

# A Diffusion model for POI recommendation

Yifang Qin

[qinyifang@pku.edu.cn](mailto:qinyifang@pku.edu.cn)

National Key Laboratory for  
Multimedia Information Processing,  
Peking University  
Beijing, China

Hongjun Wu

[dlmao3@stu.pku.edu.cn](mailto:dlmao3@stu.pku.edu.cn)

School of EECS,  
Peking University  
Beijing, China

Wei Ju

[juwei@pku.edu.cn](mailto:juwei@pku.edu.cn)

National Key Laboratory for  
Multimedia Information Processing,  
Peking University  
Beijing, China

Xiao Luo

[xiaoluo@cs.ucla.edu](mailto:xiaoluo@cs.ucla.edu)

Department of Computer Science,  
University of California  
Los Angeles, USA

Ming Zhang

[mzhang\\_cs@pku.edu.cn](mailto:mzhang_cs@pku.edu.cn)

National Key Laboratory for  
Multimedia Information Processing,  
Peking University  
Beijing, China

## ABSTRACT

Next Point-of-Interest (POI) recommendation is a critical task in location-based services that aim to provide personalized suggestions for the user's next destination. Previous works on POI recommendation have laid focused on modeling the user's spatial preference. However, existing works that leverage spatial information are only based on the aggregation of users' previous visited positions, which discourages the model from recommending POIs in novel areas. This trait of position-based methods will harm the model's performance in many situations. Additionally, incorporating sequential information into the user's spatial preference remains a challenge. In this paper, we propose Diff-POI: a Diffusion-based model that samples the user's spatial preference for the next POI recommendation. Inspired by the wide application of diffusion algorithm in sampling from distributions, Diff-POI encodes the user's visiting sequence and spatial character with two tailor-designed graph encoding modules, followed by a diffusion-based sampling strategy to explore the user's spatial visiting trends. We leverage the diffusion process and its reversed form to sample from the posterior distribution and optimized the corresponding score function. We design a joint training and inference framework to optimize and evaluate the proposed Diff-POI. Extensive experiments on four real-world POI recommendation datasets demonstrate the superiority of our Diff-POI over state-of-the-art baseline methods. Further ablation and parameter studies on Diff-POI reveal the functionality and effectiveness of the proposed diffusion-based sampling strategy for addressing the limitations of existing methods.

## CCS CONCEPTS

- Information systems → Recommender systems.

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/XXXXXXX.XXXXXXX>

## KEYWORDS

Next POI Recommendation, Graph Neural Network, Diffusion Model

### ACM Reference Format:

Yifang Qin, Hongjun Wu, Wei Ju, Xiao Luo, and Ming Zhang. 2018. A Diffusion model for POI recommendation. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 14 pages. <https://doi.org/XXXXXXX.XXXXXXX>

## 1 INTRODUCTION

The prosperity of Location-Based Social Networks (LBSN) and location-based services apps has led to a surge of interest in building Point-of-Interest (POI) recommender systems, which provide users with personalized and location-aware recommendations on various services and products. POI recommendation models have become the core of numerous check-in apps, such as Yelp and Foursquare. The location-based apps record the check-ins made by specific users, and recommend the most appropriate POIs to users to alleviate the overloaded information provided by online platforms. Unlike other sequential recommendation tasks, POI recommendation focuses on modeling the geographical feature of the POIs, and how their locations influence user behavior [19, 22, 42].

The success of POI recommendation depends on accurately determining which POI a specific user is most likely to visit based on their previous behavior. This requires joint modeling of both locality features and temporal patterns of the user's visiting sequences. There has been a variety of attempts to find the ideal method to incorporate this spatio-temporal feature, such as distance between POIs [5, 19, 45], or time gap between successive visits [14, 44]. Above that, existing works proposed different methods to encode the observed sequences, such as Markov Chains [2, 8]. Recurrent Neural Networks (RNNs) and their variants are also widely adopted by previous works [4, 6, 33, 44] and benefited from their capability of capturing the sequential dependencies and long-term preference from visiting sequences. With the development of Graph Neural Networks (GNNs) in recent years, the expressiveness and rich syntax of graph-structured data raises concerns among researchers. Lots of graph-based methods have been proposed [7, 17, 21, 38] to depict the transition and location relationship between POIs.

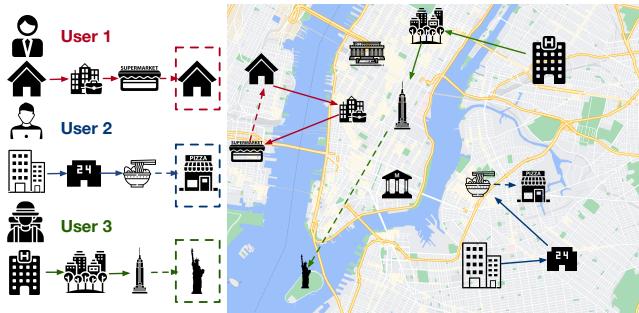


Figure 1: A toy example of different visiting patterns..

Despite the variety of model structures, how to leverage the obtained sequence encodings remains a fundamental problem. The sequential representations contain the temporal and spatial information from the user’s previous behavior, yet it is still unclear where the user would move on to the next. Some contrastive learning methods are proposed [12] to encourage consistency between the two representations, while other works propose to disentangle the sequential and spatial influences [26, 37].

Although the effectiveness of existing works, an important yet common situation is neglected, which is when the user tries to explore POIs in a novel, hasn’t visited area. For historical visits, a common practice is to aggregate the corresponding locational representations via a pooling method or through the attention mechanism [18, 23, 37]. These methods successfully summarize the geographical feature of visiting history and tend to recommend nearby POIs for the user [26], which will fail for users who want to explore novel areas. As illustrated in the toy example in Figure 1, there are several types of users that prefer diversified visiting tour routes. User 1 is a typical conservative visitor, who is more likely to revisit the same POIs or POIs in familiar areas, while user 2 and 3 are more adventurous users. User 2 is a curious traveler that loves to step out of his/her comfort zone from time to time, and User 3 represents a long-distance tourist who is touring around novel areas in the city. Though there are diversified visiting patterns along users, an aggregation-based representation learning method could only collect and aggregate the historical geographical information, and make recommendations based on the locational similarity between POIs. This trivial method could achieve promising results for cases with regular routines like User 1, yet will lead to sub-optimal solutions for users with adventurous spirits or tourists who want to explore the city, like User 2 and User 3. An effective method to determine whether to visit novel areas remains unexplored.

The rapid development of diffusion-based methods since the Denoising Diffusion Probabilistic Model (DDPM) [10, 29] is attracting increasing attention. By modeling the diffusion process with Stochastic Differential Equations (SDEs), the reverse diffusion process can be viewed as a Langevin sampling process from a given prior distribution [31]. Further studies on score-matching and Langevin dynamics bring more insights into the application of existing diffusion algorithms. Notably improvements against traditional methods have been achieved in other fields, like image editing [24, 43], natural language processing [16], and graph generation [34, 41]. The

combination of diffusion process and score-based models empowers models to sample from any distributions modeled by specific score functions, which extends the application of diffusion-based methods to more than generative tasks.

Inspired by the idea of Langevin sampling from specific distributions, we propose to address the aforementioned problem in location-based POI recommendation with a tailored-designed geo-preference diffusion process. In a nutshell, we propose **Diff-POI**: a **D**iffusion-based model that samples the user’s spatial preference for the next **P**OI recommendation. Specifically, we design an attention-based graph encoder to integrate spatio-temporal information with traditional sequential graphs to obtain fine-grained user embeddings. Moreover, a graph convolution module is applied for generating geographical representations for POIs based on their locations. Subsequently, we adopt a context-conducted attention layer to obtain location prototypes for corresponding users. Finally, a score function is applied to model the gradient of the posterior distributions, which are parameterized by location prototypes. From the posterior distributions, we sample the target location embeddings via a reversed diffusion process. The location and user embeddings are jointly considered to make POI recommendations. Empirical studies show the superiority of Diff-POI against state-of-the-art baseline methods.

To summarize, the contributions of our work are listed as follows:

- We propose an attention-based graph encoder that extends traditional sequential GNNs by integrating extra spatio-temporal information from observed visiting sequences. The novel sequence graph encoder can generate fine-grained embeddings for users.
- We propose Diff-POI, a diffusion-based model that samples from the posterior distribution that reflects the user’s geographical preference. With the help of the diffusion process, Diff-POI is able to fully exploit the potential of the obtained location and user embeddings.
- Comprehensive experiments on four real-world LSBN datasets demonstrate not only the effectiveness and robustness of the proposed Diff-POI against the state-of-the-art baselines but also its capability of depicting different locational visiting patterns.

## 2 RELATED WORK

In this section, we will introduce the related works from two aspects, namely the next POI Recommendation, graph neural networks, and SDEs with denoising diffusion.

### 2.1 Next POI Recommendation

The next POI recommendation lies at the core of numerous location-based services, which aim to recommend the most possible POI for the user to visit. Previous attempts manage to integrate the spatio-temporal information with existing recommendation methods, such as matrix factorization [25, 28] and Markov chains [2, 8]. Further studies extend the model by explicitly considering the temporal or locational similarity between POIs [20, 35].

