# Fusion Privacy Protection of Graph Neural Network Points of Interest Recommendation

Yong Gan[1], ZhenYu Hu[2]
School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China

***Abstract*—For the rapidly developing location-based web recommendation services, traditional point-of-interest(POI) recommendation methods not only fail to utilize user information efficiently, but also face the problem of privacy leakage. Therefore, this paper proposes a privacy-preserving interest point recommendation system that fuses location and user interaction information. The geolocation-based recommendation system uses convolutional neural networks (CNN) to extract the correlation between user and POI interactions and fuse text features, and then combine the location check-in probability to recommend POIs to users. To address the geolocation leakage problem, this paper proposes an algorithm that integrates k-anonymization techniques with homogenized coordinates (KMG) to generalize the real location of users. Finally, this paper integrates location-preserving algorithms and recommendation algorithms to build a privacy-preserving recommendation system. The system is analyzed by information entropy theory and has a high privacy-preserving effect. The experimental results show that the proposed recommendation system has better recommendation performance on the basis of privacy protection compared with other recommendation algorithms.**

***Keywords*—Recommendation algorithms; location protection; graph convolutional neural networks; k-anonymity**

## I. Introduction

With the rapid development of mobile Internet, a series of location-based social network (LBSN) services such as e-commerce and social software have attracted millions of users, who establish online links with other users on LBSN, share experiences and comments of visited points of interest[1], and the service platform analyzes user information and mines user preferences to recommend POIs to users. However, as the volume of data and the complexity of information, the traditional text-based collaborative filtering method is no longer applicable to the development of recommendation systems. In addition, the naturally existing geographic distance factor also has an impact on user activities, as human activity areas have the phenomenon of geographic aggregation[2]. Therefore, the geographic impact between users and POIs and between POIs is as important as the impact between users. Therefore, it makes sense to incorporate the geographic distance factor into a collaborative user or content-based filtering approach.

Compared with traditional methods, neural network-based recommendation algorithms can tap deeper features with local relevance and location invariance to extract potentially valid information from Euclidean spatial data[3]. Therefore, it is the mainstream practice in recommendation systems to use CNNs to extract hidden features from data such as users' social relationships, comment texts and visit frequencies and fuse them, and then combine the check-in probabilities of geographic distance factors[4]. At this stage, the emergence of graph neural networks (GNNs) further improves the performance of recommendation systems[5]. As a natural bipartite graph structure (user-interest point nodalization), traditional collaborative filtering methods based on neural networks, although having powerful feature representation, are weak in handling higher-order interaction information of graph-structured data in non-Euclidean space[6]. Therefore, introducing GCN into the recommendation system to extract the connection features of user-interest point interactions through information transfer between graph nodes brings better entity representation and stronger interpretation capability to the recommendation system, thus improving the prediction accuracy of the model[7][8].

Since the recommendation system in LBSN lacks location protection measures for users when recommending POIs to them, they may face the threat of location leakage. Existing location protection methods for LBSN are mainly divided into three types: generalization[9][10], *k*-anonymity and encryption[11][12]. *k*-anonymity is a method that constructs an anonymous region containing *k*-1 users after removing the identification information from their real locations and query contents, so that the probability of identifying a user does not exceed 1/*k*. *k*-anonymity is widely studied and applied because of its high privacy protection effect compared with other methods. However, existing approaches to location privacy protection exhibit significant limitations. First, recommendation systems utilize trusted third-party anonymous servers to protect user location information. This operation inherently leads to information leakage problems. Second, the generalization-based location protection mechanism generalizes the user's real location so that the user's virtual location exists at any point in the constructed larger virtual area, and the recommendation system calculates the recommendation result based on the user's virtual location, which leads to inaccurate recommendation results. *k*-anonymity technique, although it is effective for location protection, is less capable of protecting the user's location during continuous requests, because an attacker will analyze the user's action trajectory based on the user's location requests at multiple moments, and then infer the location visited by the user, exposing the user to security threats. Moreover, *k*-anonymity is achieved by sacrificing the quality of service in exchange for privacy protection [13].

Since the existing location-based point-of-interest recommendation system mainly calculates recommendation

results by analyzing users' social relationships and geographical factors, the deep interaction information between users and POI and users' comment information have more user characteristic information compared with the case of sparse social relationships of users, which can effectively improve the accuracy of the recommendation system. Therefore, how to improve both location privacy protection capability and recommendation system performance is the focus of this paper.

The main contribution of this paper is the design and analysis of a recommendation system that takes location protection into account. Because previous recommender systems did not focus on the protection of user location information, after continuous exploration it is found that privacy protection does not need to sacrifice the greater accuracy of the recommender system. User location information as the most important factor of the recommendation system calculation results, there exists the attack method against location information. The KMG method proposed in this paper is a controllable generalized geolocation algorithm, which calculates virtual location based on all user locations in the anonymous region and divides subanonymous regions based on the distance between real location and virtual location to achieve effective protection of user location without significantly reducing recommendation accuracy. The contribution of this paper in the recommendation system is mainly to fuse the deep interaction information and comment information between users and POI to obtain a more efficient and deeper recommendation algorithm, and integrate it with the location protection algorithm into one system.

