Computer Communications 157 (2020) 116–123

Contents lists available at ScienceDirect

Computer Communications

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

Multi-dimensional quality-driven service recommendation with privacy-preservation in mobile edge environment

Weiyi Zhong [a], Xiaochun Yin [b], Xuyun Zhang [c], Shancang Li [d], Wanchun Dou [e], Ruili Wang [f], Lianyong Qi [a,*]

[a] School of Information Science and Engineering, Qufu Normal University, China
[b] Facility Horticulture Laboratory of Universities in Shandong, WeiFang University of Science & Technology, China
[c] Department of Computing, Macquarie University, Australia
[d] Computer Science and Creative Technologies Department, University of the West of England, UK
[e] State Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, China
[f] Institute of Natural and Mathematical Sciences, Massey University, New Zealand

ARTICLE INFO

Keywords:
Service recommendation
Multi-dimensional QoS
Privacy-preservation
Edge service
Locality-Sensitive Hashing
TOPSIS

ABSTRACT

With the advance of mobile edge computing (MEC), the number of edge services running on mobile devices grows explosively. In this situation, it is becoming a necessity to recommend the most suitable edge services to a mobile user from massive candidates, based on the historical quality of service (QoS) data. However, historical QoS is a kind of private data for users, which needs to be protected from privacy disclosure. Currently, researchers often use the Locality-Sensitive Hashing (LSH) technique to achieve the goal of privacy-aware recommendations. However, existing LSH-based methods are only applied to the recommendation scenarios with a single QoS dimension (e.g., *response time* or *throughput*), without considering the multi-dimensional QoS (e.g., *response time* and *throughput*) ensemble, which narrow the application scope of LSH in privacy-preserving recommendations significantly. Considering this drawback, this paper proposes a multi-dimensional quality ensemble-driven recommendation approach named $Rec_{LSH\text{-}TOPSIS}$ based on LSH and TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) techniques. First, the traditional single-dimensional LSH recommendation approach is extended to be a multi-dimensional one, through which we can obtain a set of candidate services that a user may prefer. Second, we use TOPSIS technique to rank the derived multiple candidate services and return the user an optimal one. At last, a case study is presented to illustrate the feasibility of our proposal to make privacy-preserving edge service recommendations with multiple QoS dimensions.

1. Introduction

With the advent of the "Internet of Things (IoT)" and "Artificial Intelligence (AI)" Era [1–5], many smart mobile devices (e.g., Smartphone, Smart bracelets, Tablets) have been ubiquitously popularized. As a result, people have witnessed a rapidly increase of mobile data and edge services [6–8]. In this situation, it is increasingly challenging to select and recommend an appropriate service from a great number of available candidates for users. Recently, a widely used classic service recommendation technology, i.e., collaborative filtering (CF) [9] has been introduced, which can effectively help users find suitable services and filter out useless services, typically based on historical QoS (Quality of Services) data [7,10].

However, traditional CF-based service recommendation approaches often take the centralized historical QoS data as the major recommendation basis [11]. As a consequence, they often fall short in handling the distributed mobile recommendation scenarios where the QoS data are fragmented across different edge servers [12,13]. To perform the service recommendations in mobile edge environment, it is required to integrate and process the distributed QoS data generated from mobile edge terminals and stored in various edge servers, so as to help a recommender system to make a comprehensive recommendation decision.

However, there are still some problems unsolved in the mobile edge environment, such as the leakage of user privacy [14], the decline of recommendation efficiency [15], and so on. Concretely, when integrating users' historical QoS data from multiple edge servers, it is inevitable

* Corresponding author.
E-mail addresses: weiyi_zhong@outlook.com (W. Zhong), xiaochunyin@wfust.edu.cn (X. Yin), xuyun.zhang@mq.edu.au (X. Zhang), Shancang.Li@uwe.ac.uk (S. Li), douwc@nju.edu.cn (W. Dou), ruili.wang@massey.ac.nz (R. Wang), lianyongqi@gmail.com (L. Qi).

https://doi.org/10.1016/j.comcom.2020.04.018
Received 20 January 2020; Received in revised form 2 April 2020; Accepted 10 April 2020
Available online 15 April 2020
0140-3664/

that all edge servers need to share their data; however, an edge server is often reluctant to share its data with others because of the risk of users' privacy leakage, which often renders the distributed recommendations infeasible. Moreover, with the continuous growth of QoS data in each edge server, the communication cost among edge servers will increase, which indirectly decreases the recommendation efficiency and further sacrifices the user experience.

To tackle the problems mentioned above, researchers commonly use Locality-Sensitive Hashing (LSH) strategy for recommendations to secure user privacy [11]. While current LSH recommendation solutions are usually based on single-dimensional QoS data (e.g., *response time* or *throughput*). As a non-functional attribute of service, quality of service (QoS) is a crucial criterion of service recommendation, which contains multiple dimensions (e.g., response time and throughput). If we neglect the multi-dimensional attributes of QoS itself, the accuracy and authenticity of the recommendation results would be reduced considerably. Furthermore, multi-dimensional scenarios are more common in reality [16–25]. Therefore, it is of practical significance to consider the multiple QoS dimensions in recommendation system, even if the multi-dimensional case is more complex. In this paper, we extended the traditional single-dimensional LSH recommendation approach to be a multi-dimensional one. However, multiple recommended services are often available according to the LSH-based recommendation solutions, among which we need to find out the optimal one based on the candidates' multiple QoS performances. Therefore, the objective evaluation of candidate services with multiple QoS dimensions is a challenging task as the weights for different QoS dimensions are often fuzzy and difficult to determine [26].

In view of the above challenges, this paper proposes a method (named $Rec_{LSH-TOPSIS}$) to realize multi-dimensional quality-driven service recommendation with privacy preservation based on LSH (for protecting user privacy) and the multi-attribute decision making (MADM) technique TOPSIS (for evaluating candidate services with multiple QoS dimensions which are not assigned concrete weights) [27]. Overall, three contributions of our work are summarized as follows.

(1) We extend traditional LSH recommendation solutions with a single QoS dimension to the multi-dimensional cases, which not only achieve a good compromise between user privacy and recommendation accuracy, but also enlarge the applicability of LSH in privacy-aware recommender systems.

(2) We use TOPSIS technique to evaluate and rank all the candidate services returned by LSH-based recommendation solution, so as to achieve an objective and rational evaluation result even the multiple QoS dimensions of candidates are not assigned concrete weights.

(3) A case study is provided to illustrate the execution process of our proposed $Rec_{LSH-TOPSIS}$ approach, through which we prove the feasibility of our proposal.

The rest of this paper is organized as follows. Related work is summarized in Section 2. In Section 3, we formulate the multi-dimensional quality driven service recommendation problem with privacy and illustrate the motivation of this paper. In Section 4, we describe $Rec_{LSH-TOPSIS}$ in detail. Section 5 demonstrates the feasibility of $Rec_{LSH-TOPSIS}$ by a case study. Finally, in Section 6, we conclude the paper and point out the future research work.

2. Related work

As mentioned in Section 1, this study focuses on realizing the multi-dimensional quality-driven recommendation scenario considering user privacy disclosure risks. Therefore, in this section, we summarize the up-to-date research from the below two perspectives.

