Journal of Artificial Intelligence Research 76 (2023) 1019-1041 Submitted 11/2022; published 04/2023

Decentralized Gradient-Quantization Based Matrix Factorization for Fast Privacy-Preserving Point-of-Interest Recommendation

Xuebin Zhou ZHOUXUEBINSCUT@GMAIL.COM
South China University of Technology
Guangzhou, Guangdong, 510000, China

Zhibin Hu HUZHIBINSCUT@GMAIL.COM
South China Normal University
Guangzhou, Guangdong, 510000, China

Jin Huang HUANGJIN@M.SCNU.EDU.CN
(Corresponding Author)
South China Normal University
Guangzhou, Guangdong, 510000, China

Jian Chen ELLACHEN@SCUT.EDU.CN
(Corresponding Author)
South China University of Technology
Guangzhou, Guangdong, 510000, China

Abstract

With the rapidly growing of location-based social networks, point-of-interest (POI) recommendation has been attracting tremendous attentions. Previous works for POI recommendation usually use matrix factorization (MF)-based methods, which achieve promising performance. However, existing MF-based methods suffer from two critical limitations: (1) Privacy issues: all users' sensitive data are collected to the centralized server which may leak on either the server side or during transmission. (2) Poor resource utilization and training efficiency: training on centralized server with potentially huge low-rank matrices is computational inefficient. In this paper, we propose a novel decentralized gradient-quantization based matrix factorization (DGMF) framework to address the above limitations in POI recommendation. Compared with the centralized MF methods which store all sensitive data and low-rank matrices during model training, DGMF treats each user's device (e.g., phone) as an independent learner and keeps the sensitive data on each user's end. Furthermore, a privacy-preserving and communication-efficient mechanism with gradient-quantization technique is presented to train the proposed model, which aims to handle the privacy problem and reduces the communication cost in the decentralized setting. Theoretical guarantees of the proposed algorithm and experimental studies on real-world datasets demonstrate the effectiveness of the proposed algorithm.

1. Introduction

Location-based social networks such as *Foursquare* and *Facebook Places* have gained more and more popular due to explosive increase of smart terminals (e.g., mobile phones and pads) in recent decades (Yin et al., 2015, 2013, 2015). Like most of the recommender systems, point-of-interest (POI) recommendation has attracted many e-commerce companies' atten-

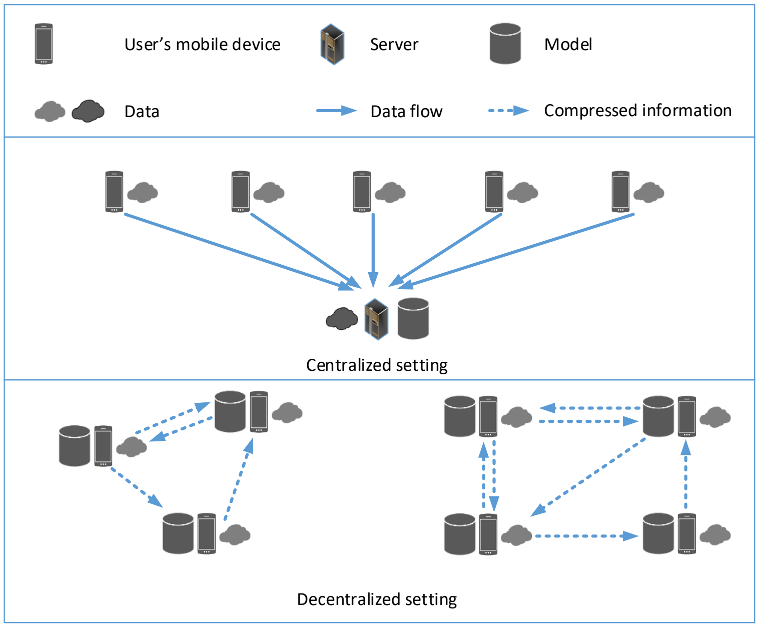


Figure 1: Comparison with centralized setting and decentralized setting. Under the centralized setting, sensitive data stored in user's device is collected to server to train the recommendation model. In contrast, under the decentralized setting, models are trained and stored in user's device separately using his/her own data. The training process does not expose user's sensitive data and only compressed updates that carry very limited information are sent to other users to train the global model collaboratively.

tion for improving user experience and business profit by exploiting location information (Yin et al., 2016; Feng et al., 2015; Scellato et al., 2011). Concretely, POI recommendation alleviates information overload problem in the way that assisting users with better decision making by modeling users' visiting preferences and recommending new POIs (e.g., hotels, restaurants and stores) for them.

As shown in Figure 1, most of the existing methods use collaborative filtering techniques to handle the scenario of POI recommendation, especially matrix factorization (MF) (Koren et al., 2009; Cheng et al., 2011, 2012). Although MF-based methods have achieved promising performance in POI recommendation, such *centralized* training mechanism becomes problematic due to the privacy risks. For example, the recently proposed General Data Protection Regulation (GDPR) from European Union restricts the collection, storage and use of personal data that companies must obey. The regulation only allows companies to collect a minimum of data for specific purpose, which makes it harder to obtain users' data required by the centralized MF methods. Additionally, centralized MF methods also suffer from data leakage risk, which may happened on the server side, or during the data transmission process (Chen, Liu, Zhao, Zhou, & Li, 2018).

One way to address the privacy issues brought by centralized MF methods is to make it *decentralized*. Unlike *distributed* methods that collect data centrally and perform training collaboratively on a cluster of machines using distributed computing frameworks (Blot et al., 2016; Konečný et al., 2017), decentralized MF methods protect users' private data without being collected to centralized server and only transmit necessary data [e.g., model weights (Tang, Liang, Yan, Zhang, & Liu, 2018) or gradients (Duriakova, Tragos, Smyth, Hurley, Peña, Symeonidis, Geraci, & Lawlor, 2019)] to train the model. The decentralized MF methods, which treat each user's own device as an individual learner, not only resolve the aforementioned privacy issues, but also improve resource utilization and training efficiency in the way that model-training and weight-saving are performed individually and parallelly on each user's own device. Therefore, collaboration between different individual devices is of utmost importance since all individual learners must share the unique data (e.g., raw ratings, model weights, or gradients) that they hold, where the communication mechanism plays an important role. However, there are still many challenges to implement such decentralized algorithm, mainly for the following two critical considerations: (1) communication overhead, (2) information leakage during communication. What type of data should be transferred to whom, and how to minimize the message size and information leakage must be carefully considered while designing such communication mechanism.

To solve the above challenges, we propose a novel Decentralized Gradient-quantization based Matrix Factorization (DGMF) framework for privacy-preserving mobile POI recommendation. In order to train the proposed DGMF model efficiently, here we adopt a privacy-preserving learning method. We first decompose item-related latent vectors into two parts: shared (global) and specific (local) latent vectors, and then apply *location-based communication technique* to exchange quantized gradients among geographical neighbors. Except for making use of the basic abilities of decentralized method like decentralized storage and computation, other major advantages of our proposed framework can be summarized as follows:

DGMF is privacy-aware. First, the privacy data (e.g., user's check-in history and ratings) is processed on each device without being collected or exposed. Second, to learn the model collaboratively, it is essential to exchange data between each user. To this end, in contrast to exchanging the gradients directly, we exchange *quantized gradients* by leveraging a gradient quantization technique using stochastic rounding (Uryasev & Pardalos, 2006; Hubara et al., 2016) to further compress gradient information since the gradients are still linear with respect to the ratings and the rating can be reconstructed after grabbing enough samples (Baraniuk, 2007). By contrast, the quantized gradients keep the statistical trending of the original gradients but carry less information, which make it harder to recover and further enhance the privacy. Third, inspired by Bayesian Personalized Ranking (Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, 2009), we adopt the indirect pairwise preference between two different POIs as the optimization objective. Even the leakage does happened during the gradient exchange process, the gradient itself only reflects the trending for a specific preference, which is empirically better than exposing the direct rating that many decentralized MF methods have used (Chen et al., 2018; Duriakova et al., 2019). Fourth, unlike many rating-oriented communication schemes that expose user's rating unintentionally, our communication scheme depends only on the cities where user located or selected. With the above design, DGMF has the ability to protect user's privacy. It resolves the challenge about what type of data should be transferred to whom, and minimizes the information leakage during communication.