## II. Related Work

First LBSN-based recommendation systems usually considers three factors: user similarity, social influence, and geographic influence[14]. The traditional method of recommendation by extracting user ratings and review features suffers from cold start and data sparsity. Park et al.[15] proposed ConvMF recommendation model, which integrates CNN into probability matrix decomposition can effectively capture contextual information and improve recommendation accuracy. Zhao et al.[16] proposed hierarchical dichotomous graph neural network, by stacking multiple graph neural networks(GNN) and alternating with clustering algorithm to obtain the potential preference information of users, which substantially improves the recommendation performance compared to the recommendation system with fused CNNs. Lin et al.[17] fused multilayer graph convolutional models with recurrent neural networks to capture user interaction graph information, which can accurately capture the rich potential implicit information between users-items. Shafqat et al.[18] transformed the similarity into the interaction probability between the neighbors of user nodes with different probability distributions using KL scatter to find the distance between them and perform clustering operations on neighboring nodes, and this method improves the efficiency of neighborhood aggregation for GCN models. The graph neural collaborative filtering model (NGCF) proposed by Wang et al.[19] is one of the classical recommender systems incorporating the GCN model, which embeds and models the user interaction information to obtain the higher-order connectivity between nodes and recommends items of interest to users. He et al.[20] proposed the LightGCN model, which eliminates the feature transformation and nonlinear activation in GCN and improves the efficiency of neighborhood aggregation. However, existing location-based point-of-interest recommendation algorithms generally use fused social relationships without deep mining of user and POI node interaction information, and although GCN can extract the association relationship between users and POI well, most location-based recommendation algorithms do not take into account the advantages of deep learning for processing textual information. Therefore, this paper will use both GCN and deep learning methods to extract interaction information and text information respectively.

There is a correlation between user check-in to points of interest and geographic distance. Ye et al.[21] analyzed the effect of distance on user check-in on the Foursquare and Whrrl datasets and constructed probability distributions based on the POI distance of the same user check-in. Most of the points of interest checked-in by the same user are in a range of small mutual distance, i.e., the check-in locations have a geographic cluster area effect. They fit the check-in probabilities to a power-law distribution, and found that they could cover most (90%) of the POI pairs after eliminating those that did not fit the power-law distribution. Therefore, the power-law distribution can be used to model the distance between POI visited by the same user and calculate the check-in probability.

*k*-anonymity is one of the main methods for privacy preservation. Song et al.[22] proposed k-anonymity method, whose basic idea is to remove user identification information from data so that each user in the same equivalence class is identified with probability A, so it has high privacy preserving ability. Gruteser et al.[23] introduced the idea of *k*-anonymity into LBSN privacy preserving by generalizing the real location of the user into an anonymous region and fuzzy the spatial location information of the user. Liu et al.[24] proposed the B-privacy method, which delineates the area of the region according to the user's location no less than s and generates *k* virtual locations. Since the B-privacy method statically sets the parameters *k* and *s*, it leads to weak protection in areas with small population density. Since the above methods use virtual locations to obtain query results, which leads to low precision of results. Ji et al.[25] proposed to split the anonymized region into several dispersed subanonymized regions, which not only reduces the scope of the anonymized region, but also improves the precision of user queries. Although all these location protection methods effectively protect the user's location, they construct as large a virtual region as possible in order to enhance the security strength, resulting in uncertainty in the distance between the user's virtual location and the real location, which will have an uncertain impact on the recommendation effect when combined with the recommendation algorithm. Therefore, in this paper, by limiting the distance between the real location and the virtual location and fusing the construction of subanonymous regions, the virtual location can meet the security and ensure that it does not have a great impact on the accuracy of the recommendation system.

We construct an anonymous region containing *k* users based on their locations, and replace the center location with

the user location, and the recommendation algorithm recommends users based on the center location. If the distance between the center location and the user location is large, the anonymous region is divided into sub-anonymous regions, and the interest points are recommended again based on the center location of the sub-anonymous region. The recommendation algorithm extracts and fuses high-order interaction features, text features and rating features by LightGCN, CNN and latent factor model (LGCL) respectively, and then recommends POIs jointly with geographic distance factor check-in probability.

## III. RECOMMENDATION ALGORITHM

The recommendation algorithm in this paper consists of two parts: the LGCL module and the geographic distance probability. The LGCL module is a nodalization of user-POI through the LightGCN model, which uses GCN to propagate aggregated user-item interactions on the graph and obtain an embedded representation of all user-item association relationships on the graph; the text feature extraction module is to process the text data into a collection of embedding vectors by BERT pre-training model, and then extract the text features by CNN; the rating data processing module obtains the implicit features of the rating matrix by the implicit semantic model (LFM). The geographical distance probability $p^g$ is calculated using the naive Bayesian method based on the user's historical check-in distance. Finally, the features processed by the similarity module are fused and normalized to obtain $\hat{f}'_{u_i}$, and the POI is recommended to the user jointly with $p^g$. The structure of the recommended algorithm is shown in Fig. 1.

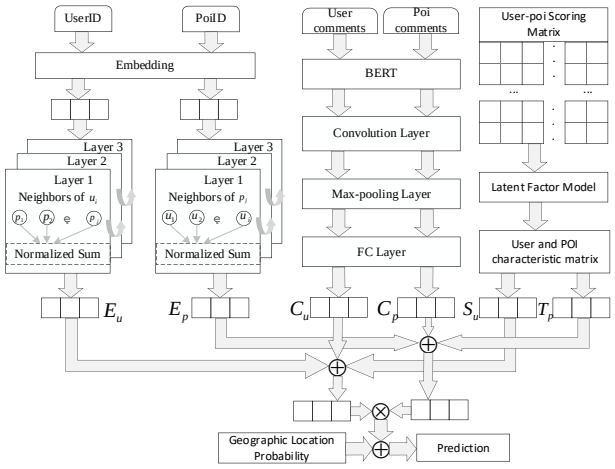



Fig. 1. Recommended algorithm structures.