2.1. Multi-dimensional service recommendation

Plenty of researchers have studied the influence of multiple quality dimensions when performing recommendations. Skyline technique is regarded as a feasible manner to select optimal services from massive candidates with multiple QoS dimensions. For example, Alrifai et al. [18] first recruited skyline to make service selections, reducing the search space of candidate services. Zhang et al. [28] proposed a skyline-based approach to find diversified services that are representative in different quality dimensions for service recommendation. Besides, Wang et al. [29] proposed to recommend services by integrating both quantitative and qualitative preferences of users based on multi-attributes of service. However, all the above multi-dimensional quality-aware approaches seldom consider the risk of privacy leakage. Although Gong et al. [26] presented an approach to protect user privacy with multi-dimensional QoS data, their solution converts the multiple-dimensional recommendation into multiple single-dimensional recommendation, without considering the multiple QoS data in an integrated and comprehensive manner. As a result, the reliability and accuracy of the recommender system still cannot be guaranteed very well in their study.

2.2. Privacy-preserving service recommendation

As a continuously hot topic of service recommendation, how to improve the accuracy of recommendations has been extensively studied by a number of researchers [30–32] at the early stage. However, with the development of service recommendation in the distributed environment, recent work shifts their concerns to user privacy protection issues. For example, Fu et al. [33] proposed a multi-keyword search scheme for personalized user preferences based on encryption technique; Xia et al. [34] proposed a scheme used for privacy-preserving information search over cloud data based on keyword vector encryption, both of which used encryption technology. Actually, in the realm of information retrieval, encryption technology is often used to protect user's privacy. However, it is not suitable to use encryption in the case of no-heavyweight service recommendations, because it often results in substantial computational costs. Casino et al. [35] proposed to protect sensitive information by employing k-anonymity method. However, if the anonymous data are used to make service recommendation, it is possible that the accuracy of the recommendation cannot be guaranteed. Dou et al. [36] suggested making only a small fraction of QoS data public. However, no matter how small fraction of QoS data is disclosed, users' private data cannot be fully protected. Zhu et al. [37] presented an approach to make service recommendations based on data obfuscation technique. However, this approach often leads to the decrease in the accuracy of recommendations because the real QoS data is blurred. Qi et al. [11,38] applied the Locality-Sensitive Hashing (LSH) technique in service recommendations to protect user privacy, even in a distributed environment. Nevertheless, existing LSH-based recommendation solutions are often based on single-dimensional QoS data, neglecting the general recommendation scenarios where multiple QoS dimensions are present. Even though privacy-preservation is well-realized in existing LSH approach, the applicability of service recommendation cannot be guaranteed.

We note here that all of the existing service recommendations researches seldom take into account the capability of privacy-preservation in multi-dimensional quality-driven recommendation scenarios. Therefore, a novel multi-dimensional quality-driven service recommendation approach with privacy preservation, named $Rec_{LSH-TOPSIS}$, is proposed in this paper to make up deficiencies of existing approaches.

3. Formulation and motivation

In this section, we formulate the problems and then better illustrate our motivation in this paper through a vivid example. The problem formulation is as follows:

Table 1
Summary of notations.

Notation	Description	Appears in
$QoS(\ )$	The original QoS data matrix	(1)
qos_{IJ}	The QoS value of Ith dimension of Jth service	(2)
qos'_{IJ}	The normalized QoS value of Ith dimension of Jth service	(2) (3)
$Qos'()$	The normalized QoS data matrix	(3) (4)
V	A matrix including k n-dimensional vectors	(4)
h_{IJ}	The hash value of Ith dimension of Jth service	(5) (6)
$H(\ )$	A hash value matrix	(6)
SU_Set	The similar users set of the key user	(8)
$P(\ )$	The prediction candidate service matrix	(9)
p_{ij}	The predicted QoS value of candidate service in ith dimensions of jth service	(9) (11)
$CS()$	The normalized prediction candidate service matrix	(10)
cs_{ij}	The normalized predicted QoS value of candidate service in ith dimensions of jth service	(10) (11) (14) (15)
J_1 (J_2)	The benefit-type (cost-type)	(11)
CS^+	The positive-ideal candidate service solutions	(12) (14)
CS^-	The negative-ideal candidate service solutions	(13) (15)
D_j^+	The distance of the jth candidate service from the positive-ideal candidate service solutions	(14) (16)
D_j^-	The distance of the jth candidate service from the negative-ideal candidate service solutions	(14) (16)
R_j	The relative closeness value about jth candidate service	(16)

3.1. Problem formalization

For facilitation, we formalize the multi-dimensional quality-driven service recommendation in edge environment as a five-tuple (U^*, Dim, Ser, ES, u^*_{key}) which can be specified as follows:

(1) $U^* = \{u^*_1, \ldots, u^*_m\}$: the set of users who have executed the services hosted on Edge servers.

(2) $Dim = \{d_1, \ldots, d_w\}$: the set of quality dimensions. To ease the discussion, we assume that each service includes w dimensions.

(3) $Ser = \{ser_1, \ldots, ser_n\}$: the set of services. Here, $ser_J = \{qos_{1J}, \ldots, qos_{wJ}\}$ $(1< J \leq v)$ where qos_{IJ} $(1< I \leq w)$ denotes the values of Ith dimension of service ser_J.

(4) $ES = \{es_1, \ldots, es_z\}$: the set of Edge Servers that record the multi-dimensional QoS data of service in set Ser.

(5) u^*_{key}: a key user requiring recommended services.

For the sake of simplifying the following discussion, the other notations recruited in our proposed approach are specified in Table 1.

3.2. Motivation

We illustrate the motivation of our paper with the example in Fig. 1. Suppose that there are n edge servers ($es_1, \ldots, es_z$) for a cloud platform, and two users *Jack* (key user) and *John*. Users can invoke services ($ser_1, \ldots, ser_n$) each with w QoS dimensions ($d_1, \ldots, d_w$). The historical QoS data qos_{IJ} produced by mobile terminals are monitored and recorded by $es_1, \ldots, es_z$, respectively. In this situation, to recommend appropriate services to the key user *Jack*, the similarity between *Jack* and *John* (denoted by *sim* (*Jack*, *John*)) should be calculated first. However, in the similarity calculation process, we will face the following challenges:

(1) When integrating the historical QoS data of *Jack* and *John* from multiple edge servers, it is necessary for all edge servers to share their data with others. However, such an integration process may disclose user privacy, which decreases the edge servers' data sharing willingness and renders the distributed recommendations infeasible.

(2) When considering the multi-dimensional historical QoS data of *Jack* and *John*, the weight of each dimension indicator is often uncharted. Therefore, it is challenging to balance the different QoS dimensions to pursue an accurate *sim* (*Jack*, *John*) value.