With the development of sequential recommendation models and deep learning methods, researchers have put more attention on deep learning-based methods to model the sequential evolution in POI recommendation. Recurrent Neural Networks (RNNs) and their variants are widely used to encode visiting sequences. ST-RNN [22]

proposes to use different transition matrices to model contexts from different perspectives in RNNs. STGN [44] introduces two extra time and distance gates to LSTM structures to better capture the spatio-temporal gap in visiting sequences. LSTPM [33] proposes to model the temporal and geographical sequences separately, by encoding POI sequences with a standard LSTM and a geo-dilated LSTM correspondingly. Apart from improving RNN structures, the attention mechanism is also widely adopted by recent works to discover multiple and flexible factors behind visits, such as ARNN [6]. There are also attempts to leverage transformer-like self-attention structure for POI recommendation tasks. GeoSAN [18] equips the self-attention layer with a geography encoder to encode locations with GPS coordinates. STAN [23] further improves the location encoder by leveraging relative spatio-temporal information in user trajectories.

## 2.2 Graph Neural Networks

A popular trend in recommender system research is the application of Graph Neural Network (GNN) structures [11]. Since the success of graph convolution networks [13], graph-structured data in POI recommendation tasks have been widely studied. GE [40] collects information from various kinds of POI and user graphs. STGCN [7] exploits the idea of the Relational Graph Convolution Network (RGCN) to depict the multiple relationships between users and POIs. Later works like GSTN [38] lay their focus on the transition and distance POI graphs. DisenPOI [26] further proposes to disentangle the geographical and sequential effects with a contrastive regularization to fully exploit the different factors behind visits.

Generally, existing graph-based methods all suffer from the limitations brought by the location embeddings learned from the POI distance graph, and models tend to recommend nearby POIs as previous researches suggested [12, 26].

## 2.3 SDEs with Denoising Diffusion

Diffusion models [10, 29] have achieved great success in multiple fields. Further researches reveal the relationship between the denoising diffusion process and score-based methods [30, 31], which bridge the gap between two generative paradigms. Models are using SDEs to calculate and simulate the Langevin sampling process from specific diffusion process for image generation [31] and are gaining great advantage against traditional generative methods such as GANs [3]. The diffusion process is also viewed as transportation between distributions and can be used in domains including image edit [24, 43], natural language processing [16], audio generation [15] and graph generation [34, 41]. However, other applications of the denoising diffusion process remains unexplored.

In a nutshell, the strong capability of denoising diffusion process and the view of Langevin sampling widely extend the application of diffusion-based models. Inspired by its success in other fields, we believe learning SDEs with the denoising diffusion process is the key to exploiting the rich syntax from historical location embeddings.

## 3 PRELIMINARY

In this section, we first define the notations and formulate the POI recommendation problem, then introduce the main idea of

constructing spatio-temporal dual graphs, followed by the basis of the SDE-based diffusion process.

### 3.1 Sequential POI Recommendation

Considering the typical settings of POI recommendation. For the POI set  $\mathcal{L} = \{l_1, l_2, \dots, l_{|\mathcal{L}|}\}$  and user set  $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ , the task can be formulated as: given a user  $u$ , its trajectory of POIs and visiting timestamps  $H(u) = \{(l_1^u, t_1^u), (l_2^u, t_2^u), \dots, (l_n^u, t_n^u)\}$ , the target is to recommend target POI  $l_{n+1}^u$  at current timestamp  $t_{n+1}^u$ .

In the case of sequential POI recommendation, the temporal and distance gap between visits is an important factor to make recommendations. For POI set  $\mathcal{L}$ , we can obtain the distance matrix  $A_d \in \mathbb{R}^{N \times N}$ , where  $A_d(i, j) = \text{haversine}(l_i, l_j)$  denotes the distance between  $l_i$  and  $l_j$  in km, calculated by haversine formula.

### 3.2 Spatio-temporal Dual Graph

Now we construct dual graph pair for each visiting trajectory to depict the spatial and temporal relationship between POIs respectively. For a trajectory  $H(u)$ , we obtain the corresponding transition graph  $\mathcal{G}_u = \{\mathcal{V}_u, \mathcal{E}_u\}$ , where vertex set  $\mathcal{V}_u$  includes all the POIs in  $H(u)$ , each edge  $e = \langle l_i^u, l_{i+1}^u \rangle \in \mathcal{E}_u$  indicates a successive visit from  $l_i^u$  to  $l_{i+1}^u$  in user's visiting history  $H(u)$ .

To model the locational influence, we obtain the distance graph  $\mathcal{G}_d = \{\mathcal{V}_d, \mathcal{E}_d, \mathcal{A}_d\}$  for all POIs, where node set  $\mathcal{V}_d = \mathcal{L}$ . The distance edge  $(l_i, l_j) \in \mathcal{E}_d$  indicates the distance between  $l_i$  and  $l_j$  is within a specific threshold, e.g. 1km as previous works suggested [38] and the edge weight  $\mathcal{A}_d$  represents the geographical distance.

### 3.3 SDE-based Diffusion Process

Consider a data point sampled from a specific prior  $x_0 \sim p_{data}(x)$ . Given a series of noise scales  $\beta_0, \beta_1, \dots, \beta_T$ , the discrete diffusion process is considered as a Markov Chain with the transition probability and the conditional probability is formulated as:

$$p(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t I). \quad (1)$$

More generally, we can formulate any discrete one-step diffusion into continuous form and reform the process as the solution to specific Itô SDE [31]:

$$dx = f(g, t)dt + g(t)dw, \quad (2)$$

through which we can sample arbitrary  $x(t) \sim p_t$ . As previous research stated [1], the reverse process of Eq. 2 is also a diffusion process that can be formulated as:

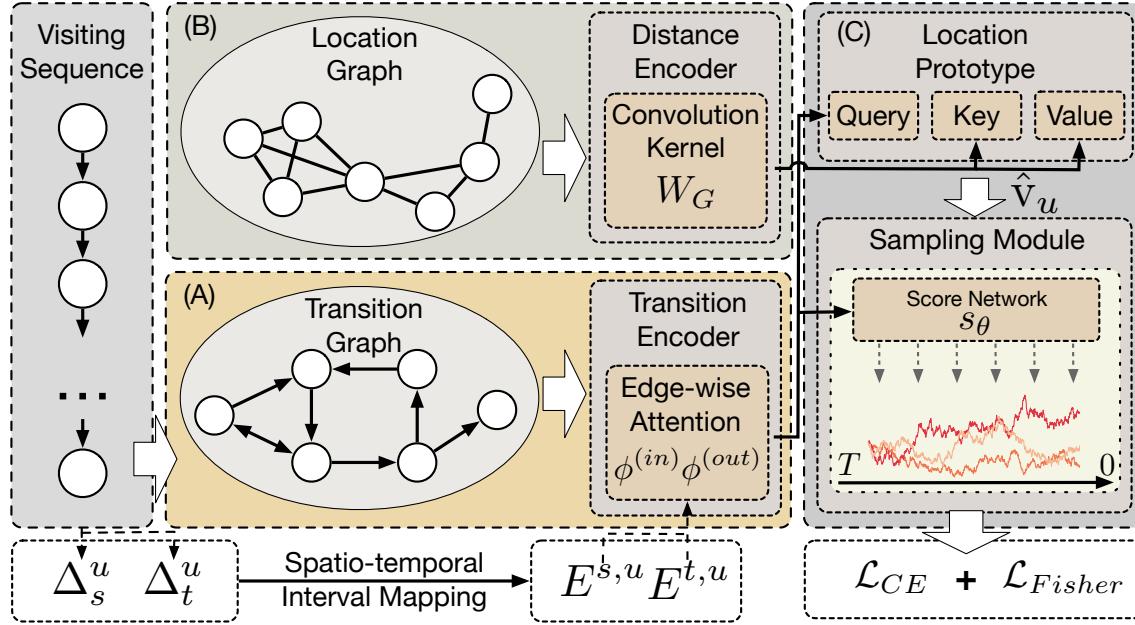
$$dx = [f(x, t) - g^2(t) \nabla_x \log p_t(x)]dt + g(t)d\bar{w}. \quad (3)$$

Where the  $dw$  and  $d\bar{w}$  in Eq. 2 and Eq. 3 are sampled independently from standard Wiener process at each step of the SDEs.

With a specified score function  $s_\theta(x)$  that is parameterized by a neural network  $\theta$ , we can estimate the gradient of the log marginal distribution  $\nabla_x \log p_t(x)$  and sample from any  $p_t$  for the target data point  $x_0$  via the reversed SDE in Eq. 3.

## 4 METHODOLOGY

In this section, we provide the detailed structure of the proposed Diff-POI. As illustrated in Figure 2, Diff-POI is composed of three key modules, which are: (A) a spatio-temporal sequence graph



**Figure 2: Illustration of the proposed framework Diff-POI.** Diff-POI is intended for making personalized POI recommendations based on the user’s spatial preference and history visiting patterns and is composed of three main parts: (A) a spatio-temporal graph encoder, (B) a distance-based graph convolution encoder, and (C) a diffusion-based sampling module.

encoder that incorporates the distance and time gap between successive visits with sequential graph encodings, (B) a distance-based POI graph encoder that performs graph convolution on the distance graph that yields location embeddings for each POI, and (C) a context-driven diffusion module which obtains user-specific locational prototypes, followed by a diffusion-based sampling strategy to sample the spatial preference for users.