### A. LGCL Model

The ID features of the user and POI are first fed into the embedding layer, and the set of embedding vectors is represented as:

$$e_u^{(0)} = [e_{u_1}^{(0)}, e_{u_2}^{(0)}, \cdots, e_{u_N}^{(0)}], \quad e_p^{(0)} = [e_{p_1}^{(0)}, e_{p_2}^{(0)}, \cdots, e_{p_M}^{(0)}] \tag{1}$$

where $N$ users, $M$ POI, $e_{u_i}^{(0)}$ and $e_{p_j}^{(0)}$ denote the $c$ dimensional embedding vectors of user $i$, POI $j$, respectively.

Then the IDs of all POI nodes in all neighboring nodes of user $u_i$ are embedded in the vector for aggregation operation. The propagation rule for the embedding representation of user $u_i$ in the LightGCN model for layers $L$ to $L$+1 by:

$$e_{u_i}^{(l+1)} = \sum_{p_j \in N_u} \frac{1}{\sqrt{|N_u|}\sqrt{|N_p|}} e_{p_j}^{(l)} \tag{2}$$

$$e_{p_i}^{(l+1)} = \sum_{u_i \in N_p} \frac{1}{\sqrt{|N_p|}\sqrt{|N_u|}} e_{u_i}^{(l)} \tag{3}$$

where $N_u$ and $N_p$ denote the set of neighboring nodes containing all $u_i$ and $p_j$, respectively; $e_{u_i}^{(l+1)}$ and $e_{p_j}^{(l+1)}$ denote the expression of the embedding of $u_i$ and $p_j$ in LightGCN obtained after the propagation of $L$ layers in $L$+1 layers.

In order to obtain better embedding aggregation, this paper introduces an attention mechanism to get the importance weight of each layer of propagation:

$$Attention_{u_i}^{l} = softmax(\mu_1 \otimes sigmoid(\mu_2 \otimes e_{u_i})) \tag{4}$$

where $\mu_1$, $\mu_2$ are adjustable hyperparameters, $Attention_{u_i}^{l}$ is the layer $L$ weight; weighted summation of the embedding vectors for each layer yields the embedding expression for the association between $u_i$ and $p_j$:

$$E_{u_i} = \sum_{l=0}^{L} Attention_{u_i}^{(l)} e_{u_i}^{(l)}, \quad E_{p_j} = \sum_{l=0}^{L} Attention_{p_j}^{(l)} e_{u_j}^{(l)} \tag{5}$$

### B. Text Feature Extraction Module

All comments of the same user or POI constitute the comment collection, respectively. Because the extraction of user and POI comment features is composed of two similar parallel network structures, only the extraction of user comment features is described in this paper. To simplify the calculation, let the number of comment texts of each user and POI be $n$ and the length of each comment text be $q$. The comment set $r_{u_i}$ of user $u_i$ is input to the BERT pre-training model to obtain the comment embedding vector $e_{u_i} \in R^{n \times q \times d}$, where $d$ is the output dimension of the BERT pre-training model.

The user comment embedding vector $e_{u_i}$ is input to CNN for extracting semantic information features. Let the $m$ convolutional kernels with step size $s$ extract contextual features:

$$g_j = Relu(M_i * R + b) \tag{6}$$

where $Relu()$ is the activation function, $*$ denotes the convolution operation, and $b_i$ denotes the bias.

After the convolution operation, the features $g_j$ are fed into the max-pooling layer to generate features with higher values, which take the maximum vector $w_i$ for the region $max(g_1, g_2, \cdots, g_{d-s+1})$ corresponding to the convolution kernel, preserving the characteristics of the original feature vector and reducing the dimensionality.

Due to the fact that each comment in the comment collection expresses user preference features to a different degree, this paper introduces an attention mechanism that applies different weights to the comments and normalizes them using the softmax function:

$$h_i = softmax(Relu(\mu_3 \otimes g_i + b)) \tag{7}$$

Finally, the text information feature representation of the user $u_i$ is obtained after processing in the fully connected layer:

$$C_{u_i} = \sum_{i=1}^{n} a_i h_i \omega_i + b_i \tag{8}$$

where $a_i$ and $b_i$ denote the amount of deviation from the weights of the fully connected layer, respectively.

### C. Rating Data Processing Module

The rating can be regarded as a direct feedback, and the level of rating can express the user's liking of the POI. In this paper, we use the Latent Factor Model (LFM) to process the user rating matrix, the essence of which is to decompose the rating matrix to get the user feature matrix and the POI feature matrix, so that the implicit features can be expressed.

The LFM model is to decompose the scoring matrix R into a user feature matrix $S_{Nt}$ and a POI feature matrix $T_{tM}$ such that the matrix multiplication of $S_{Nt}$ and $T_{tM}$ is approximately equal to R, where *t* is the hidden feature vector dimension:

$$\hat{R}_{NM} = \sum_{t=1}^{t} S_{Nt} T_{tM} \tag{9}$$

To obtain accurate potential feature matrices of users and POIs, this paper obtains the feature matrices $S_{Nt}$ and $T_{tM}$ by iteratively minimizing the loss function, and the expressions are shown below:

$$Loss(S,T) = \sum_{i}^{N}\sum_{j}^{M}(R_{ij} - S_i T_j)^2 + \omega(\sum_{i}^{N}\| S_i \|^2 + \sum_{j}^{M}\| T_j \|^2) \tag{10}$$

To avoid overfitting add the regularization term, $\omega$ is the regularization parameter. To minimize the loss function, this paper iteratively optimizes the parameters by iteratively moving the variables along the direction of the negative gradient of the loss function through the gradient descent method until convergence to the true scoring matrix.