DGMF is communication-efficient. Note that the number of users is very large in practical POI recommendation scenarios, applying the aforementioned location-based user communication technique directly is still challenging due to the real-time communication bottleneck. In order to reduce the communication cost, we take the advantages of quantized gradient by replacing real-valued gradient to the quantized one to tackle this problem, as many distributed quantization-based optimization frameworks do (Alistarh, Grubic, Li, Tomioka, & Vojnovic, 2017; Wu, Huang, Huang, & Zhang, 2018). The discrete gradient requires less bits to transfer than 32 bit floating points, which can reduce the size of communication payload to a large extent. Besides, we also restrict the maximum number and the distance of neighbors to be communicated with to have a better tradeoff between overall recommendation accuracy and communication efficiency.

Our proposed method successfully solves the limitations of centralized or decentralized MF based methods. (1) The computation and storage are completely decentralized, which reduces training cost and improves efficiency. The sensitive data (e.g., ratings, latent vectors) is kept locally without any exposure, which ensures user privacy. (2) To train our proposed model collaboratively, we make use of geographical information to exchange updates to users in the same city. (3) We leverage user's pair-wise preferences instead of ratings as model objective, which empirically makes it more difficult to restore the original user data. Quantized gradients is used to significantly reduce the communication cost and further enhance the privacy. We summarize our main contributions as follows:

- We propose a novel decentralized gradient-quantization based matrix factorization (DGMF) method for mobile POI recommendation. To handle both privacy and efficiency properly, we propose a novel decentralized training method to train our proposed method.
- We make use of a location-based communication scheme that only depends on the city where user located or selected. We further extend it by leveraging a gradient-quantization based communication mechanism to reduce communication cost during training. We also derive theoretical proofs of the variance quantization bound of the proposed mechanism.
- Experimental results conducted on real-world POI datasets demonstrate the effectiveness of the proposed model and its theory.

2. Related Works

In this section, we review some necessary backgrounds of our work, i.e., (1) MF models in POI recommendation, (2) privacy-preserving techniques in POI recommendation.

2.1 MF Models in POI Recommendation

Centralized matrix factorization has been extensively applied to POI recommendation due to its effectiveness and scalability. Previous attempts focus on improving accuracy of POI recommendation by modeling side information (a.k.a. content-aware MF methods) such as user social networks, time and space that interactions are made. FMFMGM (Cheng et al., 2012) combines multi-center gaussian model and MF to capture the geographical influence of users' behavior. GT-BNMF (Liu, Fu, Yao, & Xiong, 2013) uses geographical probabilistic latent factor model to exploit user mobility patterns and further improve performance. FPMC-LR (Cheng, Yang, Lyu, & King, 2013) introduces temporal information into MF to address successive personalized POI recommendation. GeoMF (Lian, Zhao, Xie, Sun, Chen, & Rui, 2014) utilizes geographical information to generate latent vectors. CAPRF (Gao, Tang, Hu, & Liu, 2015) makes use of content information to explain user behavior and improve recommendation performance.

Nevertheless, the above methods are trained by centralized mechanism, which suffer from low resource utilization and poor training efficiency for the reason that the growth of hardware cannot keep up with the growth of data. Therefore, these traditional centralized MF methods are no longer suitable for practical recommendation. At this point, the *distributed* and *decentralized* learning frameworks are proposed to tackle the training efficiency issue and has been widely and successfully applied in many tasks in recent years such as web mining (Lai, Liu, Lo, Kao, & Yiu, 2018), hash function learning (Spring & Shrivastava, 2017), and deep learning (McMahan, Moore, Ramage, Hampson, & y Arcas, 2017; Blot et al., 2016).

Distributed MF methods mainly focus on accelerating computation with the use of divide-and-conquer algorithms that splitting ratings or user/item latent matrices into several small sub-matrices to exploit the parallel computation ability (Mackey, Talwalkar, & Jordan, 2015; Zhu, Li, Yang, Tang, & Wakin, 2019). However, ratings and model parameters of such methods are still stored in a centralized way. Aiming to further address this shortage, later on, decentralized MF methods are developed that ratings and parameters can also be distributed stored without being centrally collected as well as the computation can be done separately using decentralized stochastic gradient descent (SGD) frameworks like Federate learning (FL) (Konečný et al., 2017) and Gossip learning (Blot et al., 2016).The decentralized SGD frameworks compute gradient locally using the data stored in each node and exchange the gradient to other nodes to collaboratively train the model, which are the core module of existing decentralized MF methods (Hegedundefineds, Berta, Kocsis, Benczúr, & Jelasity, 2016; Hegedűs, Danner, & Jelasity, 2020; Zhu et al., 2019). Some recent researches extend FL-based recommendation model to leverage the geographical information (Huang, Tong, & Feng, 2022).

Although the above decentralized SGD methods are efficient, the communication cost becomes another bottleneck that limits the performance because of the complex communication pattern between nodes. To this end, two types of SGD variants are proposed to reduce communication cost: quantization-based SGD and sparsification-based SGD. The first type of SGD like Quantized SGD (Alistarh et al., 2017) and HSQ (Dai, Yan, Zhou, Yang, Ng, Cheng, & Fan, 2019) has illustrated its dramatic power to lower the communication cost and make the training even faster by replacing the exchange of real-valued gradients to quantized gradients. The second type is to make gradient sparse by pruning away small gradients like DGC (Lin, Han, Mao, Wang, & Dally, 2018) and TernGrad (Wen, Xu, Yan, Wu, Wang, Chen, & Li, 2017). However, these methods are rarely investigated in the field of recommender systems.

2.2 Privacy-Preserving Techniques in POI Recommendation

Limited by data protection regulations and public awareness on privacy issues, today, privacy-unaware methods are gradually replaced by many privacy-aware methods. There are many kinds of works to explicitly address the privacy issue, including cryptography techniques, noise perturbation techniques and data-sharing limitation. Cryptography based techniques such as new secure aggregation protocol for federated learning (Bonawitz, Ivanov, Kreuter, Marcedone, McMahan, Patel, Ramage, Segal, & Seth, 2016) and gossip learning (Danner, Berta, Hegedűs, & Jelasity, 2018) exchange encrypted gradients during communication to preserve privacy. Noise perturbation techniques like differential privacy (Dwork, 2008) preserve user privacy by adding noise when exchanging gradients (Meng, Wang, Shu, Li, Chen, Liu, & Zhang, 2018; Agarwal, Suresh, Yu, Kumar, & McMahan, 2018). Data share limitation approaches keep a portion of interactions as sensitive data and the rest are treated as non-sensitive data (Meng et al., 2018; Duriakova et al., 2019). The non-sensitive data can be shared with other learners when sensitive data is invisible for others. PREFER combines FL with edge learning that changes centralized server to multiple edge servers to enhance the privacy protection (Guo, Liu, Cai, Zeng, Chen, Zhou, & Xiao, 2021). DCLR introduces a two-stage training method that trains a global model using public POI data first, then distributes it to users and uses users' own data to train the final model (Long, Chen, Hung, & Yin, 2022). The most relevant existing work DMF proposes a decentralized matrix factorization framework (Chen et al., 2018) using stochastic gradient descent inspired by SVD-based decentralized matrix completion (Yun, Yu, Hsieh, Vishwanathan, & Dhillon, 2014). It explicitly addressed the privacy problem by imposing the nearby communication mechanism for gradient exchanging.

However, there are some major differences between our proposed method and these decentralized methods, which are summarized as follows: (1) Although decentralized MF claims to address privacy issues, the uncompressed gradient exchanging is still lack of protection and the original ratings can be approximately reconstructed (Baraniuk, 2007). In this paper, except for applying gradient quantization to eliminate the linearity, we further adopt a pair-wise objective to hide the ratings, i.e., the malicious user can only obtain the preference relation between two items even if the leakage is still happened instead of rating value itself. We further introduce a gradient-quantization method to reduce communication cost as well as the carried information during exchanging. (2) Most of the existing communication schemes choose user's neighbors according to the user-item rating matrix or item-based similarity, which may expose rating data unintentionally. In this paper, we introduce a location-based scheme to exchange gradients, inspired by the experience that users are more likely to interact with nearby POIs. Focusing on these limitations, we propose a decentralized gradient-quantization based algorithm aiming to jointly improve both privacy and efficiency for mobile POI recommendation.

3. The Proposed Model

In this section, we first describe the preliminary knowledge and then introduce the model formulation and optimization. Next, we introduce a location-based user communication scheme and propose an efficient gradient exchange method. Finally, we analyze the model complexity.