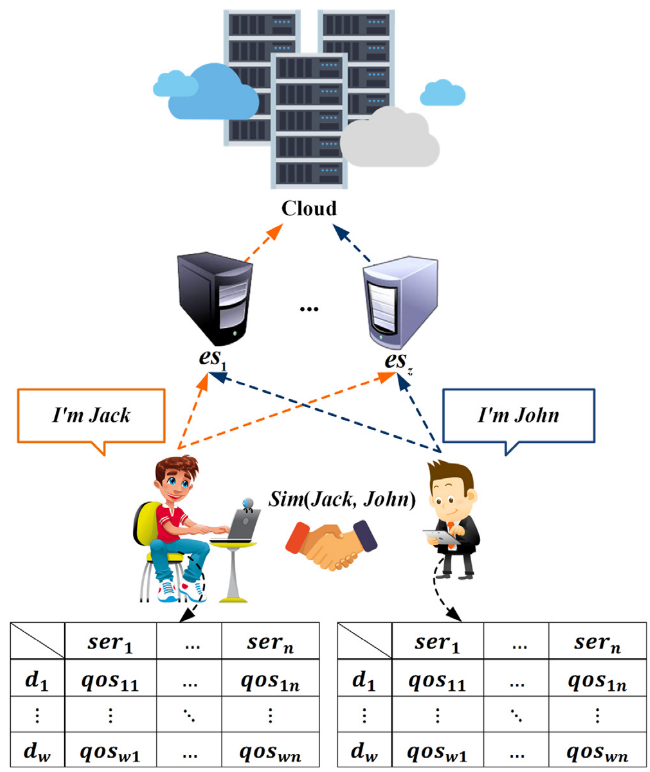

Fig. 1. Distributed service recommendation in Edge environment: an example.

(3) The similarity calculation process may consume more and more time with the increase of QoS data distributed across different edge servers, which will reduce the user service experience significantly.

In light of the aforementioned challenges, a multi-dimensional quality-driven and privacy-preserving service recommendation approach named $Rec_{LSH-TOPSIS}$ is proposed, which will be presented in detail in the following sections.

4. Privacy-aware and multi-dimensional service recommendation

4.1. Framework

In this section, the framework of our proposed two-phase recommendation approach $Rec_{LSH-TOPSIS}$ is presented in Fig. 2.

Phase 1: We extend existing single-dimensional LSH recommendation approach to a multi-dimensional one consisting of three steps. First, the normalized QoS data matrix $QoS'(u^*)$ is transformed into user index value matrix $H(u^*)$ offline in Step 1. Second, similar users of u^*_{key} is determined online based on user indices and a similar user set is generated in Step 2. Third, we select TOP-3 candidates from the similar users of u^*_{key} for each QoS dimension and then derive a QoS matrix $P(u^*_{key})$ in Step 3.

Phase 2: We utilize the multiple-criteria-decision-making method TOPSIS to enable objective evaluation of the candidates with 4 steps. In Step 4.1, we normalize matrix $P(u^*_{key})$ to be $CS(u^*_{key})$. In Step 4.2, according to $CS(u^*_{key})$, we obtain a positive-ideal solution CS^+ and a negative-ideal solution CS^-. In Step 4.3, for each candidate, we calculate its distances to CS^+ and CS^-, respectively. In Step 4.4, we rank all the candidate services based on their distances and return the optimal services to the key user.

4.2. Service recommendation approach based on LSH and TOPSIS $Rec_{LSH-TOPSIS}$

Step 1: Data pre-processing and multi-dimensional user indices building offline

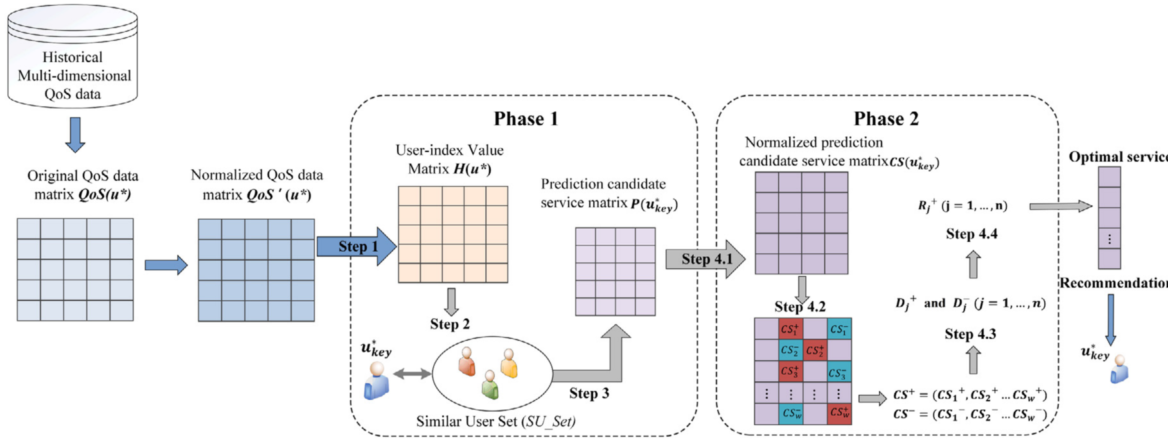

Fig. 2. The whole framework of the proposed approach.

In this step, our main task is to build user indices offline. We divide the indices generation process into two parts. First, we normalize the users' original historical QoS data matrix that contains multi-dimensional QoS information observed by users. Here, for a user u^*, we can model her/his multi-dimensional QoS data with a matrix $QoS(u^*)$ of size $w * n$ (see Eq. (1)), in which each row represents a service and each column represents a QoS dimension.

$$QoS(u^*) = \begin{bmatrix} qos_{11} & \cdots & qos_{1n} \\ \vdots & \ddots & \vdots \\ qos_{w1} & \cdots & qos_{wn} \end{bmatrix} \tag{1}$$

where qos_{IJ} $(1 < I \le w,\ 1 < J \le n)$ represents QoS value of Ith dimension of Jth service. In order to facilitate subsequent calculations, the original QoS data matrix $QoS(u^*)$ is normalized by formula (2):

$$qos'_{IJ} = \frac{qos_{IJ}}{\sqrt{\sum_{I=1}^{w} qos_{IJ}^2}} \tag{2}$$

After normalization, the original qos_{IJ} value is transformed into qos'_{IJ} which belongs to the range [0, 1]. Then, we get a normalized matrix as $QoS(u^*)$ presented in (3), where each row represents the QoS values of n services by u^* for the same dimension, and each column represents the QoS values of a service by u^* for the w dimensions (i.e., qos'_{22} is the QoS value of user in dimension d_2 of service ser_2). If u^* has never invoked ser_J previously, it can be denoted by $qos'_{IJ} = 0$.

$$QoS'(u^*) = \begin{bmatrix} qos'_{11} & \cdots & qos'_{1n} \\ \vdots & \ddots & \vdots \\ qos'_{w1} & \cdots & qos'_{wn} \end{bmatrix} \tag{3}$$

Next, we choose appropriate LSH function family to calculate users' hash values which can be regarded as the users' indices. Here, we adopt the LSH function family corresponding to the Pearson Correlation Coefficient (PCC) [39] distance for privacy-preserving similar users' finding, as PCC is frequently used as the user similarity measurement in service recommender systems and LSH is a technique with the property of "similarity retention" in privacy-aware information retrieval. Concretely, user u^*'s QoS matrix is projected by k LSH functions, which is represented by (4). Here, V is a $n*k$ matrix consisting of k n-dimensional vectors, where v_{IJ} $(1< j \le v,\ 1< i \le w)$ is a random value in range [−1,1], and symbol "∘" denotes dot product between two matrices.