### D. Feature Integration

The final feature representations of users and POIs are obtained by fusing the associative relationship feature vectors $E_u$ , $E_p$ and text feature vectors $C_u$ , $C_p$ and scoring feature vectors $S_u$ , $T_p$ obtained from the LightGCN processing module, the text feature extraction module and the scoring data processing module:

$$U_{u_i} = E_{u_i} \oplus C_{u_i} \oplus S_{u_i}, \quad P_{p_j} = E_{p_j} \oplus C_{p_j} \oplus T_{p_j} \tag{11}$$

The prediction of user $u_i$ rating of interest point $p_j$ is :

$$\hat{f}_{u_i} = U_{u_i} \otimes P_{p_j}^T \tag{12}$$

which is normalized to obtain $\hat{f}'_{u_i}$ .

To minimize the difference between the predicted score $\hat{f}_{u_i}$ and the true score $f_{u_i}$ in the dataset, this paper uses a loss function to adjust the model parameters, which is calculated as follows:

$$loss = \sum_{i,j \in O} (\hat{f}_{i,j} - f_{i,j}) + \lambda \| \Theta \|^2 \tag{13}$$

where $O$ denotes the number of samples in the training set; $\lambda \| \Theta \|^2$ is the canonical term and $\Theta$ is all trainable model parameters in the model, where $\lambda$ is the adjustable coefficient of the canonical term, which is used to control the model parameters to prevent overfitting. This paper uses Adam to optimize the model parameters and minimize the loss function.

### E. Geographical Distance Probability

The probability of a user signing up to a POI follows a power-law distribution of geographic distance probabilities, so this paper calculates the probability of a user signing up to a POI by modeling the distance between POIs with a power-law distribution, which is calculated as follows:

$$p[d(l_m, l_n)] = a \cdot d(l_m, l_n)^b \tag{14}$$

where $a, b$ are the parameters of the power-law distribution and $d(l_m, l_n)$ is the distance between the point of interest $l_m$ and $l_n$ .

For the POI set $l_k (k = 1, 2, \cdots)$ around $l_i$ , it has an effect on the probability of a user checking in to $l_i$ . Therefore, under the influence of the set $l_k$ , the probability of a user visiting $l_i$ is calculated using the plain Bayesian method:

$$p^g = p[l_i \mid l_k] = \frac{p[l_k] \cdot \Pi_{k=1,2,\ldots} p[d(l_i, l_k)]}{p[l_k]} \tag{15}$$

where the total number of check-ins is greater than $l_i$ and the closest interest point is judged to have some influence on the set $l_k$ of user check-ins to $l_i$ . Finally, this paper normalizes the check-in probabilities as follows:

$$S_{i,j}^{g} = \frac{p_{i,j}^{g}}{z_i^{g}},\ where\ \ z_i^{g} = \max_{j \in L_k} p_{i,j}^{g} \tag{16}$$

### F. Joint Recommendation Result

This paper integrates user similarity factors and geographic distance probabilities into a linear function, and then calculate the combined probability P of user $u_i$ signing up to $p_j$:

$$P = (1-\alpha)\hat{f}'_{i,j} + \alpha S_{i,j}^{g} \tag{17}$$

where $\alpha$ and $1-\alpha(0 \le \alpha \le 1)$ represent the weights of the two factors, respectively, and when $\alpha$ is 0 means that the recommendation result is independent of the geographical distance factor.

## IV. Geolocation Protection Algorithm

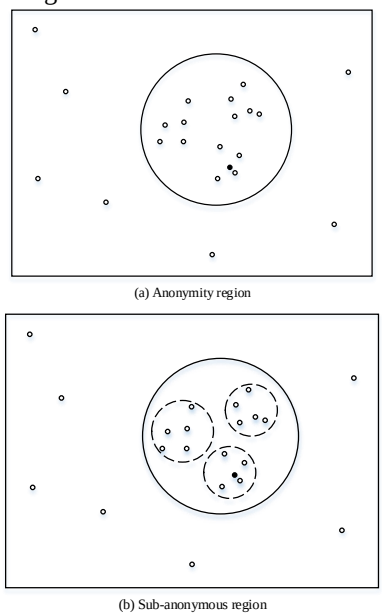
The KMG algorithm in this paper inherits the ideas of k-anonymity and location generalization, and changes the user's real location dynamically according to the locations of other users around. The individual location *k*-anonymity method has better protection for users who make location request services non-continuously, and poor protection for requesting continuous services, such as attackers who connect check-in records at different times to form trajectories, and can infer the user's true location by combining the direction and distance of the trajectories. Therefore, this paper combines *k*-anonymity techniques with location generalization techniques to provide better privacy-preserving capabilities for both individual service requests and consecutive requests.

