Received: 11 February 2022 | Revised: 10 March 2022 | Accepted: 22 March 2022

DOI: 10.1002/dac.5161

SPECIAL ISSUE ARTICLE

WILEY

Geographical POI recommendation for Internet of Things: A federated learning approach using matrix factorization

Jiwei Huang | **Zeyu Tong** | **Zihan Feng**

Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum - Beijing, Beijing, China

Correspondence
Jiwei Huang, Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum - Beijing, No. 18 Fuxue Road, Beijing 102249, China.
Email: huangjw@cup.edu.cn

Funding information
Beijing Natural Science Foundation, Grant/Award Number: 4202066; Beijing Nova Program, Grant/Award Number: Z201100006820082; National Natural Science Foundation of China, Grant/Award Number: 61972414

Summary

With the popularity of Internet of Things (IoT), Point-of-Interest (POI) recommendation has become an important application for location-based services (LBS). Meanwhile, there is an increasing requirement from IoT devices on the privacy of user sensitive data via wireless communications. In order to provide preferable POI recommendations while protecting user privacy of data communication in a distributed collaborative environment, this paper proposes a federated learning (FL) approach of geographical POI recommendation. The POI recommendation is formulated by an optimization problem of matrix factorization, and singular value decomposition (SVD) technique is applied for matrix decomposition. After proving the nonconvex property of the optimization problem, we further introduce stochastic gradient descent (SGD) into SVD and design an FL framework for solving the POI recommendation problem in a parallel manner. In our FL scheme, only calculated gradient information is uploaded from users to the FL server while all the users manage their rating and geographic preference data on their own devices for privacy protection during communications. Finally, real-world dataset from large-scale LBS enterprise is adopted for conducting extensive experiments, whose experimental results validate the efficacy of our approach.

KEYWORDS

federated learning, geographical recommendation, Internet of Things, matrix factorization, Point-of-Interest

1 | INTRODUCTION

With the emergence of mobile computing and Internet of Things (IoT), there is an exponentially growing popularity of mobile devices. It has been reported that there are nearly 7 billion IoT devices and 3 billion smartphones in the world, equipped with advanced sensors as well as computing and communication capabilities.[1] It has led to the emergence of location-based services (LBS), in which geographic data and information are fully utilized for providing personalized IT services to mobile devices. In LBS, Point-of-Interests (POI) recommendation, which is to provide personalized recommendation of places to mobile users in IoT, is an important issue.[2] Both POI check-in and device mobility data as well as the timing effects have to be fully considered, making POI recommendation an extremely challenging task and thus a hot topic in both academia and industry.

For decades, collaborative filtering (CF) technology is one of the most widely applied solutions for recommendation systems.[3,4] Its basic idea is to make a recommendation according to the users' "preference" history by finding similar users (user-based) or similar services (item-based), where the similarity can be calculated by Pearson correlation or cosine similarity. However, with the increasing amount of data in large-scale IoT environments,[5] the data sparsity problem has emerged and even has become nearly inevitable, resulting in the unsatisfactoriness of the recommendation results in several scenarios. Meanwhile, the CF-based approaches can only deal with low dimensional and linear features between users and services. Since both user preference and geographic factors should be analyzed simultaneously in POI recommendation, CF faces significant challenges and becomes even unpractical in IoT. An alternate approach for POI recommendation is machine learning (ML) technology, such as convolutional neural network,[6] and graph neural network.[7] Multiple attributes can be aggregated as the input of an ML model, and one can achieve better recommendation results by adjusting various parameters. However, due to the well-known interpretability problem of neural networks, one has to train the models very carefully by feeding them extremely large amounts of data, and thus, the models might be quite sensitive to their training dataset. To solve the above problems, in this paper, we apply matrix factorization techniques[8] to POI recommendation.

The privacy of personal geographic data during communication is another important issue in POI recommendation for IoT.[9,10] With the increase of the amount of data, the training tasks require significantly high computational resources, and thus, distributed collaborative systems such as cloud computing have to be involved. However, malicious nodes may intercept or discard data packets, thus interfering with the communication process and causing privacy disclosure.[11] Meanwhile, sharing data among multiple systems or business process may also cause the problem of privacy disclosure.[12] Therefore, in order to protect the privacy of user data meanwhile completing complex ML tasks in a distributed computing environment, a novel framework called federated learning (FL) has been proposed.[13] It was first proposed by Google in 2016, originally used to solve the problem of Android mobile phone end users updating ML models locally. It is designed to protect the privacy of terminal data and personal data in efficient ML between multiple participants or multiple computing nodes. The FL is a general framework applicable to most AI algorithms including neural networks and random forest and is expected to become the basis of the next generation of artificial intelligence collaborative algorithms and collaborative networks.

The original matrix factorization in recommendation systems reveals two types of the privacy information of users[14]: (I) *the original preference data of users* and (II) *the potential feature vector learned by users*. Previous studies have shown that both original preference data and potential features may reveal users' sensitive attributes, such as age, geographical locations, relationship status, political views, and sexual orientation. Therefore, it is important to protect users' privacy information in communication when performing matrix grading tasks. Some existing research work dedicating to the privacy protection in matrix decomposition for recommendation studied this issue by either (I) *fuzzy processing*[15] or (II) *information encryption*.[16] However, for both of them, users and/or system managers have to spend extremely expensive computational overhead and hence sacrifice the performance. To attack this challenge, this paper introduces FL into matrix factorization for POI recommendation in IoT environments. On the one hand, since the ML models are trained in the mobile devices locally, the privacy can be protected essentially by avoiding transferring any user private data to the servers via the Internet or local area network (LAN). On the other hand, FL is a distributed collaborative framework, where multiple devices and servers can process the training and recommendation tasks in a parallel way, and thus, the performance can be improved after introducing FL to POI recommendation.

The contributions of this paper are summarized as follows.

- We propose an approach of constructing geographic information matrix to quantify the location information in IoT. The users' activity matrix and POIs' influence matrix are defined. After constructing the geographic information matrix, we integrate it with rating matrix and theoretically formulate the problem of matrix factorization for POI recommendation.
- We propose an FL framework, namely, FED-RGMF and its corresponding matrix factorization algorithm. In our approach, each client locally calculates the gradient and updates the user potential vector and sends the gradient information to the server with privacy guaranteed. The server deals with the returned gradient using the federal averaging method and then updates the POI information for recommendation.
- We conduct experiments using real-world Yelp dataset to evaluate the performance of our approach. Comparing with other two well-known recommendation schemes, the experimental results illustrate that our approach can not only obtain better recommendation results but also enhance the performance in running time.

The remainder of this paper is organized as follows. Section 2 surveys the related work. In Section 3, the POI recommendation problem is theoretically formulated, and the FED-RGMF framework is presented. In Section 4, the geographic information matrix is defined and calculated, and singular value decomposition (SVD) technique is applied for matrix decomposition. In Section 5, the FL framework is applied and detailed schemes are presented. Extensive experiments based on real-world dataset are conducted in Section 6. Finally, we summarize this paper in Section 7.

2 | RELATED WORK

POI recommendation has been an important application of LBS in IoT. There have been several existing works dedicating to designing effective and efficient schemes of POI recommendation. Earlier research work mainly relied on user trajectory data to infer user preferences. For example, previous studies applied CF-based methods to recommend locations and travel packages based on user trajectory data.[17] However, these methods often only considered the dimensional factor of user check-in decision. More recent studies began to explore the influence of user preference, social impact, and geographic impact on POI recommendation. Ye et al[4] tailored the CF model for POI recommendations, aiming at improving the recommendation accuracy. Some research focused on recommending some specific types of locations. Park et al[18] designed a system based on Bayesian learning with both users' preferences and location contexts to recommend restaurants. Similarly, Horozov et al[19] developed a user-based CF system to recommend restaurants to a user by finding which restaurants similar users have visited before. Zheng et al[20] designed a random walk style model to do tourism hot spot recommendation by taking into account both users' travel experiences and location attractiveness. Zheng et al[21] considered location recommendation and activity recommendation together and formulated a location activity matrix for CF in recommendation.

