper, a piece of behavior record can be represented as R = {U, I, C, ΔT, O}, indicating that a user U operates (O, includes *Create, Revise* and *Browse*) on a knowledge item I under a problem-solving context C with a duration of ΔT. Some sample behavior records are shown in Table 5.

Table 5. Sample behavior records sequentially logged by KMS

Record	User	Knowledge Item	Context	Duration (*s*)	Operation
1	U1: Jiang	I46: Prevent a clogged extruder (173 words)	[2, 3, 3, 2, 0, 2, 2, 1]	96	Browse
2	U3: Li	I33: Signs of overheating (44 words)	[3, 2, 2, 1, 1, 1, 2, 2]	114	Browse
…	…	…	…	…	…
76	U4: Hao	I21: First layer issues (154 words)	[2, 1, 1, 0, 0, 1, 1, 3]	330	Revise
…	…	…	…	…	…
105	U6: Chen	I43: Alleviate vibrations in fast printing (212 words)	[2, 3, 3, 1, 1, 1, 1, 3]	1147	Create
…	…	…	…	…	…

In modelling user preference, users' sequential browsing records are considered. Specifically, for a user U_i, a vector $BRH_S(U_i)$ represents his/her total Stimes browsing history, namely:

$$BRH_S(U_i) = \{(I_{1\text{st}}, C_{1\text{st}}, \Delta T_{1\text{st}}), (I_{2\text{nd}}, C_{2\text{nd}}, \Delta T_{2\text{nd}}), \ldots, (I_{S\text{th}}, C_{S\text{th}}, \Delta T_{S\text{th}})\} \quad \text{(Eq. 3)}$$

Since the user's preference changes over time, his/her latest L-times browsing history $BRH_L(U_i)$ will be leveraged to calculate his/her recent preference $Pref(U_i, I_j)$ on the knowledge item I_j, namely:

$$BRH_L(U_i) = \{(I_{S-L+1\text{th}}, C_{S-L+1\text{th}}, \Delta T_{S-L+1\text{th}}), (I_{S-L+2\text{th}}, C_{S-L+2\text{th}}, \Delta T_{S-L+2\text{th}}), \ldots, (I_{S\text{th}}, C_{S\text{th}}, \Delta T_{S\text{th}})\}$$

$$Pref(U_i, I_j) = \sum_{I_{S(I_j)\text{th}} = I_j} \frac{\Delta T_{S(I_j)\text{th}} / |I_j|}{S - S(I_j) + 1}, S(I_j) \in [S - L + 1, S] \quad \text{(Eq. 4)}$$

where $S(I_j)$ is the sequential number of his/her browsing on I_j in the history (i.e. $I_{S(I_j)\text{th}} = I_j$), and $|I_j|$ is the word count of I_j. L decides the range of the user's recent preference, which will be fine-tuned subsequently.

In this paper, the whole browsing history is utilized to evaluate the relevance among the knowledge items (Section 3.3.2), while the recent browsing history is leveraged to fill/prefill the preference matrix (Section 3.4.1) and recommend items with diversification (Section 3.4.2). As for the rest *Creating* and *Revising* records, they are adopted to evaluate the user's profession towards a specific engineering domain (Section 3.3.3).

3.3. Context-aware diversity-oriented data analytics

3.3.1 Represent knowledge items and compute their similarity

As shown in Eq. 1, the key consideration in diversification is to compute the similarity between two items. In this paper, the similarity between two knowledge items will be computed based on their words and phrases. Referring to Li et al. [41, 45], an NLP-based processing is detailed as follows:

Step 1 (Sentence segmentation): For a knowledge item *I*, split its content into sentences with punctuations;

Step 2 (Part-of-Speech tagging): For a sentence *s* in *I*, tag the part-of-speech and stem each word, and remain *notional words* labelled as *NN* (normal noun, singular), *VB* (verb, base form), *ADJ* (adjective), and *ADV*(adverb);

Step 3 (Syntactic parsing): Organize the tagged words of *s* with their syntactic relations, and remain relations labelled as *nsubj* (nominal subject), *dobj* (direct object), *amod* (adjectival modifier), and *advmod* (adverbial modifier).

Step 4 (Phrase compiling): Compile a phrase *p* in *s*, if *p* matches with the syntactic patterns of *NN*-(*amod*)-*NN*, *NN*-(*nsubj*)-*VB*, *VB*-(*dobj*)-*NN*, *ADJ*-(*amod*)-*NN*, or *ADV*-(*advmod*)-*VB*;

Step 5 (Concept mapping): Referring to the domain-specific ontology established in Section 3.2.1, if *p* matches with a term in the ontology, then map *p* with this term and regard it as a *compound word*; Also map other single words;

Step 6 (Keyword Extraction): Proceed to the next sentence with Step 2-5 until all the sentences have been processed; Extract all the *notional/compound words* in an item;

Step 7 (Item representation): A knowledge item is represented with containing keywords and their term frequency – inversed document frequency (TF-IDF):

$$I = \{(w_1, f_1), (w_2, f_2), \ldots, (w_n, f_n)\}, \; f_i = \frac{Count(w_i)}{\sum_{w \in I} Count(w)} \times \log \frac{|\mathbf{I}|}{1 + |\mathbf{I}_{w_i}|} \qquad \text{(Eq. 5)}$$

where $|\mathbf{I}|$ is the count of knowledge items stored in the knowledge base, and $|\mathbf{I}_{wi}|$ is the count of items that contain w_i;

Step 8 (Item similarity computing): For two knowledge items, $I_1 = \{(w_1^1, f_1^1), (w_2^1, f_2^1), \ldots, (w_{n_1}^1, f_{n_1}^1)\}$ and $I_2 = \{(w_1^2, f_1^2), (w_2^2, f_2^2), \ldots, (w_{n_2}^2, f_{n_2}^2)\}$, their similarity is computed with Eq. 6.

$$ItemSim(I_1, I_2) = \frac{1}{2}\left(\begin{array}{l} \frac{1}{n_1} \sum_{w_i^1 \in I_1} f_i^1 f_j^2 \max_{w_j^2 \in I_2} WordSim(w_i^1, w_j^2) + \\ \frac{1}{n_2} \sum_{w_k^2 \in I_2} f_k^2 f_l^1 \max_{w_l^1 \in I_1} WordSim(w_k^2, w_l^1) \end{array} \right) \qquad \text{(Eq. 6)}$$

where word similarity in Eq. 6 is calculated with the weight of relation defined in the ontology (see Table 3), the *JCn* similarity in *WordNet*, or the normalized pointwise mutual information [45, 48].

$$WordSim(w_1, w_2) = \begin{cases} OntoWeight(w_1, w_2) & \text{Case I} \\ JCn(w_1, w_2) & \text{Case II} \\ \max\left\{0, \log \frac{p(w_1, w_2)}{p(w_1)\,p(w_2)} / \log |\mathbf{I}| \right\} & \text{Case III} \end{cases} \qquad \text{(Eq. 7)}$$

Case I. If two words can be both mapped to the ontology;

Case II. Not fit for Case I, but two words can be both found in *WordNet*;

Case III. Not fit for Case I and II. $p(w_1)$ is the proportion of knowledge items containing w_1 among all the items (namely **I**), and $p(w_1, w_2)$ is the proportion of items that simultaneously contain w_1 and w_2.

3.3.2 Establish knowledge relevance network and divide problem-solving context-based topics

A commonly-used data analytics manner for diversity-oriented recommendation is item-clustering [3], which partitions several item communities from a content similarity-based item network. However, high similarity only reflects the belonging of the engineering domains, but it doesn't indicate a strong relevance in solution design.

In fact, the users of KMS will probably browse a series of items during the problem-solving process, and hence their sequential browsing history can be leveraged to model the relevance of items. Specifically, if two knowledge items, I_1 and I_2, are browsed by a user in a row under two similar or same problem-solving contexts (not limited to the recent history), a weighted relevance is accumulated. The total relevance is then computed by traversing the whole user set **U**.

$$REL\left(I_1,I_2\right)=\sum_{U\in\mathbf{U}}\sum_{\substack{\forall I_1,I_2\in BRH_S(U)\\ \left|S(I_1)-S(I_2)\right|=1}} ItemSim\left(I_1,I_2\right)\times CxtSim\left(C_{S(I_1)\text{th}},C_{S(I_2)\text{th}}\right)\times\left(\frac{\Delta T_{S(I_1)\text{th}}}{2\left|I_1\right|}+\frac{\Delta T_{S(I_2)\text{th}}}{2\left|I_2\right|}\right) \quad \text{(Eq. 8)}$$

A knowledge relevance network, KRN = <**V**, **E**>, can be accordingly established, where the vertices are all the browsed knowledge items, and weighted edges indicate their relevance.