### A. Description of KMG Algorithm

The KMG protection algorithm is shown in Algorithm 1. The anonymous parameters *k* and *L* are set to 15 and 10, respectively. The specific scheme is as follows:

*1) After receiving the location* $l(x,y)$ of user *u* , the KMG protection algorithm finds the nearest remaining *k*-1 users according to his location and obtains their location information to generate the anonymous region, as in Fig. 2(a), and if the number of users in the anonymous region is less than *k*, a virtual user is generated. The center coordinates of *k* users in the anonymous region are calculated, and when the distance *l* between user *u* and the center coordinates is less than *L*, the center coordinates are replaced with the real locations of all users in the anonymous region and sent to the recommendation algorithm. Where *x* and *y* denote longitude and dimension respectively.

*2) If there exists a user* whose distance from the center coordinate is greater than *L*, the anonymous region is divided into *n* subanonymous regions, as shown in Fig. 2(b). Suppose the anonymous region is divided into 3 subanonymous regions. The number of users $k'$ in each subanonymity region is 5, and the first subanonymity region is formed by finding the nearest 4 users with user *u* as the center, and repeating the process of forming a subanonymity region by selecting a user as the center among the remaining users at random. The center coordinates are calculated based on the position of the user in each subanonymous region and sent to the recommendation algorithm instead of the position of the user in the subanonymous region.



(a) Anonymity region

(b) Sub-anonymous region

Fig. 2. Constructing anonymous regions and sub-anonymous regions.

*3) Virtual user generation:* Generating virtual users satisfies the principles of user distribution similarity and distance similarity. Firstly, the distance between existing users in the anonymous region is calculated and sorted, and the minimum distance between users is *s*. Virtual users are added in the space of any two users so that the distance between virtual users and users is not less than *s*/2 and does not overlap with the real user position.

Algorithm 1. KMG Algorithm

Input: k user locations $(L_1(x_1,y_1),L_2(x_2,y_2),\ldots,L_k(x_k,y_k))$

Output: Returns the generalized location of all users in the anonymous region or sub-anonymous region $(x',y')$ ;

1:if $sum(L_i)<k$:

2: $sort \leftarrow dis(L_i,L_j), add(L_{k-i+1},\ldots,L_k)$

3: $AR_0 \leftarrow Gen(L_i(x_i,y_i))$

4:if $L \ge dis(L_i(x_i,y_i), AR_0(x_o^{(0)},y_o^{(0)}))$

5: return $(x_o^{(0)},y_o^{(0)})$

6:else:

7: $k\%n==0$ and $n \ge 2$

8: $(AR_1,AR_2,\ldots,AR_n) \leftarrow Gen(L_a(x_a,y_a),L_b(x_b,y_b),\ldots,L_n(x_n,y_n))$

9: return $(x_1^{(1)},y_1^{(1)}),(x_2^{(2)},y_2^{(2)}),\ldots,(x_n^{(n)},y_n^{(n)})$

10:end

### B. Privacy Analysis based on Information Entropy

The uncertainty of information is proportional to the information entropy. Therefore, this paper uses Shannon entropy as a measure of the privacy-preserving ability of the model, and a larger Shannon entropy value indicates that it is more difficult for an attacker to distinguish the user's true location from the *k* locations. Shannon entropy is defined as follows:

Definition 1. Assuming that the random variable *x* takes values on a finite set *X*, the entropy of the random variable *x* is defined as

$$H(x) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{18}$$

where *p(x)* denotes the probability when the variable takes the value *x*.

The attacker successfully attacks and obtains the user location information in the anonymous region as an event set *X*. The successful attack on a single user's location information is an event *x* within the event set, and *p(x)* denotes the probability of obtaining the user location information, which has a high uncertainty. The location protection method against this paper, this process needs to be divided into two cases. In the first case, if there is no sub-anonymous region, the attacker successfully obtains the anonymity region corresponding to the user and successfully identifies the user; in the second case, if there is a sub-anonymous region, the attacker successfully identifies the user on top of successfully obtaining the anonymity region and the sub-anonymous region. Since anonymous regions cannot be distinguished directly, the probability that the attacker successfully obtains the anonymous region where the user is located is 1/*N* (assuming there are *N* anonymous regions). If there is a sub-anonymous region, the probability that the attacker succeeds in obtaining the sub-anonymous region is 1/*n*. The probability that the attacker succeeds in identifying the user from the anonymized region or sub-anonymous region in the second step is $1/k^{'}$.

Therefore the probability of an attacker successfully obtaining user information is

$$p_u = \begin{cases} \dfrac{1}{kN} \\ \dfrac{a}{nk'N} \end{cases} \tag{19}$$

where *a* denotes the weight value for dividing sub-anonymous regions, and in this model if there exists *l*>*L*, then sub-anonymous regions are divided.

The entropy of the user can be obtained from (18) as:

$$H(u) = -p_{u_r} \log_2 p_{u_r} \tag{20}$$

The entropy of all users in the anonymous region is:

$$H(u) = \begin{cases} \sum_i H(u_r) \\ \sum_i \sum_j H(u_r) \end{cases} \tag{21}$$

Therefore, when a higher value of anonymization *k* is chosen, the higher the entropy value of the whole anonymization region, the higher the degree of privacy.

## V. Analysis of Experimental Results

### A. Dataset Description

In this paper, we use the Yelp dataset, which is widely used for location-based social network research, as the experimental data. In this paper, the data set is divided by the filter condition that the total number of check-ins is greater than 10 and the number of comments is greater than 3. The data sparsity is $2.63\times10^{-3}$, and the check-in dataset contains user ID, POI ID, longitude, latitude, comments, ratings and time information. This paper randomly selects 80% of the check-in data from the dataset as training data, and the other 20% as test data.