Since Netflix challenge,[22] matrix factorization has been proved to be one of the most accurate recommendation methods. Yin et al[23] propose a holistic personalized recommendation framework that contains two individual models and one ensemble model, which are based on joint matrix factorization and cognitive knowledge mining. Yin et al. designed their model to leverage users hidden relationships, items hidden relationships, and both types of hidden relationships, respectively. In the POI recommendation scenario, numerous previous works applied the collaborative location activity filtering (CLAF) algorithm proposed by Zhang et al in[21] for general recommendation. CLAF was a matrix factorization method, which makes recommendations based on the correlation between location features and POIs. The different regularization matrix factorization methods proposed in Ye et al[4] applied personalized CF method, which were used to reduce the dimension of user-POI matrix to minimize the square of regularization error. Sattari et al[24] proposed improved feature combination (IFC), which is based on the extended matrix decomposition model and integrates additional data resources before applying SVD technology to the extended model. Many studies show that IFC is better than CLAF in prediction accuracy. Lian et al[25] proposed a GeoMF model, which decomposes the weighted matrix (WMF) and introduces the impact of geographic information on sparse check-in data to be processed. Nonzero check-in was set to large weight, and zero check-in (unreached POI) was set to small weight. Yuan et al[17] considered the time factor into the matrix factorization. They proposed a method called user-based CF with temporal preference and smoothing enhancement with spatial influence-based recommendation with popularity enhancement (UTE+SE), which combines temporal and geographic influence into user-based CF. Xu et al[26] proposed a multifactor influencing POI recommendation model based on martix factorization, which considers the influences of both the geographic factor and the user factor (GeoUMF). They also considered the difference between the ranking produced in the recommendation model and the actual ranking in the check-in data.

FL is an ML technology that collectively learns intelligent models by protecting the privacy of scattered user data.[27,28] Different from the existing ML methods based on centralized storage of user data, FL stores user data on local user equipment.[13] Each device maintains a local model and calculates the update of the local model according to the user data stored on the device. Local model updates from multiple users are uploaded to a central server that coordinates the model training process. These updates are aggregated into a unified update to update the global model maintained by the server. The updated model will be further distributed to all user equipment to update the local model. This process is performed iteratively until the model converges. Because the model update usually contains less privacy information and the original user data never leaves the device, it can effectively reduce the risk of privacy disclosure.[29]

There are some research works that have applied the FL framework in the recommendation system. Ammad et al[30] proposed a collaborative filtering algorithm (FCF) under the FL framework. In FCF, each user device locally computed the gradients of the user and item embeddings based on the personal ratings stored on this device. The user embeddings were locally updated, and the gradients of item embeddings were uploaded to a central server. The server aggregated the item gradients from massive clients to update the global item embeddings. The updated item embeddings were further distributed to user clients for local embedding updates. However, the gradients of item embedding may leak some information on the private ratings. Therefore, Chai et al[31] proposed a matrix factorization algorithm (FedMF) under FL to encrypt the information of gradient interaction between server and client. In addition, Chuhan et al[32] proposed a privacy based FL GNN recommendation algorithm (FedGNN), which trains the GNN model from the decentralized user dataset and protects the interactive information with FL while considering the high-order interaction between users and items.

In our work, we quantify the geographic impact and build a geographic information matrix. The geographic information and user preferences are integrated into the target matrix in our FL framework of Geographical POI Recommendation using Matrix Factorization (Fed-GRMF), and the recommendation results are obtained by SVD, and the user privacy is effectively protected under the FL framework.

3 | THE FRAMEWORK OF FED-RGMF FOR GEOGRAPHICAL POI RECOMMENDATION

In this section, we formulate the POI recommendation problem mathematically and then present the FL based framework of matrix factorization for geographic POI recommendation.

3.1 | Problem formulation

Similar to traditional recommendation scenarios, POI recommendation is to predict POIs (or ratings of POIs) that the user may have an interest in, helping users discover POIs they might not have visited otherwise. We define the user set consisting of H users in our system as $U=\{U_1,...U_h,...,U_H\}$ and the POI set including B points as $I=\{I_1,...I_b,...,I_B\}$. Each user has a rating score set for POIs constructing the user-POI rating matrix $R=[r_{i,j}]$. The value $r_{i,j}$ in the matrix R represents the score of POI j rated by user i. Each POI j can be represented by its geographical location as $\{longitude_j, latitude_j\}$.

Our goal is to precisely predict the score of POIs that the user has not visited. Mathematically, the objective of the POI recommendation is to minimize the sum of squared errors (SSE) which is the sum of the squared differences between our predictions R_{pred} and the ground truth R_{real} (i.e., the inherent real values of POI ratings), as shown in Equation (1) as follows:

$$SSE = min\|R_{pred} - R_{real}\|_F^2 \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius Norm of the matrix.

The main notations used in the following discussions of this paper are described in Table 1.

3.2 | Framework

Our POI recommendation approach is to predict the score in the blank of the original matrix according to the existing data. To do so, we construct the geographic information matrix and propose a matrix factorization algorithm. In order to accelerate the algorithm in a distributed learning manner with high performance meanwhile ensuring the privacy of user data, we propose Fed-GRMF as shown in Figure 1.

In our framework, there are multiple user devices, namely, Fed-Clients connecting to a Fed-Server. When the system starts, the Fed-Server initializes various parameters and POI profiles, while Fed-Clients initialize user profiles. The server transfers the latent vectors to Fed-Clients requesting for gradients computation. Each client keeps updating user-POI visiting and rating "raw" data in private and use them to calculate gradient values and then send back to the server. On receiving the "cooked" data from the clients, the Fed-Server aggregates all the data from the clients using a federated-averaging algorithm and updates their POI profiles for global POI recommendation.

TABLE 1 Notations and definitions

Notations	Definitions
R	User-POI rating matrix.
G	User-POI geographic information Matrix.
$r_{i,j}$	Non-negative number in R.
$g_{i,j}$	Non-negative number in G.
K	The number of rating matrix singular value.
P	The user matrix after R SVD.
Q	The POI matrix after R SVD.
X	The user matrix after G SVD.
Y	The POI matrix after G SVD.
U	The set of users.
I	The set of POIs.
b_u	The bias of user.
b_i	The bias of POI.
θ	The average score of R.
$\mathbb{P}_h$	The POI set rated by user h.
g_k	The gradient of SGD.
λ	The parameter of SGD.
Υ	The parameter of SGD.
H	The number of users.
B	The number of POI.
N	The number of recommended POIs.
L	The number of regional divided grids.
$\mathbb{L}$	The set of region after divided.
l_l	The lth grid of $\mathbb{L}$.
SSE	The objective function
v_l^p	The influence of POI p in grid l_l
w_l^u	The possibility of user u show up in grid l_l

Abbreviations: POI, Point-of-Interest; SGD, stochastic gradient descent; SVD, singular value decomposition.

The Fed-Clients interact with the server in parallel, and they are independent of each other. The Fed-Server is always ready to accept users' requests, send, and receive data. For convenience, we assume that the Fed-Server has unlimited capacity and sufficient computing power, and all Fed-Clients interact with the server.