#### 4.1 User Preference Encoding

**4.1.1 Spatio-temporal Embedding Layer.** The transition graph  $\mathcal{G}_u$  reflects the sequential relationship between POIs within a user’s visiting history, making it important for the sequence graph encoder to encode the extra spatio-temporal information between successive visits. An intuitive observation is that the interval between successive visits implies the relationship between these two visits. For instance, a series of tight visits reflect the trajectory of one single trip, long distance between two visits indicates the user interests may have changed. Thus we expect to find a way to model the intervals between visits from both spatial and temporal perspectives. For a specific transition graph  $\mathcal{G}_u$  and its corresponding visiting sequence  $H(u)$ ,  $|H(u)| = n$ , we obtain the relative spatial and temporal interval matrices  $\Delta_s^u, \Delta_t^u \in \mathbb{R}^{n \times n}$ :

$$\Delta_s^u = \begin{bmatrix} s_{11}^u & s_{12}^u & \dots & s_{1n}^u \\ s_{21}^u & s_{22}^u & \dots & s_{2n}^u \\ \dots & \dots & \dots & \dots \\ s_{n1}^u & s_{n2}^u & \dots & s_{nn}^u \end{bmatrix}, \quad \Delta_t^u = \begin{bmatrix} t_{11}^u & t_{12}^u & \dots & t_{1n}^u \\ t_{21}^u & t_{22}^u & \dots & t_{2n}^u \\ \dots & \dots & \dots & \dots \\ t_{n1}^u & t_{n2}^u & \dots & t_{nn}^u \end{bmatrix}, \quad (4)$$

where  $s_{ij}^u = \lfloor \frac{A_d(t_i^u, t_j^u)}{s_{min}^u} \rfloor$  is the relative distance between the  $i$ -th and  $j$ -th POI in  $H(u)$ , normalized by the minimal transition distance

in the visiting sequence  $s_{min}^u$ . Similarly, the temporal intervals are defined with  $t_{ij}^u = \lfloor \frac{t_j^u - t_i^u}{t_{min}^u} \rfloor$ .

We maintain two trainable embedding matrices to represent the intervals respectively, after clipping the relative interval matrices  $\Delta_s^u, \Delta_t^u$  with two threshold values  $\delta_s, \delta_t$ :

$$s_{ij}^u = \max(s_{ij}^u, \delta_s), \quad t_{ij}^u = \max(t_{ij}^u, \delta_t), \quad \forall i, j, \quad (5)$$

where  $\delta_s$  and  $\delta_t$  are hyper-parameters. The intervals are further transferred into trainable embedding vectors via interval embedding matrices  $E^s \in \mathbb{R}^{\delta_s \times d}, E^t \in \mathbb{R}^{\delta_t \times d}$ . We can obtain the interval embeddings for arbitrary interval matrices via an interval mapping operation:

$$e_{ij}^{s,u} = E_{s_{ij}^u,:}^s, \quad e_{ij}^{t,u} = E_{t_{ij}^u,:}^t, \quad (6)$$

where  $e_{ij}^{s,u}, e_{ij}^{t,u} \in \mathbb{R}^d$  are interval embeddings,  $d$  represents the embedding size.

**4.1.2 Spatio-temporal Sequence Graph Encoder.** Now we design the attention-based graph encoder to process the sequence graphs. For input sequence graph and POI embeddings  $e_i^l \in \mathbb{R}^d, i = 1, 2, \dots, |\mathcal{L}|$ , the graph encoder should output the user encoding  $x_u \in \mathbb{R}^d$  for the corresponding user  $u$ . A simple solution is to apply variants of Gate-controlled GNNs (GGNNs) to aggregate the sequence information. However, such a method only receives messages from precursor nodes when generating node representations. Therefore, we propose a tailor-designed attention-based graph encoder to encode spatio-temporal information from both directions on the transition graph.

Since bi-directional information is equally important for sequential recommendation cases [27, 32], we expect the message function

could reflect both the feature from previous visits and future choices. Specifically, the message function is applied to receive from incoming and outgoing edges to collect messages from both directions.

$$m_i = m_i^{(in)} + m_i^{(out)} = \sum_{\langle l_j, l_i \rangle \in \mathcal{E}_u} \alpha_{ij}^{(in)} e_j^l + \sum_{\langle l_i, l_k \rangle \in \mathcal{E}_u} \alpha_{ik}^{(out)} e_k^l, \quad (7)$$

the coefficients  $\alpha_{ij}^{(in)}$  and  $\alpha_{ik}^{(out)}$  are attention weights of the corresponding neighbors. We calculate them with an edge-wise attention layer, formulated as:

$$\begin{cases} a_{ij}^{(in)} = \phi^{(in)}(e_i^l \odot e_j^l + e_{ij}^{s,u} + e_{ij}^{y,u}) \\ a_{ik}^{(out)} = \phi^{(out)}(e_i^l \odot e_k^l + e_{ik}^{s,u} + e_{ik}^{y,u}) \\ \alpha_{ij} = \text{Softmax}(\text{CONCAT}([a_{ij}^{(in)}, a_{ik}^{(out)}])), \end{cases} \quad (8)$$

where  $\phi^{(in)}, \phi^{(out)} : \mathbb{R}^d \rightarrow \mathbb{R}$  are two trainable projection matrices to obtain the attention logits,  $\odot$  denotes element-wise product to model the affinity between neighboring nodes.

So far, with the message function defined in Eq. 7, we can iteratively update the node representation in  $\mathcal{G}_u$  with the weighted neighborhood influence, and the outputs noted as  $E_u^h = [e_1^h, e_2^h, \dots, e_n^h]$ . To effectively model the user's historical preference, we propose to leverage a self-attention as the readout function for the encoded node representations of sequence graph  $\mathcal{G}_u$ , noted as  $x_u \in \mathbb{R}^d$ :

$$x_u = \text{MEAN}\{\text{Attention}(W_Q^h E_u^h, W_K^h E_u^h, W_V^h E_u^h)\}, \text{with} \quad (9)$$

$$\text{Attention}(Q, K, V) \triangleq \text{FFN}(V + \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V) \quad (10)$$

where  $W_Q^h, W_K^h, W_V^h : \mathbb{R}^d \rightarrow \mathbb{R}^d$  are learnable projection matrices, FFN indicates a layer of feed-forward neural network. While the sequence graph encoder depicts the local interval features between adjacent nodes in  $\mathcal{G}_u$ , the self-attention readout layer captures the long-term global dependencies between visits to further generate fine-grained sequence embeddings.

## 4.2 User-specific Spatial Posterior

Recall that a POI geographical graph  $\mathcal{G}_g$  is constructed based on the distance between POIs, we consider a graph convolution module that encodes  $\mathcal{G}_g$  and outputs the location embeddings of POIs, marked as  $e_i^g \in \mathbb{R}^d$  for any POI  $l_i$ . We further obtain the user-specific spatial posterior based on  $e^g$  and  $x_u$  to sample the locational preference from.

**4.2.1 GNN-based Location Graph Encoder.** To extract the locational relationship between POIs from  $\mathcal{G}_g$ , we adopt a graph convolution network [13] as the geographical graph encoder. Specifically, the update function of each layer is built with:

$$h_i^{(l)} = \sum_{j \in N_i} \frac{w_{ij}}{\sqrt{|N_i||N_j|}} W_G^{(l-1)} h_j^{(l-1)}, \quad (11)$$

where the node representation  $h_i^0$  is initialized with  $e_i^l$  for each node,  $W_G^{(l)}$  represents the convolution kernel of  $l$ -th layer. To dynamically simulate the attenuation of geographical influence as the distance increases, we adapt  $w_{ij} = e^{-A_d(l_i, l_j)}$  as the distance edge weight. Benefiting from GCN's capability of aggregating and smoothing neighborhood information, we expect the updated node

representation  $h^{(l)}$ 's to reflect the implicit communities and local structures of POIs based on their locational distribution. With the fixed graph topology of  $\mathcal{G}_g$  and the number of convolution layers  $L$ , we readout the locational embeddings  $E^g = [e_1^g, e_2^g, \dots, e_{|\mathcal{L}|}^g]$  with a mean pooling function on the outputs of each layer to capture the high-order connectivity between nodes:

$$e_i^g = \text{MEAN}\{h_i^{(0)}, h_i^{(1)}, \dots, h_i^{(L)}\}, i = 1, 2, \dots, |\mathcal{L}|. \quad (12)$$

**4.2.2 User-specific Location Prototype.** Since we have obtained the locational embeddings for POIs, we aim to figure out a method to calculate the probability  $p(l = l_i | H(u), \mathcal{G}_g)$  for arbitrary user  $u$  to visit  $l_i$  according to  $u$ 's locational preference. To achieve this, we parameterize the probability with a user-specific prototype, formulated as:

$$p(l | H(u), \mathcal{G}_g) = p(l | v_u) p(v_u | H(u), \mathcal{G}_g), \quad (13)$$

where  $v_u \in \mathbb{R}^d$  denotes the geographical prototype for the user  $u$ . The first term of Eq. 13 can be further interpreted as:

$$p(l = l_i | v_u) = \text{sim}(e_i^g, v_u), \quad (14)$$

$\text{sim}(\cdot, \cdot)$  is a similarity function and here we adopt the inner product. Now we only need to sample  $v_u$  from the posterior distribution  $p(v_u | H(u), \mathcal{G}_g)$  to calculate the probability in Eq. 13.