3.1 Preliminary

Let $\mathcal{U}$ and $\mathcal{I}$ be the user and item (i.e., POI) set. Let U and I be the number of users and items, respectively. For centralized MF method, it learns a user latent matrix $W \in \mathbb{R}^{K\times U}$ and an item latent matrix $H \in \mathbb{R}^{K\times I}$. The K-dimensional latent vectors $\mathbf{w}_u$ and $\mathbf{h}_i$ mean the column vectors of W and H, respectively. For decentralized MF method, let $H \in \mathbb{R}^{U\times K\times I}$ denote the item latent tensor where $H^u \in \mathbb{R}^{K\times I}$ denotes the item latent matrix for user u. Thus each user u only needs to store u's own K-dimensional latent vector $\mathbf{w}_u$ and u's item latent matrix H^u. Decentralized MF aims to learn $\mathbf{w}_u$ and H^u for each user in the way that training the model on each user's device and exchanging updates to others.

3.2 Model Formulation and Optimization

In order to align the goal of practical POI recommender systems, i.e., recommending top-k preferred POIs to users, we adopt the pairwise ranking-based objective function, which is based on each user's pair-wise preference. A general pair-wise objective function for ranking-based MF model is as follows:

$$\min_{W,H} \sum_{(u,i,j)\in D} \ell\left(\mathbf{w}_u^\top \mathbf{h}_i^u - \mathbf{w}_u^\top \mathbf{h}_j^u\right) + \lambda\left(\|\mathbf{w}_u\|^2 + \|\mathbf{h}_i^u\|^2 + \|\mathbf{h}_j^u\|^2\right), \tag{1}$$

where ℓ is a convex loss function such as exponential loss $\ell(t) = e^{-t}$, hinge loss $\ell(t) = \max(0, 1-t)$, etc. λ is a regularization parameter. $\|\cdot\|$ denotes the Frobenius norm. And $\mathbf{h}_i^u, \mathbf{h}_j^u$ denote the i-th and j-th column of H^u for user u respectively. The training data is generated as: $D = \{(u,i,j) | i \in \mathcal{I}_u^+ \wedge j \in \mathcal{I}\backslash\mathcal{I}_u^+\}$, where $\mathcal{I}_u^+$ represents the set of items that is interacted by the user u. The semantic meaning of $(u,i,j) \in D$ is that user u is assumed to prefer item i over item j. Following (Rendle et al., 2009), we adopt a bayesian personalized ranking optimization criterion. Specifically, we use a logistic sigmoid function to model the predicted probability of user u preferring item i than item j. Then we use maximum posterior estimator to derive the objective function of the ranking-based MF model:

$$\min_{W,H} \sum_{(u,i,j)\in D} -\ln\sigma\left(\mathbf{w}_u^\top \mathbf{h}_i^u - \mathbf{w}_u^\top \mathbf{h}_j^u\right) + \lambda\left(\|\mathbf{w}_u\|^2 + \|\mathbf{h}_i^u\|^2 + \|\mathbf{h}_j^u\|^2\right), \tag{2}$$

where $\sigma(x) = 1/(1+e^{-x})$ is the logistic sigmoid function. In addition, we suppose that for user u, the corresponding latent vector $\mathbf{h}_i^u$ of item i can be decomposed as: $\mathbf{h}_i^u = \mathbf{p}_i + \mathbf{q}_i^u$, which means that the latent vector of item i is made up of two parts, shared (global) latent vector $\mathbf{p}_i$ and specific (local) latent vector $\mathbf{q}_i^u$. The shared vector represents the shared preference of all the users, and the specific vector represents the personal favor of user u. Under this assumption, the loss function can be formulated as,

$$\begin{aligned} \mathcal{L} = \min_{W,H} \sum_{(u,i,j)\in D} & -\ln\sigma\left(\hat{r}_{ui} - \hat{r}_{uj}\right) + \frac{\alpha}{2}\|\mathbf{w}_u\|^2 \\ & + \frac{\beta}{2}\|\mathbf{p}_i\|^2 + \frac{\gamma}{2}\|\mathbf{q}_i^u\|^2 + \frac{\delta}{2}\|\mathbf{p}_j\|^2 + \frac{\mu}{2}\|\mathbf{q}_j^u\|^2 \\ & \text{s.t.} \quad \mathbf{h}_i^u = \mathbf{p}_i + \mathbf{q}_i^u, \quad \mathbf{h}_j^u = \mathbf{p}_j + \mathbf{q}_j^u, \end{aligned} \tag{3}$$

where $\hat{r}_{ui} = \mathbf{w}_u^\top \mathbf{h}_i^u$, and $\hat{r}_{uj} = \mathbf{w}_u^\top \mathbf{h}_j^u$. The last five terms in Equation (3) are regularizers. In the decentralized learning setting, not only the specific latent vectors $\mathbf{q}_i^u$ and $\mathbf{q}_j^u$ are saved on each user's device, but also the shared latent vectors $\mathbf{p}_i$ and $\mathbf{p}_j$ are saved separately for each user u, which are denoted as $\mathbf{p}_i^u$ and $\mathbf{p}_j^u$ respectively. Thus, in training process, we need to exchange $\mathbf{p}_i^u$ and $\mathbf{p}_j^u$ to learn the shared $\mathbf{p}_i$ and $\mathbf{p}_j$. In light of this, we adopt a communication scheme which sends the gradient of Equation (3) with respect to $\mathbf{p}_i^u$ and $\mathbf{p}_j^u$ from user u to his neighbors. The shared $\mathbf{p}_i$ and $\mathbf{p}_j$ are implicitly updated by this scheme in a fully decentralized manner. Stochastic gradient descent algorithm is used to solve the optimization problem, and the gradient of $\mathcal{L}$ with respect to the model parameters is:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \Theta} &= \sum_{(u,i,j)\in D} \frac{\partial}{\partial \Theta} - \ln \sigma\left(\hat{r}_{ui} - \hat{r}_{uj}\right) + \frac{\lambda_\Theta}{2}\frac{\partial}{\partial \Theta}\|\Theta\|^2 \\ &= \sum_{(u,i,j)\in D} -\frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \frac{\partial}{\partial \Theta}\left(\hat{r}_{ui} - \hat{r}_{uj}\right) + \lambda_\Theta \Theta,\end{aligned} \tag{4}$$

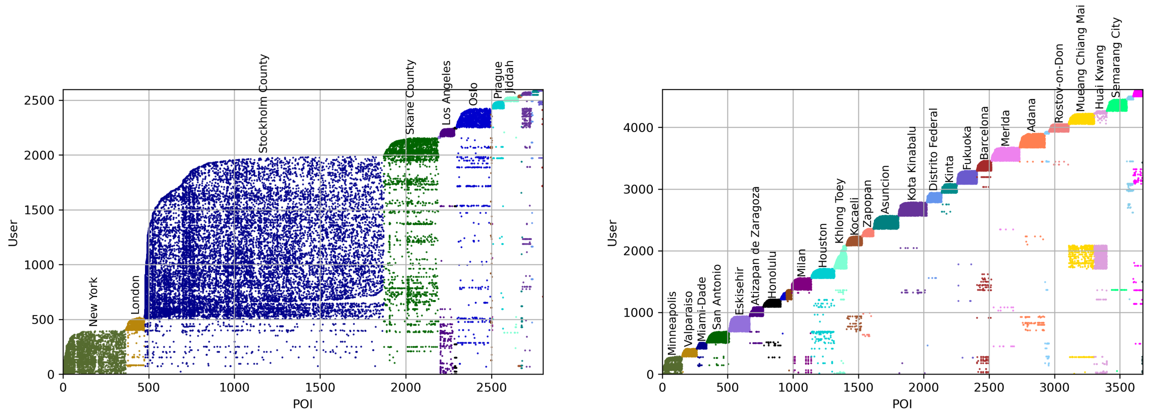
where Θ means the model parameters, such as W and H. λ_Θ denotes the regularization parameter for Θ. Then the model parameters are updated with the learning rate η: $\Theta = \Theta - \eta\frac{\partial \mathcal{L}}{\partial \Theta}$. In our decentralized framework, the gradients for user u with respect to $\mathbf{w}_u$, $\mathbf{p}_i^u$, $\mathbf{q}_i^u$, $\mathbf{p}_j^u$ and $\mathbf{q}_j^u$ are as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{w}_u} &= -\frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \left(\mathbf{h}_i^u - \mathbf{h}_j^u\right) + \alpha \mathbf{w}_u, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{p}_i^u} &= -\frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \mathbf{w}_u + \beta \mathbf{p}_i^u, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{q}_i^u} &= -\frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \mathbf{w}_u + \gamma \mathbf{q}_i^u, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{p}_j^u} &= \frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \mathbf{w}_u + \delta \mathbf{p}_j^u, \\ \frac{\partial \mathcal{L}}{\partial \mathbf{q}_j^u} &= \frac{1}{1+e^{(\hat{r}_{ui}-\hat{r}_{uj})}} \cdot \mathbf{w}_u + \mu \mathbf{q}_j^u.\end{aligned} \tag{5}$$

Based on the above gradient information, users can collaboratively learn the shared $\mathbf{p}_i$ and $\mathbf{p}_j$. For example, user u will send $\partial\mathcal{L}/\partial\mathbf{p}_i^u$ and $\partial\mathcal{L}/\partial\mathbf{p}_j^u$ to neighbors, to collaboratively learn $\mathbf{p}_i$ and $\mathbf{p}_j$. In this way, the proposed model only exchanges the gradients of shared item latent vectors, which can protect the privacy of sensitive data. Next, we focus on how to choose neighbors for each user in the decentralized setting.