To divide the knowledge relevance network into several topics, the Louvain algorithm is adopted in this paper [51]. It is a robust graph-based community-partitioning approach seeking the highest modularity partition, where the modularity (Q) is defined as follows. One can refer to [50-51] for the detailed processes for optimizing Q and evaluating robustness in partitioning.

$$\begin{aligned}
&Q=\frac{1}{2m}\sum_{I_i,I_j\in\mathbf{I}}\delta\left(I_i,I_j\right)\left(A_{ij}-\frac{k_ik_j}{2m}\right)\\
&A_{ij}=REL\left(I_i,I_j\right),k_i=\sum_{I_j\in\mathbf{I}}REL\left(I_i,I_j\right),m=\frac{1}{2}\sum_{I_i\in\mathbf{I}}\sum_{I_j\in\mathbf{I}}REL\left(I_i,I_j\right) \qquad \text{(Eq. 9)}\\
&\delta\left(I_i,I_j\right)=\begin{cases}1 & I_i,I_j\in TP_k\\ 0 & \text{else}\end{cases}
\end{aligned}$$

With partitioning results, knowledge items are divided into t problem-solving context-based topics, namely, **TP** = {TP_1, TP_2, …, TP_t}. Items clustered to the same topic have strong content, context, and browsing behavior relations simultaneously, which reflects a more comprehensive relevance in engineering design.

3.3.3. Evaluate user profession to a specific problem-solving topic

In order to avoid recommending an already-known knowledge item to a domain-expert, user profession to a specific problem-solving topic, as a specified *User Context*, is further evaluated with their behavior records of creating or revising related items. Specifically, the user profession is evaluated by a profession distribution on the keywords, namely, $Pf\left(U\right)=\left\{\left(w_1,pf_1\right),\left(w_2,pf_2\right),\ldots,\left(w_n,pf_n\right)\right\}$.

Similar to Eq. 3, a user's S_P-times creating history and S_Q-times revising history are firstly retrieved and represented as follows:

$$CRH\left(U\right)=\left\{\left(I_{1\text{st}}^{CR},C_{1\text{st}}^{CR},\Delta T_{1\text{st}}^{CR}\right),\left(I_{2\text{nd}}^{CR},C_{2\text{nd}}^{CR},\Delta T_{2\text{nd}}^{CR}\right),\ldots,\left(I_{S_P\text{th}}^{CR},C_{S_P\text{th}}^{CR},\Delta T_{S_P\text{th}}^{CR}\right)\right\} \quad \text{(Eq. 10)}$$

$$REH(U)=\left\{\left(I_{1\text{st}}^{RE},C_{1\text{st}}^{RE},\Delta T_{1\text{st}}^{RE}\right),\left(I_{2\text{nd}}^{RE},C_{2\text{nd}}^{RE},\Delta T_{2\text{nd}}^{RE}\right),\ldots,\left(I_{S_Q\text{th}}^{RE},C_{S_Q\text{th}}^{RE},\Delta T_{S_Q\text{th}}^{RE}\right)\right\} \quad \text{(Eq. 11)}$$

The quality of created and revised items is also concerned in measuring the user profession. It is evaluated by the relative durations ($\Delta T/|I|$) in other users' browsing records, as computed in Eq. 12.

$$q\left(I_j\right)=\begin{cases}0 & \mathbf{U}\left(I_j\right)=\varnothing \\ \dfrac{1}{\left|\mathbf{U}\left(I_j\right)\right|}\sum_{U\in\mathbf{U}\left(I_j\right)}\left(\max_{I_{S(I_j)\text{th}}=I_j}\left\{\dfrac{\Delta T_{S(I_j)\text{th}}}{\left|I_j\right|}\right\}-\dfrac{1}{S}\sum_{i=1}^{S}\dfrac{\Delta T_{i\text{th}}}{\left|I_{i\text{th}}\right|}\right) & \mathbf{U}\left(I_j\right)\neq\varnothing\end{cases} \quad \text{(Eq. 12)}$$

$\mathbf{U}(I_j)$ in Eq. 12 is the set of users who have browsed I_j. Note that one item can be repetitively browsed by one user, so only the longest relative duration is selected.

Based on the creating and revising history and quality of items, the corresponding weight for each word w_i in profession evaluation is computed as follows:

$$pf_i=\alpha\sum_{\substack{w_i\in I_C\\ I_C\in CRH(U)}}q\left(I_C\right)f_i^C+\sum_{\substack{w_i\in I_R\\ I_R\in REH(U)}}q\left(I_R\right)f_i^R \quad \text{(Eq. 13)}$$

α in Eq. 13 is a coefficient considering the originality in creating items, which is set to 3 in the calculation.

The representation of $Pf(U)$ is in the same form of knowledge items (i.e., Eq. 5), thus enabling the calculation using *ItemSim* (i.e. Eq. 6). Specifically, the user profession to a specific problem-solving topic is evaluated with Eq. 14, where $|TP_k|$ is the count of knowledge items clustered to topic TP_k.

$$Pro\left(U_i,TP_k\right)=\begin{cases}0 & Pf\left(U_i\right)=\varnothing \\ \dfrac{1}{\left|TP_t\right|}\sum_{I_j\in TP_k} ItemSim\left(Pf\left(U_i\right),I_j\right) & Pf\left(U_i\right)\neq\varnothing\end{cases} \quad \text{(Eq. 14)}$$

Based on Eq. 14, the similarity between two users' professions can be calculated with the cosine similarity:

$$\begin{aligned}&ProSim\left(U_1,U_2\right)\\&=\begin{cases}0 & Pf\left(U_1\right)=\varnothing \text{ or } Pf\left(U_2\right)=\varnothing \\ \dfrac{\sum_{TP\in\mathbf{TP}}Pro\left(U_1,TP\right)Pro\left(U_2,TP\right)}{\sqrt{\sum_{TP\in\mathbf{TP}}\left(Pro\left(U_1,TP\right)\right)^2}\sqrt{\sum_{TP\in\mathbf{TP}}\left(Pro\left(U_2,TP\right)\right)^2}} & \text{else}\end{cases}\end{aligned} \quad \text{(Eq. 15)}$$

3.4. Recommend knowledge for engineering solution design

3.4.1 Prefill preference matrix with the topic and profession information

Since the number of knowledge items is much larger than users in engineering solution design, a user-based collaborative filtering algorithm (UCF) is leveraged in this paper. Based on a user-item preference matrix shown in Eq. 16, UCF firstly finds K-nearest neighbors for the current user, and then predicts the preference of an unfilled knowledge item. However, since one user's preference may only focus on one or a few problem-solving topics, he/she will only browse limited numbers of knowledge items. This situation results in the high sparsity of $Pref(\mathbf{U},\mathbf{I})$, namely, plenty of slots cannot be computed with Eq. 4 and nor filled. Therefore, before conducting UCF, $Pref(\mathbf{U},\mathbf{I})$ should be pre-filled as much as possible.

$$Pref(\mathbf{U},\mathbf{I}) = \begin{bmatrix} Pref(U_1,I_1) & Pref(U_1,I_2) & \ldots & Pref(U_1,I_M) \\ Pref(U_2,I_1) & Pref(U_2,I_2) & \ldots & Pref(U_2,I_M) \\ \ldots & \ldots & \ldots & \ldots \\ Pref(U_N,I_1) & Pref(U_N,I_1) & \ldots & Pref(U_N,I_M) \end{bmatrix} \quad \text{(Eq. 16)}$$

With topic division and user profession information achieved in Sections 3.3.2 and 3.3.3, two prefilling cases are considered. Firstly, if a knowledge item I_j hasn't been browsed by a user, but the items in the same problem-solving topic TP_i have been browsed recently, then the user may need this item with the same degree, and hence the user's preference on I_j will be prefilled with the average preference on the browsed items in TP_i:

$$\begin{gathered} Pref_1^*(U_i,I_j) = \frac{1}{\left|TP_j \cap BRH_L(U_i)\right|} \sum\nolimits_{I \in TP_j \cap BRH_L(U_i)} Pref(U_i,I) \\ I_j \in TP_j, I_j \notin BRH_L(U_i), TP_j \cap BRH_L(U_i) \neq \varnothing \end{gathered} \quad \text{(Eq. 17)}$$

Secondly, for the rest items, if a user achieves the highest profession among all the users in a specific problem-solving topic TP_j, it's possible that he/she will no longer need the related knowledge items in this topic, and hence the preference on the items in TP_j will be prefilled with the minimum preference in the recent browsing history:

$$Pref_2^*(U_i,I_j) = \min\nolimits_{I \in BRH_L(U_i)} \left\{Pref(U_i,I)\right\}, \forall I_j \in TP_j, U_i = \arg\max\nolimits_U Pro(\mathbf{U},TP_j) \quad \text{(Eq. 18)}$$

Beyond these two prefilling cases, the remaining slots in Eq. 16 are still left unfilled, which will be predicted by the collaborative filtering algorithm.