We initialize the prototype  $v_u$  to assist and accelerate the sampling process from the posterior. Directly apply mean pooling on visited POIs could be a simple solution, yet it neglects the user's own characteristics. We expect the generated prior could reflect the user's preference when exploring areas and make it able to bring more personalized information. To achieve this, we leverage the target attention mechanism to initialize the prototype  $v_u$ , before further feeding it into the sampling module. In particular, the user encoding  $x_u$  is denoted as the key vector to calculate the relativity of each POIs in  $u$ 's visiting history. The initial of  $v_u$  is formulated as:

$$\hat{v}_u = \text{Attention}(W_Q^g x_u, W_K^g E_{H(u)}^g, W_V^g E_{H(u)}^g), \quad (15)$$

where  $W_Q^g, W_K^g, W_V^g$  are trainable parameters.

## 4.3 Diffusion-based Sampling Module

To sample from the posterior distribution in Eq. 13, a common practice is to use stochastic gradient Langevin dynamics [39] to iteratively update the sampled parameter. However, the gradient of log probability density, i.e. the score of the posterior, in our case is intractable to calculate. Inspired by the Score-based SDEs [31], we propose to estimate the score with a score function and model the sampling process with the Variance Preserving (VP) SDE. Though diffusion processes with other perturbation kernels may achieve better performance, experimental results demonstrate that VP-SDE already shows promising results in modeling spatial preferences. Details of the proposed sampling module are illustrated in figure 3.

Additionally, the proposed diffusion-based sampling strategy can be adapted to any other POI recommendation frameworks with explicit POI geographical encoders and used for generating an expressive locational prototype with enriched information on the user's spatial performance.

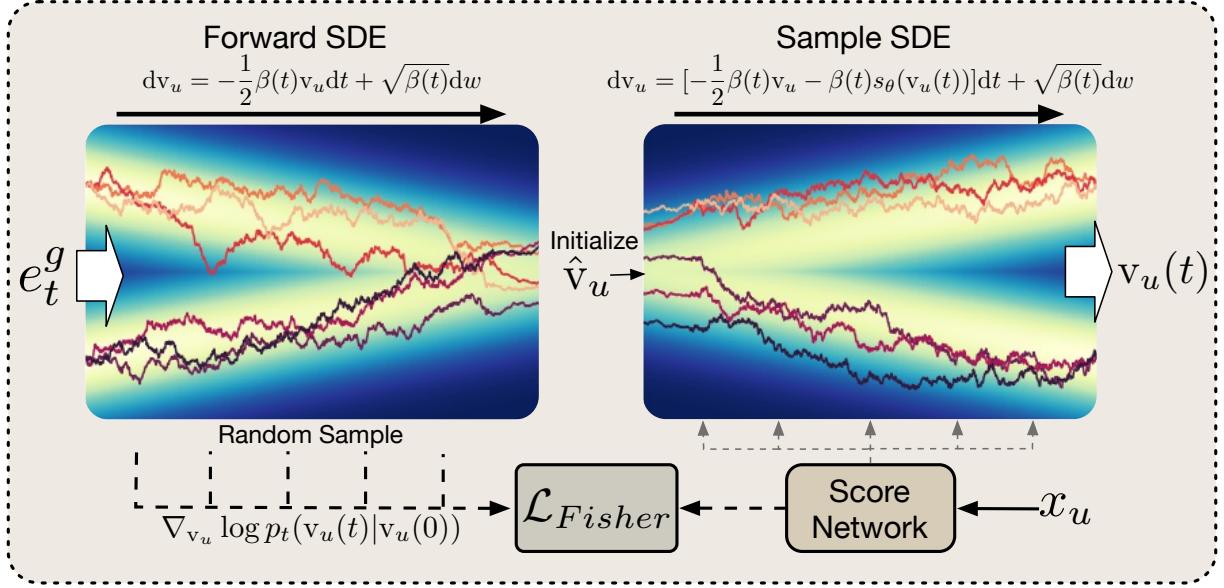


Figure 3: Illustration of the diffusion-based sampling strategy.

**4.3.1 Forward SDE for Denoising Diffusion.** The aim of the sampling module is to sample the locational embedding of target POI from the user-specific spatial posterior, marked with  $e_t^g$  which represents the location embedding of the ground truth target POI  $l_t$ . In other words, the continuous diffusion process from time  $0 \sim T$  starts with  $v_u(0) = e_t^g$  and ends with  $v_u(T) = \hat{v}_u$ . Specifically, we model the diffusion process with the following SDE:

$$dv_u = -\frac{1}{2}\beta(t)v_u dt + \sqrt{\beta(t)}dw, \quad (16)$$

here we parameterize the noise schedule  $\beta(t) = \beta_{min} + \frac{(\beta_{max} - \beta_{min})t}{\beta_{min}T}$  with hyper-parameters  $\beta_{min}, \beta_{max}$ , and  $w$  is sampled from a standard Wiener Process at each step of the SDE.

So far, by solving the Itô SDE defined by Eq. 16 starting from the target data point  $v_u(0) = e_t^g$ , we can obtain the perturbed data, which in our case is the denoised spatial posterior.

**4.3.2 Reversed SDE for Preference Sampling.** The sampling process of the location prototype  $v_u$  from the spatial posterior  $p(v_u|H(u)\mathcal{G}_t)$  is equivalent to the reverse process of Eq. 16. As previous research concluded [1], the reverse SDE is formulated with:

$$dv_u = [-\frac{1}{2}\beta(t)v_u - \beta(t)\nabla_{v_u} \log p_t(v_u)]dt + \sqrt{\beta(t)}dw. \quad (17)$$

Specifically, the spatial preference for  $u$  can be sampled with the reversed SDE in Eq. 16 and we are expecting that the sampled  $v_u(0)$  should be similar to the location embedding of the ground-truth target POI, i.e.  $e_t^g$ , thus the score function we used should be able to finely estimate real scores.

However, note that the gradient of the log probability, i.e. the score, is intractable to calculate, thus we estimate the score with a score network  $s_\theta(v_u(t))$  parameterized by  $\theta$ :

$$s_\theta(v_u(t)) = \text{MLP}(\text{CONCAT}(v_u(t), x_u)). \quad (18)$$

We model the scoring network with a multi-layer perceptron and take the concatenation of user historical embedding and the prototype as input. By introducing the user context condition  $x_u$  to the score function, the sampling process is finely controlled by the user's historical behavior, which ensures the sampled prototype fits the user-specific location preference.

The score function is optimized by the Fisher divergence between  $s_\theta$  and the score of real samples as previous work suggested [30]:

$$\mathcal{L}_{Fisher} = \mathbb{E}_t[\|s_\theta(v_u(t)) - \nabla_{v_u} \log p_t(v_u(t)|v_u(0))\|_2^2], \quad (19)$$

where  $t \sim U(0, T)$  is uniformly sampled. With  $\mathcal{L}_{Fisher}$  as an auxiliary loss, the  $s_\theta$  is optimized to better depict the gradient field of the posterior distribution for any given user.

#### 4.4 Model Inference and Optimization

**4.4.1 Target POI Prediction.** Recall that we have obtained two representative embeddings, i.e.  $x_u$  and  $v_u(0)$ , that reflect a specific user's transition history and spatial preference respectively. To comprehensively consider the user's preference for the next-to-visit POI, the probability for user  $u$  to visit POI  $l_i$  is formulated with:

$$\hat{y}_i^u = \text{Softmax}(\alpha e_i^L x_u^T + (1 - \alpha) e_i^g (v_u(0))^T), \quad (20)$$

where  $\alpha$  is a weight coefficient that reflects the importance of the two similarity terms. By default, we set  $\alpha = 0.5$  and value these two factors with equal importance.

**4.4.2 Model Optimization.** Given the user's visiting history  $H(u)$  and the corresponding ground truth POI  $y_i^u$ , the model is optimized with a supervised cross-entropy loss:

$$\mathcal{L}_{CE} = -\frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} y_i^u \log(\hat{y}_i^u) + \lambda \|\Theta\|_2^2, \quad (21)$$

where  $\Theta$  represents model parameters and  $\lambda$  is the corresponding weight for the L2 regularization term.

**Table 1: Descriptive statistics of the used datasets.**

Dataset	#User	#POI	#Interaction	#Avg. Visit
Gowalla	10162	24237	456820	44.95
SIN	2321	5596	194108	83.63
TKY	2293	15177	494807	215.79
NYC	1083	9989	179468	165.71

Recall that the score function in the diffusion-based sampling module is optimized via an extra Fisher loss, the overall loss function of Diff-POI is further written as:

$$\mathcal{L}_{Total} = \mathcal{L}_{CE} + \gamma \mathcal{L}_{Fisher}, \quad (22)$$

where  $\gamma$  is a hyper-parameter to balance the optimization targets.

## 5 EXPERIMENT

To evaluate the performance of our Diff-POI and the effectiveness of the proposed ideas, we conduct a series of experiments and empirical studies on four real-world LBSN datasets. The aim of the experiments is to answer the following research questions.