### B. Evaluation Metrics

To assess the recommendation quality of the recommendation *Top-K*, this paper uses *Precision@K* and *Recall@K* as evaluation criteria, where *K* denotes the number of recommended POIs (*K*=5,10,15,20).

$$Precision@K = \frac{A \cap B}{B} \tag{22}$$

$$Recall@K = \frac{A \cap B}{A} \tag{23}$$

Where *A* denotes the POI of user check-in in the test set and *B* denotes the POI in *Top-K*.

### C. Comparison Approach

To verify the performance of the algorithm proposed in this paper, this paper compares the algorithm with the following three algorithms, all of which involve location privacy protection and point-of-interest recommendation. In this paper, the experimental dataset operation of the compared methods is the same as the method proposed in this paper, while generalizing the user location according to the location protection algorithm and using virtual location to make recommendations to users.

*1) USG[21]:* This method blends user preferences for POIs, social relationship influence and geographic influence, and calculates the probability of recommended POIs by the Naive Bayes method. Note that the USG model does not have location privacy protection.

*2) USD[26]:* This method is based on k-coordinate generalized user location and then POI recommendation, where the recommendation system is similar to the USG method. However, in terms of geolocation influencing factors, USD calculates the recommendation POI probability based on the check-in frequency.

*3) GLP[14]:* The GLP approach generalizes user locations based on population density (i.e., check-in density) and uses virtual locations to recommend POIs to users.

### D. Experimental Parameters Adjustment

In order to achieve the best recommended performance, the experiment-related parameters are set in this paper as follows:

embedding vector dimension 64; BERT model output dimension $d$ is 128; number of graph convolution iterations is 3; attention mechanism dimensions are 64 and 128, respectively; regularization coefficient $\lambda$ is $1\times10^{-4}$. In addition, the experimental comparison model uses the parameter settings with the best results in the corresponding literature.

### E. Regularization Coefficient

The regularization coefficient $\lambda$ takes a range of $[1\times10^{-1},1\times10^{-2},1\times10^{-3},1\times10^{-4},1\times10^{-5}]$, which is essentially to adjust the hyperparameters in the LGCL model to make feature extraction more accurate and prevent overfitting. The parameters K, L and α are set to 5, 10 and 0.2, respectively. As shown in Table I, the values of *Precision@K* and *Recall@K* of the recommendation algorithm show a trend of increasing and then decreasing with the decrease of the regularization coefficient. When $\lambda$ is $1\times10^{-4}$, both *Precision@K* and *Recall@K* reach the maximum; when $\lambda$ is $1\times10^{-5}$, *Precision@K* and *Recall@K* decrease, and considering the overfitting problem, the regularization coefficient is $1\times10^{-4}$.

TABLE I. REGULARIZATION COEFFICIENT COMPARISON EXPERIMENT

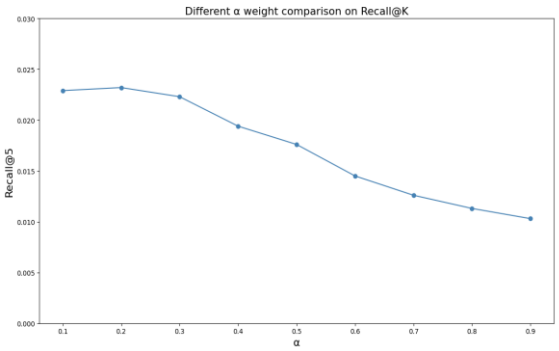
| Recommended performance | Regularization coefficient | | | | |
|---|---|---|---|---|---|
| | $1\times10^{-1}$ | $1\times10^{-2}$ | $1\times10^{-3}$ | $1\times10^{-4}$ | $1\times10^{-5}$ |
| Precision@K | 0.1473 | 0.1754 | 0.2041 | 0.2452 | 0.2270 |
| Recall@K | 0.0158 | 0.0182 | 0.0214 | 0.0232 | 0.0226 |

### F. Geographical Distance Factor Weighting Analysis

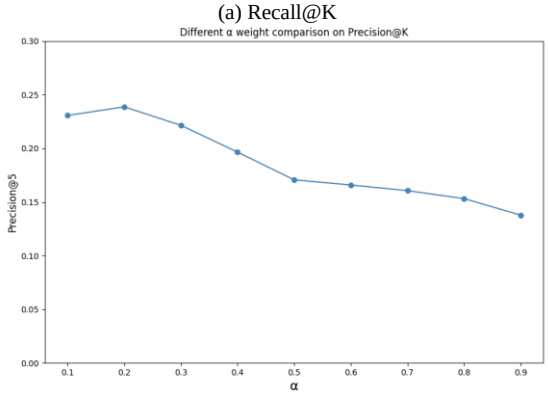
The weight of the geographic distance factor has a significant effect on the performance of the recommendation algorithm. As shown in Fig. 3, the recommendation parameters $K$ and $L$ are set to 5 and 10, respectively. As the geographic weight $\alpha$ increases, *Precision@K* and *Recall@K* gradually decrease, and the recommendation algorithm achieves the best performance when $\alpha$ is 0.2.