3.4.2 User-based collaborative filtering with context-based diversification strategies

Users' background information usually impacts their actual knowledge needs and preferences on knowledge items [22]. In this paper, user professions on different problem-solving topics are concerned in finding K-nearest-neighbors in UCF. The similarity between two user's preference is hence calculated with a profession-weighted *Pearson correlation coefficient* (*PCC*):

$$UserSim(U_1,U_2) = \begin{cases} 0 & \mathbf{I}(U_1) \cap \mathbf{I}(U_2) = \varnothing \\ \left(1 + ProSim(U_1,U_2)\right) PCC(U_1,U_2) & \mathbf{I}(U_1) \cap \mathbf{I}(U_2) \neq \varnothing \end{cases}$$

$$PCC(U_1,U_2) = \frac{\sum\nolimits_{I \in \mathbf{I}(U_1) \cap \mathbf{I}(U_2)} \left(Pref(U_1,I) - \overline{Pref(U_1,\mathbf{I}(U_1))}\right)\left(Pref(U_2,I) - \overline{Pref(U_2,\mathbf{I}(U_2))}\right)}{\sqrt{\sum\nolimits_{I \in \mathbf{I}(U_1) \cap \mathbf{I}(U_2)} \left(Pref(U_1,I) - \overline{Pref(U_1,\mathbf{I}(U_1))}\right)^2} \sqrt{\sum\nolimits_{I \in \mathbf{I}(U_1) \cap \mathbf{I}(U_2)} \left(Pref(U_2,I) - \overline{Pref(U_2,\mathbf{I}(U_2))}\right)^2}} \quad \text{(Eq. 19)}$$

where $\mathbf{I}(U_1)$ represents the set of items that have filled/prefilled preference by the user U_1 in Eq. 16. $\overline{Pref(U_1,\mathbf{I}(U_1))}$ indicates U_1's average preference among all the items in $\mathbf{I}(U_1)$. $ProSim(U_1,U_2)$ is computed with Eq. 15, which amplifies the preference similarity using the profession similarity and hence achieves a higher precision in finding K-nearest neighbors. When the set of neighbors $\mathbf{U}_N$ are determined, the preference of the unfilled slots can be predicted:

$$Pref^*\left(U_i,I_j\right)=\overline{Pref\left(U_i,\mathbf{I}\left(U_i\right)\right)}+\frac{\sum_{U\in\mathbf{U}_N}UserSim\left(U_i,U\right)\left(Pref\left(U,I_j\right)-\overline{Pref\left(U_i,\mathbf{I}\left(U\right)\right)}\right)}{\sum_{U\in\mathbf{U}_N}\left|UserSim\left(U_i,U\right)\right|} \qquad \text{(Eq. 20)}$$

The last step in the knowledge recommendation is to select and rank the knowledge items, in order to achieve accuracy and diversity simultaneously. This paper copes with a post-filtering strategy. For a specific user, recently un-browsed knowledge items with the Top N+N_D preference are firstly selected as an optional set, which guarantees the accuracy. Then, a context-based diversification strategy is conducted to consider his/her facing problems. Specifically, for an item I_j in the optional set, all users' recent L-times browsing histories are traversed to fetch all the browsing records on this item (i.e., $BRH_L(\mathbf{U}, I_j)$), so as to determine its usability under the current context C_0:

$$Usa\left(I_j,C_0\right)=\begin{cases}0 & BRH_L\left(\mathbf{U},I_j\right)=\varnothing\\ \dfrac{1}{\left|BRH_L\left(\mathbf{U},I_j\right)\right|}\sum_{\left(I_j,C_i,\Delta T_i\right)\in BRH_L\left(\mathbf{U},I_j\right)}CxtSim\left(C_0,C_i\right)\dfrac{\Delta T_i}{\left|I_j\right|} & BRH_L\left(\mathbf{U},I_j\right)\neq\varnothing\end{cases} \qquad \text{(Eq. 21)}$$

Re-ranking the optional set with the usability, and then deleting the last N_D items in repetitive problem-solving topics, a Top-N recommendation list is finally generated. It will provide both accurate and diverse knowledge items, thus supporting engineering solution design in a smarter way.

4. Case study

In this section, we first describe the background of Smart 3D printer platform and the dataset collected in a real solution design task. Then, the performance of *CDKR* is evaluated and compared with several state-of-the-art recommendation approaches. At last, we further discuss the efficacy of *CDKR* in providing accurate and diverse knowledge items, as well as the limitations in data preparation and pre-processing.

4.1 Background and datasets

Nowadays, 3D printer platforms can be bundled with multiple customized services, for example, remote printing configuring/monitoring and maintenance scheduling. Therefore, it's able to co-create value for multiple stakeholders, including designers, manufacturers, customers, maintenance staff, and online service providers. However, due to the lack of essential engineering knowledge about 3D printers and printing processes, it often contributes to unexpected failures and high costs in new product/service innovation.

To mitigate this, a mobile app-based KMS was developed to manage the knowledge items accumulated in 3D printer platform solution design. As shown in Figure 4, this KMS was able to collect the sensing data from the 3D printer platform and perceive the corresponding problem-solving contexts. Meanwhile, user behavior records were also collected and stored in KMS. As an important module in KMS, the proposed *CDKR* approach was responsible for recommending the highly related knowledge items. It would continuously fulfill users' knowledge needs, until a suitable solution for reconfiguring the 3D printer platform was finally come up (e.g. upgrade parameter setting for an e-service module).

To compare the performance of *CDKR* with several state-of-the-art recommendation approaches, this section conducted an experiment in a solution design task of *printing troubleshooting* in *remote printing configuring/monitoring*. Aiming to innovate a timely service of *printing tracking*, total 11 stakeholders (including designers, maintenance staff, and customers) were involved. Before this solution design task, 2638 historical behavior records (including 2448 browsing records, 46 creating records, and 144 revising records) on 87 knowledge document items of *Guides to Printing Quality Assurance* were collected, which served as the training dataset. Some sample records are shown in Table 5. Besides, during the completion of this task, another 272 browsing records were self-directedly generated by these stakeholders (i.e. without receiving any recommendations from *CDKR*), and these records were used as the testing dataset. The proposed approach was coded with Python 3.7 and ran on a PC with 16GB RAM and Core i7 CPU.

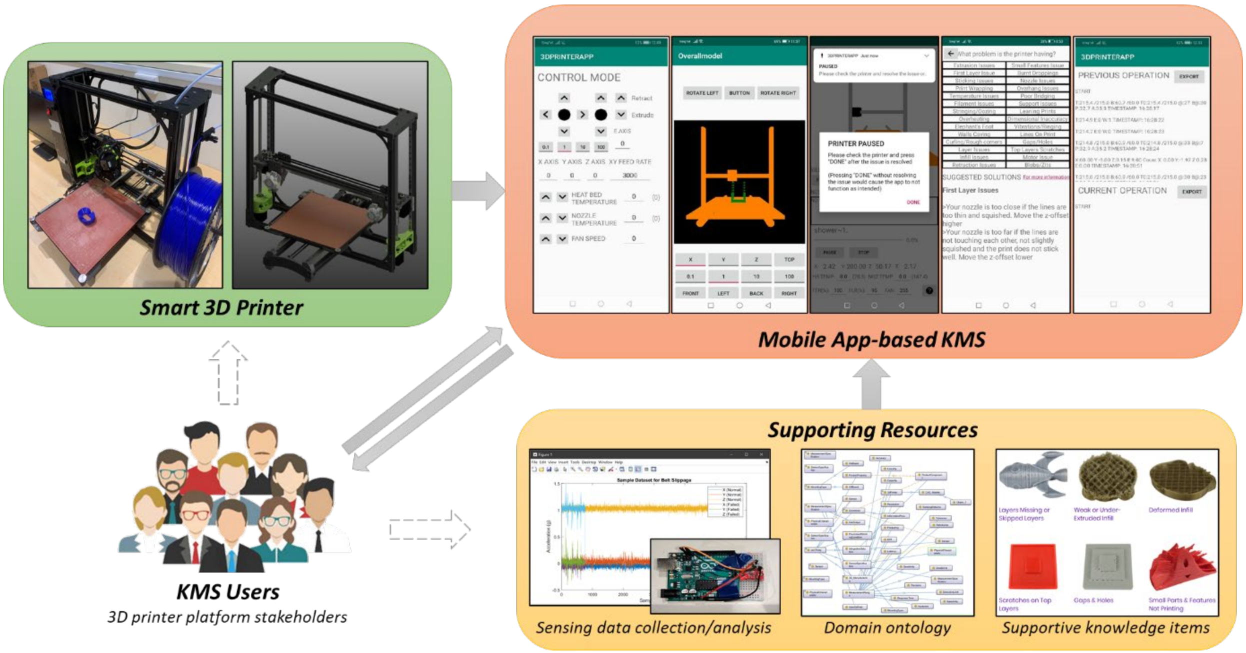

Figure 4. A mobile app-based KMS for the Smart 3D printer platform

4.2 Evaluation metrics

For each non-repetitive pair of problem-solving context and user (U,C) in the testing dataset, an N-item recommendation list $RL(U,C)$ was generated. To evaluate the performance of accuracy and diversity, as well as the time complexity, four widely-used evaluation metrics were selected.