**RQ1:** Compared with the current state-of-the-art baseline methods, how well does the proposed Diff-POI perform? Is sampling the user’s spatial preference with the diffusion process an effective method for POI recommendation?

**RQ2:** How do the proposed sequence graph encoder and the diffusion-based sampling module boost the performance of Diff-POI? Is the model sensitive to hyper-parameters?

**RQ3:** Can Diff-POI correctly clarify different types of spatial preference? How does the diffusion-based sampling strategy respond to different users?

### 5.1 Datasets and Experimental Setup

**5.1.1 Evaluation Datasets.** We evaluate the proposed Diff-POI on four datasets collected from two real-world check-in platforms, namely **Gowalla**<sup>1</sup> and **Foursquare**<sup>2</sup>. The Gowalla dataset contains users’ check-in records on the Gowalla website from February 2009 to October 2010. The Foursquare dataset includes three subsets, which are collected from Singapore, Tokyo, and New York respectively. The Singapore subset is collected from August 2010 to July 2011, and the rest two subsets are from 12 April 2012 to 16 February 2013.

Following previous work [37], we process the aforementioned datasets with the 5-core cleaning strategy, which means the users and POIs with less than 5 visits are filtered out of the dataset. The detailed statistics of the used datasets are listed in Table 1. All the check-in records are sorted chronologically and split by the ratio of 80%, 10%, 10% into train, valid and test set respectively, which is a common practice in next POI recommendation [33, 37]. We train and tune all models on the train set, and choose the best epoch on the validation set as the model to be tested on the test set and report the experimental results.

<sup>1</sup><http://snap.stanford.edu/data/loc-gowalla.html>

<sup>2</sup><https://sites.google.com/site/yangdingqi/home/foursquare-dataset>

**5.1.2 Compared Baselines.** To comprehensively evaluate and demonstrate the effectiveness of Diff-POI, we compare its performance with the following baseline methods from three aspects: (a) traditional graph-based user-item recommendation methods, (b) sequence-based POI recommendation methods that considers spatial factors in recommendation, and (c) graph-based POI recommendation methods. The compared baselines and their brief introductions are listed as follows:

- (a) MF [25]: It is a classical collaborative filtering method that based on low-rank matrix factorization and optimized with gradient decent method.
- (a) LightGCN [9]: It is a CF method that adopts graph convolution networks to node representation learning for recommendation.
- (a) DGCF [36]: It is a variant of LightGCN that proposes to disentangle the node representations for better performance.
- (b) GeoIE [35]: It is a classical POI recommendation method that learns geographical and interaction influence separately.
- (b) LSTPM [33]: It is a state-of-the-art LSTM-based POI recommendation method that models the visiting trajectories with a nonlocal network and a geo-dilated LSTM.
- (b) STAN [23]: It is a state-of-the-art attention-based model that collect sequetal information with a spatio-temporal attention network with consideration of personalized item frequency.
- (c) SGRec [17]: It is a GNN-based POI recommendation method that propose a Seq2graph augmentation to propogate collaborative signals and learn effective sequential patterns.
- (c) DRAN [37]: It is the state-of-the-art GNN-based method that leverages a disentangled representation-enhanced attention network for next POI recommendation.

**5.1.3 Evaluation Protocol.** We adopt Recall and Normalized Discounted Cumulative Gain (a.k.a NDCG) with a cutoff at top-K-rated items as the evaluation metric, which is a common practice. Specifically, K is enumerated from {2, 5, 10} for Recall and NDCG. Though some previous works choose to evaluate the model with a certain amount of negative samples, we choose all the POIs as the candidate POI at the evaluation stage to get stable and more convincing results.

**5.1.4 Implementation Detail.** We implement the proposed Diff-POI and all the baseline models in Pytorch. The implementations of baseline methods are based on the released open-sourced projects or from the authors. The embedding size of all models is fixed to 64, the learning rate is fixed as  $lr = 0.001$  and all models are optimized with Adam Optimizer. For Diff-POI and DRAN, the distance-based POI graph is constructed with a 1km distance threshold. For Diff-POI, we set the hyper-parameters with  $\alpha = 0.5$ ,  $\gamma = 0.2$ ,  $\lambda = 10^{-3}$ . The dropout rate is searched from {0.1, 0.2, 0.3, 0.4}. The two interval thresholds  $\delta_s$ ,  $\delta_t$  are fixed to 256 for Diff-POI, DRAN, and STAN. For all sequence-based methods, we preserve the latest 100 visits for sequences with more than 100 POIs. For the diffusion-based sampling module in Diff-POI, we adopt a step size of 0.01, which means the reversed SDE for the sampling module would be solved from time step 0 to 1 with  $dt = 0.01$  at each step. Particularly, we adopt an early-stop mechanism when training and evaluating models with the patience of 10, which means the training process will be stopped when there is no performance gain on validation set in the last 10 epochs.

## 5.2 Overall Comparison (RQ1)

We conduct the overall experiments for the aforementioned baselines on four datasets and the experimental results are reported in table 2. We make the following observations:

- Compared with traditional recommendation methods like MF, LightGCN and DGCF, models that incorporate locational factors achieve significant improvements. The result demonstrates the importance of explicitly considering the geographical influence in POI recommendation. Moreover, models that leverage the time and distance intervals between successive visits (LSTPM, STAN, DRAN, and Diff-POI) outperform the regular location-based baselines, which implies that the key to optimal POI recommendation lies in the spatio-temporal features in the user’s transition history.
- Graph-based methods (SGRec, DRAN, and Diff-POI) have achieved more promising results among all datasets. The observation reflects the effectiveness and rich syntax of graph-structured data. With the help of different expressive GNNs, the models are empowered with the capability of modeling the high-order similarity between POIs revealed from sequences.
- The proposed Diff-POI achieves significant improvements on all four datasets compared with the current state-of-the-art baseline methods. In particular, the performance gain on Recall@10 is improved over the best baseline by 11.9%, 9.5%, 7.6%, and 5.3% on four datasets respectively, and the gain w.r.t. NDCG@10 by 10.8%, 10.5%, 9.9%, and 10.1%. The result shows the effectiveness of the proposed Diff-POI to encode user transition graphs with a spatio-temporal graph encoder and a diffusion-based spatial preference sampling module. The attention-based graph encoder depicts the intervals between successive visits and generates fine-grained user preference embeddings specifically, and the diffusion-based sampling strategy samples the corresponding locational embedding to reveal the spatial trends for the next-to-visit POI to recommend.

## 5.3 Ablation and Parameter Study (RQ2)

In this section, we explore the functionality and effectiveness of different modules in the proposed Diff-POI via a series of ablation studies, as well as Diff-POI’s sensitivity to the hyper-parameters by parameter studies.

**5.3.1 Functionality of Sequence Graph Encoder.** As we have proposed to design an attention-based graph encoder to encode the spatio-temporal signals from sequence graphs, it’s necessary to explore whether the proposed attention module and the message function help to obtain fine-grained user embeddings. Specifically, we compare the performance of the following variants of Diff-POI:

- Diff-POI<sub>GCN</sub>: It replaces the transition graph encoder with a plain Graph Convolution Network (GCN) that neglects the spatio-temporal intervals of the edges.
- Diff-POI<sub>ATT</sub>: It replaces the transition graph encoder with a self-attention layer. Although this variant can aggregate information from all visited POIs, it loses the transition relationship and high-order similarities between POIs.

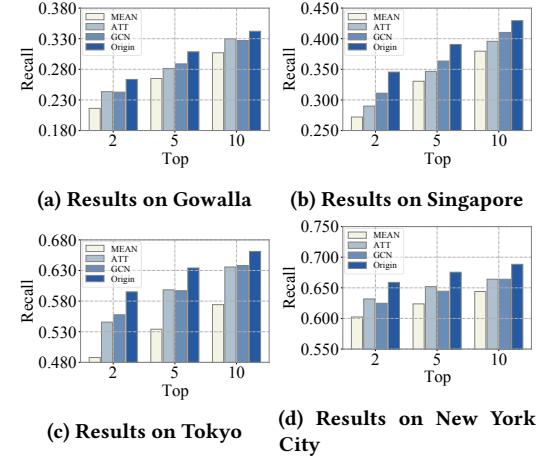


Figure 4: Performance comparison w.r.t. different types of sequence graph encoder.

- Diff-POI<sub>MEAN</sub>: It drops the graph encoder and yields the user embedding merely with a mean pooling function. This variant suffers from neglecting most of the useful information.

We conduct ablation studies on the aforementioned four datasets as well. From the results illustrated in Figure 4, we can observe that:

- The tailor-designed graph encoder module is necessary to fully exploit the spatio-temporal information in transition graphs. The model performance suffers from information loss when the graph encoder is replaced with other structures that neglect the spatiotemporal factors. This proves the effectiveness of our idea to design an attention-based transition graph encoder to leverage spatio-temporal intervals.
- The attention mechanism plays an essential role in generating fine-grained user embeddings. Diff-POI suffers from a significant decline when the spatio-temporal aware attention layer is removed from the model (i.e., Diff-POI<sub>GCN</sub> and Diff-POI<sub>MEAN</sub>).
- The neighborhood aggregation boosts the effect of the generated user embedding  $x_u$ , since Diff-POI<sub>GCN</sub> outperforms Diff-POI<sub>MEAN</sub>. The successive visiting pattern reveals the sequential similarities between POIs, which is helpful in modeling a user. Thus the optimal choice is to combine the spatio-temporal-based attention module with the GNN-based aggregation module to capture useful information from visiting sequences.