4 | GEOGRAPHICAL MATRIX FACTORIZATION FOR POI RECOMMENDATION

In this section, we combine the location information of POI with rating information to make accurate recommendations for users. Section 4.1 constructs geographic location information matrix, and Section 4.2 proposes the corresponding SVD method after integrating geographic information and rating information.

4.1 | Geographic information matrix

Before modeling the geographic location information matrix, we divide the whole region into L grids as $\mathbb{L} = \{l_1, \ldots l_l, \ldots, l_L\}$, and then we clarify two definitions which are the POIs' influence matrix and the users' activity matrix.

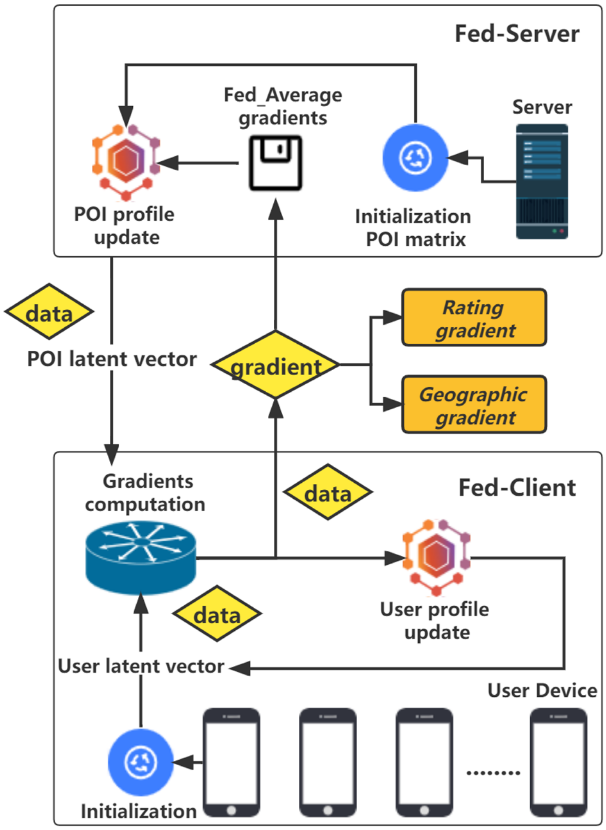

FIGURE 1 Overall framework

The values in POIs' influence matrix refer to the degree of publicity or attraction of the POI in a certain area, while the values in users' activity matrix refer to the degree of interest of the user in a certain area or the possibility of visiting to the area. Detailed definitions are as follows.

Definition 1. (POIs' INFLUENCE MATRIX) We define the POIs' INFLUENCE MATRIX expressed as $V=[v_{b,l}]_{B\times L}$, where b represents the POI I_b in I and l represents the gird l_l in $\mathbb{L}$. The nonnegative value $v_{b,l}$ indicates the influence degree of POI I_b on grid l_l.

Commonly, it can be assumed that the POI influence to a grid is normally distributed,[25] with the maximum value at the geographical center of the POI. Formally, one can quantitatively calculate the value of $v_{b,l}$ by

$$v_{b,l}=\frac{1}{\sigma}K(\frac{d(l,b)}{\sigma}) \tag{2}$$

where $K(\cdot)$ is standard normal distribution and σ is the standard deviation, $d(l,b)$ represents the Euclidean distance between the l_l grid and POI I_b.

Definition 2. (USERs' ACTIVITY MATRIX) We define the USERs' ACTIVITY MATRIX expressed as $W=[w_{h,l}]_{H\times L}$, where h represents the user U_h in U and l represents the gird l_l in $\mathbb{L}$. The nonnegative value $w_{h,l}$ indicates the possibility that the user U_h visits grid l_l.

With Definition 1, one can obtain the value of user activity to a grid. Since a user U_h may have rated multiple POIs expressed as the set $\mathbb{P}_h$, the possibility of such user visiting the grid l_l can be calculated by averaging all the POI influence values within the set of $\mathbb{P}_h$. Therefore, we quantify the value $w_{h,l}$ in users activity matrix as Equation (3) as follows.

$$w_{h,l} = \frac{1}{\sigma|\mathbb{P}_h|}\sum_{i \in \mathbb{P}_h} K(\frac{d(l,i)}{\sigma}) \tag{3}$$

Then the geographic information matrix G can be obtained by inner product of V and W^T, which can be considered as the probability that the user U_h visits POI I_b under geographic conditions only. We have obtained the user rating matrix R and the geographic information matrix G, then the corresponding SVD is proposed to make recommendation in next part.

4.2 | SVD for rating and geographic information matrix

SVD is a method for matrix decomposition. With the rating matrix R, we obtain the geographic information matrix G by calculating the inner product of V and W^T and then we decompose R and G with SVD using the following equations,

$$R' = PQ^T \tag{4}$$

$$G' = XY^T \tag{5}$$

where X and P are both $H \times L$ matrices ans Q and Y are both $B \times L$ matrices. R' represents the rating matrix R after SVD, while G' represents the geographic information matrix G after SVD.

SVD can be used for information retrieval, but currently, it has been widely applied in recommendation systems. For the rating matrix, we can perform low-rank matrix decomposition on it. The user and POI are mapped to a joint potential space K, where $K \ll min(H,B)$, so that the user's preference for POI is modeled as their inner product in the potential space. For the geographic information matrix, the potential space is the divided grid L, where $L \ll min(H,B)$. The mapping is achieved by decomposing the following value in predicted matrix to approximate the real rating matrix and geographic information matrix, as follows:

$$r_{h,b} = \sum_{k=1}^{K} p_{h,k} q_{b,k}^T \tag{6}$$

$$g_{h,b} = \sum_{l=1}^{L} x_{h,l} y_{b,l}^T \tag{7}$$

Here, the matrix $P = [p_{h,k}]$ represents the latent vector matrix of all users in the joint potential space K, and the matrix $Q = [q_{b,k}]$ represents the latent vector matrix of the POIs in rating information. The matrix $X = [x_{h,l}]$ represents the latent vector matrix of all users in the joint potential divided grid L, and the $Y = [y_{b,l}]$ matrix represents the latent vector matrix of the POIs in geographic information. Through this transformation, we can transform the objective function into the following expression as follows:

$$min\|\widetilde{C} - PQ^T - XY^T\|_F^2 \tag{8}$$

where $\|\cdot\|_F$ is the Frobenius Norm of the matrix and $\widetilde{C}$ is the sum of matrix R and G. Geographic information matrix is added to the objective function in this way. It is necessary to clarify that there is still no evidence to prove that the rating information potential space K contains geographic location information; that is, we cannot extract relevant geographic information from the potential space K of the rating matrix. Therefore, we construct the geographic information matrix G to include geographic information in the recommendation factors. The processing of the objective function can ensure that the user's preference for POI includes both interest in potential spatial attributes and the POI location.

We convert the objective function as following regularized least squares minimization as Equation (9), where θ is the average score of the rating matrix and b_u and b_i are the bias values to avoid unbalanced scores from certain users from resulting in unfair recommendation outputs.

$$SSE = min(\widetilde{C} - \sum_{l=1}^{L} X_{i,l} Y_{j,l}^{T} - \sum_{k=1}^{K} P_{i,k} Q_{j,k}^{T}) + \theta + b_u + b_i \tag{9}$$

The gradient descent is used to solve this unconstrained optimization problem. Gradient is a vector, which indicates that a function changes the fastest along the direction of gradient at this point, and the change rate is the largest. The direction of gradient descent refers to the direction of negative gradient. However, gradient descent requires the objective function must be convex.