- *F-Score*

F-score was adopted to evaluate the accuracy of the recommended results. For each context, compared the recommended list $RL(U,C)$ with the actually-browsed non-repetitive items in the testing dataset (i.e., $BRH_{test}(U,C)$) to compute *F-score* with the *precision* and *recall*.

$$F - Score = \begin{cases} 0 & RL(U,C) \cap BRH_{test}(U,C) = \varnothing \\ \dfrac{2 \times Pre \times Rec}{Pre + Rec} & RL(U,C) \cap BRH_{test}(U,C) \neq \varnothing \end{cases}$$
$$Pre = \frac{\left|RL(U,C) \cap BRH_{test}(U,C)\right|}{\left|RL(U,C)\right|}, Rec = \frac{\left|RL(U,C) \cap BRH_{test}(U,C)\right|}{\left|BRH_{test}(U,C)\right|} \quad \text{(Eq. 22)}$$

- *NDCG (Normalized discounted cumulative gain)*

As another metric to further evaluate the accuracy, *NDCG* score was computed with the rankings in the *N*-item recommendation list.

$$NDCG = \sum_{i=1}^{N} \frac{2^{rel_i} - 1}{\log_2(i+1)} \Big/ \sum_{i=1}^{\left|BRH_{test}(U,C)\right|} \frac{1}{\log_2(i+1)} \quad \text{(Eq. 23)}$$
$$rel_i \in \left\{1: \text{The } i\text{th item in } RL(U,C) \text{ occurred in } BRH_{test}(U,C); 0: \text{Not occurred}\right\}$$

- *Personalized diversity*

Personalized diversity (see Eq. 1) was adopted to evaluate the diversity of the recommended results. In this case, the similarity between two knowledge items was computed by $ItemSim(I_i, I_j)$ in Eq. 6.

$$Div\left(RL(U,C)\right) = \frac{\sum_{I_i \in RL(U,C)} \sum_{I_j \in RL(U,C), I_j \neq I_i} \left(1 - ItemSim\left(I_i, I_j\right)\right)}{\frac{1}{2}\left|RL(U,C)\right|\left(\left|RL(U,C)\right| - 1\right)} \quad \text{(Eq. 24)}$$

- *Online response time for the one-time recommendation*

Since different baseline algorithms relied on different training methods and various machine learning framework, the time consumed for offline model construction was not comparable. Therefore, this paper only measured the online response time *T* for generating one $RL(U,C)$ under one problem-solving context.

4.3 Implementation and fine-tuning of the proposed approach

As introduced in Section 3.2.1, a pre-defined ontology containing 267 concepts in the 3D printer domain was leveraged in this paper. The relationship among these concepts was quantified with Table 3. Then, to complete NLP-based processing on the contents of knowledge items (Section 3.3.1), *NLTK*, a prestigious NLP tool, was adopted to extract the keywords for each knowledge item. For calculation simplicity, 30 words with the highest TF-IDF values were remained for representing each item and computing the item similarity between a pair of items.

Dividing problem-solving topics for the knowledge items was based on the perceiving of context information. In this case, 9 context features were considered and valued, as shown in Figure 3. Based on all the browsing history in the training dataset, the adopted Louvain algorithm converged with a result of 14 partitioned communities. Table 6 shows some problem-solving topics. Based on these topics, the most professional user on each topic was determined using their creating and revising history.

Table 6. Extracted problem-solving topics and the user with the highest profession

Topic	Containing Knowledge Items	Representative Keywords	Possible troubleshooting issues	Most professional User
TP1	I1, I2, I8, I18, I34, I35, I46, I60, I76	Filament, Bed, Nozzle, First layer, Bed temperature, …	Print Sticking	U10
TP2	I3, I4, I7, I21, I25, I37, I47, I54, I64, I84, I87	Extrusion, Blockage, Filament, Feed rate, Material, …	Inconsistent Extrusion	U6
TP3	I5, I31, I41, I42, I56, I61, I86	Temperature, Fan, Fan speed, Hot end, Thermistor, …	Overheating	U6
…	…	…	…	…

An important parameter of *CDKR* was *L* in Eq. 4, which indicated the range of the user's recent preference. This section will fine-tune it to achieve the best recommendation performance. Specifically, for each length of *L*, Top 9 items were firstly recommended by 5 nearest neighbors, and 4 items were then filtered out by the diversification strategy (i.e., $K = 5$, $N = 5$, and $N_D = 4$, referring to [31]). The average scores of *F-Score*, *NDCG*, *Div*, and *T* on total 22 non-repetitive pairs of (*U*, *C*) were shown in Figure 5. When *L* increased to 30, *F-Score* and *NDCG* kept increasing, while *Div* kept decreasing. Three metrics became stable when *L* was larger than 30. Besides, the response time for the one-time recommendation became slightly larger when *L* increased. This paper preferred a rather small *L* and hence set it to 30, so as to reduce the calculations in model constructing.

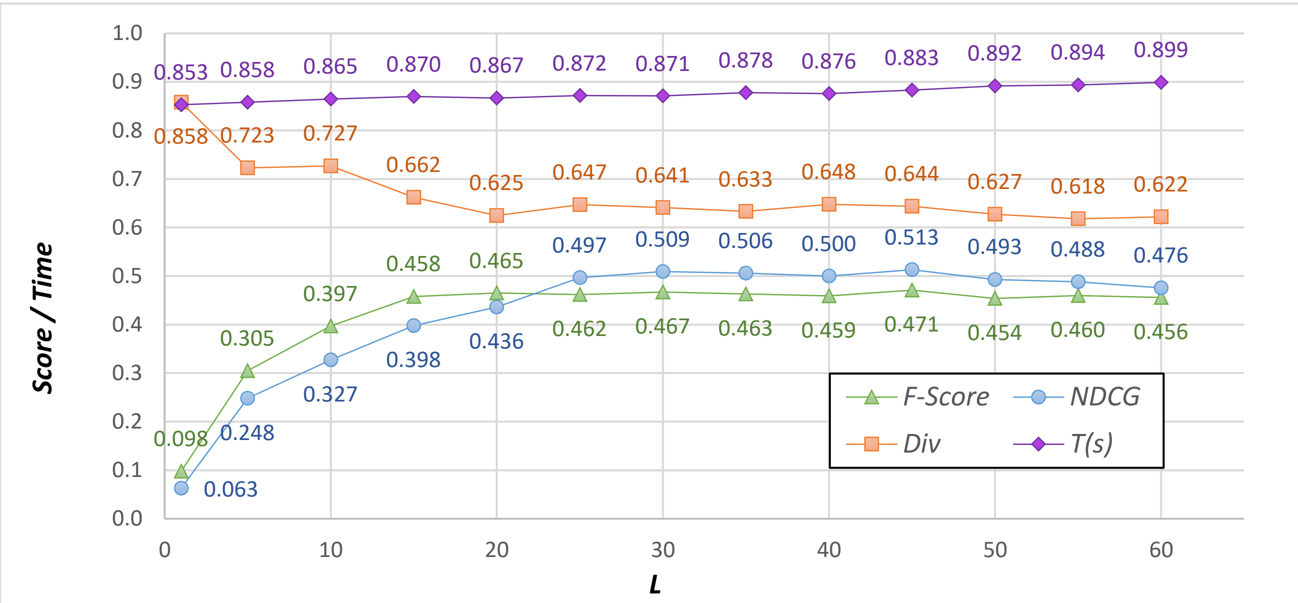

Figure 5. Performance of the proposed approach under different *L*

To showcase the validity of the recommendation results, a troubleshooting case was reported as an example. This case happened on user U6, who had the highest profession on the topics of *Inconsistent Extrusion* (TP2), *Overheating* (TP3), and *Supporting Structures* (TP7). His problem-solving context information was encoded as [3,3,1,1,1,2,1,2,3], which indicated that this expert customer encountered a stringing issue when he conducted a precise printing with the overheated nozzle and bed. Table 7 reported the recommended list for this case. His actual browsing list, with the ratio between browsing duration and word counts, was also reported.