**5.3.2 Functionality of Spatial Preference Sampling.** The diffusion-based sampling strategy is intended for sampling from the user-specific posterior distribution. We expect the model to be capable of modeling different locational preferences and generating more instructive representations for recommending a next-to-visit POI. Thus it is necessary to figure out whether the sampling module improves the recommendation results. Specifically, we consider three variants of Diff-POI, which are Diff-POI without the user-specific context condition, Diff-POI without the entire sampling module, and Diff-POI without any locational embeddings at all,

**Table 2: The test results of Diff-POI and all baselines, where R@K and N@K are short for Recall@K and NDCG@K. The highest performance is emphasized with bold font and the second highest is marked with underlines.★ indicates that Diff-POI outperforms the best baseline model at the significance level with a p-value<0.05 level of unpaired t-test.**

<b>Gowalla</b>									
Metrics	MF	LightGCN	DGCF	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.0414	0.1802	0.1147	0.1767	0.1904	<u>0.2195</u>	0.1932	0.2133	<b>0.2635★</b>
Recall@5	0.0869	0.1348	0.1396	0.2123	0.2049	0.2364	0.2286	<u>0.2466</u>	<b>0.3086★</b>
Recall@10	0.1473	0.1699	0.1721	0.2436	0.2618	0.2994	0.2718	<u>0.3056</u>	<b>0.3421★</b>
NDCG@2	0.0428	0.0914	0.0995	0.1670	0.1431	0.1917	0.1573	<u>0.2038</u>	<b>0.2527★</b>
NDCG@5	0.0810	0.1162	0.1319	0.1829	0.1588	0.2152	0.1662	<u>0.2168</u>	<b>0.2716★</b>
NDCG@10	0.1078	0.1432	0.1561	0.1929	0.1742	0.2268	0.1704	<u>0.2367</u>	<b>0.2872★</b>

<b>Singapore</b>									
Metrics	MF	LightGCN	DGCF	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.1103	0.1606	0.1798	0.2521	0.2704	0.2634	0.3058	<u>0.3086</u>	<b>0.3455★</b>
Recall@5	0.1766	0.2144	0.2040	0.2867	0.3253	0.2940	0.3507	<u>0.3554</u>	<b>0.3906★</b>
Recall@10	0.2059	0.2419	0.2285	0.3162	0.3791	0.3280	0.3884	<u>0.3921</u>	<b>0.4294★</b>
NDCG@2	0.0744	0.1720	0.1973	0.2428	0.2610	0.2831	0.2274	<u>0.2972</u>	<b>0.3316★</b>
NDCG@5	0.0791	0.1789	0.2162	0.2583	0.2697	0.2995	0.2697	<u>0.3175</u>	<b>0.3518★</b>
NDCG@10	0.0868	0.1892	0.2269	0.2679	0.2749	0.2892	0.2916	<u>0.3297</u>	<b>0.3643★</b>

<b>Tokyo</b>									
Metrics	MF	LightGCN	DGCF	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.2896	0.3917	0.4204	0.4975	0.5029	0.5105	0.5091	<u>0.5225</u>	<b>0.6031★</b>
Recall@5	0.3141	0.4473	0.4913	0.5408	0.5513	0.5489	0.5488	<u>0.5570</u>	<b>0.6401★</b>
Recall@10	0.3866	0.4936	0.5310	0.5726	0.5795	0.6167	0.6173	<u>0.6210</u>	<b>0.6681★</b>
NDCG@2	0.1984	0.4207	0.4299	0.4842	0.4724	0.4990	0.4715	<u>0.5089</u>	<b>0.5886★</b>
NDCG@5	0.2071	0.4354	0.4376	0.5037	0.4881	0.5264	0.4905	<u>0.5390</u>	<b>0.6062★</b>
NDCG@10	0.2327	0.4403	0.4558	0.5141	0.4962	0.5554	0.5112	<u>0.5602</u>	<b>0.6160★</b>

<b>New York City</b>									
Metrics	MF	LightGCN	DGCF	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.2361	0.3789	0.3368	0.5655	0.5754	<u>0.6027</u>	0.5734	0.5859	<b>0.6589★</b>
Recall@5	0.2792	0.4363	0.4167	0.5994	0.6020	<u>0.6358</u>	0.6175	0.6253	<b>0.6755★</b>
Recall@10	0.3046	0.4519	0.5653	0.6251	0.6312	<u>0.6533</u>	0.6424	0.6478	<b>0.6884★</b>
NDCG@2	0.2002	0.3819	0.4847	0.5629	0.5524	<u>0.5887</u>	0.5490	0.5702	<b>0.6536★</b>
NDCG@5	0.2318	0.3867	0.4922	0.5716	0.5596	<u>0.6092</u>	0.5559	0.5881	<b>0.6616★</b>
NDCG@10	0.2426	0.3903	0.5041	0.5805	0.5681	<u>0.6124</u>	0.5613	0.5956	<b>0.6745</b>

**Table 3: The test results of Diff-POI its two variants, which are W/O-L., W/O-S. and W/O-C., shorts for Diff-POI W/O-Location, Diff-POI W/O-Sampling and Diff-POI W/O-Condition respectively. Ori. represents the original performance of Diff-POI.**

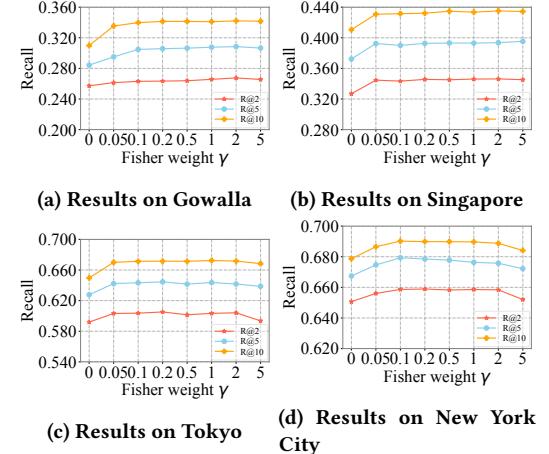
Model	Gowalla				Singapore			
	W/O-L.	W/O-S.	W/O-C.	Ori.	W/O-L.	W/O-S.	W/O-C.	Ori.
R@2	0.2375	0.2441	0.2557	<b>0.2635</b>	0.3126	0.3302	0.3427	<b>0.3455</b>
R@5	0.2785	0.2907	0.3069	<b>0.3086</b>	0.3604	0.3720	0.3895	<b>0.3906</b>
R@10	0.3037	0.3288	0.3401	<b>0.3421</b>	0.3914	0.4028	0.4279	<b>0.4294</b>
N@2	0.2182	0.2363	0.2509	<b>0.2527</b>	0.2789	0.3140	0.3302	<b>0.3316</b>
N@5	0.2365	0.2544	0.2694	<b>0.2716</b>	0.3002	0.3321	0.3517	<b>0.3518</b>
N@10	0.2477	0.2662	0.2810	<b>0.2872</b>	0.3234	0.3428	0.3649	<b>0.3643</b>

Model	Tokyo				New York City			
	W/O-L.	W/O-S.	W/O-C.	Ori.	W/O-L.	W/O-S.	W/O-C.	Ori.
R@2	0.5623	0.5891	0.5947	<b>0.6031</b>	0.6261	0.6308	0.6538	<b>0.6589</b>
R@5	0.6041	0.6278	0.6344	<b>0.6401</b>	0.6359	0.6587	<b>0.6761</b>	0.6755
R@10	0.6367	0.6582	0.6605	<b>0.6681</b>	0.6507	0.6704	<b>0.6893</b>	0.6884
N@2	0.5492	0.5769	0.5816	<b>0.5886</b>	0.6098	0.6247	<b>0.6568</b>	0.6536
N@5	0.5681	0.5944	0.6017	<b>0.6062</b>	0.6284	0.6429	<b>0.6639</b>	0.6616
N@10	0.5787	0.6042	0.6104	<b>0.6160</b>	0.6323	0.6594	<b>0.6784</b>	0.6745

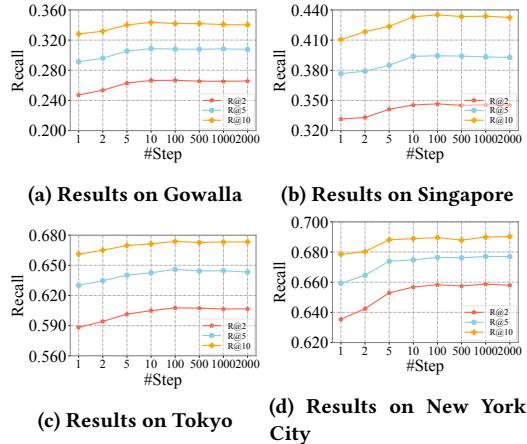
namely W/O-Condition, W/O-Sampling, and W/O-Location respectively. The comparison result between the original Diff-POI and the variants is reported in Table 3 and we can observe that:

- The extra locational embedding plays a vital role in location-based POI recommendations for models without any locational embeddings reach the worst performance among all variants. Explicitly representing the geographical information brings significant improvements to model performance. The model is able to assess whether the target item is similar to the user’s locational preference by the encodings of the distance-based POI graph.
- The diffusion-based sampling process is necessary to improve the recommendation results for the model without the sampling process under-performs the original model and the model without conditional sampling. The sampling strategy brings over % and % improvement on Recall@10 and NDCG@10 respectively, which proves the effectiveness of the sampling module. The diffusion-based sampling process could generate a more relevant and user-specific locational preference and boost the recommendation process.
- The user-specific context condition can bring positive influences for sampling an effective locational preference. Though compared with the model without sampling, Diff-POI could still benefit directly from the locational posterior, yet it suffers a performance decline for neglecting the user-specific information provided during the controlled diffusion process. However, sometimes (e.g. on Tokyo and New York City datasets) the extra condition won’t bring too many benefits to depicting the user’s locational preference since the initialized location prototype already contains sufficient information about the user.