$$S(u^*) = QoS'(u^*) \circ V = \begin{bmatrix} qos'_{11} & \cdots & qos'_{1n} \\ \vdots & \ddots & \vdots \\ qos'_{w1} & \cdots & qos'_{wn} \end{bmatrix} \circ \begin{bmatrix} v_{11} & \cdots & v_{1k} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nk} \end{bmatrix}$$
$$= \begin{bmatrix} s_{11} & \cdots & s_{1k} \\ \vdots & \ddots & \vdots \\ s_{w1} & \cdots & s_{wk} \end{bmatrix} \tag{4}$$

$$h_{IJ}(u^*) = \begin{cases} 1 & If\ s_{IJ} > 0 \\ 0 & If\ s_{IJ} \le 0 \end{cases} \tag{5}$$

Next, through the conversion function in (5), we transform matrix $S(u^*)$ to a Boolean hash value matrix in (6), i.e., $H(u^*)$, where h_{IJ} $(1 < I \le w,\ 1 < J \le k)$ is a binary value of 0 or 1. Here, $H(u^*)$ can be regarded as multi-dimensional user index for u^*. Comparatively, less sensitive information is involved in $H(u^*)$ than that in original historical QoS data matrix $QoS(u^*)$. In this way, u^*'s multi-dimensional sensitive QoS data is successfully protected. All the users in set U^* and their respective indices constitute a LSH table, i.e., $\{\{{u^*}_1 \to H({u^*}_1), \ldots, {u^*}_m \to H({u^*}_m)\}\}$.

$$H(u^*) = \begin{bmatrix} h_{11} & \cdots & h_{1k} \\ \vdots & \ddots & \vdots \\ h_{w1} & \cdots & h_{wk} \end{bmatrix} \tag{6}$$

Step 2: Find similar users of ${u^*}_{key}$ online based on user indices

As LSH is a probability-based retrieval strategy, multiple LSH tables rather than one should be created to relax the conditions for similar search to overlook as few similar as possible [35]. In concrete, suppose there are two users ${u^*}_1$ and ${u^*}_2$, Step 1 is repeated L times to generate L LSH table $\{\{{u^*}_1 \to H_1({u^*}_1), \ldots, {u^*}_m \to H_1({u^*}_m)\}, \ldots, \{{u^*}_1 \to H_L({u^*}_1), \ldots, {u^*}_m \to H_L({u^*}_m)\}\}$. If equation in (7) holds, ${u^*}_1$ and ${u^*}_2$ can be regarded as similar users (denoted by ${u^*}_1 \overset{sim}{\leftrightarrow} {u^*}_2$).

$$\exists\, \beta,\ \text{satisfy } H_\beta({u^*}_1) = H_\beta({u^*}_2)(\beta \in \{1, \ldots, L\}) \tag{7}$$

Similarly, we obtain the LSH value matrix $H_\beta({u^*}_{key})$ of ${u^*}_{key}$, and if ${u^*}_{key} \overset{sim}{\leftrightarrow} {u^*}_1$, put ${u^*}_1$ into the similar user set of ${u^*}_{key}$ (SU_Set).

Step 3: Select TOP-3 services for ${u^*}_{key}$ from the similar candidates

In Step 2, we obtain a similar user set SU_Set of ${u^*}_{key}$. In this step, we predict the missing QoS values of services that have never been invoked by ${u^*}_{key}$, based on the QoS values of the users in set SU_Set. Concretely, we utilize the equation in (8) to calculate the predicted QoS value of different services in terms of dimensions $(d_1, \ldots, d_w)$ by ${u^*}_{key}$.

$$qos_{key,I,J} = \overline{{u^*}_{key,I}} + \frac{\sum_{u_a^* \in SU\_Set} \left| qos_{1,I,J} - \overline{{u^*}_{1,I}} \right|}{|SU\_Set|} \tag{8}$$

where $qos_{key,I,J}$ denotes the predicted value of Ith dimension of Jth service by ${u^*}_{key}$, $\overline{{u^*}_{key,I}}$ denotes the average QoS value of Ith quality dimension of all services by ${u^*}_{key}$. Similarly, $\overline{{u^*}_{1,I}}$ denotes the average QoS value of all services in Ith quality dimension by the similar user ${u^*}_1$.

Next, according to predicted value in (8), the candidate services are ranked. Finally, we select the TOP-3 services for each QoS dimension and then take their union to generate a predictive candidate service matrix $P(u^*_{key})$ as in (9). Here, p_{ij} denotes the predicted QoS value of the ith dimension of jth service $(1 < i \leq w,\ 1 < j \leq v)$ by u^*_{key}.

$$P(u^*_{key}) = \begin{bmatrix} p_{11} & \cdots & p_{1v} \\ \vdots & \ddots & \vdots \\ p_{w1} & \cdots & p_{wv} \end{bmatrix} \tag{9}$$

Step 4: TOP-3 services evaluation and ranking based on TOPSIS

Despite the above three steps, we have finished the service prediction process with privacy protection based on multi-dimensional QoS data and obtained the candidate recommendation list (denote $P(u^*_{key})$). However, determining the optimal services from candidate list to recommended to the key user is still not an easy task, which requires a suitable evaluate approaches to consider the characteristics and weights of all QoS dimensions. Thus, in this step, we employ the multi-attribute decision making technique TOPSIS [27] to comprehensive evaluate and rank the candidate services derived in Step 3 based on their respective multi-dimensional QoS data and return the optimal services. According to TOPSIS, the optimal solution is the one that has the shortest distance from the positive-ideal solution and has the longest distance from the negative-ideal solution [40]. Concretely, in this step, our proposed service evaluation approach based on TOPSIS consists of the following four sub-steps.

Step 4.1: It is necessary to normalize the matrix $P(u^*_{key})$ in (9) as the w dimensions of service are of either cost-type or benefit-type. Concretely, the normalization process is enacted according to the rules in (10)–(11). Here, the normalized value cs_{ij} in (10) can be calculated by (11). Here, J_1 and J_2 represent the benefit-type dimensions and cost-type dimensions, respectively.

$$CS(u^*_{key}) = \begin{bmatrix} cs_{11} & \cdots & cs_{1v} \\ \vdots & \ddots & \vdots \\ cs_{w1} & \cdots & cs_{wv} \end{bmatrix} \tag{10}$$

$$cs_{ij} = \begin{cases} \dfrac{p_{ij} - min\{p_{ij}\}}{max\{p_{ij}\} - min\{p_{ij}\}}, d_i \in J_1 \\ \dfrac{max\{p_{ij}\} - p_{ij}}{max\{p_{ij}\} - min\{p_{ij}\}}, d_i \in J_2 \end{cases} \tag{11}$$

Step 4.2: According to matrix $CS(u^*_{key})$ in (10), we determine the positive-ideal candidate service solution CS^+ by (12) and the negative-ideal candidate service solution CS^- by (13). Where $CS_i{}^+ = (max\_cs_{ij},\ j = 1, \ldots, v)$, $CS^- = (min\_cs_{ij},\ j = 1, \ldots, v)$. cs_{ij} represents the value of the ith dimension of the jth candidate service.

$$CS^+ = (CS_1{}^+, CS_2{}^+, \ldots, CS_w{}^+) \tag{12}$$

$$CS^- = (CS_1{}^-, CS_2{}^-, \ldots, CS_w{}^-) \tag{13}$$

Step 4.3: We calculate the distances of the jth $(j = 1, \ldots, v)$ candidate service from CS^+ and CS^-, respectively. The distance from the jth candidate service to CS^+ can be defined as in (14). Likewise, the distance of the jth candidate service from CS^- can be defined as in (15).