### G. L-value Analysis

The $L$ value indicates the critical value of the distance between the real location and the center location of the anonymous region, which has a certain influence on the performance of the recommendation algorithm. The parameter α is set to 0.2. The range of the value of $L$ is set to $[10,60,100,150,300]$. As shown in Fig. 4, with the fixed *Top-K*, the *Precision@K* and *Recall@K* of the recommendation algorithm gradually decrease as the parameter $L$ increases. When the value range of $L$ is $[10,60,100,150]$, the recommendation performance is weakened, but there is no significant impact on the accuracy of the recommendation results. When $K$ is 5, the Recall values when $L$ is 60, 100 and 150 are 2.5%, 6.89%, 15.9% and 38.8% lower than the Recall values when $L$ is 10, and the Precision is 0.9%, 1.2%, 2.3% and 5.8% lower, respectively. Therefore, [10,150] is a reasonable interval for the parameter $L$.

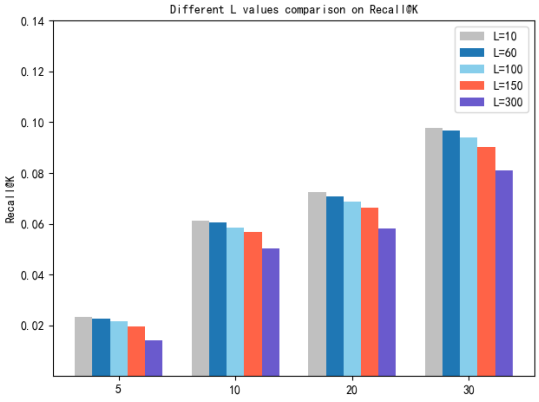



(a) Recall@K

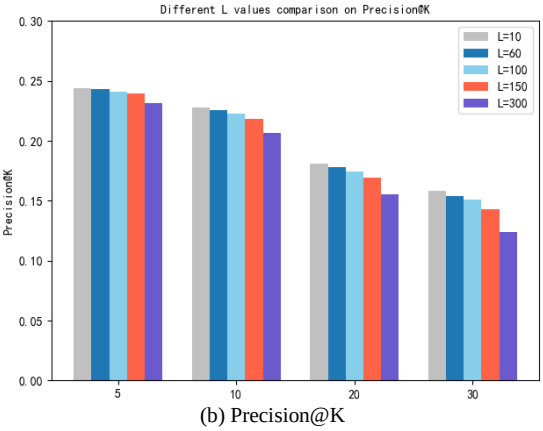



(b) Precision@K

Fig. 3. Recall@K and precision@K of recommendation algorithm.




(a) Recall@K

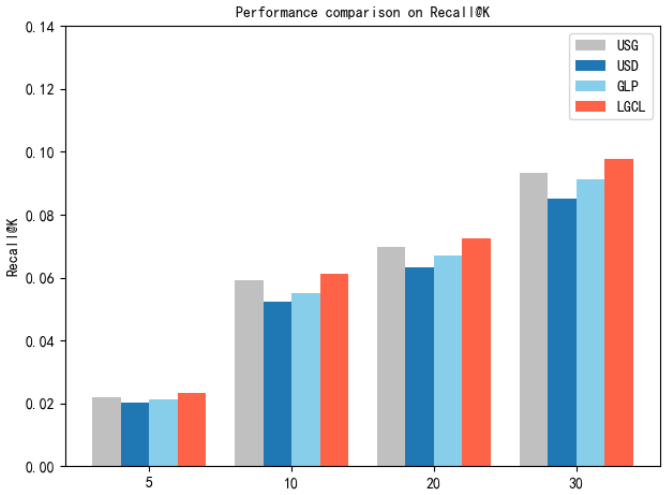



(b) Precision@K

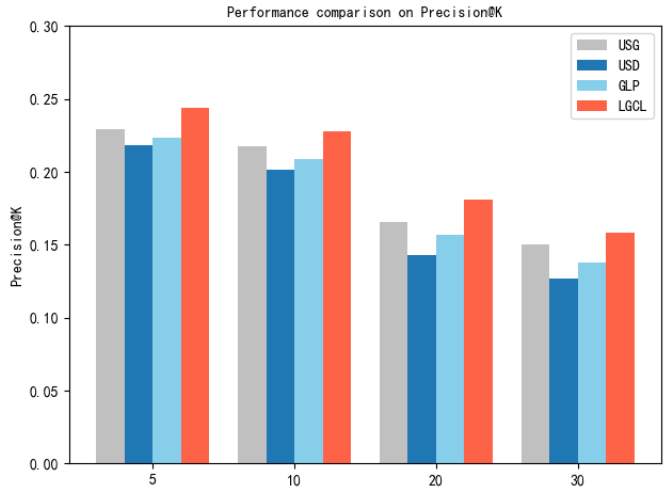
Fig. 4. Recall@K and precision@K of different l values.

### H. Performance Comparison

Fig. 5 shows the recommended performance of Top-K (5,10,20,30) for all methods with $\alpha$ set to 0.2, respectively. Fig. 5(a) shows the *Recall@K* performance and Fig. 5(b) shows the effect of *Precision@K*. The performance of our method degrades as *K* increases. From the figure, it can be seen that the method proposed in this paper outperforms USG, USG and GLP in terms of accuracy and recall. By analyzing Fig. 5, it can be seen that the performance of the recommendation algorithms proposed in this paper are both higher than other comparison algorithms. Although the performance improvement of the method proposed in this paper is small compared with the USG method when *K* is taken as 5 (Recall and Precision are improved by 5.4% and 6.3%, respectively), the method proposed in this paper generalizes the processing for user location, and from the perspective of information entropy, the method in this paper has a better privacy protection ability. Thus a good balance between recommendation performance and privacy protection can be achieved by sacrificing a small recommendation performance.