Table 7. Recommended list and actual browsing list for an exemplary case

Recommended List generated by *CDKR*	**Actual browsing list** (Sorted by the browsing sequence)
I40: Clean the outside of the nozzle	I82: Burnt stuff dropping on the print (0.48s/word)
I45: Repair the leaking hot end	I45: Repair the leaking hot end (3.69s/word)
I76: Tips for printing with PETG filament	I40: Clean the outside of the nozzle (3.22s/word)
I62: Popping noise coming from the extruder	I25: Increase retraction distance and speed (0.79s/word)
I16: Printing slower, printing better	I26: Increase travel speed to fix oozing (0.62s/word)
I82: Burnt stuff dropping on the print (Deleted in diversification)	I45: Repair the leaking hot end (0.43s/word, re-occurred)
I60: Excessive adhesion of PETG filament (Deleted in diversification)	I24: Arrange prints closer (0.56s/word)
I9: Dimensional accuracy of models (Deleted in diversification)	I16: Printing slower, printing better (0.56s/word)
I32: Clicking motors (Deleted in diversification)	I76: Tips for printing with PETG filament (2.13s/word)
Performance in this case:	I40: Clean the outside of the nozzle (0.81s/word, re-occurred)
***F-score**:* 0.615*;* ***NDCG**:* 0.637*;* ***Div**:* 0.660*;* ***T**:* 0.834s	I16: Printing slower, printing better (0.20s/word, re-occurred)

Commonly recommended troubleshooting knowledge items for *Overheating* was replacing the thermistor and adjusting the temperature setting to conduct a cooler printing. However, as an expert customer, he meant to conduct precise printing with very slow printing speeds, where the heat dissipation conditions were good enough. The common items were hence not suitable for his context (actually, he didn't browse any item in the topic of *Overheating* as well). Instead, knowledge items about *components maintenance* (I40 and I45) and *filament selection* (I76) aroused U6's interest (longer browsing duration per wordcount), which might be more useful. Reflected in the proposed recommendation approach, his profession on TP3 (*Overheating*) and the values of the first four context features were fully concerned. Knowledge items in TP3 were largely filtered out in the final recommendation list, while items belonged to other topics were remained and re-ranked to the top places, thus achieving high accuracy and diversity. Meanwhile, through a follow-up interview, U6 also agreed that the recommended knowledge items would precisely and comprehensively solve his facing troubles.

4.4 Comparisons and discussions

4.4.1 Performance comparison of CDKR over baseline algorithms

Based on the user-based collaborative filtering approaches, the proposed approach introduces two sources of information into the recommendation process, namely, user problem-solving context and user profession. Besides, to achieve a higher diversity in the results, a context-based diversification strategy is also proposed. To validate the advantages, a comparison was conducted on the following algorithms:

- Basic user-based collaborative filtering approach (*UCF*);
- Bayesian personalized ranking (*BPR*), which treats users' and their nearest neighbors' feedback as relative preferences rather than absolutely like or not [25];

- A diversity-balanced collaborative filtering approach (*Div-CF*) [4];
- The proposed approach with a diversity-maximizing strategy (*CDKR-MaxDiv*, post-filter the results with the *Total Diversity Effect Ranking* model proposed in [31]);
- The proposed approach without context information (*CDKR w/o Cxt*, set *CxtSim* in Eq.2 to 1);
- The proposed approach without user profession information (*CDKR w/o Pro*, set pf_i in Eq. 13 to 0);
- The proposed approach (*CDKR*).

All the algorithms considered 5 nearest neighbors (i.e. $K = 5$) in the recommendation and finally retrieved Top-5 items as the recommendation list. In *BPR*, stochastic gradient descent (SGD) manner was leveraged to optimize the model with a preferred regularization parameter of 0.01 and a learning rate of 0.1 [25]. In *Div-CF*, the trust matrix between users was formed using *PCC* in Eq.19, and the balancing parameter was set to 0.4 as recommended in [4]. The performance improvement of *CDKR* over other baseline algorithms was measured using all 22 contexts, as reported in Figure 6 and Table 8. The pairwise T-test was also performed on the results of *CDKR* and each baseline algorithm. To illustrate the difference in the performance of algorithms more clearly, recommendation lists generated for the example in Table 7 were also compared in Table 9.

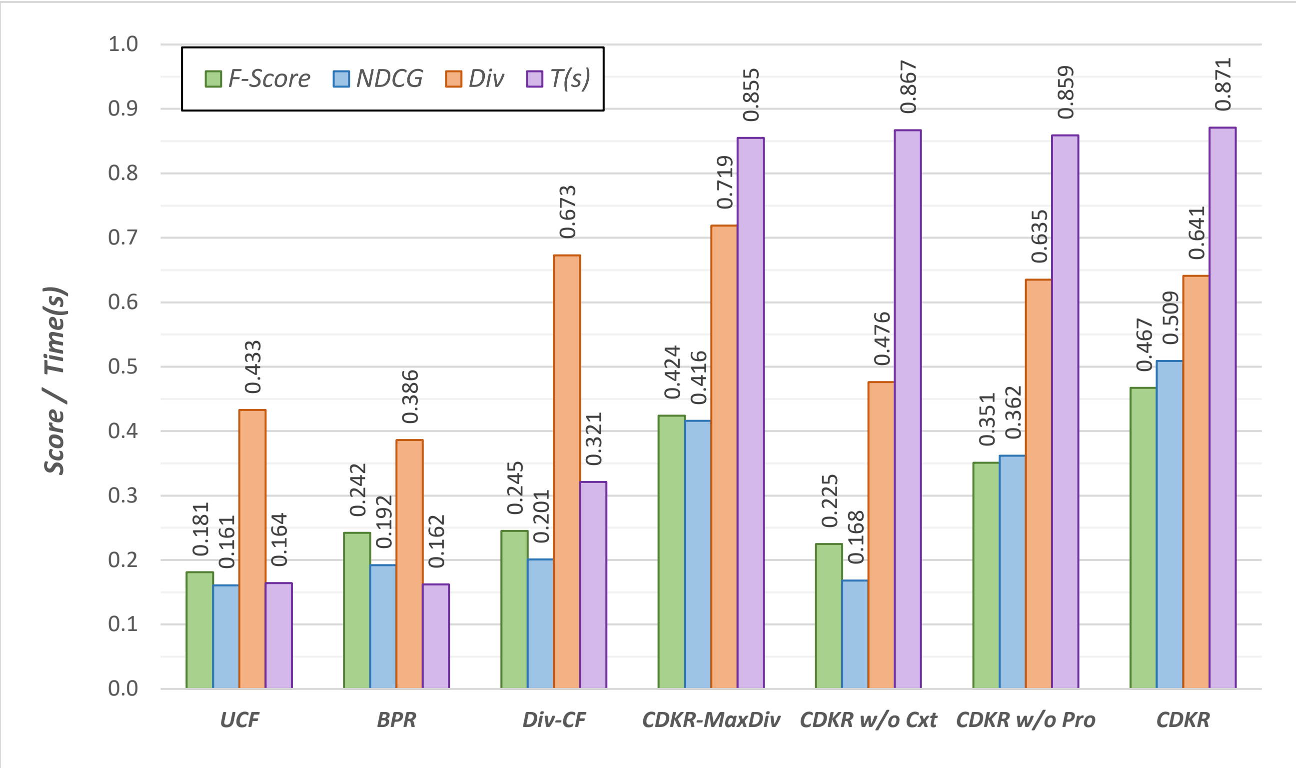

Figure 6. Performance comparison of *CDKR* over baseline algorithms

Table 8. Statistics of the performance comparison

	F-Score		*NDCG*		*Div*		*T* (s)	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
UCF	0.181 ***	0.073	0.161 ***	0.066	0.433 ***	0.140	0.164 ***	0.034
BPR	0.242 ***	0.092	0.192 ***	0.095	0.386 ***	0.159	**0.162** ***	0.042
Div-CF	0.245 ***	0.106	0.201 ***	0.099	0.673	0.151	0.321 ***	0.031
CDKR-MaxDiv	0.424	0.091	0.416 **	0.108	**0.719**	0.188	0.855	0.081
CDKR w/o Cxt	0.225 ***	0.083	0.168 ***	0.069	0.476 **	0.178	0.867	0.050
CDKR w/o Pro	0.351 ***	0.089	0.362 ***	0.102	0.635	0.165	0.859	0.047
CDKR	**0.467**	0.071	**0.509**	0.103	0.641	0.129	0.871	0.077