5 | FEDERATED STOCHASTIC GRADIENT DESCENT (SGD)

In this section, we propose the corresponding SGD method to solve the SVD problem with FL. In Section 5.1, we mathematically prove that our problem is nonconvex which cannot be solved by conventional SVD gradient descent, and thus, we apply SGD. In Section 5.2, we propose SGD matrix factorization algorithm with the framework of FL. The basic procedures are demonstrated by Figure 2.

5.1 | SGD for SVD

We propose **Theorem** 1 to demonstrate that the objective function is nonconvex. For multivariate function $f(x)$, we can judge whether $f(x)$ is a convex function by its Hessian matrix which is a square matrix composed of the second derivative of multivariate function. If Hessian matrix is a positive semidefinite matrix, $f(x)$ is convex function.

Theorem 1. *The SSE minimization problem expressed as Equation (9) is a nonconvex objective optimization problem.*

Proof. There are four variables P, Q, X, and Y in the objective function; $P \cdot Q$ and $X \cdot Y$ are independent of each other. $P \cdot Q$ and $X \cdot Y$ are similar in structure. The constant θ and bias b_u, b_i do not affect the convexness of the objective function. Therefore, in order to facilitate the analysis in the following discussions, we can simplify our proof by proving the nonconvexness of the function SSE' defined as follows:

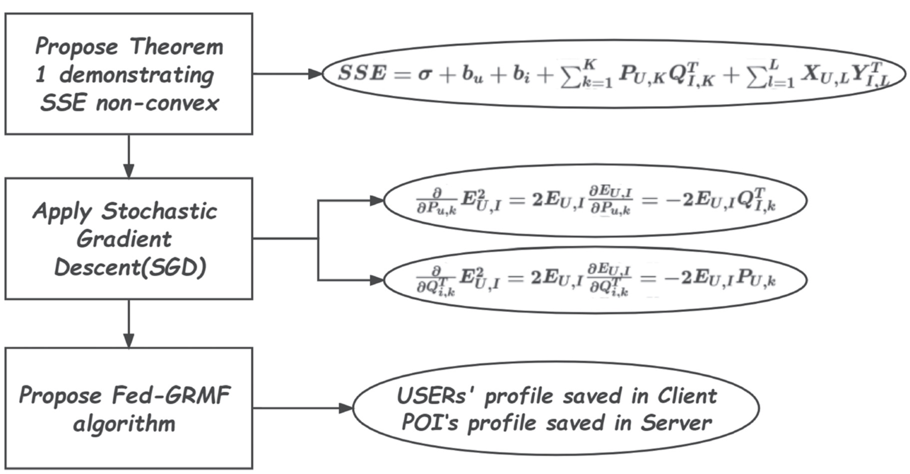

FIGURE 2 Brief flow chart of federated stochastic gradient descent (SGD)

$$SSE' = \min_{(PQ)} \frac{1}{2} (\sum_{(i,j)} R_{i,j} - PQ^T)^2 \tag{10}$$

where R is a matrix and P, Q are two matrices obtained by R decomposition using SVD. For a multivariate function $f(\cdot)$, its Hessian matrix is defined as follows:

$$H_{i,j}(f(\vec{x})) = D_i D_j f(\vec{x}) \tag{11}$$

where D_i represents the differential operator for the i variable.

Here, we let $\vec{x} = (P, Q^T)$. The first derivatives of SSE' are expressed as follows:

$$\frac{\partial SSE'}{\partial P_i} = \frac{\partial}{\partial P_i}[(R_{i,j} - P_i Q_j^T)] = -Q_j^T (R_{i,j} - P_i Q_j^T) \tag{12}$$

$$\frac{\partial SSE'}{\partial Q_j^T} = \frac{\partial}{\partial Q_j^T}[(R_{i,j} - P_i Q_j^T)] = -P_i (R_{i,j} - P_i Q_j^T) \tag{13}$$

The second derivative of SSE' can be calculated as follows.

$$\frac{\partial^2 SSE'}{\partial P_i \partial P_i} = \frac{\partial}{\partial P_i}(-Q_j^T (R_{i,j} - P_i Q_j^T)) = (Q_j^T)^2 \tag{14}$$

$$\frac{\partial^2 SSE'}{\partial P_i \partial Q_j^T} = \frac{\partial}{\partial Q_j^T}(-Q_j^T (R_{i,j} - P_i Q_j^T)) = -R_{i,j} + 2P_i Q_j^T \tag{15}$$

$$\frac{\partial^2 SSE'}{\partial Q_j^T \partial P_i} = \frac{\partial}{\partial P_i}(-P_i (R_{i,j} - P_i Q_j^T)) = -R_{i,j} + 2P_i Q_j^T \tag{16}$$

$$\frac{\partial^2 SSE'}{\partial Q_j^T \partial Q_j^T} = \frac{\partial}{\partial Q_j^T}(-P_i (R_{i,j} - P_i Q_j^T)) = (P_i)^2 \tag{17}$$

Thus, we obtain the Hessian matrix of our objective function as follows.

$$\begin{aligned} H(SSE') &= \begin{bmatrix} \frac{\partial^2 SSE'}{\partial P_i \partial P_i} & \frac{\partial^2 SSE'}{\partial P_i \partial Q_j^T} \\ \frac{\partial^2 SSE'}{\partial Q_j^T \partial P_i} & \frac{\partial^2 SSE'}{\partial Q_j^T \partial Q_j^T} \end{bmatrix} \\ &= \begin{bmatrix} (Q_j^T)^2 & -R_{i,j} + 2P_i Q_j^T \\ -R_{i,j} + 2P_i Q_j^T & (P_i)^2 \end{bmatrix}. \end{aligned} \tag{18}$$

Supposing a nonzero vector $\vec{X} = [x_1, x_2]^T$, we then obtain the deduction as follows.

$$X^T H(SSE')X = [x_1(Q_j^T)^2 + x_2(-R_{i,j} + 2P_iQ_j^T),\\ x_1(-R_{i,j} + 2P_iQ_j^T) + x_2(P_i)^2][x_1, x_2]^T\\ = (x_1Q_j^T + x_2P_i)^2 + 2x_1x_2(P_iQ_j^T - R_{i,j}) \tag{19}$$

If the error between $P_iQ_j^T$ and $R_{i,j}$ is large, it cannot be guaranteed that Equation (19) is always positive so that the Hessian matrix is not a positive semidefinite matrix. Therefore, we can get the result that the objective function SSE' is a nonconvex function.

Theorem 1 can demonstrate that the our objective function SSE is nonconvex. Therefore, simply applying the gradient descent method to our problem is not able to always obtain the optimal solution. Although the possibility of finding the minimum (optimality) is high, there is still possibility of traditional optimization algorithms stopping at the saddle points or maximum points. To attack this challenge, various improved algorithms have been derived based on the gradient descent method, such as dynamic adjustment step size (i.e., learning rate), momentum method, and SGD. We select SGD which is good at solving the optimization problems in summation form like our problem. The idea of SGD is to randomly select one of the calculated gradients as the total gradient.