(a) Recall@K




(b) Precision@K

Fig. 5. Performance comparison of different methods.

## VI. Conclusion

In this paper, we propose a recommendation algorithm that integrates GNN and geographic distance based on geographic location privacy protection. The algorithm learns implicit features from interaction information, comment information, and geographic distance for user recommendations, and integrates k-anonymity and generalization techniques for user location privacy protection. Finally, the information entropy theory analysis and experiments on real datasets show that our proposed recommendation system with integrated location protection can better adapt to interest in LBSN while effectively protecting user location privacy compared to other algorithms point recommendation. Therefore, it is meaningful and feasible to integrate privacy protection in point-of-interest recommendation systems. Since recommendation systems in location services involve geographic location and information records, the next step of research will focus on improving information record protection on the basis of ensuring recommendation performance.

## References

[1] Hao P Y, Cheang W H, Chiang J H. Real-time event embedding for POI recommendation[J]. Neurocomputing, 2019, 349: 1-11.

[2] Jannach D, Manzoor A, Cai W, et al. A survey on conversational recommender systems[J]. ACM Computing Surveys (CSUR), 2021, 54(5): 1-36.

[3] Da'u A, Salim N. Recommendation system based on deep learning methods: a systematic review and new directions[J]. Artificial Intelligence Review, 2020, 53(4): 2709-2748.

[4] Seo S, Huang J, Yang H, et al. Interpretable convolutional neural networks with dual local and global attention for review rating prediction[C]//Proceedings of the eleventh ACM conference on recommender systems. 2017: 297-305.

[5] Sun J, Guo W, Zhang D, et al. A framework for recommending accurate and diverse items using bayesian graph convolutional neural networks[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020: 2030-2039.

[6] Hu K, Wu J, Li Y, et al. Fedgcn: Federated learning-based graph convolutional networks for non-euclidean spatial data[J]. Mathematics, 2022, 10(6): 1000.

[7] Ying R, He R, Chen K, et al. Graph convolutional neural networks for web-scale recommender systems[C]//Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 2018: 974-983.

[8] Wang S, Hu L, Wang Y, et al. Graph learning approaches to recommender systems: A review[J]. arXiv preprint arXiv:2004.11718, 2020.

[9] Wu S, Sun F, Zhang W, et al. Graph neural networks in recommender systems: a survey[J]. ACM Computing Surveys, 2022, 55(5): 1-37.

[10] Yu Z, Wong R K, Chi C H. Efficient role mining for context-aware service recommendation using a high-performance cluster[J]. IEEE transactions on services computing, 2015, 10(6): 914-926.

[11] Wu W, Liu J, Wang H, et al. Secure and efficient outsourced k-means clustering using fully homomorphic encryption with ciphertext packing technique[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 33(10): 3424-3437.

[12] Zhang G, Qi L, Zhang X, et al. Point-of-interest recommendation with user's privacy preserving in an iot environment[J]. Mobile Networks and Applications, 2021, 26(6): 2445-2460.

[13] Wang H, Huang H, Qin Y, et al. Efficient location privacy-preserving k-anonymity method based on the credible chain[J]. ISPRS International Journal of Geo-Information, 2017, 6(6): 163.

[14] Huo Y, Chen B, Tang J, et al. Privacy-preserving point-of-interest recommendation based on geographical and social influence[J]. Information Sciences, 2021, 543: 202-218.

[15] Kim D, Park C, Oh J, et al. Convolutional matrix factorization for document context-aware recommendation[C]//Proceedings of the 10th ACM conference on recommender systems. 2016: 233-240.

[16] Li Z, Shen X, Jiao Y, et al. Hierarchical bipartite graph neural networks: Towards large-scale e-commerce applications[C]//2020 IEEE 36th International Conference on Data Engineering (ICDE). IEEE, 2020: 1677-1688.

[17] Lin S, Runger G C. GCRNN: Group-constrained convolutional recurrent neural network[J]. IEEE transactions on neural networks and learning systems, 2017, 29(10): 4709-4718.

[18] Shafqat W, Byun Y C. Incorporating similarity measures to optimize graph convolutional neural networks for product recommendation[J]. Applied Sciences, 2021, 11(4): 1366.

[19] Wang X, He X, Wang M, et al. Neural graph collaborative filtering[C]//Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. 2019: 165-174.

[20] He X, Deng K, Wang X, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation[C]//Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020: 639-648.

[21] Ye M, Yin P, Lee W C, et al. Exploiting geographical influence for collaborative point-of-interest recommendation[C]//Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. 2011: 325-334.

[22] Song F, Ma T, Tian Y, et al. A new method of privacy protection: random k-anonymous[J]. IEEE Access, 2019, 7: 75434-75445.

[23] Gruteser M, Grunwald D. Anonymous usage of location-based services through spatial and temporal cloaking[C]//Proceedings of the 1st international conference on Mobile systems, applications and services. 2003: 31-42.

[24] Liu J, Jiang X, Zhang S, et al. FADBM: Frequency-aware dummy-based method in long-term location privacy protection[C]//2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS). IEEE, 2019: 384-391.

[25] Ji Y, Gui R, Gui X, et al. Location privacy protection in online query based-on privacy region replacement[C]//2020 10th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, 2020: 0742-0747.

[26] NING Xueli, LUO Yonglong, XING Kai, ZHENG Xiaoyao. Frequent location privacy-preserving algorithm based on geosocial network[J]. Journal of Computer Applications, 2018, 38(3): 688-692.