Note: Numbers in bold indicate the best performance; SD: standard deviation; * indicates the significance in T-test, where * p-value < 0.05, ** p-value < 0.01 and *** p-value < 0.001

Table 9. Comparison of the recommended lists for the case reported in Table 7

Approaches	***UCF***	***BPR***	***Div-CF***	***CDKR***
Top-5 Items	I36: Fail to print supports	**I16: Printing slower, printing better**	I67: Causes of droopy overhangs	**I40: Clean the outside of the nozzle**
	I39: Print thinner layers to prevent wrapping	I68: Print overhangs and bridges	**I16: Printing slower, printing better**	**I45: Repair the leaking hot end**
	I19: Layer shifting	I2: Settings for printing the first layer	I39: Print thinner layers to prevent wrapping	**I76: Tips for printing with PETG filament**
	I16: Printing slower, printing better	I13: Aviod molten & deformed printing	I20: Surface quality above supports	I62: Popping noise coming from the extruder
	I68: Print overhangs and bridges	I19: Layer shifting	**I24: Arrange prints closer**	**I16: Printing slower, printing better**
Performance in this case	***F-Score***: 0.154 ***NDCG***: 0.109 ***Div***: 0.403 ***T***: 0. 207s	***F-Score***: 0.154 ***NDCG***: 0.253 ***Div***: 0.399 ***T***: 0.175s	***F-Score***:0.308 ***NDCG***: 0.257 ***Div***: 0.676 ***T***: 0.291s	***F-Score***: 0.615 ***NDCG***: 0.637 ***Div***: 0.660 ***T***: 0.834s
Approaches	***CDKR w/o Cxt***	***CDKR w/o Pro***	***CDKR-MaxDiv***	
Top-5 Items	I19: Layer shifting	I7: Temperature and over extrusion	**I82: Burnt stuff dropping on the print**	
	I16: Printing slower, printing better	**I40: Clean the outside of the nozzle**	I9: Dimensional accuracy of models	
	I6: Scratches on the top layers	**I16: Printing slower, printing better**	I32: Clicking motors	
	I68: Print overhangs and bridges	I5: Part cooling fan setting	**I76: Tips for printing with PETG filament**	
	I2: Settings for printing the first layer	**I76: Tips for printing with PETG filament**	**I40: Clean the outside of the nozzle**	
Performance in this case	***F-Score***: 0.154 ***NDCG***: 0.160 ***Div***: 0.425 ***T***: 0.848s	***F-Score***: 0.462 ***NDCG***: 0.384 ***Div***: 0.631 ***T***: 0.888s	***F-Score***: 0.462 ***NDCG***: 0.460 ***Div***: 0.783 ***T***: 0.920s	

- Accuracy improvement brought by considering problem-solving context information

The problem-solving context information has long been regarded as the key element in both engineering solution design [1, 40, 50] and knowledge needs fulfillment [2]. The comparison in Figure 6 and statistics results in Table 8 also concords with this statement: *CDKR* significantly outperformed *UCF*, *BPR*, *Div-CF*, and *CDKR w/o Cxt* on accuracy metrics of *F-Score* and *NDCG*.

Without considering the context information, the recommendation approaches would only consider the user's past browsing records and tend to recommend some general knowledge items that have been browsed under a larger range of problem-solving contexts. As shown in Table 9, I16 (*Printing slower, printing better*), I19 (*Layer shifting*), and I68 (*Print overhangs and bridges*) were some frequently browsed knowledge items. Including these popular items without pertinence, *F-score* and *NDCG* scores of the recommendation list generated by non-context-aware algorithms were largely impaired. In contrast, considering the context similarity in item relevance evaluating (Eq. 8) and the diversification (Eq. 21), *CDKR* regarded some popular items as irrelevant and hence recommended more context-related ones instead.

- Further accuracy improvement brought by considering user profession information

Comparing the results of *CDKR w/o Pro* and *CDKR*, the user profession information is also proved to improve the accuracy one step further.

As analyzed before, a user with specific expertise may not need knowledge items in corresponding domains anymore. In the proposed approach, on one hand, users' profession was articulately modelled using his/her actual behavior records (i.e. creating and revising knowledge items); on the other hand, familiar items, as well as their similar items in the same problem-solving topic (like I5 and I7 to U6), were discovered based on the content analysis and filtered out in the prefilling process. Therefore, recommended items would concentrate more on topics that the user was unfamiliar, and better fill his/her possible knowledge gap in smart engineering solution design.

- Accuracy-Diversity balance achieved by the context-based diversification strategy

Comparing the results of *CDKR-MaxDiv* and *CDKR*, the context-based diversification strategy in *CDKR* also shown effectiveness in balancing accuracy and diversity. As reported in Table 8, even though the highest *Div* was achieved by *CDKR-MaxDiv*, the difference was not significant. However, the *NDCG* score of *CDKR-MaxDiv* was significantly lower, which impaired its validity in the practice.

In fact, for two lists separately generated by *CDKR-MaxDiv* and *CDKR*, they shared the same items before diversification. However, neglecting users' real contexts in the diversification, some high context-relevance knowledge items (like I45 in Table 8) were randomly deleted in the re-ranking of *CDKR-MaxDiv*, while some low context-relevance ones (I9 and I32) occupied the top places. Reversely, through the context-based diversification proposed in *CDKR*, items' usability under the facing problem-solving context was considered, and low-usable items in repetitive problem-solving topics were deleted. It guaranteed enough accuracy first, then pursued higher diversity to achieve better balance.

4.4.2 Time complexity of CDKR

Seen from Figure 6 and Table 8, *CDKR* consumed the longest time in the one-time recommendation, compared to baseline algorithms. Hence this section will further discuss the time complexity of *CDKR* and offer some mitigations to reduce the computing time.

In the worst-case (i.e. every user has browsed and created/revised all items, and a recommendation is made using all users and all records), the time complexity for item similarity computation is $O(|\mathbf{I}|^2)$, and $O(|\mathbf{U}|\times|\mathbf{R}|+|\mathbf{U}|^2)$ for profession-weighted user similarity. As for online computation, $O(|\mathbf{U}|\times|\mathbf{I}|)$ is required for preference pre-filling and CF, and $O(|\mathbf{R}|)$ for the context-based diversification. $|\mathbf{U}|$ is the number of users, $|\mathbf{I}|$ is the number of items, and $|\mathbf{R}|$ is the number of records (in this case, $O(|\mathbf{R}|) \approx O(|\mathbf{U}|\times|\mathbf{I}|)$).

In fact, due to the high sparsity in $Pref(\mathbf{U},\mathbf{I})$, the actual time complexity for online computation is much lower. The ideal complexity for prefilling and CF will only be $O(|\mathbf{U}|)$, as $\mathbf{I}(U_1) \cap \mathbf{I}(U_2)$ in Eq. 19 just contains a limited portion of items. Meanwhile, since only recent L times records of each user will be traversed in diversification, $O(|\mathbf{R}|)$ of the context-based diversification can be regarded as $L\times O(|\mathbf{U}|)$. Thereby, a smaller L (but still guarantees the performance) will be more appropriate to reduce the time of online computing.

Besides, with a periodical and incremental update manner, the time complexity for offline model construction is also not as high as expected. For the matrix of item similarity, it is rather stable and can be directly reused, since the contents of knowledge items don't change frequently. For the matrix of user similarity and topic division, they can be incrementally updated with an approximate complexity of $O(|\mathbf{U}|+|\mathbf{I}|)$, when few new browsing records are collected. Therefore, for a well-established *CDKR* model, the time complexity for the incremental update is acceptable.

4.4.3 Improvement to the engineering solution design process

To further discuss the improvement of *CDKR* in assisting stakeholders in engineering solution design, we also qualitatively compared the solution design process with and without *CDKR*, and the usual process assisted by common recommendation approaches.

After the experiment in 3D printer platform solution design, referring to concerns proposed in [2], a survey was conducted on 11 involved stakeholders. The Delphi method was leveraged to arrive at a consensus. As shown in Table 10, the panel of stakeholders agrees that the engineering solution design process will be hard to proceed without any assistance from the knowledge recommendation approach. As the most common recommendation approach nowadays, the content-based/keyword-based recommendation approach [13-15] will provide some highly context-relevant knowledge items. However, these items are usually not comprehensive and sometimes already known, thus performing poorly in covering users' knowledge needs in the solution design process. Another common approach, CF [16-19], can rapidly supply knowledge items according to similar users' behaviors, but it lacks a pertinence to the facing context. As a result, the generated design solution tends to be generic and sometimes it becomes infeasible due to special constraints in particular contexts. With rather high sensitivity to context details and high diversity in covering multi-facet knowledge needs, the proposed *CDKR* approach is regarded to overcome the above shortcomings, thereby assisting stakeholders to generate satisfying and even innovative design solutions.