3.3 Location-Based User Communication

The essence of MF-based methods is that the latent vectors are learnt collaboratively. Therefore, which user should be communicated with is a key problem in the decentralized setting. Let $d_{u,u'}$ be the distance between user u and user u', and the relationship degree between u and u' is defined as: $w_{u,u'} = f(d_{u,u'})$, where $w_{u,u'} \in [0, 1]$ and $f(\cdot)$ is a mapping function of distance and relationship degree. The smaller the distance between u and u' is, the bigger their relationship degree is. Then existing communication schemes select user



(a) Gowalla

(b) Foursquare

Figure 2: Check-in distribution for two real-world datasets, Gowalla (left) and Foursquare (right). Different colors are used to distinguish POIs located in different regions. It is obvious that the check-in records are geographically aggregated.

according to the probability

$$P(u, u') = \frac{w_{u,u'}}{\sum_{u' \in N_u} w_{u,u'}}, \tag{6}$$

where N_u denotes the neighbor sets for user u, and $P(u, u')$ means the probability between user u and u'.

However, to obtain $P(u, u')$, existing communication schemes expose user's raw ratings unintentionally, e.g., random walking on user-item rating bipartite graph or making use of user similarity based on their common items. To address this problem, we adopt a novel location-based user communication scheme that determining neighbors only depends on the city where user locates in, inspired by the phenomenon that the interactions between users and POIs are highly regional aggregated. As shown in Figure (2), most users are only active in a certain city and each POI is usually interacted by the users in the same city where it locates.

Specifically, we define the neighbor set N_u for user u as other users in the same city, and let $w_{u,u'} = 1$ such that each user can be chosen evenly with same probability since it is not the focus of our work. Since N_u can be large in practical, we further limit the maximum number of neighbors N so that at most N neighbors are chosen to be communicated with.

3.4 Efficient Gradient Exchange

Armed with the location-based user communication scheme, we can find the necessary collaborative neighbors. Now we move our focus on the communication efficiency of decentralized learning. Aiming to alleviate the communication cost among users' devices, we propose a gradient quantization method to compress the gradients of shared item latent vectors in DGMF. By quantizing the values of gradient into several values instead of transmitting the original full-precision floating-point gradient, we can use fewer bits to represent the gradients and reduce the communication cost.

We first introduce a general stochastic quantization function: for any gradient vector $\mathbf{g} \in \mathbb{R}^n$ with $\mathbf{g} \neq 0$, the stochastic quantization function $Q_s(\mathbf{g})$ is defined as

$$Q_s(\mathbf{g}) = v \cdot \mathrm{sgn}(\mathbf{g}) \ \circ \ \xi(\mathbf{g}, v, s), \tag{7}$$

where $\circ$ is the Hadamard product, v is a scaling factor, and s is the number of discrete values. $\mathrm{sgn}(\mathbf{g})$ returns the sign value of the each element in $\mathbf{g}$, and $\xi(\cdot)$ is a stochastic function which maps a scalar to some element in $\{0, \frac{1}{s}, \dots, 1\}$ according to

$$\xi(\mathbf{g}, v, s) = \begin{cases} l/s & \text{with probability } 1 - p\left(\dfrac{|g_k|}{v}, s\right), \\ (l+1)/s & \text{otherwise,} \end{cases} \tag{8}$$

where g_k is the k-th element of $\mathbf{g}$, and $0 \leq l < s$ is an integer such that $|g_k|/v \in [l/s, (l+1)/s]$ and $p(a, s) = as - l$ for any $a \in [0, 1]$. The stochastic rounding has an unbiased expectations satisfying $\mathbf{E}[\xi(\mathbf{g}, v, s)] = \mathrm{abs}(\mathbf{g})/v$. Based on the above general stochastic quantization function, many stochastic quantization methods (Alistarh et al., 2017; Wen et al., 2017; Wu et al., 2018) can be obtained.

We set $v = \|\mathbf{g}\|_{+\infty} = \max(\mathrm{abs}(\mathbf{g}))$ and $s = 1$ in our quantization technique by default. The compressed gradient $\hat{\mathbf{g}}$ is computed as $\hat{\mathbf{g}} = Q(\mathbf{g}) = v \cdot \mathrm{sgn}(\mathbf{g}) \ \circ \ \mathbf{b}$. Each element of $\mathbf{b}$ follows the Bernoulli distribution as

$$\begin{cases} P(b_k = 1|\mathbf{g}) = |g_k|/v, \\ P(b_k = 0|\mathbf{g}) = 1 - |g_k|/v, \end{cases} \tag{9}$$

where b_k is the k-th element of $\mathbf{b}$. In the way that compressing the gradient into a ternary vector with values in $\{-1, 0, +1\}$, we can use just $\log_2 3$ bits to encode each element of gradients. Compared with the original gradients saved by the floating-point form (32 bits), the proposed quantization technique can aggressively reduce the communication cost, and further enhances the privacy of information. Although our proposed model is trained by transferring gradient information instead of the raw data, the gradient exchange protocol may also lead to privacy leakage, i.e., untrusted observers in the decentralized network may still violate the privacy by manipulating the protocol. As the gradients are linear transformations of the data matrix, it is also possible to recover the original data after collecting enough gradients by using sensing techniques (Baraniuk, 2007). The quantization technique is equivalent to the encryption of the gradients. While preserving the statistical properties of the gradients, the randomness is also introduced to the gradients. We summarize our proposed method in Algorithm 1.

3.5 Complexity Analysis

In this section, we analyze the communication and computation complexities of Algorithm 1. Here we assume that all real values are represented as 32-bit floating-point numbers.

3.5.1 Communication Complexity

The communication cost depends on the length of item gradient K and the maximum number of neighbors N. Before gradient quantization, each original value of item gradient

Algorithm 1 DGMF Optimization

```
Input: training data ($D$), learning rate ($\eta$), regularization strength ($\alpha, \beta, \gamma, \delta, \mu$), the number of neighbors ($N$), and maximum iterations ($T$).
Output: user latent matrix ($W$), shared item latent tensor ($P$), and specific item latent tensor ($Q$).
For $u = 1$ to $U$ do
    Initialize $\mathbf{w}_u$, $P^u$, $Q^u$.
End For
For $t = 1$ to $T$ do
    Randomly draw $(u, i, j) \in D$.
    $\mathbf{w}_u \leftarrow \mathbf{w}_u - \eta(\frac{\partial \mathcal{L}}{\partial \mathbf{w}_u})$.
    $\mathbf{p}_i^u \leftarrow \mathbf{p}_i^u - \eta(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_i^u})$.
    $\mathbf{q}_i^u \leftarrow \mathbf{q}_i^u - \eta(\frac{\partial \mathcal{L}}{\partial \mathbf{q}_i^u})$.
    $\mathbf{p}_j^u \leftarrow \mathbf{p}_j^u - \eta(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_j^u})$.
    $\mathbf{q}_j^u \leftarrow \mathbf{q}_j^u - \eta(\frac{\partial \mathcal{L}}{\partial \mathbf{q}_j^u})$.
    For neighbor $u'$ do
        Receive $Q(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_i^u})$, and $Q(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_j^u})$.
        Update $\mathbf{p}_i^{u'}$ with $Q(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_i^u})$.
        Update $\mathbf{p}_j^{u'}$ with $Q(\frac{\partial \mathcal{L}}{\partial \mathbf{p}_j^u})$.
    End For
End For
```

contains $32K$ bits information. For user u, the maximum number of neighbors to be communicated is $\min(|C_u|, N)$, where $|C_u|$ is the actual number of neighbors for user u . Thus, for passing the whole training data, the communication cost is $|D| \times \min(|C_u|, N) \times 32K \times 2$ bits, where $|D|$ denotes the number of instances in dataset D. After compressing the gradients, we can use $\log_2 3$ bits to store each element of the gradients, and plus another 32 bits to save the scaler v. Hence, the communication cost for passing the training data is reduced to $|D| \times \min(|C_u|, N) \times (32 + K \log_2 3) \times 2$ bits. The communication cost can be aggressively reduced.