For the rating matrix, we can get its update gradient as follows.

$$\nabla_{p_i} R(P,Q) = -2\sum_{(i,j)} q_j(r_{i,j} - (p_i, q_j)) + 2\lambda p_i \tag{20}$$

$$\nabla_{q_j} R(P,Q) = -2\sum_{(i,j)} p_i(r_{i,j} - (p_i, q_j)) + 2\lambda q_j \tag{21}$$

Then we update the P and Q matrix as follows:

$$p_i^t = p_i^{t-1} - \Upsilon \nabla_{p_i} R(P^{t-1}, Q^{t-1}) \tag{22}$$

$$q_j^t = q_j^{t-1} - \Upsilon \nabla_{q_j} R(P^{t-1}, Q^{t-1}) \tag{23}$$

where λ and Υ are the parameters of SGD, which are used to adjust the step size. Similarly, we can calculate the gradient and update of geographic information matrix as follows.

$$\nabla_{x_i} G(X,Y) = -2\sum_{(i,j)} y_j(g_{i,j} - (x_i, y_j)) + 2\lambda x_i \tag{24}$$

$$\nabla_{y_j} G(X,Y) = -2\sum_{(i,j)} x_i(g_{i,j} - (x_i, y_j)) + 2\lambda y_j \tag{25}$$

$$x_i^t = x_i^{t-1} - \Upsilon \nabla_{x_i} G(X^{t-1}, Y^{t-1}) \tag{26}$$

$$y_j^t = y_j^{t-1} - \Upsilon \nabla_{y_j} G(X^{t-1}, Y^{t-1}) \tag{27}$$

According to the size of the dataset and operation situation, we initialize various types of parameters. Next, we draw into the framework of FL and put the above process into it.

5.2 | Horizontal FL for SGD

FL is a distributed ML framework model, which allows clients to obtain the ML model on the server through receiving model parameters. The clients send back their own parameters to the server after local training. The server integrates the parameters of each client to update the model. The main feature of FL is to enhance privacy. For each client, its personal data are only used to train its local ML model. Since there is no transmission on the raw data, neither servers nor other clients can obtain the personal local data.

According to the distribution type of data, FL can be divided into horizontal FL, vertical FL, and transfer FL. In our model, different users have different data, but they all learn under the same POI information; therefore, we can apply horizontal federation learning.

We use the federated-averaging (FedAvg) to deal with the gradients from the clients. The FedAvg is a method for the server to process each parameter in horizontal FL. It is suitable for the cumulative error calculation of finite samples $minf(\omega)$. The FedAvg is an efficient algorithm for the distributed training with an enormous number of clients.[33] In FedAvg, clients keep their data locally for privacy protection; a central parameter server is used to communicate between clients. This central server distributes the parameters to each client and collects the updated parameters from clients. In our algorithm, we have H user devices participating in training at the same time, and P_h represents the user h's training sample, namely, the set of POIs scored by the user h, and we have $n_h = |P_h|$. We define our FedAvg as follows:

$$f(\omega) = \sum_{h=1}^{H} \frac{n_h}{h} F_h(\omega) \tag{28}$$

where $F_h(\omega)$ represents the result of training in the user h's device, namely, the returned gradient, shown as Equation (29) where $gradient_h^i$ represents the gradient for the POI i in the user device h.

$$F_h(\omega) = \frac{1}{n_h} \sum_{i \in P_h} gradient_h^i \tag{29}$$

In order to protect user privacy, the tasks of gradient calculation and updating are placed on the user's local machine, and the tasks of POI updating are placed on the federated server. The POI data on the server are accessible to the user, while only the gradient of user's data is transmitted to the server. The specific framework is shown in Figure 3.

Based on the above methods, we propose our Fed-GRMF algorithm as Algorithm 1. The specific algorithm flow chart is shown in Figure 4. The Fed-Server prepares the POI profile and calculates the bias. The Fed-Clients prepare the

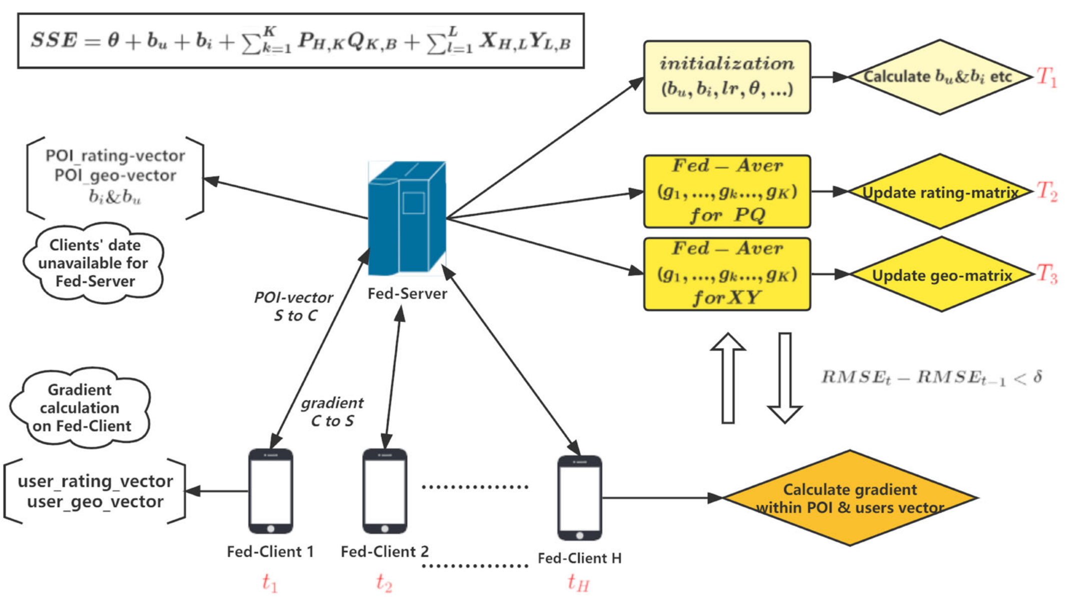

FIGURE 3 Federated learning stochastic gradient descent of matrix factorization

local user profile. The Fed-Clients connect to the Fed-Server and receive the POI profile which is used to calculate the gradient, then update the local user profile and return gradient to Fed-Server. The Fed-Server obtains returned gradients and deal with them by the federal average method, then update the local POI profile. The Fed-Server determines whether the training is over and then get the recommended results.

Algorithm 1 Federated Learning approach of Geographical POI Recommendation using Matrix Factorization (Fed-GRMF)

Require: Server initializes POI rating (geographic) profile matrix $Q\,(Y)$; Clients initialize Users' own rating (geographic) profile vector $P\,(X)$;
Ensure: optimal converged $P, X,\ Q, Y,\ b_u$ and b_i

Initialize $g_0 = 0,\ b_u = 0,\ b_i = 0$ and $K(L), lr, reg, EPOCH$.
Fed-Server keeps latest POI-profile for all clients download
Initialize $q_0\ (y_0)$
repeat
  for each user h's $\in U$ client **in parallel do**
    for POI $b = 1, 2, \dots, B$
      for each $k(l)$ in $K(L)$:
        $q^b_{t+1} = q^b_t$ - $\Upsilon \cdot$ *ClientUpdate* $(h, q^b_t, k(l))$
  $Q_{t+1} = Federated\_Average\ (q^1_{t+1}, q^2_{t+1}, \dots, q^B_{t+1})$
until $RMSE_{t+1} - RMSE_t < \Theta$
Fed-Client calculate the gradient and return to the server
ClientUpdate $(h, q^b_t, k(l))$ run on client h
  Compute gradient: $g_h = \nabla R(p^h_t, q^b_t, k(l))$;
  Local update: $p^h_{t+1} = p^h_t - \Upsilon \nabla R(P_t, Q_t, k(l))$;
Return g_h to Fed-Server