Table 10. Qualitative comparison in the aspect of engineering solution design process

	Without recommendation (Manual searching)	With content-based/ keyword-based recommendation	With ordinary CF like *UCF*	With the proposed *CDKR*
Direct usefulness of retrieved items	**Poor**: Only a few are usable knowledge	**Medium**: Some task relevant knowledge	**Medium**: Some task relevant knowledge	**Good**: Direct solution of an emerging problem
Pertinence to the facing context	**Poor**: Only a few are related to the context	**Good**: Sensitive to details in the context	**Poor**: Regardless of the facing context	**Good**: Sensitive to details in the context
Coverage of knowledge needs	**Medium**: Some required knowledge is not easy to find	**Poor**: Only focus on one or few already known disciplines	**Medium**: Can be rather exhaustive, regarding similar users' behaviors	**Good**: Diversified topics that can fully cover the needs
Prerequisites before querying	**Poor**: Need to describe the problem clearly, which is hard	**Medium**: Need historical behaviors in solving similar problems	**Good**: No specific requirement for an individual user	**Good**: No specific requirement for an individual user
Time consumed in solving problem	**Poor**: Keep searching and keep trying for a long time	**Medium**: Need repetitive queries if retrieved items are not inclusive	**Good**: Quick to find some necessary items	**Good**: Quick to find some necessary items
Satisfaction of the generated design solution	**Poor**: Sometimes hard to generate a feasible design solution	**Medium**: A precise but not comprehensive design solution	**Medium**: Usually a generic design solution, sometimes infeasible	**Good**: Solutions can be quite innovative sometimes

4.4.4 Limitations in the proposed approach

Besides the improvements, there are still two limitations in *CDKR*. Firstly, in pre-processing work, *CDKR* depends on respective techniques of context modelling and knowledge representations/content-analysis [2, 26, 33, 35]. Especially for domain-specific ontology modelling (Section 3.2.1) and context encoding (Section 3.2.2) addressed in this paper, if there are no previous research outcomings in Smart 3D printer platform development served as the basis for the proposed approach, it will require a large quantity of expert knowledge and human intervention. This issue refrains the prospection of rapidly transplanting the proposed approach to other domain-specific cases. Regard this, a cost-effective procedure better leveraging massive heterogeneous data and information resources in smart engineering solution design should be integrated as the basis of the proposed approach (e.g. domain-level knowledge fusion [52] and text-based feature and relation extraction [53]), thus enabling rapid domain-specific ontology modelling and precise context feature value determination.

Secondly, although a profession-based prefilling process is leveraged to mitigate the sparsity of the preference matrix, inherited from the basic user-based collaborative filtering, the *cold start* issue of new user still exists in the proposed approach. For a new KMS user (i.e., a stakeholder in engineering solution design) who has few or none browsing/creating/revising records, his/her preference and profession cannot be modelled.

To handle this, two mitigation plans might be useful. On one hand, from a perspective of information management, KMS can be further integrated with the human resources management system (HRMS), project management system (PMS), and customer relationship management system (CRMS), so as to collect stakeholders' demographic information and historical participation information in engineering solution design. Similar to clustering knowledge items and prefilling preference with problem-solving topics, a new user can be clustered/classified to a particular user group, with his/her preference information and profession information prefilled by the known members in the same group. On the other hand, from a perspective of open innovation, an incentive mechanism (e.g. monetary incentives or KPIs) can be deployed to encourage KMS users to create new knowledge items, improve their quality, and share them with their partners in engineering solution design. As more behavior information collected, the models inside the proposed approach can be periodically evolved, so as to overcome the *cold start* issue.

5. Conclusion and future work

To support the knowledge-intensive process in engineering solution design, KMS is leveraged to proactively provide domain-specific engineering knowledge items to the stakeholders, where the knowledge recommendation approach serves as the basis for this core functionality. Aiming to meet the high standard of in-context accuracy and diversity simultaneously, this paper comprehensively considers item-diversity, context-diversity, and user-diversity in the scenario of engineering solution design, and then proposes a context-aware diversity-oriented knowledge recommendation approach. The main contributions are summarized into three aspects below:

1) *Identified three diversity requirements for the knowledge recommendation approach used in engineering solution design*. Based on the features in leveraging the accumulated engineering knowledge items and the analysis of the diversity and context-aware concerns in engineering solution design, three requirements, namely, item-diversity, context-diversity, and user-diversity, are identified as the key requirements in supporting engineering solution design.

2) *Introduced two effective information sources to support engineering knowledge recommendation*. Two information sources, problem-solving context information and user profession information, are perceived and evaluated from the massive product-sensed data and stakeholder-generated data collected from engineering solution design. Compared to only considering the browsing records (or rating records) in the conventional collaborative filtering, these two information sources are proved to largely improve the in-context accuracy and diversity in the recommendation.

3) *Proposed a user profession-based prefilling process and a context-based diversification strategy*. They fundamentally reduce the possibility of recommending an already known knowledge item to a stakeholder, and also balance the accuracy and diversity in the final results, thus improving the user experience in receiving the recommended list in smart engineering solution design.

Based on these contributions, future research directions lie in two aspects. On one hand, advanced context-aware manners and knowledge representation manners can be integrated into the proposed manner. It will guarantee the quality and quantity of raw datasets collected from engineering solution design, and further improve the cost-efficiency of the proposed approach in pre-processing these multi-source, heterogeneous data. On the other hand, it recommends integrating with management information systems in the enterprises, and also deploy an incentive mechanism to trigger stakeholders' participation in using/reusing knowledge resources. A further enhancement to the applicability and performance of the proposed approach is hence expected, fitting for more practical cases of smart engineering solution design.

Acknowledgement

This work is conducted within the Delta-NTU Corporate Lab for Cyber-Physical Systems with funding support from Delta Electronics Inc and the National Research Foundation (NRF) Singapore under the Corporate Laboratory @ University Scheme (Ref. RCA-16/434; SCO-RP1) at Nanyang Technological University, Singapore. The authors also acknowledge the funding support from the National Natural Science Foundation of China (No. 71671113).

References

[1] P. Zheng, Z. Wang, C.-H. Chen, L. Pheng Khoo, A survey of smart product-service systems: Key aspects, challenges and future perspectives, ADV ENG INFORM, 42 (2019) 100973.

[2] B. Song, Z. Jiang, X. Li, Modeling knowledge need awareness using the problematic situations elicited from questions and answers, KNOWL-BASED SYST, 75 (2015) 173-183.

[3] M. Kunaver, T. Požrl, Diversity in recommender systems - A survey, KNOWL-BASED SYST, 123 (2017) 154-162.

[4] L. Zhang, Q. Wei, L. Zhang, B. Wang, W. Ho, Diversity Balancing for Two-Stage Collaborative Filtering in Recommender Systems, Applied Sciences, 10 (2020) 1257.

[5] X. Li, Z. Wang, C.-H. Chen, P. Zheng, A data-driven reversible framework for achieving Sustainable Smart product-service systems, J CLEAN PROD, 279 (2021) 123618.

[6] E. Malekzadeh Hamedani, M. Kaedi, Recommending the long tail items through personalized diversification, KNOWL-BASED SYST, 164 (2019) 348-357.

[7] P. Zheng, X. Xu, C.-H. Chen, A data-driven cyber-physical approach for personalised smart, connected product co-development in a cloud-based environment, J INTELL MANUF, 31 (2020) 3-18.

[8] J.A. Pereira, P. Matuszyk, S. Krieter, M. Spiliopoulou, G. Saake, Personalized recommender systems for product-line configuration processes, Computer Languages, Systems & Structures, 54 (2018) 451-471.

[9] J.K. Tarus, Z. Niu, G. Mustafa, Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning, ARTIF INTELL REV, 50 (2018) 21-48.