3.5.2 Computation Complexity

The computation cost mainly relies on three parts, (1) calculating gradients, (2) quantizing gradients, and (3) updating user and item latent vectors. For a single pass of the training data, the time complexity of (1) is $|D| \times K$, the time complexity of (2) is $|D| \times K$, and the time complexity of (3) is $|D| \times \min(|C_u|, N) \times K$. In summary, the total computational complexity is $|D| \times \min(|C_u|, N) \times K$, which is linear with the training data size $|D|$. However, in practical scenario, D has the complexity of $O(UI^2)$ which will be a burden during training. To address the efficiency of training, we restrict the number of negative samples j for each rating (u, i) by sampling to reduce the complexity to $O(UI)$. The above communication and computation complexity analysis shows that our proposed approach is efficient and can scale up to large datasets.

4. Variance Bound of Quantization Error

In this section, we analyze the variance bound of quantization error. We define the quantization error ϵ as the difference between the original gradient $\mathbf{g}$ and its quantization result $\hat{\mathbf{g}}$, $\epsilon = \hat{\mathbf{g}} - \mathbf{g}$. Then we derive the variance bound of ϵ.

Theorem 1. *For any gradient vector* $\mathbf{g} \in \mathbb{R}^n$ *and its quantization result* $\hat{\mathbf{g}} \in \mathbb{R}^n$*, we have*

$$\mathbf{E}[\|\epsilon\|_2^2] \leq n \cdot \|\mathbf{g}\|_{+\infty}^2 \tag{10}$$

Proof. Since the quantization tuning parameter $s = 1$, $\mathbf{E}[\xi(\mathbf{g}, v, s)]$ has minimal variance over distributions with support $\{0, 1\}$, and its expectation satisfies $\mathbf{E}[\xi(g_k, v, s)] = |g_k|/\|\mathbf{g}\|_{+\infty}$. We first have the following bound:

$$\begin{aligned}\mathbf{E}[\xi(g_k, v, s)^2] &= \mathbf{E}[\xi(g_k, v, s)]^2 + \mathbf{E}[(\xi(g_k, v, s) - \mathbf{E}[\xi(g_k, v, s)])^2] \\ &= \frac{|g_k|^2}{\|\mathbf{g}\|_{+\infty}^2} + \frac{1}{s^2} p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s) \cdot (1 - p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s)) \\ &\leq \frac{|g_k|^2}{\|\mathbf{g}\|_{+\infty}^2} + \frac{1}{s^2} p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s).\end{aligned} \tag{11}$$

Under this bound, we have

$$\begin{aligned}\mathbf{E}[\|Q(\mathbf{g}, s)\|^2] &= \sum_{k=1}^{n} \mathbf{E}[\|\mathbf{g}\|_{+\infty}^2 \xi(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s)^2] \\ &\leq \|\mathbf{g}\|_{+\infty}^2 \sum_{k=1}^{n} \mathbf{E}[\frac{|g_k|^2}{\|\mathbf{g}\|_{+\infty}^2} + \frac{1}{s^2} p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s)] \\ &= \left(\frac{\|\mathbf{g}\|_2^2}{\|\mathbf{g}\|_{+\infty}^2} + \frac{1}{s^2} \sum_{k=1}^{n} p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s) \right) \|\mathbf{g}\|_{+\infty}^2.\end{aligned} \tag{12}$$

Combining with the previous condition $p(a, s) < 1$ and $s = 1$, we have

$$\left(\frac{\|\mathbf{g}\|_2^2}{\|\mathbf{g}\|_{+\infty}^2} + \frac{1}{s^2} \sum_{k=1}^{n} p(\frac{|g_k|}{\|\mathbf{g}\|_{+\infty}}, s) \right) \|\mathbf{g}\|_{+\infty}^2 \leq \|\mathbf{g}\|_2^2 + n \cdot \|\mathbf{g}\|_{+\infty}^2. \tag{13}$$

□

This immediately implies that $\mathbf{E}[\|Q(\mathbf{g}) - \mathbf{g}\|_2^2] \leq n \cdot \|\mathbf{g}\|_{+\infty}^2$. We can see the second moment of quantization error is bounded. In addition, since the quantization method relies on an unbiased stochastic rounding technique, we have $\mathbf{E}[\hat{\mathbf{g}}] = \mathbf{g}$, which helps preserve the statistical properties of original gradients.

Table 1: Dataset statistics.

Datasets	#users	#POIs	#check-ins	#cities
Gowalla	2,598	2,801	29,923	20
Foursquare	4,615	3,675	41,294	30

5. Experiments

In this section, we evaluate our proposed algorithm aiming at answering the following questions: (1) How does DGMF perform compared with existing centralized MF models and decentralized MF models? (2) How does gradient quantization affect the DGMF, and would it damage the recommendation accuracy? (3) Whether or not the location-based user communication scheme is effective?

5.1 Experiment Datasets and Settings

We conduct experiments on two real-world POI datasets, Gowalla and Foursquare (Cho, Myers, & Leskovec, 2011; Yang, Zhang, & Qu, 2016). For both datasets, we use two-month check-in history (May 2010 to June 2010 for Gowalla and April 2012 to May 2012 for Foursquare). To remove outliers and clean up the data, we impose that each user/item has at least 5 interactions. The reason why we use subsets of datasets is: we alleviate the memory usage since we are simulating decentralized learning during our experiments. The simulation will produce two huge $U \times \mathbb{R}^{K\times I}$ item latent matrices in total, which should be evenly distributed on each user's device in practical. After that, we select 20 cities which have most check-in records for Gowalla dataset and 30 cities for Foursquare. Table 1 shows the statistics after processing two datasets, with which we randomly sample 90% as training set and the rest 10% as test set, and Figure 2 illustrates their geological aggregation patterns.

5.1.1 Evaluation Metrics

POI recommendation aims to recommend top-k highest ranked POIs to a targeted user. Hence, we evaluate the recommendation performance with widely used ranking-based metrics, i.e., *Precision@k* (P@k), *Recall@k* (R@k) and Area Under Receiver operating Characteristic Curve (AUC). @k means that the ranked list is truncated at position k.

5.1.2 Baselines and Parameters Settings

We compare our proposed DGMF with the following MF based models:

- **MF** (Koren et al., 2009): the most classic centralized MF model.
- **MF-BPR** (Rendle et al., 2009): a classic centralized ranking-based MF model.
- **DMF** (Chen et al., 2018): a recently proposed decentralized MF model, and achieves the state-of-the-art in POI recommendation.

- **PDMF** (Duriakova et al., 2019): another similar state-of-the-art privacy-preserving decentralized MF model.

To analyze the contribution of our design, we perform ablation study with three variants of our method:

- **DGMF-W**: a special case of our proposed model, which removes the gradient quantization technique and directly exchanges the real-valued gradients.

- **DGMF-G**: a special case of our proposed model that users do not save their specific latent vectors and only depend on the shared vector.

- **DGMF-L**: a special case of our proposed model that users do not exchange preferences and learn the model only based on their own data.

Some state-of-the-art POI recommendation methods have not been compared with. This is mainly because: (1) Most of them are either the improvement of the classic MF model by using additional information or non-MF models like factorization machines, which are not fair to be compared with. Our proposed decentralized and gradient quantization techniques are less related to these centralized and real-valued gradient models, which makes the comparison meaningless. (2) Our focus is to compare the effectiveness among existing centralized or decentralized MF models and our decentralized MF model.

For all experiments, we set the length of ranked POI list $k \in \{5, 10\}$, and search the learning rate η in $[10^{-3}, 10^{-1}]$. User regularizer α and the item regularizers β, γ, δ, μ are determined from $[10^{-5}, 10^{-1}]$. For the latent vector dimension K, we vary its values in $\{5, 10, 15\}$. We simply set $w_{u,u'} = 1$ for Equation (6) to eliminate the effect of mapping function on model performance, since this is not the focus of this paper. We vary the maximum number of neighbors N in $\{0, 1, 10, 50, 100\}$. Based on the training set, we use 5-fold cross validation to choose the hyper-parameters. The experimental results are obtained by averaging on the metrics of 5 independent trials.

5.2 Accuracy Comparison with Baselines

We report the results of Precision, Recall and AUC on both Gowalla and Foursquare datasets in Table 2. We use the bold font to show the best results and the underline to show the second best results. According to the results, we can draw several interesting observations: (1) All decentralized methods achieve better performance than centralized methods in most cases, which proves the decentralized learning is more suitable for POI recommendation scenarios. (2) Our proposed DGMF consistently outperforms existing centralized MF method and decentralized MF method, which shows the effectiveness of the proposed model. Due to the success design of DGMF, our model is capable of learning both items' global preferences and users' local preferences. The discrepancy between DMF and our DGMF also demonstrates that ranking-based objective and location-based user communication scheme are the key components that can compute and deliver the gradients more accurately and effectively.