$$D_j{}^+ = \sqrt{\sum_{i=1}^{w} (CS_i{}^+ - cs_{ij})^2} \tag{14}$$

$$D_j{}^- = \sqrt{\sum_{i=1}^{w} (CS_i{}^- - cs_{ij})^2} \tag{15}$$

Step 4.4: We calculate the relative closeness R_j (comprehensive indicators for evaluating candidate services) of each candidate service ser_j to the positive-ideal candidate service solutions CS^+ by (16). Here, R_j belongs to [0, 1] (the larger the better, $(j = 1, \ldots, v)$). Then we rank service ser_j based on R_j and finally return the optimal services to the key user u^*_{key}.

$$R_j = \frac{D_j{}^-}{D_j{}^+ + D_j{}^-} \tag{16}$$

Through the above four steps of our approach $Rec_{LSH-TOPSIS}$, the optimal services can be recommended to u^*_{key} in a privacy-preserving manner. Next, we use the pseudo-code in Algorithm 1 and Algorithm 2 to specify our proposal more clearly. Concretely, in Algorithm 1, we build the multi-dimensional user indices (Lines 1–16) offline. Then, we find similar users of u^*_{key} based on user indices (Lines 17–23) online. After that, we calculate the predicted QoS value and generate a matrix $P(u^*_{key})$ of u^*_{key} (Lines 24–26). In Algorithm 2, through calculating cs_{ij}, we generate the normalized matrix $CS(u^*_{key})$ (Lines 1–5). Then, we determine CS^+ and CS^- (Lines 6–7). Finally, through calculating the relative closeness R_j, the candidate services are ranked and the optimal service is returned to the key user (Lines 8–14).

Algorithm 1: TOP-3 services selection

Input: u^*_{key}: a key user
$U^* = \{u^*_1, \ldots, u^*_m\}$: the set of users
$Dim = \{d_1, \ldots, d_w\}$: the set of quality dimensions
$Ser = \{ser_1, \ldots, ser_n\}$: the set of services
$ES = \{es_1, \ldots, es_z\}$: the set of Edge Servers
Output: u^*_{key}'s prediction candidate service matrix $P(u^*_{key})$

Process
1 **For** each $u_i^* \in U^*$ **do**
2 Normalize QoS data matrix $Qos(u^*_i)$ to $Qos'(u^*_i)$ according to (2)
3 **End For**
4 **For** $k = 1$ to L **do** // L hash table
5 **For** $i = 1$ to r **do** // r LSH function in each LSH table
6 **For** $j = 1$ to n **do** // n-dimensional vector depicting a user
7 $V[k]_{ij}$ = random [-1, 1]
8 **End For**
9 **End For**
10 **End For**
11 **For** each $u_i^* \in U^*$ **do**
12 **For** $k = 1$ to L **do**
13 $S(u^*_i) = Qos'(u^*_i) \circ V[k]$ // LSH mapping
14 **Mapping** $H_k(u^*_i)$ to 0-1 matrix according to (4) - (5)
15 **End For**
16 **End For**
17 **For** each $u_i^* \in U^*$ **do**
18 **For** $k = 1$ to L **do**
19 **If** $H_k(u^*_i) = H_k(u^*_{key})$
20 **Then** put u^*_i into SU_Set
21 **End If**
22 **End For**
23 **End For**
24 Calculate the predicted QoS value $qos'_{key,\ l,\ J}$ based on equation (8)
25 Generate prediction candidate service matrix $P(u^*_{key})$ as in (9)
26 **Return** $P(u^*_{key})$

Algorithm 2: Service evaluation and optimal services selection

Input: Prediction candidate services matrix $P(u^*_{key})$
Output: $ser_{optimal}$

Process
1 **For** i =1 to w **do** // w quality dimensions
2 **For** j =1 to v **do** // v candidate services
3 Calculate cs_{ij} based on (11)
4 **End for**
5 **End for**
6 Determine CS^+ based on (12)
7 Determine CS^- based on (13)
8 **For** $j = 1$ to v **do**
9 Calculate $D_j{}^+$ based on (14)
10 Calculate $D_j{}^-$ based on (15)
11 Calculate R_j based on (16)
12 **End For**
13 Rank candidate services based on R_j
14 **Return** $ser_{optimal}$

Table 2
Historical QoS data of users.

		d_1	d_2			d_1	d_2
u^*_{key}	ser_1	0.32	16.42	u^*_1	ser_1	2.76	4.71
	ser_2	0.49	24.09		ser_2	5.63	20.62
	ser_3	0.65	10.85		ser_3	0.47	20.68
	ser_4	Null	Null		ser_4	0.25	6.82
	ser_5	Null	Null		ser_5	1.11	12.98
	ser_6	0.12	13.1		ser_6	2.70	16.24
	ser_7	Null	Null		ser_7	3.39	17.14
	ser_8	Null	Null		ser_8	Null	Null
	ser_9	0.25	13.23		ser_9	3.54	15.36
	ser_{10}	Null	Null		ser_{10}	0.85	21.52
u^*_2	ser_1	0.52	17.01	u^*_3	ser_1	0.28	15.71
	ser_2	0.38	24.33		ser_2	0.51	23.06
	ser_3	0.58	11.01		ser_3	0.71	11.02
	ser_4	0.09	19.18		ser_4	0.27	16.71
	ser_5	0.40	14.33		ser_5	0.33	13.77
	ser_6	0.08	12.41		ser_6	Null	Null
	ser_7	Null	Null		ser_7	0.31	18.27
	ser_8	0.14	15.99		ser_8	0.34	5.88
	ser_9	0.19	12.98		ser_9	0.30	12.59
	ser_{10}	0.48	16.62		ser_{10}	0.54	17.25
u^*_4	ser_1	1.98	9.47	u^*_5	ser_1	0.27	17.54
	ser_2	4.75	206.42		ser_2	0.44	25.11
	ser_3	2.56	9.47		ser_3	0.70	12.24
	ser_4	0.89	9.34		ser_4	Null	Null
	ser_5	6.15	9.47		ser_5	0.24	5.98
	ser_6	3.16	8.67		ser_6	0.09	18.12
	ser_7	1.56	14.03		ser_7	0.28	12.94
	ser_8	2.14	8.44		ser_8	0.11	9.14
	ser_9	0.25	13.23		ser_9	0.18	14.33
	ser_{10}	Null	Null		ser_{10}	0.38	16.58

5. Case study

In order to illustrate the feasibility of our approach, a case study is provided in this section. We assume that there are 10 services (i.e., ser_n $(1 < n \le 10)$) invoked by 6 users (i.e., u^*_{key} and u^*_i $(1 < i \le 5)$) and 2 quality dimensions (i.e., d_1 = *response time*; d_2 = *throughput*) for each service. According to the above design, the original QoS values of users is tabulated in Table 2, if service ser_n has never been invoked by a user, the QoS values will be marked null in the table. The specific steps of our approach are introduced as follows.

Step 1. Data pre-processing and multi-dimensional user indices building offline

In this section, we illustrate the process of building u^*_{key}'s index with only one hash table. First, we normalize the original QoS data matrix to $QoS'(u^*_{key})$ by Eq. (2), the results are shown in (17). Next, we randomly generate a 10*6 matrix V_1, whose elements range from [-1, 1] as in (18). After that, according to (4), $S(u^*_{key})$ is obtained and shown in (19). Finally, through the conversion function (5), we get the hash value matrix $H(u^*_{key})$ which can be regarded as the index for u^*_{key} and is shown in (20).

$$QoS'(u^*_{key}) = \begin{bmatrix} 0.009 & 0.014 & 0.018 & \text{Null} & \text{Null} & 0.003 & \text{Null} & \text{Null} & 0.007 & \text{Null} \\ 0.446 & 0.676 & 0.295 & \text{Null} & \text{Null} & 0.356 & \text{Null} & \text{Null} & 0.359 & \text{Null} \end{bmatrix} \quad (17)$$

$$V_1 = \begin{bmatrix} -0.166 & 0.441 & -1 & -0.395 & -0.706 & -0.815 \\ -0.627 & -0.309 & -0.206 & 0.078 & -0.162 & 0.37 \\ -0.591 & 0.756 & -0.945 & 0.341 & -0.165 & 0.117 \\ -0.719 & -0.604 & 0.601 & 0.937 & -0.373 & 0.385 \\ 0.753 & 0.789 & -0.83 & 0.922 & -0.66 & 0.756 \\ -0.803 & -0.158 & 0.916 & 0.066 & 0.384 & 0.369 \\ 0.373 & 0.669 & -0.963 & 0.5 & 0.978 & 0.496 \\ -0.439 & 0.579 & -0.794 & -0.104 & 0.817 & -0.413 \\ -0.424 & -0.74 & -0.961 & 0.358 & -0.577 & -0.469 \\ -0.017 & -0.893 & 0.148 & -0.707 & 0.179 & 0.4 \end{bmatrix} \quad (18)$$

$$S(u^*_{key}) = QoS'(u^*_{key}) \circ V_1 = \begin{bmatrix} -0.026 & 0.008 & -0.032 & 0.006 & -0.014 & 0.012 \\ -1.111 & -0.112 & -0.884 & 0.129 & -0.544 & 0.611 \end{bmatrix} \quad (19)$$

$$H(u^*_{key}) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \quad (20)$$

Step 2: Find similar users of u^*_{key} based on user indices online

After obtaining the index of u^*_{key}, we repeat the above process to build indices for the remaining users. The user indices are displayed in Table 3.

As indicated in Table 3, the index values of u^*_2, u^*_3, u^*_5 are the same as that of u^*_{key}; therefore, they are similar users of u^*_{key} according to the LSH theory.

Step 3: Select TOP-3 services for u^*_{key} from the similar candidates

For each service which has never been invoked by u^*_{key}, its QoS data is predicted based on Eq. (8). The result of the predicted value is tabulated in Table 4. Next, we take a union of TOP-3 services in each dimension to generate a matrix $P(u^*_{key})$, as in (21).

$$P(u^*_{key}) = \begin{array}{c} \\ d_1 \\ d_2 \end{array} \begin{array}{c} \begin{array}{cccc} ser_5 & ser_7 & ser_8 & ser_{10} \end{array} \\ \begin{bmatrix} 0.612 & 0.420 & 0.507 & 0.494 \\ 19.367 & 18.076 & 20.393 & 21.388 \end{bmatrix} \end{array} \quad (21)$$

Step 4: TOP-3 services evaluation and ranking based on TOPSIS

In this step, we perform a comprehensive ranking of candidates in matrix $P(u^*_{key})$ according to the TOPSIS approach. Here, *response time* and *throughput* are considered as two indicators of service quality, in which *response time* is a benefit-type (J_1) dimension while *throughput* is a cost-type (J_2) dimension. Therefore, we normalize matrix $P(u^*_{key})$ to transform it into a dimensionless one, i.e., $CS(u^*_{key})$ in (22). Concretely, the normalized matrix $CS(u^*_{key})$ can be calculated by cs_{ij} based on Eq. (11).

$$CS(u^*_{key}) = \begin{array}{c} \\ d_1 \\ d_2 \end{array} \begin{array}{c} \begin{array}{cccc} cs_5 & cs_7 & cs_8 & cs_{10} \end{array} \\ \begin{bmatrix} 0 & 1 & 0.547 & 0.615 \\ 0.394 & 0 & 0.700 & 1 \end{bmatrix} \end{array} \quad (22)$$

According to matrix $CS(u^*_{key})$, the positive-ideal candidate service solutions CS^+ and negative-ideal candidate service CS^- can be obtained by Eqs. (12) and (13), respectively, as shown in Table 5. Namely, $CS^+ = CS_1{}^+, CS_2{}^+ = \{1, 1\}$, $CS^- = CS_1{}^-, CS_2{}^- = \{0, 0\}$.

Finally, the distances of each candidate services from CS^+ and CS^- are calculated by Eqs. (14)–(15), whose results are shown in (23)–(26).

$$D_5{}^+ = \sqrt{\sum_{i=1}^{2}(CS_i{}^+ - cs_{i5})^2} = 1.169, \quad D_5{}^- = \sqrt{\sum_{i=1}^{2}(CS_i{}^- - cs_{i5})^2} = 0.394 \quad (23)$$

$$D_7{}^+ = \sqrt{\sum_{i=1}^{2}(CS_i{}^+ - cs_{i7})^2} = 1.000, \quad D_7{}^- = \sqrt{\sum_{i=1}^{2}(CS_i{}^- - cs_{i7})^2} = 1.000 \quad (24)$$

$$D_8{}^+ = \sqrt{\sum_{i=1}^{2}(CS_i^+ - cs_{i8})^2} = 0.543, \quad D_8{}^- = \sqrt{\sum_{i=1}^{2}(CS_i^- - cs_{i8})^2} = 0.888 \quad (25)$$

$$D_{10}{}^+ = \sqrt{\sum_{i=1}^{2}(CS_i{}^+ - cs_{i10})^2} = 0.385 \quad D_{10}{}^- = \sqrt{\sum_{i=1}^{2}(CS_i{}^- - cs_{i10})^2} = 1.173 \quad (26)$$

The relative closeness (R) is calculated by Eqs. (16), whose results are shown in (27)–(30).

$$R_5 = \frac{D_5{}^-}{D_5{}^+ + D_5{}^-} = 0.252 \quad (27)$$

Table 3
Indices of users.

	User index		User index
u^*_{key}	$H(u^*_{key}) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$	$\boldsymbol{u^*_3}$	$H(u^*_3) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$
u^*_1	$H(u^*_1) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$	u^*_4	$H(u^*_4) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$
$\boldsymbol{u^*_2}$	$H(u^*_2) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$	$\boldsymbol{u^*_5}$	$H(u^*_5) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$

Table 4
The multi-dimensional predicted QoS value of u^*_{key}.

	ser_4	ser_5	ser_7	ser_8	ser_{10}
d_1	0.544	0.621	0.420	0.507	0.494
d_2	18.031	19.367	18.076	20.393	21.388

Table 5
Positive-ideal solution and Negative-ideal solution.

	d_1	d_2
CS^+	1	1
CS^-	0	0

Table 6
Candidate services ranking based on TOPSIS.

	d_1	d_2	D^+	D^-	R	*Rank*
ser_5	0	0.394	1.169	0.394	0.252	4
ser_7	1	0	1.000	1.000	0.500	3
ser_8	0.547	0.700	0.543	0.888	0.620	2
$\boldsymbol{ser_{10}}$	0.615	1	0.385	1.173	0.753	1

$$R_7 = \frac{D_7{}^-}{D_7{}^+ + D_7{}^-} = 0.500 \tag{28}$$

$$R_8 = \frac{D_8{}^-}{D_8{}^+ + D_8{}^-} = 0.620 \tag{29}$$

$$R_{10} = \frac{D_{10}{}^-}{D_{10}{}^+ + D_{10}{}^-} = 0.753 \tag{30}$$

In order to compare and rank the candidate services more clearly, we summarized the relevant calculation results into Table 6.

As Table 6 indicates, candidate service ser_{10} has the smallest distance to CS^+ and the largest distance to CS^-; the relative closeness of ser_{10} is closer to 1. Therefore, ser_{10} is ranked No.1 among all the candidate services and should be recommended to u^*_{key}.

6. Conclusions and future work

In this paper, we propose a multi-dimensional quality-driven recommendation approach with privacy-preservation, named $Rec_{LSH-TOPSIS}$, based on LSH and TOPSIS techniques. Different from existing approaches, we extend the traditional single-dimensional quality-driven LSH recommendation approach to the multi-dimensional scenario, to make the recommendation solution more comprehensive. Furthermore, for the candidate services returned by LSH recommendations, we use TOPSIS technique to evaluate them objectively so as to avoid the inappropriate and fuzzy weight assignment by key users; thus, user's burden can be alleviated considerably. Finally, an optimal service is returned to the key user. To validate the feasibility of $Rec_{LSH-TOPSIS}$, a case study is presented to clarify the detailed recommendation process.

In the future, we will further improve our proposal by launching a set of real-world experiments and compare its performances with other related approaches. Besides, the available data for recommendation decision-makings are often very sparse [41–43] and context-aware [44–47], we will continue to refine our work by considering these uncertain influencing factors.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Weiyi Zhong: Conceptualization, Methodology, Software. **Xiaochun Yin:** Software, Validation. **Xuyun Zhang:** Formal analysis. **Shancang Li:** Validation. **Wanchun Dou:** Visualization, Investigation. **Ruili Wang:** Writing - review & editing. **Lianyong Qi:** Supervision.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2017YFB1400600), the National Natural Science Foundation of China (No. 61872219, No. 61672276) and the Natural Science Foundation of Shandong Province (ZR2019MF001).

References

[1] L. Qi, Y. Chen, Y. Yuan, S. Fu, et al., A QoS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems, World Wide Web J. 23 (2020) 1275–1297, http://dx.doi.org/10.1007/s11280-019-00684-y.

[2] S. Zhang, X. Li, Z. Tan, T. Peng, et al., A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services, Future Gener. Comput. Syst. 94 (2019) 40–50, http://dx.doi.org/10.1016/j.future.2018.10.053.

[3] S. Wan, Y. Xia, L. Qi, Y. Yang, et al., Automated colorization of a grayscale image with seed points propagation, IEEE Trans. Multimed. (2020) http://dx.doi.org/10.1109/TMM.2020.2976573.

[4] S. Ding, S. Qu, Y. Xi, S. Wan, Stimulus-driven and concept-driven analysis for image caption generation, Neurocomputing (2019) http://dx.doi.org/10.1016/j.neucom.2019.04.095.

[5] Z. Gao, H.Z. Xuan, H. Zhang, S. Wan, et al., Adaptive fusion and category-level dictionary learning model for multiview human action recognition, IEEE Internet Things J. 6 (6) (2019) 9280–9293, http://dx.doi.org/10.1109/JIOT.2019.2911669.

[6] S. Wan, Z. Gu, Q. Ni, Cognitive computing and wireless communications on the edge for healthcare service robots, Comput. Commun. 149 (2020) 99–106, http://dx.doi.org/10.1016/j.comcom.2019.10.012.

[7] Y. Yin, W. Zhang, Y. Xu, H. Zhang, et al., Qos prediction for mobile edge service recommendation with auto-encoder, IEEE Access 7 (2019) 62312–62324, http://dx.doi.org/10.1109/ACCESS.2019.2914737.

[8] S. Wan, X. Li, Y. Xue, W. Lin, et al., Efficient computation offloading for internet of vehicles in edge computing-assisted 5G networks, J. Supercomput. (2019) 1–30, http://dx.doi.org/10.1007/s11227-019-03011-4.

[9] D. Goldberg, D. Nichols, B.M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Commun. ACM 35 (12) (1992) 61–70, http://dx.doi.org/10.1145/138859.138867.

[10] K.K. Fletcher, X.F. Liu, A collaborative filtering method for personalized preference-based service recommendation, in: IEEE International Conference on Web Services, 2015, pp. 400–407, http://dx.doi.org/10.1109/ICWS.2015.60.

[11] L. Qi, X. Zhang, W. Dou, C. Hu, et al., A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment, Future Gener. Comput. Syst. 88 (2018) 636–643, http://dx.doi.org/10.1016/j.future.2018.02.050.

[12] X. Xu, H. Cao, Q. Geng, X. Liu, et al., Dynamic resource provisioning for workflow scheduling under uncertainty in edge computing environment, Concurr. Comput.-Pract. Exp. (2019) e5674, http://dx.doi.org/10.1002/cpe.5674.

[13] S. Zhang, G. Wang, M.Z.A. Bhuiyan, Q. Liu, A dual privacy preserving scheme in continuous location-based services, IEEE Internet Things J. 5 (5) (2018) 4191–4200, http://dx.doi.org/10.1109/JIOT.2018.2842470.

[14] Y. Wang, Z. Cai, Z. Zhan, Y. Gong, et al., An optimization and auction-based incentive mechanism to maximize social welfare for mobile crowdsourcing, IEEE Trans. Comput. Soc. Syst. 6 (3) (2019) 414–429, http://dx.doi.org/10.1109/TCSS.2019.2907059.
[15] Y. Wang, Z. Cai, X. Tong, Y. Gao, et al., Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems, Comput. Netw. 135 (2018) 32–43, http://dx.doi.org/10.1016/j.comnet.2018.02.008.
[16] X. Zhou, W. Liang, K. Wang, S. Shimizu, Multi-modality behavioral influence analysis for personalized recommendations in health social media environment, IEEE Trans. Comput. Soc. Syst. 6 (5) (2019) 888–897, http://dx.doi.org/10.1109/TCSS.2019.2918285.
[17] X. Zhou, W. Liang, K. Wang, R. Huang, et al., Research Collaboration, Academic influence aware and multidimensional network analysis for navigation based on scholarly big data, IEEE Trans. Emerg. Top. Comput. (2018) 1, http://dx.doi.org/10.1109/TETC.2018.2860051.
[18] M. Alrifai, D. Skoutas, T. Risse, Selecting skyline services for qos-based web service composition, in: Proceedings of the 19th International Conference on World Wide Web (WWW), 2010, pp. 11–20, http://dx.doi.org/10.1145/1772690.1772693.
[19] C. Li, Z. Ma, Y. Zhou, Periodic orbits in 3-dimensional systems and application to a perturbed Volterra system, J. Differential Equations 260 (2016) 2750–2762, http://dx.doi.org/10.1016/j.jde.2015.10.018.
[20] X. Zhou, W. Liang, S. Huang, M. Fu, Social recommendation with large-scale group decision making for cyber-enabled online service, IEEE Trans. Comput. Soc. Syst. 6 (5) (2019) 1073–1082, http://dx.doi.org/10.1109/TCSS.2019.2932288.
[21] Y. Zhao, F. Luo, M. Chen, Y. Wang, et al., Evaluating multi-dimensional visualizations for understanding fuzzy clusters, IEEE Trans. Vis. Comput. Graphics 25 (1) (2019) 12–21, http://dx.doi.org/10.1109/TVCG.2018.2865020.
[22] X. Wang, M. Wang, Adaptive group bridge estimation for high-dimensional partially linear models, J. Inequal. Appl. 2017 (158) (2017) http://dx.doi.org/10.1186/s13660-017-1432-x.
[23] C. Stringer, M. Pachitariu, N. Steinmetz, et al., High-dimensional geometry of population responses in visual cortex, Nature 571 (2019) 361–365, http://dx.doi.org/10.1038/s41586-019-1346-5.
[24] S. Wan, Y. Zhao, T. Wang, Z. Gu, et al., Multi-dimensional data indexing and range query processing via voronoi diagram for internet of things, Future Gener. Comput. Syst. 91 (2019) 382–391, http://dx.doi.org/10.1016/j.future.2018.08.007.
[25] H. Tian, M. Han, Bifurcation of periodic orbits by perturbing high-dimensional piecewise smooth integrable systems, J. Differential Equations 263 (11) 7448–7474, http://dx.doi.org/10.1016/j.jde.2017.08.011.
[26] W. Gong, L. Qi, Y. Xu, Privacy-aware multi-dimensional mobile service quality prediction and recommendation in distributed fog environment, Wirel. Commun. Mobile Comput. 2018 (2018) http://dx.doi.org/10.1155/2018/3075849.
[27] C.L. Hwang, K. Yoon, Multiple Attribute Decision Making: Methods and Applications, Springer-Verlag, New York, 1981.
[28] Y. Zhang, G. Cui, S. Deng, F. Chen, et al., Efficient query of quality correlation for service composition, IEEE Trans. Serv. Comput. (2018) 1, http://dx.doi.org/10.1109/TSC.2018.2830773.
[29] H. Wang, D. Yang, Q. Yu, Y. Tao, Integrating modified cuckoo algorithm and creditability evaluation for QoS-aware service composition, Knowl.-Based Syst. 140 (2018) 64–81, http://dx.doi.org/10.1016/j.knosys.2017.10.027.
[30] S. Ding, S. Qu, Y. Xi, S. Wan, A long video caption generation algorithm for big video data retrieval, Future Gener. Comput. Syst. 93 (2019) 583–595, http://dx.doi.org/10.1016/j.future.2018.10.054.
[31] K. Chung, D. Lee, K.J. Kim, Categorization for grouping associative items using data mining in item-based collaborative filtering, Multimedia Tools Appl. 71 (2) (2014) 889–904, http://dx.doi.org/10.1007/s11042-011-0885-z.
[32] C. Jiang, R. Duan, H.K. Jain, S. Liu, et al., Hybrid collaborative filtering for high-involvement products: A solution to opinion sparsity and dynamics, Decis. Support Syst. 79 (2015) 195–208, http://dx.doi.org/10.1016/j.dss.2015.09.002.
[33] Z. Fu, K. Ren, J. Shu, X. Sun, et al., Enabling personalized search over encrypted outsourced data with efficiency improvement, IEEE Trans. Parallel Distrib. Syst. 27 (9) (2016) 2546–2559, http://dx.doi.org/10.1109/TPDS.2015.2506573.
[34] Z. Xia, X. Wang, X. Sun, Q. Wang, A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data, IEEE Trans. Parallel Distrib. Syst. 27 (2) (2015) 340–352, http://dx.doi.org/10.1109/TPDS.2015.2401003.
[35] F. Casino, J. Domingo-Ferrer, C. Patsakis, D. Puig, et al., A k-anonymous approach to privacy preserving collaborative filtering, J. Comput. System Sci. 81 (6) (2015) 1000–1011, http://dx.doi.org/10.1016/j.jcss.2014.12.013.
[36] W. Dou, X. Zhang, J. Liu, J. Chen, HireSome-II: towards privacy-aware cross-cloud service composition for big data applications, IEEE Trans. Big Data 26 (2) (2015) 455–466, http://dx.doi.org/10.1109/TPDS.2013.246.
[37] J. Zhu, P. He, Z. Zheng, M.R. Lyu, A privacy-preserving QoS prediction framework for web service recommendation, in: 22nd International Conference on Web Services, IEEE, New York, 2015, pp. 241–248, http://dx.doi.org/10.1109/ICWS.2015.41.
[38] L. Qi, X. Zhang, S. Li, S. Wan, et al., Spatial-temporal data-driven service recommendation with privacy-preservation, Inform. Sci. 515 (2020) 91–102, http://dx.doi.org/10.1016/j.ins.2019.11.021.
[39] J. Lee Rodgers, W.A. Nicewander, Thirteen ways to look at the correlation coefficient, Amer. Statist. 42 (1) (1988) 59–66, http://dx.doi.org/10.1080/00031305.1988.10475524.
[40] A.T. Gumus, Evaluation of hazardous waste transportation firms by using a two step fuzzy-AHP and TOPSIS methodology, Expert Syst. Appl. 36 (2) (2009) 4067–4074, http://dx.doi.org/10.1016/j.eswa.2008.03.013.
[41] Y. Zhang, K. Wang, Q. He, F. Chen, et al., Covering-based web service quality prediction via neighborhood-aware matrix factorization, IEEE Trans. Serv. Comput. (2019) 1, http://dx.doi.org/10.1109/TSC.2019.2891517.
[42] H. Liu, H. Kou, C. Yan, L. Qi, Link prediction in paper citation network to construct paper correlated graph, EURASIP J. Wireless Commun. Networking 2019 (2019) 233, http://dx.doi.org/10.1186/s13638-019-1561-7.
[43] X. Xu, Q. Cai, G. Zhang, J. Zhang, et al., An incentive mechanism for crowdsourcing markets with social welfare maximization in cloud-edge computing, Concurr. Comput.: Pract. Exper. (2018) e4961, http://dx.doi.org/10.1002/cpe.4961.
[44] B. Zhao, Y. Wang, Y. Li, Y. Gao, et al., Task allocation model based on worker friend relationship for mobile crowdsourcing, Sensors 19 (4) (2019) 921, http://dx.doi.org/10.3390/s19040921.
[45] S. Zhang, K.K.R. Choo, Q. Liu, G. Wang, Enhancing privacy through uniform grid and caching in location-based services, Future Gener. Comput. Syst. 86 (2018) 881–892, http://dx.doi.org/10.1016/j.future.2017.06.022.
[46] S. Wan, S. Goudos, Faster R-CNN for multi-class fruit detection using a robotic vision system, Comput. Netw. 168 (2020) 107036, http://dx.doi.org/10.1016/j.comnet.2019.107036.
[47] Y. Zhang, C. Yin, Q. Wu, Q. He, et al., Location-aware deep collaborative filtering for service recommendation, IEEE Trans. Syst. Man Cybern.: Syst. (1–12) (2019) http://dx.doi.org/10.1109/TSMC.2019.2931723.