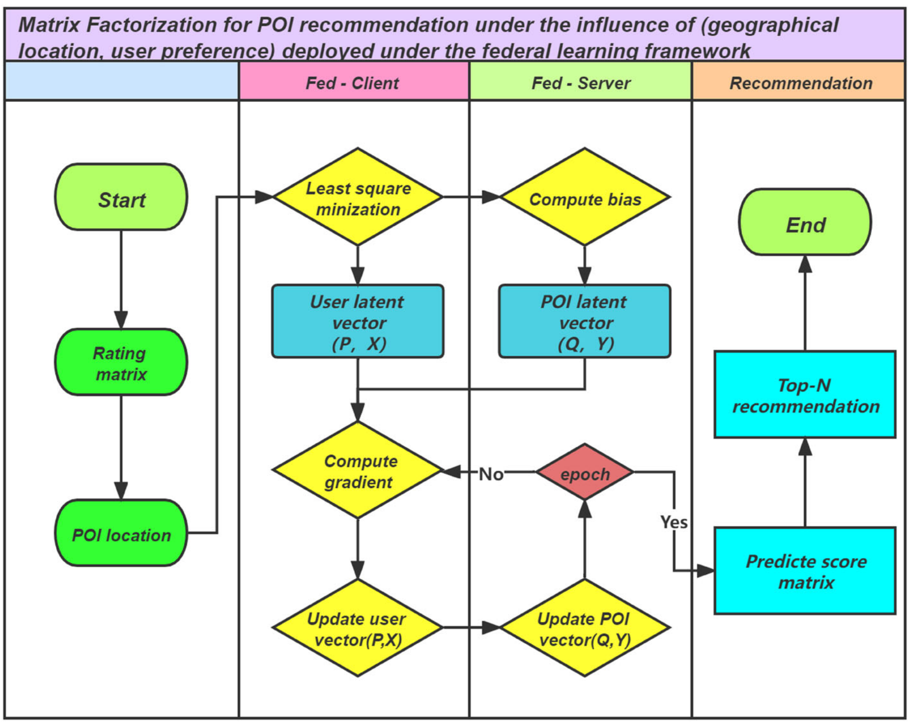

FIGURE 4 Flow chart of Federated Learning approach of Geographical POI Recommendation using Matrix Factorization (Fed-GRMF)

We analyze the time complexity of our algorithm. In Figure 3, we can see that our algorithm is distributed. We update the rating matrix and geographic information matrix in sequence. The complexity is determined by the singular value $K(L)$ and the scale of POI dataset. Therefore, the time complexity of our algorithm is $O(EB(K+L))$, where E is the number of iterations epoch, B represents the number of POI, and $(K+L)$ represents the sum of singular values of rating matrix and geographic information matrix.

6 | EXPERIMENT AND EVALUATION

6.1 | Dataset and experimental setup

We adopt the real-world Yelp dataset[34] to carry out the experiments. Yelp is the largest comment website in the United States. The dataset covers merchants, comments, and user data, including 8,635,403 user comments, more than 1,605.85 million pieces of merchant information and 200,000 pictures. We select the dataset in Henderson City, Clark County, southeast Nevada, USA, including 13,124 merchants and 2435 users, with a total of 1,342,627 rating data. The dataset was collected within the period of nearly 7 years, from February 2011 to March 2017. We construct a user-POI rating matrix with a density of 0.0420. After that, refer to the similar existing work on POI recommendation,[25] we filter merchants with score data less than 10 and users with score data less than 5. Finally, 3788 merchants, 543 users, and 126,752 scores are reserved. The matrix density is 0.0616, and each user gives an average score of 200 merchants.

For each user, we randomly select 20% of their scores as the testing data and 80% as the training data to train our model. After obtaining the trained model, we score the nonscored POI (not in the training data) of each user, rank the predicted scores in descending order, and then compare with the real data (testing data). We use two metrics widely used in the recommendation system to measure the model, namely, *Recall*@*N* and *Precision*@*N*, where N represents the number of recommended POIs. We denote $\mathbb{H}_i(N)$ as the top-N POIs recommended by the model for user i, and $\mathbb{F}_i$ represents the ranking of real POI data scored by user i in the testing data. We can get the formal expression of the above two metrics as follows:

$$Recall@N = \frac{1}{H}\sum_{i=1}^{H}\frac{|\mathbb{H}_i(N)\cap\mathbb{F}_i|}{|\mathbb{F}_i|} \tag{30}$$

$$Precision@N = \frac{1}{H}\sum_{i=1}^{H}\frac{|\mathbb{H}_i(N)\cap\mathbb{F}_i|}{N}. \tag{31}$$

In addition to the above two metrics, we also add the running time as a metric to evaluate our Fed-GRMF approach. The symbolic representation of the running time of each part is shown in Figure 3. The t_1 to t_H represent the transmission delay and gradient calculation delay at each federated client; T_1 represents the time of parameters initialization at Fed-Server, T_2 represents the time of updating for rating matrix, and T_3 represents the time of updating for geographic information matrix. After obtaining these time data, we evaluate our algorithm. We count the running time as $T=T_1+T_2+T_3$. Since our FL approach can be implemented in a parallel way, we have $T_2^f = Max_r(t_1, t_2, ..., t_H)$ and $T_3^f = Max_g(t_1, t_2, ..., t_H)$.

We use the convergence of root mean squared error (RMSE) as the criterion for stopping the iterations. The RMSE is the square root of the ratio of the squared of the deviation between the real value and the predicted value to the number of observations n, and its expression is shown by Equation (32), where n is the number of observations, y_i is the real value, and y_i' is the predicted value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2} \tag{32}$$

6.2 | Comparison experimental results

In this part, we carry out comparison experiments to evaluate the performance of our Fed-GRMF algorithm. We adopt the following two well-known schemes for comparison.

- *Sequential execution without FL (Non-Fed)*: As the same as Fed-GRMF, we construct geographic information and integrate it with rating matrix. Then for each user, the gradient calculation and profile update are executed sequentially.
- *User-based Collaborative Filtering (UCF)*: For each user, we apply Jaccard formula to calculate user similarity and then obtain its corresponding recommendations.

Table 2 shows the RMSE under different iterative epochs. We conduct the experiments with the number of iterations epoch from 10 to 100. We can find that the RMSE of Fed-GRMF and Non-Fed is close, which indicates that

TABLE 2 RMSE of each epoch

#epoch	Fed-GRMF	Non-Fed	Convergence
10	0.981275	0.9642313	No
20	0.864632	0.8675679	No
30	0.773482	0.7435367	No
40	0.631645	0.6292423	Yes
50	0.621233	0.6142455	Yes
60	0.625904	0.6231421	Yes
70	0.631395	0.6242313	Yes
80	0.604015	0.6143678	Yes
90	0.631923	0.6198852	Yes
100	0.626457	0.6067811	Yes

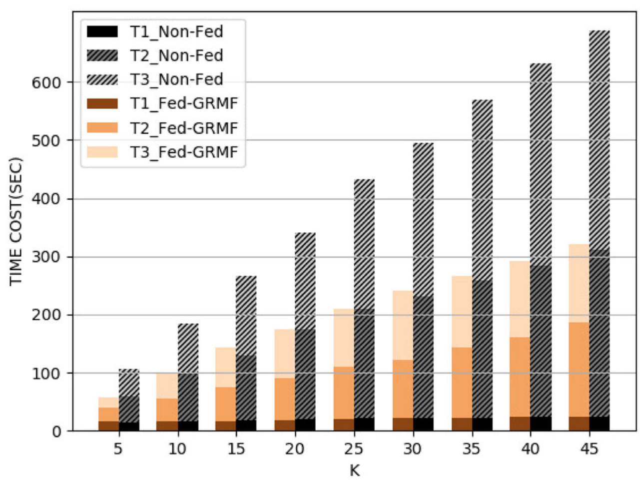

FIGURE 5 Time consumption comparison with K increasing

introducing distributed FL framework results in no performance degradation in recommendation. We can see that RMSE has converged when epoch is 40. In order to balance the effectiveness (low RMSE) and efficiency (small running time), we select 50 as the number of iterations in our experiments for the following discussions.

We tune the parameter K from 5 to 45 to investigate the running time of Fed-GRMF and Non-Fed, shown as Figure 5. We adjust the number of grid L as the same as K to compare the time consumption of updating the rating matrix and updating the geographic information matrix. The matrix factorization in our paper has a total running time of three parts. We time the three parts, respectively. It can be found that the running time of Fed-GRMF is much shorter than the Non-Fed. The T_3 part of geographic information matrix processing is the most time-consuming. At the same time, with the increase of singular value K, T_3 increases significantly faster than that in the other two, followed by the T_2, and finally T_1 can be regarded negligible. It is because the geographic information matrix in T_3 part has more corresponding amount of information than the rating matrix in T_2 part. The parameter initialization and preprocessing are performed in T_1 part; therefore, the time consumption is short and the growth is not obvious.

In Figure 6, we adjust the size of the original dataset from 10% to 100% to study the algorithm running time relationship between Fed-GRMF and Non-Fed when the amount of data increases. At the same time, we take the running time of UCF for comparison. The horizontal axis in the figure is the size of the dataset, and the logarithmic vertical axis is the running time. It can be seen that Fed-GRMF has higher computational performance than UCF and Non-Fed. Fed-GRMF is a distributed structure whose time complexity is $O(EB(K+L))$, while the time complexity of Non-Fed is $O(EB(K+L)n)$, where n is the number of users participating. UCF needs to calculate the similarity between users. Its

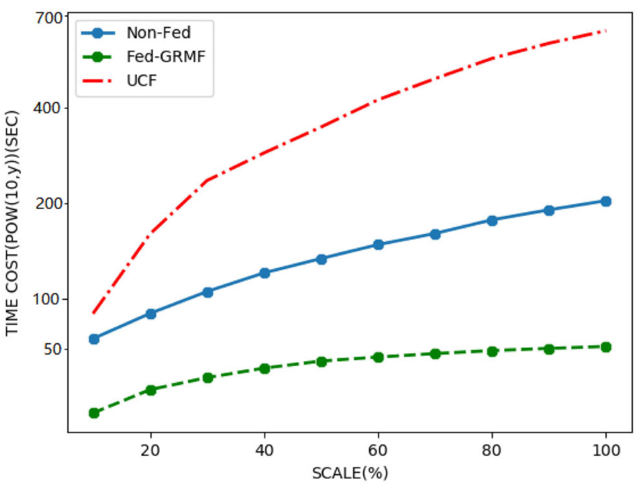

FIGURE 6 Time consumption comparison with scale of dataset increasing

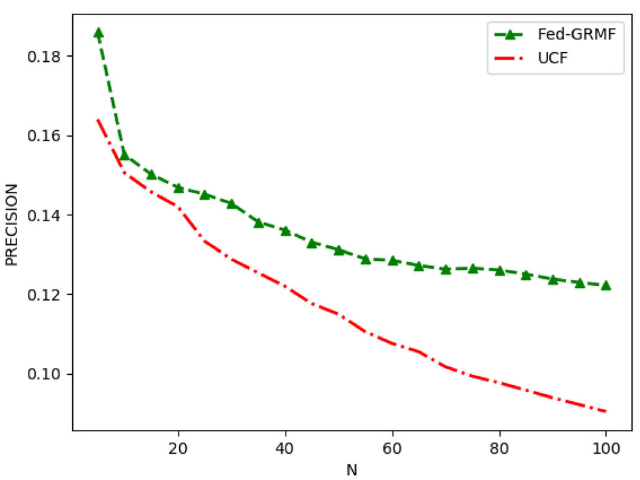

FIGURE 7 Precision comparison

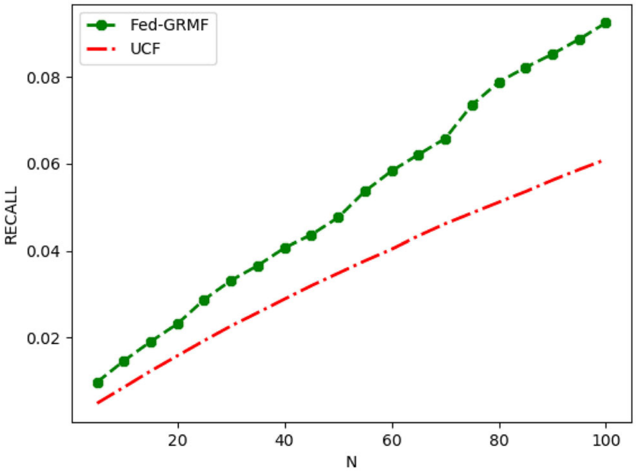

FIGURE 8 Recall comparison

complexity is $O(n^2)$, resulting in the longest running time. The experimental results correspond to our theoretical analysis in the above discussions.

In Figures 7 and 8, we compare the precision and recall between UCF and Fed-RGMF. For Fed-GRMF, we set the parameter N from 5 to 100. After 100 rounds of training, we average the 100 recalls and precisions, for both Fed-GRMF UCF. Figure 7 shows that, as N increases, the precision decreases, similar to the common phenomenon in most recommendation systems. The precision of Fed-GRMF is always higher than UCF, and the downward trend is slower than UCF. Figure 8 shows that as N increases, the recall increases. The recall of Fed-GRMF is always higher than UCF, and the growth trend is faster than UCF. According to these two figures, we can see that Fed-GRMF is better than UCF in recommendation performance.

7 | CONCLUSIONS AND FUTURE WORK

In this paper, we study the POI recommendation in IoT, aiming at guaranteeing user privacy in communication within high-performance recommendation systems. We propose an FL approach of geographic POI recommendation with matrix factorization, namely, Fed-GRMF. We quantify the geographic information to construct a matrix then integrate it with the rating matrix to construct the objective function for optimization. We decompose the objective matrix by SVD and apply corresponding SGD method under FL to solve this regulated least squares minimization problem. Real-world dataset is adopted for evaluation, whose experimental results show that our Fed-GRMF approach dominates existing approaches in recommendation and computational performance.

One of the promising directions for our future work is to investigate multiserver collaboration on the server site of our approach. The schemes of data sharing and synchronization should be carefully designed, and the time consumption should also be fully considered in large-scale POI recommendation systems. We may also introduce edge computing into our framework, and cloud-edge collaboration of FL for POI recommendation could be considered. Meanwhile, some open problems such as task scheduling[35] and resource allocation[36] should be studied in the scenarios of recommendation systems with FL framework.

ACKNOWLEDGMENTS

This work is supported by Beijing Natural Science Foundation (no. 4202066), Beijing Nova Program (no. Z201100006820082), and the National Natural Science Foundation of China (no. 61972414).

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in Yelp Dataset at https://www.kaggle.com/yelp-dataset/yelp-dataset, reference number 34.

ORCID

Jiwei Huang https://orcid.org/0000-0001-5220-6703

REFERENCES

1. Lueth K State of the IoT 2018: Number of IoT devices now at 7b-market accelerating. [Online] https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/; 2019.
2. Yao Z, Fu Y, Liu B, Liu Y & Xiong H Poi recommendation: A temporal matching between poi popularity and user regularity. In: 2016 IEEE 16th international conference on data mining (ICDM), ICDM'16 IEEE; 2016:549-558.
3. Chen Z, Ning Z, Xiong Q, Obaidat MS. A collaborative filtering recommendation-based scheme for WLANS with differentiated access service. *IEEE Syst J*. 2018;12(1):1004-1014.
4. Ye M, Yin P, Lee W-C, Lee D-L. Exploiting geographical influence for collaborative point-of-interest recommendation. In: SIGIR'11 ACM; 2011; Beijing, China: 325-334. doi:10.1145/2009916.2009962
5. Ahmadova U, Mustafayev M, Kiani Kalejahi B, Saeedvand S, Rahmani AM. Big data applications on the internet of things: A systematic literature review. *Int J Commun Syst*. 2021;34(18):e5004.
6. Liu H, Lu J, Yang H, et al. Category-specific CNN for visual-aware CTR prediction at jd. com. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, SIGKDD'20ACM; 2020:2686-2696.
7. Chen T, Wong RC-W. Handling information loss of graph neural networks for session-based recommendation. In: SIGKDD'20 ACM; 2020; Virtual, Online, United States:1172-1180. doi:10.1145/3394486.3403170
8. Raja K, Pushpa S. Novelty-driven recommendation by using integrated matrix factorization and temporal-aware clustering optimization. *Int J Commun Syst*. 2020;33(13):e3851.
9. Park JH, Obaidat MS, Chao HC. User experience and quality-of-service in internet of things: Practice and trends. *Wirel Pers Commun*. 2016;91(4):1549-1554.
10. Talat R, Obaidat MS, Muzammal M, Sodhro AH, Luo Z, Pirbhulal S. A decentralised approach to privacy preserving trajectory mining. *Future Gener Comput Syst*. 2020;102:382-392.
11. Gao H, Liu C, Yin Y, Xu Y, Li Y. A hybrid approach to trust node assessment and management for VANETs cooperative data communication: Historical interaction perspective. *IEEE Trans Intell Transp Syst*. 2021;2021:1-10.
12. Gao H, Zhang Y, Miao H, Barroso RJD, Yang X. Sdtioa: Modeling the timed privacy requirements of IoT service composition: A user interaction perspective for automatic transformation from BPEL to timed automata. *Mobile Netw Appl*. 2021;2021:1-26.
13. Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: Concept and applications. *ACM Trans Intell Syst Technol*. 2019;10(2):1-19.
14. Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer*. 2009;42(8):30-37.
15. Berlioz A, Friedman A, Kaafar MA, Boreli R, Berkovsky S. Applying differential privacy to matrix factorization. In: RecSys'15 ACM; 2015; Vienna, Austria:107-114.
16. Kim S, Kim J, Koo D, Kim Y, Yoon H, Shin J. Efficient privacy-preserving matrix factorization via fully homomorphic encryption. In: ASIA CCS 2016 ACM; 2016; Xi'an, China: 617-628.
17. Yuan Q, Cong G, Ma Z, Sun A, Thalmann NM. Time-aware point-of-interest recommendation. In: Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval, SIGIR'13 ACM; 2013; New York, NY, USA: 363-372. doi:10.1145/2484028.2484030
18. Park M-H, Hong J-H, Cho S-B. Location-based recommendation system using Bayesian user's preference model in mobile devices. In: UIC'07, Vol. 4611 LNCS IEEE; 2007; Hong Kong: 1130-1139.
19. Horozov T, Narasimhan N, Vasudevan V. Using location for personalized poi recommendations in mobile environments. In: International symposium on applications and the internet (saint'06), SAINT'06 IEEE; 2006:124-129.
20. Zheng Y, Zhang L, Ma Z, Xie X, Ma W-Y. Recommending friends and locations based on individual location history. *ACM Trans Web*. 2011;5(1):1-44.
21. Zheng VW, Zheng Y, Xie X, Yang Q. Collaborative location and activity recommendations with GPS history data. In: Proceedings of the 19th international conference on world wide web, WWW'10 ACM; 2010:1029-1038.
22. Liu Q, Ge Y, Li Z, Chen E, Xiong H. Personalized travel package recommendation. In: 2011 ieee 11th international conference on data mining, ICDM'11 IEEE; 2011:407-416.
23. Yin Y, Huang Q, Gao H, Xu Y. Personalized APIS recommendation with cognitive knowledge mining for industrial systems. *IEEE Trans Ind Inf*. 2020;17(9):6153-6161.
24. Sattari M, Manguoglu M, Toroslu IH, Symeonidis P, Senkul P, Manolopoulos Y. Geo-activity recommendations by using improved feature combination. In: Proceedings of the 2012 ACM conference on ubiquitous computing, UbiComp'12 ACM; 2012; New York, NY, USA: 996-1003. doi:10.1145/2370216.2370432
25. Lian D, Zhao C, Xie X, Sun G, Chen E, Rui Y. GEOMF: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD'14 ACM; 2014; New York, NY, USA: 831-840. doi:10.1145/2623330.2623638
26. Xu Y, Li Y, Yang W, Zhang J. A multi-factor influencing POI recommendation model based on matrix factorization. In: 2018 tenth international conference on advanced computational intelligence (icaci), ICACI'18IEEE; 2018:514-519.
27. Konečnỳ J, McMahan HB, Yu FX, Richtárik P, Suresh AT, Bacon D. Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:161005492; 2016.

28. Brendan McMahan H, Moore E, Ramage D, Hampson S, Aguera y Arcas B. Communication-efficient learning of deep networks from decentralized data. In: AISTATS'17 PMLR; 2017; Fort Lauderdale, FL, United states.
29. Hard A, Rao K, Mathews R, Ramaswamy S, Beaufays F, Augenstein S, Eichner H, Kiddon C, Ramage D. Federated learning for mobile keyboard prediction. arXiv preprint arXiv:181103604; 2018.
30. Ammad-Ud-Din M, Ivannikova E, Khan SA, Oyomno W, Fu Q, Tan KE, Flanagan A. Federated collaborative filtering for privacy-preserving personalized recommendation system. arXiv preprint arXiv:190109888; 2019.
31. Chai D, Wang L, Chen K, Yang Q. Secure federated matrix factorization. *IEEE Intell Syst*. 2021;36(5):11-20.
32. Wu C, Wu F, Cao Y, Huang Y, Xie X. FEDGNN: Federated graph neural network for privacy-preserving recommendation. arXiv e-prints, arXiv:2102.04925; 2021.
33. Deng Y, Kamani MM, Mahdavi M. Distributionally robust federated averaging. In: Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Advances in Neural Information Processing Systems*, Vol. 33: Curran Associates, Inc.; 2020:15111-15122. https://proceedings.neurips.cc/paper/2020/file/ac450d10e166657ec8f93a1b65ca1b14-Paper.pdf
34. Inc Y. Yelp dataset. [Online] https://wwwkagglecom/yelp-dataset/yelp-dataset; 2017.
35. Ma X, Xu H, Gao H, Bian M. Real-time multiple-workflow scheduling in cloud environments. *IEEE Trans Netw Serv Manag*. 2021;18(4):4002-4018.
36. Huang Y, Xu H, Gao H, Ma X, Hussain W. SSUR: An approach to optimizing virtual machine allocation strategy based on user requirements for cloud data center. *IEEE Trans Green Commun Netw*. 2021;5(2):670-681.

How to cite this article: Huang J, Tong Z, Feng Z. Geographical POI recommendation for Internet of Things: A federated learning approach using matrix factorization. *Int J Commun Syst*. 2022;e5161. doi:10.1002/dac.5161