Table 2: The performance of different centralized and decentralized methods on two datasets

Dataset	Gowalla					Foursquare				
Metrics	P@5	R@5	P@10	R@10	AUC	P@5	R@5	P@10	R@10	AUC
Dimension			$K = 5$					$K = 5$		
MF	0.0226	0.1009	0.0197	0.1651	0.8507	0.0197	0.0835	0.0180	0.1502	0.9389
MF-BPR	0.0267	0.1223	0.0225	0.1812	0.8645	0.0245	0.0970	0.0213	0.1712	0.9456
DMF	0.0345	0.1518	0.0271	0.2305	0.8567	0.0328	0.1349	0.0255	0.2187	0.9421
PDMF	0.0326	0.1463	0.0253	0.2210	0.8552	0.0284	0.1090	0.0226	0.1826	0.9418
DGMF	**0.0382**	**0.1604**	**0.0279**	**0.2369**	**0.8735**	**0.0372**	**0.1556**	**0.0274**	**0.2286**	**0.9531**
Dimension			$K = 10$					$K = 10$		
MF	0.0326	0.1392	0.0272	0.2205	0.8562	0.0259	0.1263	0.0241	0.2087	0.9425
MF-BPR	0.0362	0.1539	0.0302	0.2458	0.8763	0.0335	0.1478	0.0282	0.2305	0.9534
DMF	0.0397	0.1745	0.0313	0.2532	0.8665	0.0370	0.1602	0.0309	0.2576	0.9510
PDMF	0.0365	0.1583	0.0293	0.2395	0.8633	0.0347	0.1522	0.0290	0.2409	0.9507
DGMF	**0.0425**	**0.1832**	**0.0326**	**0.2602**	**0.8789**	**0.0403**	**0.1662**	**0.0325**	**0.2608**	**0.9548**
Dimension			$K = 15$					$K = 15$		
MF	0.0367	0.1675	0.0303	0.2582	0.8586	0.0338	0.1413	0.0290	0.2318	0.9463
MF-BPR	0.0407	0.1783	0.0332	0.2752	0.8796	0.0364	0.1517	0.0309	0.2607	0.9571
DMF	0.0458	0.1922	0.0338	0.2812	0.8763	0.0427	0.1642	0.0338	0.2702	0.9535
PDMF	0.0415	0.1820	0.0334	0.2785	0.8749	0.0392	0.1551	0.0324	0.2623	0.9524
DGMF	**0.0468**	**0.1982**	**0.0354**	**0.2886**	**0.8815**	**0.0457**	**0.1853**	**0.0353**	**0.2855**	**0.9588**

5.3 Ablation Study

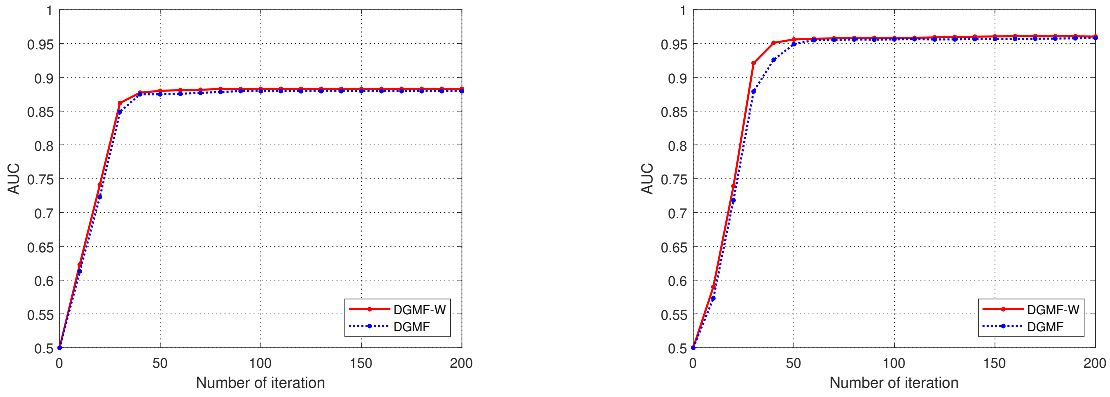
The results of DGMF and its variants on both datasets are listed in Table 3. The numbers illustrated that: (1) Compared with DGMF-W, DGMF can converge to the similar accuracy with less degradation, which is consistent with our theoretical analysis of quantization error bound. It is empirically reasonable because the quantized gradients carry less information than the real-valued gradients. (2) DGMF-L behaves the worst, since each user learns item preference only based on his own check-in data which is very sparse and is not enough to support the model learning. This phenomenon shows the necessity of location-based communication scheme during training. (3) DGMF-G achieves suboptimal performance by only using global item latent vectors. This observation indicates the effectiveness of our proposed mixed item latent vectors. In conclusion, items' global preferences and proper communication scheme are necessary to train an accurate recommendation model. Users' local preferences are also helpful for making personalized recommendation according to the global preferences.

5.4 Convergence of the Proposed Method

As we analyzed above, the computing time complexity is linear with the training data size. Therefore, the converging speed determines how long DGMF and DGMF-W should be trained. For this reason, we analyze the convergence of the proposed methods. We choose AUC as the evaluation metric, since the pair-wise objective function is similar to optimize AUC (Rendle et al., 2009). We set the latent vector dimension $K = 15$ and test DGMF and DGMF-W on both two datasets. The performance on other metrics are similar, so they are not included for the sake of saving space. From Figure 3(a) and 3(b), we can observe that DGMF and DGMF-W converge steadily with the increase of T, and it takes about 50

Table 3: The performance of our proposed method and its variants on two datasets

Dataset	Gowalla					Foursquare				
Metrics	P@5	R@5	P@10	R@10	AUC	P@5	R@5	P@10	R@10	AUC
Dimension			$K = 5$					$K = 5$		
DGMF-G	0.0265	0.1158	0.0225	0.1818	0.8683	0.0298	0.1204	0.0246	0.2018	0.9479
DGMF-L	0.0172	0.0082	0.0141	0.1228	0.8467	0.0153	0.0502	0.0119	0.1025	0.9277
DGMF-W	**0.0398**	**0.1621**	**0.0290**	**0.2413**	**0.8781**	**0.0381**	**0.1588**	**0.0285**	**0.2393**	**0.9561**
DGMF	0.0382	0.1604	0.0279	0.2369	0.8735	0.0372	0.1556	0.0274	0.2286	0.9531
Dimension			$K = 10$					$K = 10$		
DGMF-G	0.0393	0.1728	0.0315	0.2536	0.8783	0.0346	0.1393	0.0279	0.2353	0.9519
DGMF-L	0.0241	0.1065	0.0169	0.1822	0.8521	0.0186	0.0929	0.0148	0.1458	0.9357
DGMF-W	**0.0447**	**0.1862**	**0.0342**	**0.2676**	**0.8813**	**0.0413**	**0.1679**	**0.0345**	**0.2653**	**0.9572**
DGMF	0.0425	0.1832	0.0326	0.2602	0.8789	0.0403	0.1662	0.0325	0.2608	0.9548
Dimension			$K = 15$					$K = 15$		
DGMF-G	0.0453	0.1874	0.0323	0.2712	0.8787	0.0398	0.1577	0.0328	0.2613	0.9538
DGMF-L	0.0308	0.1288	0.0225	0.2032	0.8563	0.0238	0.1026	0.0204	0.1808	0.9472
DGMF-W	**0.0489**	**0.2024**	**0.0375**	**0.2924**	**0.8837**	**0.0476**	**0.1878**	**0.0362**	**0.2865**	**0.9603**
DGMF	0.0468	0.1982	0.0354	0.2886	0.8815	0.0457	0.1853	0.0353	0.2855	0.9588



(a) Convergence on Gowalla

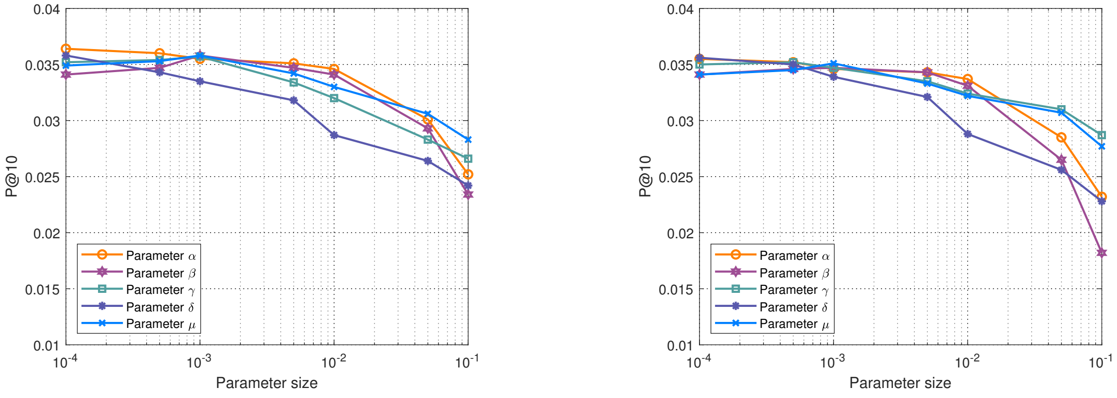
(b) Convergence on Foursquare

Figure 3: AUC convergence comparison of DGMF and DGMF-W

iterations to to converge on both two datasets, which proves that our proposed algorithms can converge quickly.