**Figure 5: Recommendation recall w.r.t. different weight coefficient  $\gamma$  on four datasets.**

**5.3.3 Effect of Fisher Loss.** Recall that we adopt a Fisher loss defined by Eq. 19 as the optimization target in Eq. 21, where the loss is controlled by a hyper-parameter  $\gamma$ . The Fisher loss  $\mathcal{L}_{Fisher}$  is proposed to optimize the score function  $s_\theta$  so the model can sample a more accurate location preference that is close to the target embedding. We wonder how  $\mathcal{L}_{Fisher}$  affects the recommendation performance, so we conduct a parameter study on  $\gamma$ . Especially, we vary  $\gamma$  from 0 (without  $\mathcal{L}_{Fisher}$ ) to 5 and record the model performance. From the results in Figure 5 we can observe that:

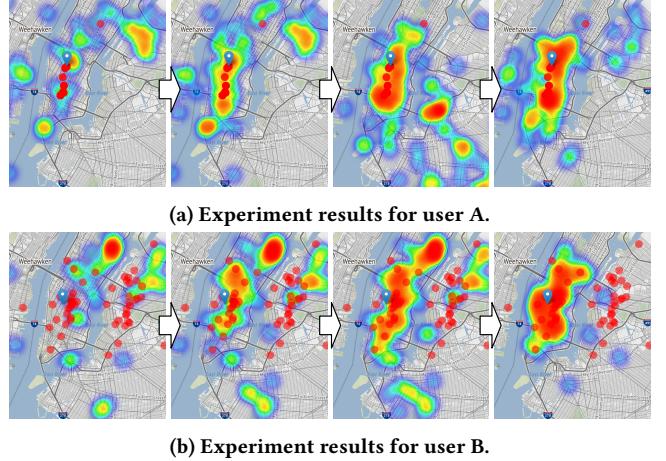


**Figure 6: Recommendation recall w.r.t. different numbers sampling steps.**

- The Fisher loss  $\mathcal{L}_{Fisher}$  is necessary for the promising model performance and  $\mathcal{L}_{Fisher}$  regularizes the sampling module to more accurate generation results. When  $\gamma$  is 0 and no regularization is applied on the score network, the model performance declines due to the degenerated score function and fails to sample a representative locational preference and influence the model performance.
- The model performance reaches the optimal when  $\gamma$  is around 0.2 to 1. When the regularization is weak, the model lacks a reasonable sampling module and fails to model the locational preference. However on some datasets, when  $\gamma$  is extremely large (over 2.0), Diff-POI would overly focus on generating an accurate locational embedding and ignore the useful information from the user's visiting history.
- Even though the model performance suffers when  $\gamma$  is reduced to 0, Diff-POI can still achieve competitive performance against most of the baseline methods. This implies that the score network  $s_\theta(\cdot)$  can be optimized with the main cross entropy loss  $\mathcal{L}_{CE}$ . As the auxiliary fisher loss  $\mathcal{L}_{Fisher}$  boosts the optimization of  $s_\theta(\cdot)$  with additional assumptions on the marginal probability throughout the diffusion process, it can bring extra benefits to model performance.

**5.3.4 Effect of Sampling Steps.** Recall that for the sampling process, we adopt a fixed step size  $dt$  for solving the reversed SDE defined in Eq. 17. Intuitively, with little step size, the model is capable of capturing fine-grained information for being better fitted to the assumption of  $\Delta t \rightarrow 0$  in diffusion process [10]. However, the specific influence of the step size  $dt$  remains to be explored. We explore the performance of Diff-POI by varying  $dt$  from 1 (where the sampling process degenerates to a single-step adjustment) to  $10^{-4}$  and the results are reported in Figure 6. We make the following observations:

- Generally  $dts$  close to 0 can lead to better results since the model performance increases when the step size goes smaller  $dt$ . By decreasing the step size, we actually extend the step number of



**Figure 7: Visualization of the top-100 recommended POIs as the diffusion-based sampling process proceeds. Each user's previously visited POIs are marked with red dots and the target POI is marked with blue markers.**

the reversed diffusion process and empower the sampling module by capturing more fine-grained gradient information.

- Even the one-step sampling module ( $dt = 1$ ) outperforms the model without sampling process. The result indicates that the model could still learn more about the user-specific locational preference. However, it doesn't hold for the assumption about  $dt \rightarrow 0$  and suffers from the performance decline.
- When step size is below a specific value (e.g. 100), the performance no obvious increase in model performance can be observed. This result makes the choosing of  $dt$  a trade-off between model performance and computational consumption since increasing sampling steps will significantly increase the time complexity of Diff-POI.

## 5.4 In-depth Study (RQ3)

Recall that we propose Diff-POI to better clarify the user's spatial preference in a more fine-grained manner. Specifically, the diffusion-based sampling module solves a reverse diffusion SDE to dynamically sample from the user's preference posterior. We wonder if the proposed sampling strategy helps to depict the user's spatial preference effectively. Thus we expect to intuitively show how the sampling result approaches the true locational preference of the user. Particularly, we choose two representative users, namely user A and user B, where A is more likely to follow regular routines and B is an adventurous user. We visualize the top-recommended POIs with the sampled location preference at different sampling steps. And we also conduct detailed case studies to figure out whether the sampling strategy of Diff-POI is capable of recommending appropriate POIs considering the locational preference of users.

**5.4.1 Visualization of Dynamic Sampling.** The diffusion-based sampling module samples the locational preference in an iterative manner, which means that the sampled data should eventually approach the real distribution, in our case is the location of the target POI. To visualize this process, we train Diff-POI on New York City dataset,

**Table 4: The top-5 recommended POIs via the sampled spatial preference of users. The category and distance (in km) between the recommended POI and target POI (i.e. ground truth) is recorded in the table. The category marked with  $\star$  represents the target POI.**

Recommended POIs for User A w.r.t. sampling step									
Top-K	0%		33%		66%		100%		
	Category	Distance	Category	Distance	Category	Distance	Category	Distance	
1	Seafood	3.55	Tattoo Parlor	2.02	Caribbean Food $\star$	0.00	Caribbean Food $\star$	0.00	
2	Mediterranean Food	5.24	Seafood	3.55	Vegetarian Food	1.84	Home (private)	1.91	
3	Bank	2.39	Bar	3.41	Coffee Shop	1.41	American Food	0.53	
4	Park	6.76	Caribbean Food $\star$	0.00	Sandwich Place	2.88	Seafood	3.55	
5	Food & Drink	4.57	Clothing Store	6.86	Bus Station	3.88	Subway	1.89	

Recommended POIs for User B w.r.t. sampling step									
Top-K	0%		33%		66%		100%		
	Category	Distance	Category	Distance	Category	Distance	Category	Distance	
1	Burger Joint	7.35	Burger Joint	7.35	Outdoor Store	1.31	Laundry Service	0.59	
2	Smoke Shop	9.25	College Building	10.15	American Food	1.83	Bar	0.69	
3	Food & Drink	15.13	Subway	4.07	Bridge	1.94	Pharmacy $\star$	0.00	
4	Medical Center	15.61	Greek Food	3.24	Pharmacy $\star$	0.00	Bakery	1.33	
5	Bar	9.62	Medical Center	15.61	Food & Drink	1.03	Deli	0.97	

and obtain the sampled locational preference  $v_u$  for users A and B when the sampling process is 0% (the initial value), 33%, 67%, and 100% complete. We then calculate the top-100 recommended POIs with the sampled  $v_u$  only and visualize them in heat maps. The history POIs are marked with red dots and the targets are marked with blue markers. From the results in Figure 7 we can observe that:

- With the help of the sampling strategy, Diff-POI is capable of finding the approximate location for the next visits. From both figures, we can observe a clear pattern that the recommended POIs are clustering around the location of the ground truth target POI as the sampling process proceeds. This intuitively shows the diffusion-based sampling process is helpful to figure out the locational preference of specific users.
- Predicting POIs for regular users like A is easier for the sampling module to depict. As illustrated in Figure 7a, the historical visits of A show clear regional characteristics (basically around the Manhattan district), making it easy to locate the possible POIs around the familiar areas at the early stage of the sampling process. In contrast, the visited POIs of user B are scattered around the city (in Manhattan, Queens and Brooklyn) so it takes more steps to sample the true preference of B and locate the target POI through the sampling process.
- Though the sampled locational preference can intuitively reflect which region the user is most likely to visit, it is unable to directly find the accurate location of target POI. As illustrated, when the sampled process converges, the predicted POIs are gathering around a relatively large area around target POI. This implies that we still need a sequence-based preference score to accurately select the target POI from a wide range of possible candidates.