[10] T. Zhou, Z. Kuscsik, J.G. Liu, M. Medo, J.R. Wakeling, Y.C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proceedings of the National Academy of Sciences, 107 (2010) 4511-4515.
[11] M. Göksedef, Ş. Gündüz-Öğüdücü, Combination of Web page recommender systems, EXPERT SYST APPL, 37 (2010) 2911-2922.
[12] V. Stantchev, L. Prieto-González, G. Tamm, Cloud computing service for knowledge assessment and studies recommendation in crowdsourcing and collaborative learning environments based on social network analysis, COMPUT HUM BEHAV, 51 (2015) 762-770.
[13] J. Shu, X. Shen, H. Liu, B. Yi, Z. Zhang, A content-based recommendation algorithm for learning resources, MULTIMEDIA SYST, 24 (2018) 163-173.
[14] N.A. Albatayneh, K.I. Ghauth, F. Chua, Utilizing Learners' Negative Ratings in Semantic Content-based Recommender System for e-Learning Forum, Journal of Educational Technology & Society, 21 (2018) 112-125.
[15] F. Narducci, P. Basile, C. Musto, P. Lops, A. Caputo, M. de Gemmis, L. Iaquinta, G. Semeraro, Concept-based item representations for a cross-lingual content-based recommendation process, INFORM SCIENCES, 374 (2016) 15-31.
[16] X. Yu, F. Jiang, J. Du, D. Gong, A cross-domain collaborative filtering algorithm with expanding user and item features via the latent factor space of auxiliary domains, PATTERN RECOGN, 94 (2019) 96-109.
[17] W. Cai, J. Zheng, W. Pan, J. Lin, L. Li, L. Chen, X. Peng, Z. Ming, Neighborhood-enhanced transfer learning for one-class collaborative filtering, NEUROCOMPUTING, 341 (2019) 80-87.
[18] C. Lai, Y. Chang, Document recommendation based on the analysis of group trust and user weightings, J INF SCI, 45 (2018) 845-862.
[19] F. Shen, S. Liu, Y. Wang, A. Wen, L. Wang, H. Liu, Utilization of electronic medical records and biomedical literature to support the diagnosis of rare diseases using data fusion and collaborative filtering approaches, JMIR medical informatics, 6 (2018) e11301.
[20] N.S. Selvan, S. Vairavasundaram, L. Ravi, Fuzzy ontology-based personalized recommendation for internet of medical things with linked open data, J INTELL FUZZY SYST, 36 (2019) 4065-4075.
[21] Z. Huang, B. Cautis, R. Cheng, Y. Zheng, N. Mamoulis, J. Yan, Entity-Based Query Recommendation for Long-Tail Queries, ACM T KNOWL DISCOV D, 12 (2018) 1-24.
[22] M. Li, Y. Li, W. Lou, L. Chen, A hybrid recommendation system for Q&A documents, EXPERT SYST APPL, 144 (2020) 113088.
[23] H. Li, H. Li, S. Zhang, Z. Zhong, J. Cheng, Intelligent learning system based on personalized recommendation technology, Neural Computing and Applications, 31 (2019) 4455-4462.
[24] D. Chae, J. Kang, S. Kim, J. Lee, CFGAN: A Generic Collaborative Filtering Framework based on Generative Adversarial Networks, Proceedings of the 27th ACM international conference on information and knowledge management, ACM, Torino, Italy, 2018, pp. 137-146.
[25] H. Liu, Z. Wu, X. Zhang, CPLR: Collaborative pairwise learning to rank for personalized recommendation, KNOWL-BASED SYST, 148 (2018) 31-40.
[26] X. Yin, B. Sheng, F. Zhao, X. Wang, Z. Xiao, H. Wang, A Correlation-Experience-Demand Based Personalized Knowledge Recommendation Approach, IEEE ACCESS, 7 (2019) 61811-61830.

[27] K. Bradley, B. Smyth, Improving recommendation diversity, Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland, Citeseer, 2001, pp. 85-94.
[28] C.L. Clarke, M. Kolla, G.V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, I. MacKinnon, Novelty and diversity in information retrieval evaluation, Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, 2008, pp. 659-666.
[29] D.M. Fleder, K. Hosanagar, Recommender systems and their impact on sales diversity, Proceedings of the 8th ACM conference on Electronic commerce, 2007, pp. 192-199.
[30] S. Wan, Z. Niu, An e-learning recommendation approach based on the self-organization of learning resource, KNOWL-BASED SYST, 160 (2018) 71-87.
[31] W. Premchaiswadi, P. Poompuang, N. Jongswat, N. Premchaiswadi, Enhancing Diversity-Accuracy Technique on User-Based Top-N Recommendation Algorithms, 2013 IEEE 37th Annual Computer Software and Applications Conference Workshops, 2013, pp. 403-408.
[32] K. Zuva, T. Zuva, Diversity and serendipity in recommender systems, Proceedings of the International Conference on Big Data and Internet of Thing, 2017, pp. 120-124.
[33] A. Jain, P.K. Singh, J. Dhar, Multi-objective item evaluation for diverse as well as novel item recommendations, EXPERT SYST APPL, 139 (2020) 112857.
[34] S. Wang, M. Gong, H. Li, J. Yang, Multi-objective optimization for long tail recommendation, KNOWL-BASED SYST, 104 (2016) 145-155.
[35] B. Geng, L. Jiao, M. Gong, L. Li, Y. Wu, A two-step personalized location recommendation based on multi-objective immune algorithm, INFORM SCIENCES, 475 (2019) 161-181.
[36] Y. Zhang, S. Ren, Y. Liu, T. Sakao, D. Huisingh, A framework for Big Data driven product lifecycle management, J CLEAN PROD, 159 (2017) 229-240.
[37] P. Zheng, Y. Liu, F. Tao, Z. Wang, C.-H. Chen, Smart Product-Service Systems Solution Design via Hybrid Crowd Sensing Approach, IEEE ACCESS, 7 (2019) 128463-128473.
[38] L. Trevisan, L. Trevisan, D. Brissaud, A system-based conceptual framework for product-service integration in product-service system engineering, J ENG DESIGN, 28 (2017) 627-653.
[39] Z. Wang, C.-H. Chen, P. Zheng, X. Li, L.P. Khoo, A novel data-driven graph-based requirement elicitation framework in the smart product-service system context, ADV ENG INFORM, 42 (2019) 100983.
[40] Z. Wang, C.-H. Chen, P. Zheng, X. Li, L.P. Khoo, A graph-based context-aware requirement elicitation approach in smart product-service systems, INT J PROD RES, (2019) 1-17.
[41] X. Li, C.-H. Chen, P. Zheng, Z. Wang, Z. Jiang, Z. Jiang, A Knowledge Graph-aided Concept-Knowledge Approach for Evolutionary Smart Product-Service System Development, J MECH DESIGN, 10 (2020) 101403.
[42] S. Wan, D. Li, J. Gao, R. Roy, F. He, A collaborative machine tool maintenance planning system based on content management technologies, The International Journal of Advanced Manufacturing Technology, 94 (2018) 1639-1653.
[43] H.J. Long, L.Y. Wang, S.X. Zhao, Z.B. Jiang, An approach to rule extraction for product service system configuration that considers customer perception, INT J PROD RES, 54 (2016) 5337-5360.

[44] C. Lim, M. Kim, J. Heo, K. Kim, Design of informatics-based services in manufacturing industries: case studies using large vehicle-related databases, J INTELL MANUF, 29 (2018) 497-508.
[45] X. Li, Z. Jiang, B. Song, L. Liu, Long-term knowledge evolution modeling for empirical engineering knowledge, ADV ENG INFORM, 34 (2017) 17-35.
[46] J.S. Liang, A process-based automotive troubleshooting service and knowledge management system in collaborative environment, ROBOT CIM-INT MANUF, 61 (2020) 101836.
[47] E. Maleki, F. Belkadi, N. Boli, B.J. van der Zwaag, K. Alexopoulos, S. Koukas, M. Marin-Perianu, A. Bernard, D. Mourtzis, Ontology-Based Framework Enabling Smart Product-Service Systems: Application of Sensing Systems for Machine Health Monitoring, IEEE INTERNET THINGS, 5 (2018) 4496-4505.
[48] X. Li, Z. Jiang, L. Liu, B. Song, A novel approach for analysing evolutional motivation of empirical engineering knowledge, INT J PROD RES, 56 (2018) 2897-2923.
[49] L. Zhen, L. Wang, J. Li, A design of knowledge management tool for supporting product development, INFORM PROCESS MANAG, 49 (2013) 884-894.
[50] P. Zheng, C.-H. Chen, S. Shang, Towards an automatic engineering change management in smart product-service systems - A DSM-based learning approach, ADV ENG INFORM, 39 (2019) 203-213.
[51] V.D. Blondel, J. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, Journal of statistical mechanics: theory and experiment, 2008 (2008) 1-12.
[52] A. Smirnov, T. Levashova, Knowledge fusion patterns: A survey, INFORM FUSION, 52 (2019) 31-40.
[53] J. Leng, P. Jiang, A deep learning approach for relationship extraction from interaction context in social manufacturing paradigm, KNOWL-BASED SYST, 100 (2016) 188-199.