5.5 Effect of Regularizer Parameter

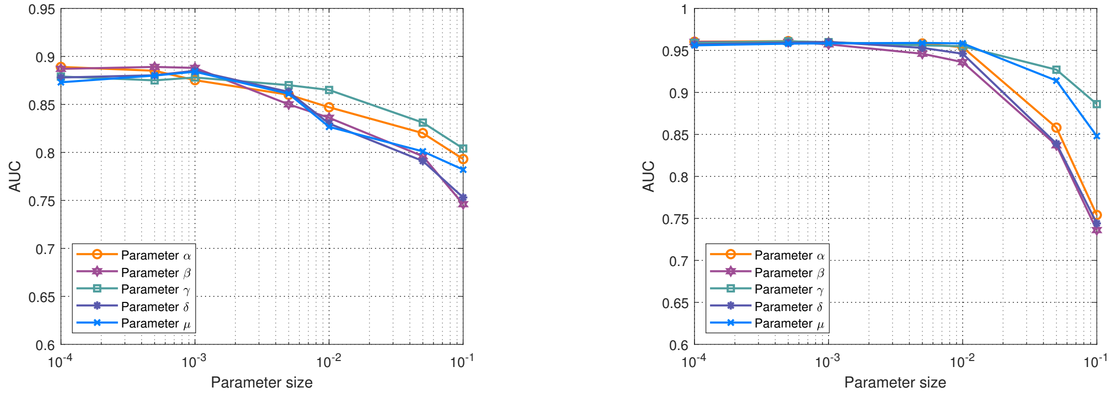
In this section, we analyze the influences of regularizer parameters. When one regularizer parameter is considered, other parameters are fixed. We show the results of P@10 and AUC with latent vector dimension $K = 15$ in Figure 4 and Figure 5. We can see that the effectiveness of the model is stable when each parameter is set in a relatively small range. The results also depict that the increase of local item regularizer parameters γ and μ are more stable than global item regularizer parameters β and δ. This phenomenon illustrates that the large local (or global) item regularizer parameters are partially similar to DGMF-G (DGMF-L), which is consistent with the observations in Table 3. Overall,



(a) Gowalla

(b) Foursquare

Figure 4: Effect of different regularizer parameters on two datasets.



(a) Gowalla

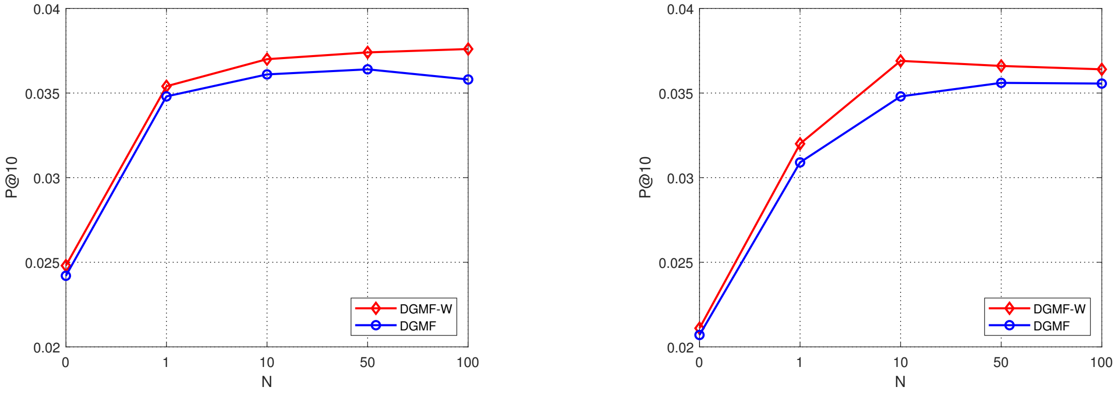
(b) Foursquare

Figure 5: Effect of different regularizer parameters on two datasets

through the carefully design of regularizer parameters, our proposed model can achieve satisfactory performance.

5.6 Effect of User Communication Scheme

In this section, for analyzing the effect of our location-based user communication scheme, we vary maximum number of neighbors N in $\{0, 1, 10, 50, 100\}$ while fixing latent vector dimension $K = 15$. The results of P@10 are illustrated in Figure 6. Note that the case of $N = 0$ means that each user only uses his own data to learn the model. From Figure 6, we can observe that the accuracy is relatively poor in such case, which is similar to the results of the DGMF-L. Additionally, when N increases as we use neighbors to learn the model collaboratively, the accuracy will be significantly improved. This proves that our location-based communication scheme is effective and is suitable for the decentralized setting in POI recommendation scenarios. The performance tends to be relative stable when N is bigger



(a) Gowalla

(b) Foursquare

Figure 6: Effect of the maximum communication neighbors N on two datasets

than 10, and it further shows that our model achieves a good performance with only a small value of N, which significantly reduce the communication complexity.

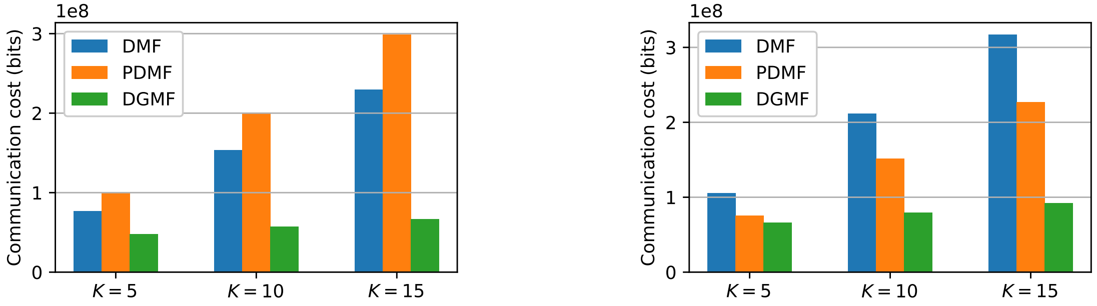
We also compare the computational cost and communication cost to other decentralized MF models in Table 4. It is clear that our model can significantly reduce the communication cost with computational cost similar to other baselines. Experiment results on two datasets in Figure 7 demonstrate that our proposed method can reduce the communication cost when K increases. This phenomenon shows that our communication scheme is more suitable for training an accurate decentralized recommendation model with less data being exchanged, because compared to computation that can be distributed to each user's device, communication usually becomes the bottleneck during training due to the network bandwidth.

Table 4: Computational and communication cost for different models

Model	Computational cost	Communication cost
DMF	$O\left(\lvert D\rvert \times \min(\lvert C_u\rvert, N) \times K\right)$	$32 \times \lvert D\rvert \times \min(\lvert C_u\rvert, N) \times K$
PDMF	$O(\lvert D\rvert \times \lvert C_u\rvert \times K)$	$32 \times \lvert D\rvert \times \lvert C_u\rvert \times K$
DGMF	$O\left(\lvert D\rvert \times \min(\lvert C_u\rvert, N) \times K\right)$	$2 \times \lvert D\rvert \times \min(\lvert C_u\rvert, N) \times (32 + K \log_2 3)$

6. Conclusions

In this paper, we proposed a decentralized gradient-quantization based matrix factorization framework for mobile POI recommendation, which keeps the data on each user's own device. Our proposed framework consists of two key techniques, i.e., location-based communication scheme and stochastic gradient quantization technique. The former technique assures the model is trained collaboratively in each user's end, while the latter one significantly reduces the communication overhead and further hides the gradient information. Our proposed framework can be deployed in a fully decentralized manner that all the users' devices can



(a) Gowalla

(b) Foursquare

Figure 7: Comparison of communication cost on two datasets

be taken as distributed learners. Hence, the efficiency problem can be well addressed in this way. Experimental results on two real-world datasets demonstrate that compared with the existing centralized or decentralized MF based models, the proposed method significantly improves the recommendation performance.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62077015, 62072186), the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2019B1515130001).