**5.4.2 Case Study of Sampling Process.** To further investigate the effectiveness of the sampling module, we conduct a detailed case study by recording the top-5 recommended POIs for both users throughout the sampling process. Particularly, we calculate the recommendation score with the sampled locational preference when the sampling process is 0%, 33%, 67%, and 100% done respectively and recall the top-5 recommended POIs sorted by the score. We report the category of the recalled POIs and the distance from recalled POIs and corresponding target POIs. From the results in Table 4 we can observe that:

- The sampled representations reflect the locational characteristics of POIs. The recalled POIs via locational preference are more likely to be in the same region, for the distance between the target and recalled POIs are similar at every step. The distances between the recalled POIs and target POIs are getting smaller when the sampling step grows, which illustrates that the sampling module is capable of locating the target POI via the reversed diffusion process.
- Generally recommending POIs to users with regular routines (e.g. user A) is more accurate since the helpful information from the previous visiting patterns can be easily leveraged. With the expressiveness patio-temporal sequence graph encoder, Diff-POI is able to recall the target POI in the early stage of the sampling process. The other recalled POIs are geographically close to the target POI, which makes them reasonable candidates as well.
- However, for users who are exploring novel areas (e.g. user B), simply aggregating the representations of historical visits would not be as effective. However, as the sampling step increases, Diff-POI successfully samples the user’s implicit locational preference, and the target POI begins to appear in the top-5 recommended

list. The experimental result illustrates that with the condition-driven sampling process, Diff-POI is able to sample from the user's locational posterior to decide where its implicit location preference lies. The result also illustrates the necessity to iteratively sample the locational preference to locate the accurate spot of the next-to-visit POIs.

## 6 CONCLUSION

In this work, we propose an autoregressive ODE-based graph recommendation framework Diff-POI, for graph recommendation on the hybrid dynamic interacting system, which is built upon two tailored designed graph propagation modules. On the one hand, a neural ODE-based edge evolving module implicitly depicts the dynamic evolving process brought by the collaborative affinity between user-item interactions. On the other hand, a temporal attention-based graph aggregating module explicitly aggregates neighboring information on the interaction graph. We define the way to build the corresponding hybrid dynamic interaction systems given a temporal interaction graph  $\mathcal{G}$ . By autoregressively applying the two modules, Diff-POI obtains node representations with temporal and dynamic features on a hybrid dynamic system. Comprehensive experiments and visualization results on several real-world datasets illustrate the effectiveness and strength of Diff-POI.

## ACKNOWLEDGMENTS

The authors are grateful to the anonymous reviewers for critically reading the manuscript and for giving important suggestions to improve their paper.

This paper is partially supported by the National Key Research and Development Program of China with Grant No. 2018AAA0101902 and the National Natural Science Foundation of China (NSFC Grant No. 62276002).

## REFERENCES

- [1] Brian DO Anderson. 1982. Reverse-time diffusion equation models. *Stochastic Processes and their Applications* 12, 3 (1982), 313–326.
- [2] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. 2013. Where you like to go next: Successive point-of-interest recommendation. In *Twenty-Third international joint conference on Artificial Intelligence*.
- [3] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems* 34 (2021), 8780–8794.
- [4] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. 2018. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference*. 1459–1468.
- [5] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, and Yeow Meng Chee. 2015. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI'15 Proceedings of the 24th International Conference on Artificial Intelligence*. ACM, 2069–2075.
- [6] Qing Guo, Zhu Sun, Jie Zhang, and Yin-Leng Theng. 2020. An attentional recurrent neural network for personalized next location recommendation. In *Proceedings of the AAAI Conference on artificial intelligence*, Vol. 34. 83–90.
- [7] Haoyu Han, Mengdi Zhang, Min Hou, Fuzheng Zhang, Zhongyuan Wang, Enhong Chen, Hongwei Wang, Jianhui Ma, and Qi Liu. 2020. STGCN: a spatial-temporal aware graph learning method for POI recommendation. In *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1052–1057.
- [8] Ruining He and Julian McAuley. 2016. Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th international conference on data mining (ICDM)*. IEEE, 191–200.
- [9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.
- [10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.
- [11] Wei Ju, Zheng Fang, Yiyang Gu, Zequn Liu, Qingqing Long, Ziyue Qiao, Yifang Qin, Jianhao Shen, Fang Sun, Zhiping Xiao, et al. 2023. A Comprehensive Survey on Deep Graph Representation Learning. *arXiv preprint arXiv:2304.05055* (2023).
- [12] Wei Ju, Yifang Qin, Ziyue Qiao, Xiao Luo, Yifan Wang, Yanjie Fu, and Ming Zhang. 2022. Kernel-based Substructure Exploration for Next POI Recommendation. *arXiv preprint arXiv:2210.03969* (2022).
- [13] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [14] Dejian Kong and Fei Wu. 2018. HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction.. In *IJCAI*, Vol. 18. 2341–2347.
- [15] Zifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761* (2020).
- [16] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B Hashimoto. 2022. Diffusion-lm improves controllable text generation. *arXiv preprint arXiv:2205.14217* (2022).
- [17] Yang Li, Tong Chen, Yadan Luo, Hongzhi Yin, and Zi Huang. 2021. Discovering collaborative signals for next POI recommendation with iterative Seq2Graph augmentation. *arXiv preprint arXiv:2106.15814* (2021).
- [18] Defu Lian, Yongji Wu, Yong Ge, Xing Xie, and Enhong Chen. 2020. Geography-aware sequential location recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 2009–2019.
- [19] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 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*. 831–840.
- [20] Defu Lian, Vincent W Zheng, and Xing Xie. 2013. Collaborative filtering meets next check-in location prediction. In *Proceedings of the 22nd International Conference on World Wide Web*. 231–232.
- [21] Nicholas Lim, Bryan Hooi, See-Kiong Ng, Yong Liang Goh, Renrong Weng, and Rui Tan. 2022. Hierarchical multi-task graph recurrent network for next poi recommendation. In *Proceedings of the 45th international ACM SIGIR conference on Research and development in Information Retrieval*.
- [22] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 30.
- [23] Yingtao Luo, Qiang Liu, and Zhaocheng Liu. 2021. Stan: Spatio-temporal attention network for next location recommendation. In *Proceedings of the Web Conference 2021*. 2177–2185.
- [24] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. 2021. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073* (2021).
- [25] Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic matrix factorization. *Advances in neural information processing systems* 20 (2007).
- [26] Yifan Qin, Yifan Wang, Fang Sun, Wei Ju, Xuyang Hou, Zhe Wang, Jia Cheng, Jun Lei, and Ming Zhang. 2023. DisenPOI: Disentangling Sequential and Geographical Influence for Point-of-Interest Recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 508–516.
- [27] Ruihong Qiu, Hongzhi Yin, Zi Huang, and Tong Chen. 2020. Gag: Global attributed graph neural network for streaming session-based recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 669–678.
- [28] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*. 811–820.
- [29] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*. PMLR, 2256–2265.
- [30] Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems* 32 (2019).
- [31] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456* (2020).
- [32] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.
- [33] Ke Sun, Tieyun Qian, Tong Chen, Yile Liang, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2020. Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 214–221.
- [34] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. 2022. DiGress: Discrete Denoising diffusion for graph

- generation. *arXiv preprint arXiv:2209.14734* (2022).
- [35] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation. In *IJCAI*. 3877–3883.
- [36] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. 2020. Disentangled graph collaborative filtering. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 1001–1010.
- [37] Zhaobo Wang, Yanmin Zhu, Haobing Liu, and Chunyang Wang. 2022. Learning Graph-based Disentangled Representations for Next POI Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1154–1163.
- [38] Zhaobo Wang, Yanmin Zhu, Qiaomei Zhang, Haobing Liu, Chunyang Wang, and Tong Liu. 2022. Graph-enhanced spatial-temporal network for next POI recommendation. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16, 6 (2022), 1–21.
- [39] Max Welling and Yee W Teh. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. 681–688.
- [40] Min Xie, Hongzhi Yin, Hao Wang, Fanjiang Xu, Weitong Chen, and Sen Wang. 2016. Learning graph-based poi embedding for location-based recommendation. In *Proceedings of the 25th ACM international conference on information and knowledge management*. 15–24.
- [41] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. 2022. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923* (2022).
- [42] Quan Yuan, Gao Cong, and Aixin Sun. 2014. Graph-based point-of-interest recommendation with geographical and temporal influences. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*. 659–668.
- [43] Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. 2022. Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations. *arXiv preprint arXiv:2207.06635* (2022).
- [44] Pengpeng Zhao, Anjing Luo, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S Sheng, and Xiaofang Zhou. 2020. Where to go next: A spatio-temporal gated network for next poi recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 5 (2020), 2512–2524.
- [45] Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael Lyu, and Irwin King. 2016. STELLAR: Spatial-temporal latent ranking for successive point-of-interest recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009