References

Agarwal, N., Suresh, A. T., Yu, F., Kumar, S., & McMahan, H. B. (2018). Cpsgd: Communication-efficient and differentially-private distributed sgd. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, p. 7575–7586, Red Hook, NY, USA. Curran Associates Inc.

Alistarh, D., Grubic, D., Li, J. Z., Tomioka, R., & Vojnovic, M. (2017). Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, p. 1707–1718, Red Hook, NY, USA. Curran Associates Inc.

Baraniuk, R. G. (2007). Compressive sensing [lecture notes]. *IEEE Signal Processing Magazine*, *24*(4), 118–121.

Blot, M., Picard, D., Cord, M., & Thome, N. (2016). Gossip training for deep learning. arXiv preprint.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., & Seth, K. (2016). Practical secure aggregation for federated learning on user-held data. arXiv preprint.

Chen, C., Liu, Z., Zhao, P., Zhou, J., & Li, X. (2018). Privacy preserving point-of-interest recommendation using decentralized matrix factorization. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1).

Cheng, C., Yang, H., King, I., & Lyu, M. R. (2012). Fused matrix factorization with geographical and social influence in location-based social networks. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, p. 17–23. AAAI Press.

Cheng, C., Yang, H., Lyu, M. R., & King, I. (2013). Where you like to go next: Successive point-of-interest recommendation. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, p. 2605–2611. AAAI Press.

Cheng, Z., Caverlee, J., Lee, K., & Sui, D. Z. (2011). Exploring millions of footprints in location sharing services.. ICWSM, pp. 81–88. AAAI Press.

Cho, E., Myers, S. A., & Leskovec, J. (2011). Friendship and mobility: User movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, p. 1082–1090, New York, NY, USA. Association for Computing Machinery.

Dai, X., Yan, X., Zhou, K., Yang, H., Ng, K. K. W., Cheng, J., & Fan, Y. (2019). Hyper-sphere quantization: Communication-efficient sgd for federated learning. arXiv preprint.

Danner, G., Berta, Á., Hegedűs, I., & Jelasity, M. (2018). Robust fully distributed minibatch gradient descent with privacy preservation. *Security and Communication Networks*, *2018*.

Duriakova, E., Tragos, E. Z., Smyth, B., Hurley, N., Peña, F. J., Symeonidis, P., Geraci, J., & Lawlor, A. (2019). Pdmfrec: A decentralised matrix factorisation with tunable user-centric privacy. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys '19, p. 457–461, New York, NY, USA. Association for Computing Machinery.

Dwork, C. (2008). Differential privacy: A survey of results. In Agrawal, M., Du, D., Duan, Z., & Li, A. (Eds.), *Theory and Applications of Models of Computation*, pp. 1–19, Berlin, Heidelberg. Springer Berlin Heidelberg.

Feng, S., Li, X., Zeng, Y., Cong, G., Chee, Y. M., & Yuan, Q. (2015). Personalized ranking metric embedding for next new poi recommendation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, p. 2069–2075. AAAI Press.

Gao, H., Tang, J., Hu, X., & Liu, H. (2015). Content-aware point of interest recommendation on location-based social networks. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, p. 1721–1727. AAAI Press.

Guo, Y., Liu, F., Cai, Z., Zeng, H., Chen, L., Zhou, T., & Xiao, N. (2021). Prefer: Point-of-interest recommendation with efficiency and privacy-preservation via federated edge learning. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, *5*(1).

Hegedundefineds, I., Berta, A., Kocsis, L., Benczúr, A. A., & Jelasity, M. (2016). Robust decentralized low-rank matrix decomposition. *ACM Trans. Intell. Syst. Technol.*, *7*(4).

Hegedűs, I., Danner, G., & Jelasity, M. (2020). Decentralized recommendation based on matrix factorization: A comparison of gossip and federated learning. In Cellier, P., & Driessens, K. (Eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 317–332, Cham. Springer International Publishing.

Huang, J., Tong, Z., & Feng, Z. (2022). Geographical poi recommendation for internet of things: A federated learning approach using matrix factorization. *International Journal of Communication Systems*, *n/a*(n/a), e5161.

Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y. (2016). Binarized neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, p. 4114–4122, Red Hook, NY, USA. Curran Associates Inc.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2017). Federated learning: Strategies for improving communication efficiency. arXiv preprint.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, *42*(8), 30–37.

Lai, Z., Liu, C., Lo, E., Kao, B., & Yiu, S.-M. (2018). Decentralized search on decentralized web. arXiv preprint.

Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E., & Rui, Y. (2014). 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, p. 831–840, New York, NY, USA. Association for Computing Machinery.

Lin, Y., Han, S., Mao, H., Wang, Y., & Dally, W. J. (2018). Deep Gradient Compression: Reducing the communication bandwidth for distributed training. In *The International Conference on Learning Representations*.

Liu, B., Fu, Y., Yao, Z., & Xiong, H. (2013). Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, p. 1043–1051, New York, NY, USA. Association for Computing Machinery.

Long, J., Chen, T., Hung, N. Q. V., & Yin, H. (2022). Decentralized collaborative learning framework for next poi recommendation..

Mackey, L., Talwalkar, A., & Jordan, M. I. (2015). Distributed matrix completion and robust factorization. *Journal of Machine Learning Research*, *16*(28), 913–960.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. arXiv preprint.

Meng, X., Wang, S., Shu, K., Li, J., Chen, B., Liu, H., & Zhang, Y. (2018). Personalized privacy-preserving social recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1).

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Con-*

ference on Uncertainty in Artificial Intelligence, UAI '09, p. 452–461, Arlington, Virginia, USA. AUAI Press.

Scellato, S., Noulas, A., & Mascolo, C. (2011). Exploiting place features in link prediction on location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, p. 1046–1054, New York, NY, USA. Association for Computing Machinery.

Spring, R., & Shrivastava, A. (2017). Scalable and sustainable deep learning via randomized hashing. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, p. 445–454, New York, NY, USA. Association for Computing Machinery.

Tang, H., Liang, X., Yan, M., Zhang, C., & Liu, J. (2018). d^2: Decentralized training over decentralized data. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 4848–4856.

Uryasev, S., & Pardalos, P. M. (2006). *Stochastic Optimization: Algorithms and Applications (Applied Optimization, Volume 54) (Applied Optimization)*. Springer-Verlag, Berlin, Heidelberg.

Wen, W., Xu, C., Yan, F., Wu, C., Wang, Y., Chen, Y., & Li, H. (2017). Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, p. 1508–1518, Red Hook, NY, USA. Curran Associates Inc.

Wu, J., Huang, W., Huang, J., & Zhang, T. (2018). Error compensated quantized sgd and its applications to large-scale distributed optimization. arXiv preprint.

Yang, D., Zhang, D., & Qu, B. (2016). Participatory cultural mapping based on collective behavior data in location-based social networks. *ACM Trans. Intell. Syst. Technol.*, *7*(3).

Yin, H., Zhou, X., Cui, B., Wang, H., Zheng, K., & Nguyen, Q. V. H. (2016). Adapting to user interest drift for poi recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *28*(10), 2566–2581.

Yin, H., Cui, B., Chen, L., Hu, Z., & Zhang, C. (2015). Modeling location-based user rating profiles for personalized recommendation. *ACM Trans. Knowl. Discov. Data*, *9*(3).

Yin, H., Sun, Y., Cui, B., Hu, Z., & Chen, L. (2013). Lcars: A location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, p. 221–229, New York, NY, USA. Association for Computing Machinery.

Yin, H., Zhou, X., Shao, Y., Wang, H., & Sadiq, S. (2015). Joint modeling of user check-in behaviors for point-of-interest recommendation. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, p. 1631–1640, New York, NY, USA. Association for Computing Machinery.

Yun, H., Yu, H.-F., Hsieh, C.-J., Vishwanathan, S. V. N., & Dhillon, I. (2014). Nomad: Non-locking, stochastic multi-machine algorithm for asynchronous and decentralized matrix completion. *Proc. VLDB Endow.*, *7*(11), 975–986.

Zhu, Z., Li, Q., Yang, X., Tang, G., & Wakin, M. B. (2019). Distributed low-rank matrix factorization with exact consensus. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., & Garnett, R. (Eds.), *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc.