A Diffusion Model for POI Recommendation

YIFANG QIN, National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University, Beijing, China

HONGJUN WU, School of EECS, Peking University, Beijing, China

WEI JU, National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University, Beijing, China

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

MING ZHANG, National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University, Beijing, China

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 focus 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 article, we propose *Diff-POI*: a *Diff*usion-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 reverse 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;**

Additional Key Words and Phrases: Next POI recommendation, graph neural network, diffusion model

54

1 INTRODUCTION

The prosperity of **Location-Based Social Networks (LBSNs)** and location-based services apps has sparked significant interest in the development of **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 have become integral to numerous check-in apps, such as Yelp and Foursquare. These location-based apps store user check-ins and provide recommendations for the most suitable POIs, aiming to alleviate the information overload of online platforms. Unlike other sequential recommendation tasks, POI recommendation focuses on capturing the geographical characteristics of the POIs, and how their locations influence user behavior [27, 31, 63].

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 [6, 27, 66] or time gap between successive visits [22, 65]. Above that, existing works proposed different methods to encode the observed sequences, such as Markov chains [3, 10]. **Recurrent Neural Networks (RNNs)** and their variants are also widely adopted by previous works [5, 8, 50, 65] and have 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 [9, 25, 29, 56] to depict the transition and location relationship between POIs.

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 [19] to encourage consistency between the two representations, whereras other works propose to disentangle the sequential and spatial influences [40, 55].

Although the the existing works are effective, an important yet common situation is neglected, which is when users aim to explore POI recommendations in unfamiliar areas. For historical visits, a common practice is to aggregate the corresponding locational representations via a pooling method or through the attention mechanism [26, 35, 55]. These methods successfully summarize the geographical feature of visiting history and tend to recommend nearby POIs for the user [40]. However, they often fail to cater to users who seek to explore novel areas. Figure 1 illustrates this issue with a toy example, showcasing different types of users and their preferences for diverse visiting tour routes. User 1 represents a typical conservative visitor who is more likely to revisit the same POIs or explore familiar areas. User 2 is a curious traveler who enjoys stepping out of their comfort zone occasionally. User 3 embodies a long-distance tourist who is exploring novel areas within the city. Despite the diverse visiting patterns among users, aggregation-based representation learning methods can only collect and aggregate historical geographical information and make recommendations based on locational similarity between POIs. This simplistic approach may yield promising results for users like User 1 who follow regular routines but can lead to
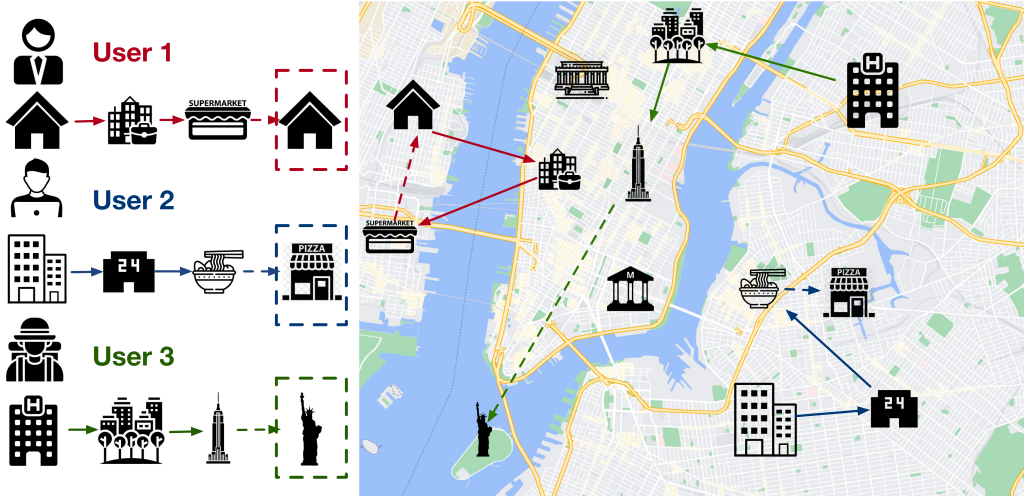
Fig. 1. A toy example of different geographical visiting patterns. User 1 usually follows a daily routine and commutes between workplace and home, and would occasionally drop by some spots like supermarkets along the way. User 2 works from home and is interested in exploring restaurants around the dwelling. User 3 has just arrived at the hotel and is about to travel around famous tourist attractions in the city. Although the various implicit locational preference lies in historical visits, previous aggregation-based representation would be insufficient for exploring POIs in unfamiliar places.

sub-optimal solutions for adventurous users like User 2 and tourists seeking to explore the city like User 3. An effective method to determine whether to visit novel areas remains unexplored. To tackle this problem, we require a method that bridges the gap between the historical visiting distribution and the target preference distribution. Although research on this inter-distribution relationship is still limited, we can rely on the diffusion generative method, which is a popular theory in the field of generative studies and effectively resolves this problem.

The rapid development of diffusion-based methods since the DDPM (Denoising Diffusion Probabilistic Model) [13, 45] has garnered increasing attention. By utilizing **Stochastic Differential Equations (SDEs)** to model the diffusion process, the reverse diffusion process can be seen as a Langevin sampling process from a given prior distribution [48]. Further studies on score matching and Langevin dynamics have provided more insights into the application of existing diffusion algorithms. Notably, these methods have shown significant improvements over traditional approaches in various fields, such as image editing [36], natural language processing [24], and graph generation [51, 60]. The combination of diffusion process and score-based models enables sampling from distributions represented by specific score functions, expanding the application of diffusion-based methods beyond generative tasks.

Recent studies on diffusion models [32, 46, 64] have shed light on the nature of the diffusion process. It empowers generative models with the ability to depict the transportation path from source distributions to target distributions. From the perspective of diffusion process, the overall transition between distributions is decomposed into a series of stochastic processes, which can be better captured with specific model structures, such as neural networks. In the context of generative methods, this benefit of the diffusion process aids models in accurately representing the complex sample space compared to traditional methods such as VAEs [20, 44] and GANs [7]. Under this assumption, layers that map between distributions are trained in a step-by-step manner, which fully explores the expressiveness of models.

In the case of POI recommendation, there exists a similar dilemma in traditional methods, where the relationship between a user's historical visits and their future preferences remains unclear. As mentioned earlier, previous location-based representation learning methods have successfully modeled the geographical features of visiting sequences but lack an effective way to indicate a user's future preference. A straightforward approach would be to apply an aggregation layer to directly transform the historical distribution into a future preference. However, these traditional solutions lack explicit modeling of the subtle relationship between these two distributions. Thereby, inspired by the idea of Langevin sampling from specific distributions, we propose addressing the aforementioned problem in location-based POI recommendation with a tailored geo-preference diffusion process.

In a nutshell, we propose *Diff-POI*: a *Diff*usion-based model that samples the user's spatial preference for the next *POI* 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 reverse 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 LBSN 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 three aspects, namely the next POI recommendation, GNNs, 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 [37, 43] and Markov chains [3, 10]. Further studies extend the model by explicitly considering the temporal or locational similarity between POIs [14, 28, 52].

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. RNNs and their variants are widely used to encode visiting sequences. ST-RNN [31] proposes to use different transition matrices to model contexts from different perspectives in RNNs. STGN [65] introduces two extra time and distance gates to LSTM structures to better

capture the spatio-temporal gap in visiting sequences. LSTPM [50] 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 [8]. There are also attempts to leverage transformer-like self-attention structure for POI recommendation tasks [38, 61]. GeoSAN [26] equips the self-attention layer with a geography encoder to encode locations with GPS coordinates. STAN [35] further improves the location encoder by leveraging relative spatio-temporal information in user trajectories.

2.2 Graph Neural Networks

The popularity of GNNs has seen tremendous success, demonstrating exceptional performance across various graph-related tasks such as node classification [15, 21, 62], graph classification [18, 33, 34], and graph clustering [2, 16]. Their immense potential has propelled the development of widespread applications, including collaborative filtering [39, 54], traffic flow prediction [67], and drug discovery [17]. A popular trend in recommender system research is the application of GNN structures. Since the success of **Graph Convolution Networks (GCNs)** [21], graph-structured data in POI recommendation tasks have been widely studied [30, 42]. GE [59] collects information from various kinds of POI and user graphs. STGCN [9] exploits the idea of the RGCN (Relational Graph Convolution Network) to depict the multiple relationships between users and POIs. Later works like GSTN [56] lay their focus on the transition and distance POI graphs. DisenPOI [40] further proposes to disentangle the geographical and sequential effects with a contrastive regularization to fully exploit the different factors behind visits. More recent studies [19, 25, 40, 55, 56] focus on the dual graph encoder structures that are capable of jointly modeling the spatio-temporal influence through the expressive graph structures.

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 research has suggested [19, 40].

2.3 SDEs with Denoising Diffusion

Diffusion models [13, 45] have achieved great success in multiple fields. Further research reveals the relationship between the denoising diffusion process and score-based methods [47, 48], 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 [48] and are gaining great advantage against traditional generative methods such as GANs [4]. The diffusion process is also viewed as transportation between distributions and can be used in domains including image edit [36, 64], natural language processing [24], audio generation [23], and graph generation [51, 60]. However, other applications of the denoising diffusion process remain unexplored.

In a nutshell, the strong capability of the 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 that 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

Consider the typical settings of POI recommendation. For the POI set $\mathcal{L} = \{l1, l2, \ldots, l_{|\mathcal{L}|}\}$ and user set $\mathcal{U} = \{u1, u2, \ldots, u_{|\mathcal{U}|}\}$, the task can be formulated as follows: given a user u, its trajectory of POIs, and visiting timestamps $H(u) = \{(l_1^u, t_1^u), (l_2^u, t_2^u), \ldots (l_n^u, t_n^u)\}$, the target is to recommend target POI l_{n+1}^u at the 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 $A_d(i, j) = haversine(l_i, l_j)$ denotes the distance between l_i and l_j in kilometers, calculated by the haversine formula.

3.2 Spatio-Temporal Dual Graph

Now we construct a 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 $H(u)$, and each edge $e = <l_i^u, l_{i+1}^u> \in \mathcal{E}_u$ 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 that the distance between l_i and l_j is within a specific threshold (e.g., 1 km as previous works suggested [56]), and the edge weight $\mathcal{A}_u$ 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 sequence of noise scales $\beta_0, \beta_1, \ldots, \beta_T$, the discrete diffusion process is considered as a Markov chain with the transition probability and the conditional probability is formulated as

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

More generally, we can formulate any discrete one-step diffusion into continuous form and reform the process as the solution to specific Itô SDE [48]:

$$dx = f(g, t)dt + g(t)dw, \tag{2}$$

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

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

where the dw and $d\overline{w}$ in Equations (2) and (3) are sampled independently from a standard Wiener process at each step of the SDEs.

With a specified score function $s_\theta(x)$ that is parameterized by a neural network θ, 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 reverse SDE in Equation (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: (A) a spatio-temporal sequence graph 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 that obtains user-specific locational prototypes, followed by a diffusion-based sampling strategy to sample the spatial preference for users.
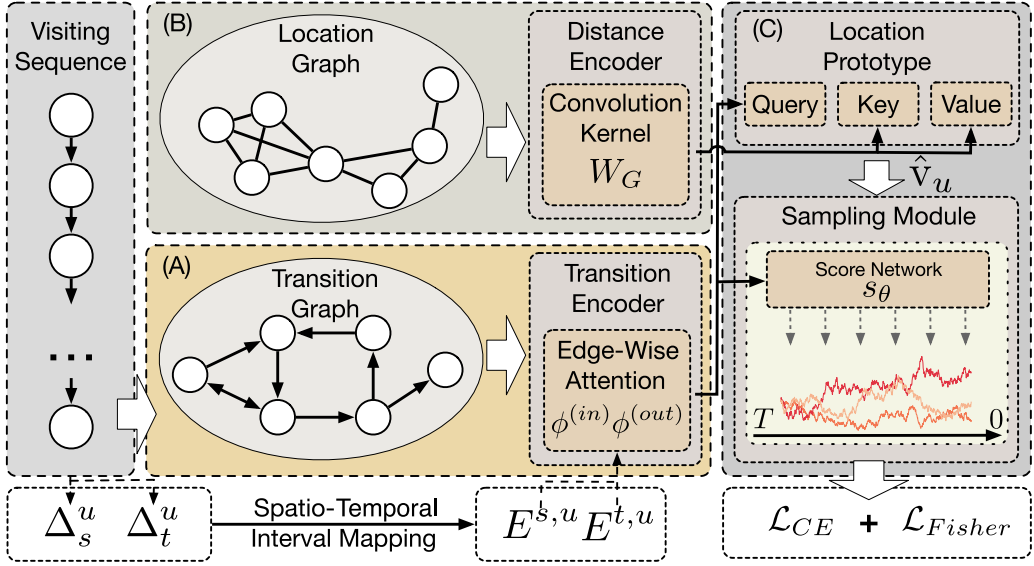
Fig. 2. Illustration of the proposed framework Diff-POI. Diff-POI is intended for making personalized POI recommendations based on the user's spatial preference and history visiting patterns and is composed of three main parts: (A) a spatio-temporal graph encoder, (B) a distance-based graph convolution encoder, and (C) a diffusion-based sampling module. The details of part (C) will be illustrated in the following sections.

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 closely spaced visits reflects the trajectory of a single trip, suggesting that the user is more likely to continue visiting nearby POIs. Conversely, a long distance between two visits indicates a potential change in the user's locational preference. This distinction calls for an explicit modeling of the intervals between visits from both spatial and temporal perspectives. To achieve this, we propose modeling the temporal and locational intervals using a trainable embedding matrix. For a transition graph $\mathcal{G}_u$ and its corresponding visiting sequence $H(u), |H(u)| = n$, we first obtain the relative spatial and temporal interval matrices $\Delta_s^u, \Delta_t^u \in \mathbb{R}^{n \times n}$:

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

where $s_{ij}^u = \lfloor \frac{A_d(l_i^u, l_j^u)}{s_{min}^u} \rfloor$ 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. We round down the normalized distance to discrete integers so that intervals at different levels can be represented via a list of embedding vectors. Similarly, the temporal intervals are defined with $t_{ij}^u = \lfloor \frac{t_j^u - t_i^u}{t_{min}^u} \rfloor$.

We then maintain two trainable embedding matrices to represent the intervals respectively, after clipping the relative interval matrices Δ_s^u, Δ_t^u with two threshold values δ_s, δ_t:

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

where δ_s and δ_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:

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

where $e_{ij}^{s,u}, e_{ij}^{t,u} \in \mathbb{R}^d$ are interval embeddings, and 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 $e_i^l \in \mathbb{R}^d, i = 1, 2, \ldots, |\mathcal{L}|$, 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 GGNNs (Gate-controlled GNNs) 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. The detailed illustration of the graph encoder is presented in Figure 3.

Since bi-directional information is equally important for sequential recommendation cases [41, 49], we expect that 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:

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

where 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

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

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

So far, with the message function defined in Equation (7), we can iteratively update the node representation in $\mathcal{G}_u$ with the weighted neighborhood influence, and the outputs noted as $E_u^h = [e_1^h, e_2^h, \ldots, e_n^h]$. 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 $x_u \in \mathbb{R}^d$:

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

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

where $W_Q^h, W_K^h, W_V^h : \mathbb{R}^d \to \mathbb{R}^d$ are learnable projection matrices, and 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.
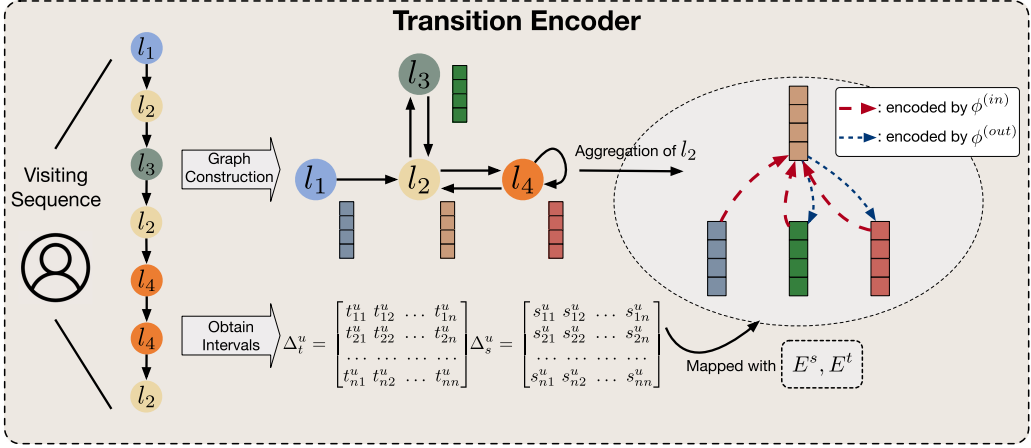
Fig. 3. Illustration of the spatio-temporal sequence graph encoder. When generating messages of a POI, we use two projection matrices to calculate the attention coefficients and reaggregate the neighborhood.

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 GCN [21] as the geographical graph encoder. Specifically, the update function of each layer is built with

$$h_i^{(l)} = \sum_{j \in \mathcal{N}_i} \frac{w_{ij}}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}} W_G^{(l-1)} h_j^{(l-1)}, \tag{11}$$

where the node representation h_i^0 is initialized with e_i^l for each node and $W_G^{(l)}$ represents the convolution kernel of the 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 the 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 $E^g = [e_1^g, e_2^g, \ldots, e_{|\mathcal{L}|}^g]$ with a mean pooling function on the outputs of each layer to capture the high-order connectivity between nodes:

$$e_i^g = \text{MEAN} \left\{ h_i^{(0)}, h_i^{(1)}, \ldots, h_i^{(l)} \right\}. \ i = 1, 2, \ldots, |\mathcal{L}|. \tag{12}$$

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

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

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

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

where $sim(\cdot, \cdot)$ 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 Equation (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

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

where W_Q^g, W_K^g, W_V^g are trainable parameters. $\hat{v}_u$ shows u's historic preference on different locations of the city.

4.3 Diffusion-Based Sampling Module

So far, we have obtained a historic locational preference $\hat{v}_u$ from previous sections. However, directly using v_u to estimate the conditional probability $p(v_u|H(u), \mathcal{G}_g)$ in Equation (13) would lead to sub-optimal results, as it does not fully reflect the user's future locational preference. We address this issue by the diffusion-based sampling module in the following section.

To sample from the posterior distribution in Equation (13), a common practice is to use stochastic gradient Langevin dynamics [57] 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 [48], we propose to estimate the score with a score function and model the sampling process with the variance-preserving SDE. Although diffusion processes with other perturbation kernels may achieve better performance, experimental results demonstrate that variance-preserving SDE already shows promising results in modeling spatial preferences. Details of the proposed sampling module are illustrated in Figure 4.

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.

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:

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

where here we parameterize the noise schedule $\beta(t) = \beta_{min} + \frac{(\beta_{max}-\beta_{min})t}{\beta_{min}T}$ with hyper-parameters β_{min}, β_{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 Equation (16) that starts 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 Reverse SDE for Preference Sampling. The sampling process of the location prototype v_u from the spatial posterior $p(v_u|H(u)\mathcal{G}_g)$ is equivalent to the reverse process of Equation (16). As
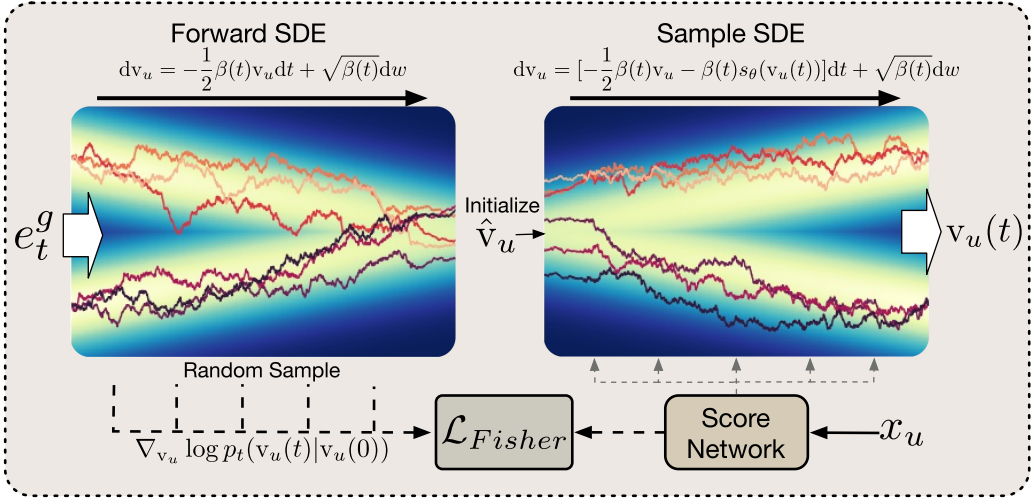
Fig. 4. Illustration of the diffusion-based sampling strategy. We use the reverse (sample) SDE, which is controlled by the user's historical representation, to sample from the user-specific posterior to generate personalized locational preferences. The marginal probabilities are sampled to optimize the score function.

previous research concluded [1], the reverse SDE is formulated with

$$\mathrm{d}v_u = \left[-\frac{1}{2}\beta(t)v_u - \beta(t)\nabla_{v_u}\log p_t(v_u)\right]\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}\overline{w}. \tag{17}$$

Specifically, the spatial preference for u can be sampled with the reverse SDE in Equation (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 θ:

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

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_θ and the score of real samples as previous work suggested [47]:

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

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

To summarize, we have developed a training and inference framework for sampling personalized geographical preferences for individual users, taking inspiration from the diffusion sampling process. By leveraging the diffusion process, our tailored sampling module has an advantage over other existing generative methods such as VAEs and GANs. It allows our model to learn a detailed transition path from the user's historical preferences to the target distribution. Therefore, the sampled preference $v_u(0)$ can comprehensively represent user's future locational preference.

ALGORITHM 1: Learning Algorithm of Diff-POI

    Input: POI set $\mathcal{L}$, User set $\mathcal{U}$, Sampling step num
    Initialize: Distance matrix A_g; Model parameters
    Construct $\mathcal{G}_g$ based on A_g;
    while *not converged* **do**
        Sample a visiting sequence $H(u) = \{(l_1^u, t_1^u), (l_2^u, t_2^u), \dots (l_n^u, t_n^u)\}$ and target POI l_t;
        Construct sequential graph $\mathcal{G}_u$ and interval matrices Δ_s^u, Δ_t^u;
        Encode $\mathcal{G}_u$ to the user encoding x_u;
        Obtain initial location prototype $\hat{v}_u$;
        Initialize $v_u(0) \leftarrow \hat{v}_u$;
        for *i in range stepnum* **do**
            Sample dw from standard Wiener process;
            Calculate dv_u via Equation (17);
            Update $v_u(i) \leftarrow v_u(i-1) + dv_u$;
        end
        Inference preference score $\hat{y}_i^u$;
        if *Train with Fisher loss* **then**
            Sample $t \sim U(0, T)$;
            Calculate Fisher loss $\mathcal{L}_{Fisher}$;
        end
        Calculate and optimize model with overall loss $\mathcal{L}_{Total}$;
    end

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

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

where α 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 target POI y_i^u, the model is optimized with a supervised cross-entropy loss:

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

where Θ represents model parameters and λ is the corresponding weight for the L2 regularization term.

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

$$\mathcal{L}_{Total} = \mathcal{L}_{CE} + \gamma \mathcal{L}_{Fisher}, \tag{22}$$

where γ is a hyper-parameter to balance the optimization targets.

We summarize the overall training and inference process in Algorithm 1.

Complexity Analysis. In the previous sections, we defined the overall propagation algorithm, which consists of two main parts: (i) the graph encoders responsible for generating sequence

embeddings and geographical prototypes, and (ii) the diffusion sampling module used to sample the user's spatial preference.

For (i), assume having a visiting sequence with n POIs, and the complexity of the transition graph encoder is primarily determined by the number of edges and the size of node representations. Computing the locational POI embedding involves edge propagation on the geographical graph $\mathcal{G}_g$, and its complexity is related to the edge size $|\mathcal{E}_g|$. The attention module complexity arises from the softmax normalization applied along the sequence. To summarize, the time complexity can be calculated as follows:

$$O(nd + |\mathcal{E}_g|d + n^2d) = O((|\mathcal{E}_g| + n^2)d). \tag{23}$$

For (ii), forward and reverse diffusion process requires the same computational complexity, which is defined by the number of sampling steps and the complexity of the scoring network, formulated as $O(Td)$.

In summary, the overall computational consumption of Diff-POI can be estimated by considering the complexities of both parts (i) and (ii):

$$O((|\mathcal{E}_g| + n^2 + T)d). \tag{24}$$

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[1] and Foursquare.[2] 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 remaining two subsets are from April 12, 2012 to February 16, 2013.

Following previous work [55], we process the aforementioned datasets with the five-core cleaning strategy, which means the users and POIs with fewer than five 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 sets, respectively, which is a common practice in next POI recommendation [50, 55]. 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.

---

[1]http://snap.stanford.edu/data/loc-gowalla.html
[2]https://sites.google.com/site/yangdingqi/home/foursquare-dataset

Table 1. Descriptive Statistics of the Used Datasets

Dataset	#User	#POI	#Interaction	#Avg. Visit
Gowalla	10,162	24,237	456,820	44.95
SIN	2,321	5,596	194,108	83.63
TKY	2,293	15,177	494,807	215.79
NYC	1,083	9,989	179,468	165.71

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 interaction-based user-item recommendation methods, (b) general graph-based recommendation methods, (c) sequence-based POI recommendation methods that consider spatial factors in recommendation, and (d) graph-based POI recommendation methods. The compared baselines and their brief introductions are listed as follows:

- (a) *MF* [37], a classical collaborative filtering method based on low-rank matrix factorization and optimized with the gradient decent method.
- (a) *GRU4Rec* [12], one of the most classical methods that applies RNNs into sequential recommendation tasks.
- (b) *LightGCN* [11], a CF method that adopts GCNs to node representation learning for recommendation.
- (b) *DGCF* [53], a variant of LightGCN that proposes to disentangle the node representations for better performance.
- (b) *SR-GNN* [58], a transition graph based method that leverages gated graph neural networks to predict the next item.
- (c) *GeoIE* [52], a classical POI recommendation method that learns geographical and interaction influence separately.
- (c) *LSTPM* [50], a state-of-the-art LSTM-based POI recommendation method that models the visiting trajectories with a non-local network and a geo-dilated LSTM.
- (c) *STAN* [35], a state-of-the-art attention-based model that collects sequential information with a spatio-temporal attention network with consideration of personalized item frequency.
- (d) *SGRec* [25], a GNN-based POI recommendation method that proposes a Seq2Graph augmentation to propagate collaborative signals and learn effective sequential patterns.
- (d) *DRAN* [55], a 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 (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. Whereas 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 source 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 the Adam optimizer. For Diff-POI and DRAN, the distance-based POI graph is constructed with a 1-km distance threshold. For Diff-POI, we set the hyper-parameters to $\alpha = 0.5, \gamma = 0.2, \lambda = 10^{-3}$. The dropout rate is searched from {0.1, 0.2, 0.3, 0.4}. The two interval thresholds δ_s, δ_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

Table 2. Test Results of Diff-POI and All Baselines, Where R@K and N@K Are Short for Recall@K and NDCG@K

					Gowalla						
Metrics	MF	GRU4Rec	LightGCN	DGCF	SR-GNN	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.0414	0.1328	0.1102	0.1147	0.1531	0.1767	0.1904	0.2195	0.1932	0.2133	**0.2635**[★]
Recall@5	0.0869	0.1754	0.1348	0.1396	0.1978	0.2123	0.2049	0.2364	0.2286	0.2466	**0.3086**[★]
Recall@10	0.1473	0.2164	0.1699	0.1721	0.2384	0.2436	0.2618	0.2994	0.2718	0.3056	**0.3421**[★]
NDCG@2	0.0428	0.1124	0.0914	0.0995	0.1402	0.1670	0.1431	0.1917	0.1573	0.2038	**0.2527**[★]
NDCG@5	0.0810	0.1367	0.1162	0.1319	0.1621	0.1829	0.1588	0.2152	0.1662	0.2168	**0.2716**[★]
NDCG@10	0.1078	0.1449	0.1432	0.1561	0.1751	0.1929	0.1742	0.2268	0.1704	0.2367	**0.2872**[★]
					Singapore						
Metrics	MF	GRU4Rec	LightGCN	DGCF	SR-GNN	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.1103	0.1775	0.1606	0.1798	0.2406	0.2521	0.2704	0.2634	0.3058	0.3086	**0.3455**[★]
Recall@5	0.1766	0.2291	0.2144	0.2040	0.2715	0.2867	0.3253	0.2940	0.3507	0.3554	**0.3906**[★]
Recall@10	0.2059	0.2774	0.2419	0.2285	0.3080	0.3162	0.3791	0.3280	0.3884	0.3921	**0.4294**[★]
NDCG@2	0.0744	0.1652	0.1720	0.1973	0.2272	0.2428	0.2610	0.2831	0.2274	0.2972	**0.3316**[★]
NDCG@5	0.0791	0.1884	0.1789	0.2162	0.2815	0.2583	0.2697	0.2995	0.2697	0.3175	**0.3518**[★]
NDCG@10	0.0868	0.2040	0.1892	0.2269	0.2549	0.2679	0.2749	0.2892	0.2916	0.3297	**0.3643**[★]
					Tokyo						
Metrics	MF	GRU4Rec	LightGCN	DGCF	SR-GNN	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.2896	0.4203	0.3917	0.4204	0.4600	0.4975	0.5029	0.5105	0.5091	0.5225	**0.6031**[★]
Recall@5	0.3141	0.4878	0.4473	0.4913	0.5257	0.5408	0.5513	0.5489	0.5488	0.5570	**0.6401**[★]
Recall@10	0.3866	0.5205	0.4936	0.5310	0.5458	0.5726	0.5795	0.6167	0.6173	0.6210	**0.6681**[★]
NDCG@2	0.1984	0.4109	0.4207	0.4299	0.4537	0.4842	0.4724	0.4990	0.4715	0.5089	**0.5886**[★]
NDCG@5	0.2071	0.4373	0.4354	0.4376	0.4788	0.5037	0.4881	0.5264	0.4905	0.5390	**0.6062**[★]
NDCG@10	0.2327	0.4520	0.4403	0.4558	0.4908	0.5141	0.4962	0.5554	0.5112	0.5602	**0.6160**[★]
					New York City						
Metrics	MF	GRU4Rec	LightGCN	DGCF	SR-GNN	GeoIE	LSTPM	STAN	SGRec	DRAN	Diff-POI
Recall@2	0.2361	0.4393	0.3789	0.3368	0.5052	0.5655	0.5754	0.6027	0.5734	0.5859	**0.6589**[★]
Recall@5	0.2792	0.4870	0.4363	0.4167	0.5465	0.5994	0.6020	0.6358	0.6175	0.6253	**0.6755**[★]
Recall@10	0.3046	0.5238	0.4519	0.5653	0.5951	0.6251	0.6312	0.6533	0.6424	0.6478	**0.6884**[★]
NDCG@2	0.2002	0.4252	0.3819	0.4847	0.5191	0.5629	0.5524	0.5887	0.5490	0.5702	**0.6536**[★]
NDCG@5	0.2318	0.4467	0.3867	0.4922	0.5319	0.5716	0.5596	0.6092	0.5559	0.5881	**0.6616**[★]
NDCG@10	0.2426	0.4587	0.3903	0.5041	0.5495	0.5805	0.5681	0.6124	0.5613	0.5956	**0.6745**

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 an unpaired t-test.

0.01, which means the reverse SDE for the sampling module would be solved from timestep 0 to 1 with $dt = 0.01$ at each step. In particular, 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 the validation set in the last 10 epochs. The code is available at https://github.com/Yifang-Qin/Diff-POI.

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, 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 of transition history.

• Graph-based methods that leverage the transition graphs (SR-GNN, 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 in depicting sequential patterns. 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 the Sequence Graph Encoder. As we have proposed to design an attention-based graph encoder to encode the spatio-temporal signals from sequence graphs, it is 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$_{Disen}$*, which replaces the transition graph encoder with the disentangled graph representation layer in DRAN [55].

• *Diff-POI$_{GCN}$*, which replaces the transition graph encoder with a plain GCN that neglects the spatio-temporal intervals of the edges.

• *Diff-POI$_{ATT}$*, which 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.

• *Diff-POI$_{MEAN}$*, which 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 5, we can observe the following:

• 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 spatio-temporal 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$_{Disen}$, Diff-POI$_{GCN}$, and Diff-POI$_{MEAN}$).

(a) Results on **Gowalla**

(b) Results on **Singapore**

(c) Results on **Tokyo**
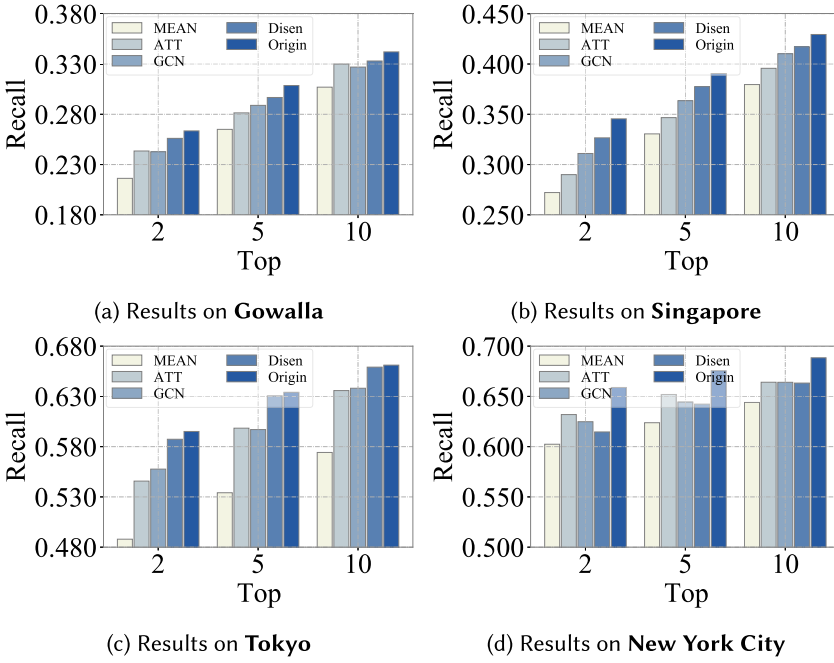
(d) Results on **New York City**

Fig. 5. Performance comparison w.r.t. different types of sequence graph encoder.

- Disentangling representations of POIs is helpful for learning more expressive POI encodings compared with vanilla GCNs. However, the disentangling method suffers from the lack of explicit spatial and temporal information, and therefore there remains a performance gap from the original graph encoder.
- The neighborhood aggregation boosts the effect of the generated user embedding x_u, as Diff-POI$_{GCN}$ outperforms Diff-POI$_{MEAN}$. 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 four variants of Diff-POI: Diff-POI without all graph encoders, Diff-POI without the user-specific context condition and the score network is replaced with an MLP with just $v_u(t)$ as input, Diff-POI without the entire sampling module where the prototype $\hat{v}_u$ is directly fed into the calculation of recommendation score, and Diff-POI without any locational embeddings at all. The aforementioned variants are referred to as W/O-Graph, W/O-Condition, W/O-Sampling, and W/O-Location, respectively. To ensure robustness and reliability of our results, we conduct each setting five times with different random seeds and report the average performance across these iterations. The comparison result between the original Diff-POI and the variants is reported in Table 3, and we can observe the following:

- The sequential and geographical graph encoders are necessary for Diff-POI to extract informative POI embeddings for downstream recommendation. The locational embedding plays

Table 3. Test Results of Diff-POI and Its Four Variants, Which Are W/O-G., W/O-L., W/O-S., and W/O-C., Short for Diff-POI W/O-Graph, Diff-POI W/O-Location, Diff-POI W/O-Sampling, and Diff-POI W/O-Condition, Respectively

Model	Gowalla					Singapore				
	W/O-G.	W/O-L.	W/O-S.	W/O-C.	Ori.	W/O-G.	W/O-L.	W/O-S.	W/O-C.	Ori.
R@2	0.2307	0.2366	0.2431	0.2536	**0.2635**★	0.3019	0.3139	0.3284	0.3408	**0.3455**★
R@5	0.2774	0.2764	0.2887	0.3007	**0.3086**★	0.3572	0.3601	0.3715	0.3891	**0.3906**
R@10	0.3177	0.3025	0.3199	0.3402	**0.3421**	0.4032	0.3921	0.4026	0.4258	**0.4294**★
N@2	0.2194	0.2177	0.2374	0.2508	**0.2527**★	0.2866	0.2787	0.3154	0.3301	**0.3316**★
N@5	0.2389	0.2374	0.2536	0.2688	**0.2716**★	0.3114	0.3012	0.3271	0.3513	**0.3518**
N@10	0.2561	0.2565	0.2658	0.2794	**0.2872**★	0.3263	0.3241	0.3396	0.3568	**0.3643**★

Model	Tokyo					New York City				
	W/O-G.	W/O-L.	W/O-S.	W/O-C.	Ori.	W/O-G.	W/O-L.	W/O-S.	W/O-C.	Ori.
R@2	0.5662	0.5631	0.5894	0.5925	**0.6031**★	0.6000	0.6258	0.6317	0.6496	**0.6589**★
R@5	0.6140	0.6047	0.6312	0.6325	**0.6401**★	0.6311	0.6361	0.6569	0.6751	**0.6755**★
R@10	0.6480	0.6365	0.6581	0.6603	**0.6681**★	0.6527	0.6509	0.6695	0.6862	**0.6884**★
N@2	0.5503	0.5483	0.5771	0.5820	**0.5886**★	0.5895	0.6080	0.6204	0.6512	**0.6536**★
N@5	0.5718	0.5687	0.5934	0.6021	**0.6062**★	0.6035	0.6273	0.6406	0.6599	**0.6616**
N@10	0.5829	0.5784	0.6027	0.6122	**0.6160**★	0.6104	0.6255	0.6575	0.6706	**0.6745**★

Ori. represents the original performance of Diff-POI. ★indicates that the corresponding method outperforms the second best at the significance level with a p-value < 0.05.

a vital role in location-based POI recommendations, given the fact that models without any locational embeddings performs worse than most of the other 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 3.3% and 2.5% 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. 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.

5.3.3 Effect of Fisher Loss. Recall that we adopt a Fisher loss defined by Equation (19) as the optimization target in Equation (21), where the loss is controlled by a hyper-parameter γ. The Fisher loss $\mathcal{L}_{Fisher}$ is proposed to optimize the score function s_{θ} 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 γ. Especially, we vary γ from 0 (without $\mathcal{L}_{Fisher}$) to 5 and record the model performance. From the results in Figure 6, we can observe the following:

- The Fisher loss $\mathcal{L}_{Fisher}$ is necessary for promising model performance, and $\mathcal{L}_{Fisher}$ regularizes the sampling module to more accurate generation results. When γ is 0 and no regularization is applied on the score network, the model performance declines due to the degenerated
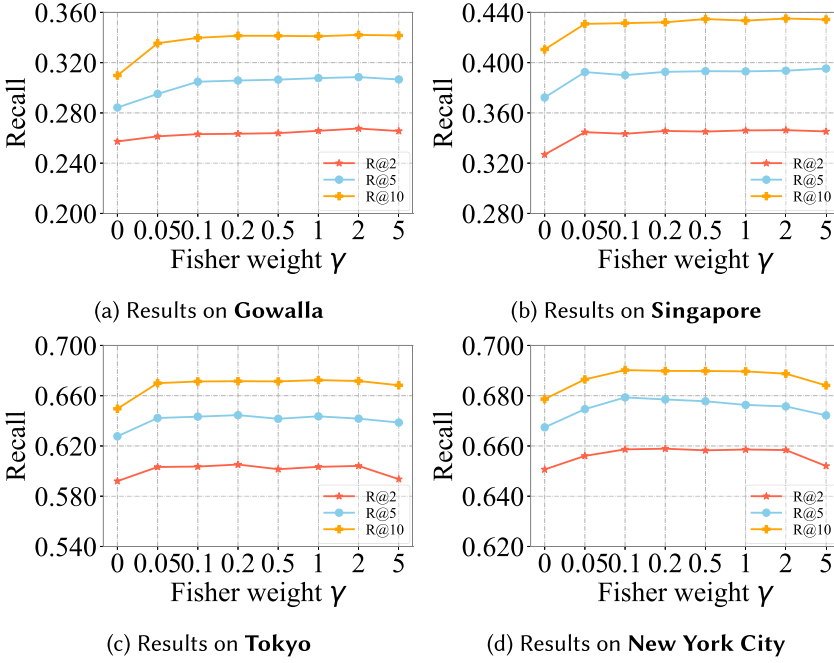
Fig. 6. Recommendation recall w.r.t. different weight coefficient γ on four datasets.

score function and fails to sample a representative locational preference and influence the model performance.

- The model performance reaches the optimal when γ 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 γ 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 γ 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 reverse SDE defined in Equation (17). Intuitively, with little step size, the model is capable of capturing fine-grained information for being better fitted to the assumption of $\Delta t \to 0$ in diffusion process [13]. 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 7. We make the following observations:

- Generally, step sizes that are close to 0 can lead to better results since the model performance increases when the number of sampling steps grows. By decreasing the step size, we actually extend the step number of the reverse diffusion process and empower the sampling module by capturing more fine-grained gradient information.

(a) Results on **Gowalla**

(b) Results on **Singapore**

(c) Results on **Tokyo**
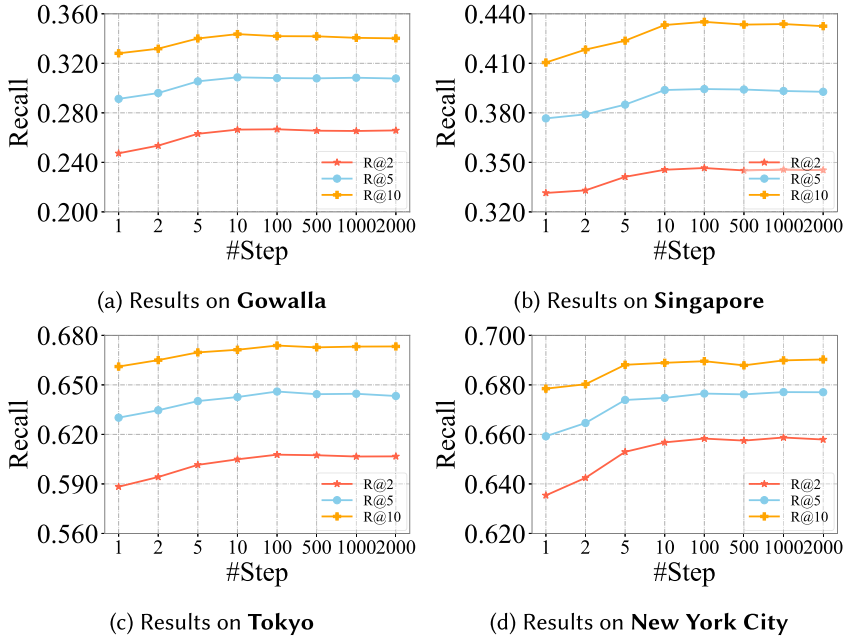
(d) Results on **New York City**

Fig. 7. Recommendation recall w.r.t. different numbers of sampling steps.

- Even the one-step sampling module ($dt = 1$) outperforms the model without the sampling process. The result indicates that the model could still learn more about the user-specific locational preference. However, it does not hold for the assumption about $dt \rightarrow 0$ and suffers from the performance decline.
- When the number of steps 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. To evaluate the effectiveness of Diff-POI in addressing different user preferences, we divide users into distinct groups based on their geographical preferences and compare Diff-POI's performance with different baseline methods for each user group. Additionally, we select three representative users, namely users A, B and C. User A represents users who tend to follow regular routines and visit the same POIs frequently. User B frequently visits two different neighborhoods, indicating a preference for diverse locations. User C is an adventurous user who seeks out new and unfamiliar areas. We visualize the top-recommended POIs with the sampled location preference at different sampling steps for these users. Furthermore, we conduct detailed case studies to examine whether the sampling strategy is capable of recommending POIs with personalized locational preferences.
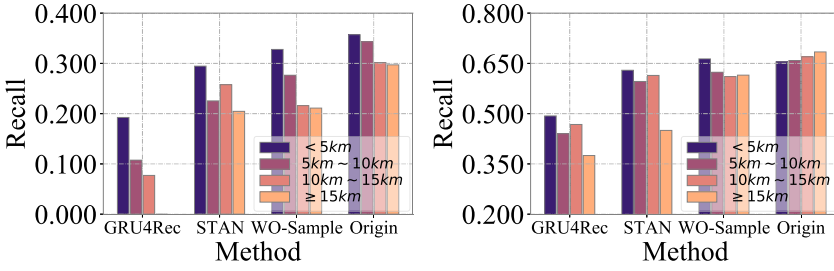
(a) Results on **Singapore**   (b) Results on **New York City**

Fig. 8.   Recommendation recall w.r.t. different user groups.

5.4.1   Model Performance on Different User Groups. To assess whether the proposed sampling strategy of Diff-POI effectively captures different locational preferences, we divide users into different groups based on their geographical visiting patterns and evaluate the model's performance for each group. Specifically, we utilize datasets from Singapore and New York City and calculate the average distance $\bar{\delta}_s$ between successive visits for each user. Larger values of $\bar{\delta}_s$ indicate more adventurous users who travel longer distances between visits. We categorize users into different groups based on their average distance between successive visits: from visitors in a limited area ($\bar{\delta}_s < 5$ km) to long-distance travelers ($\bar{\delta}_s \geq 15$ km). For each group, we measure the Recall@2 performance of four methods: GRU4Rec, STAN, Diff-POI without the sampling module, and the original Diff-POI. The results are presented in Figure 8 and listed as follows:

- In general, as the average distance $\bar{\delta}_s$ between successive visits increases, it becomes more challenging for a model to accurately recommend locations for users. This is because the locational preferences of active users who travel longer distances are harder to capture and model effectively.
- Methods that explicitly incorporate the locational influence in historical visits (STAN and Diff-POI) show improvement compared to traditional methods and mitigate the decline in performance. Notably, the proposed Diff-POI consistently outperforms other methods and maintains robustness even as the average distance increases. This highlights the importance and effectiveness of the diffusion-based sampling module in Diff-POI. For some specific cases (e.g., users who are less adventurous in the New York City dataset), Diff-POI without sampling module may achieve good performance for successfully depicting the local visiting structures.

5.4.2   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, the location of the target POI. To visualize this process, we train Diff-POI on the New York City dataset, and we obtain the sampled locational preference $\mathbf{v}_u$ for users A, B, and C 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 $\mathbf{v}_u$ 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 the results in Figure 9, we can observe the following:

- 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

(a) Experiment results for User A.

(b) Experiment results for User B.

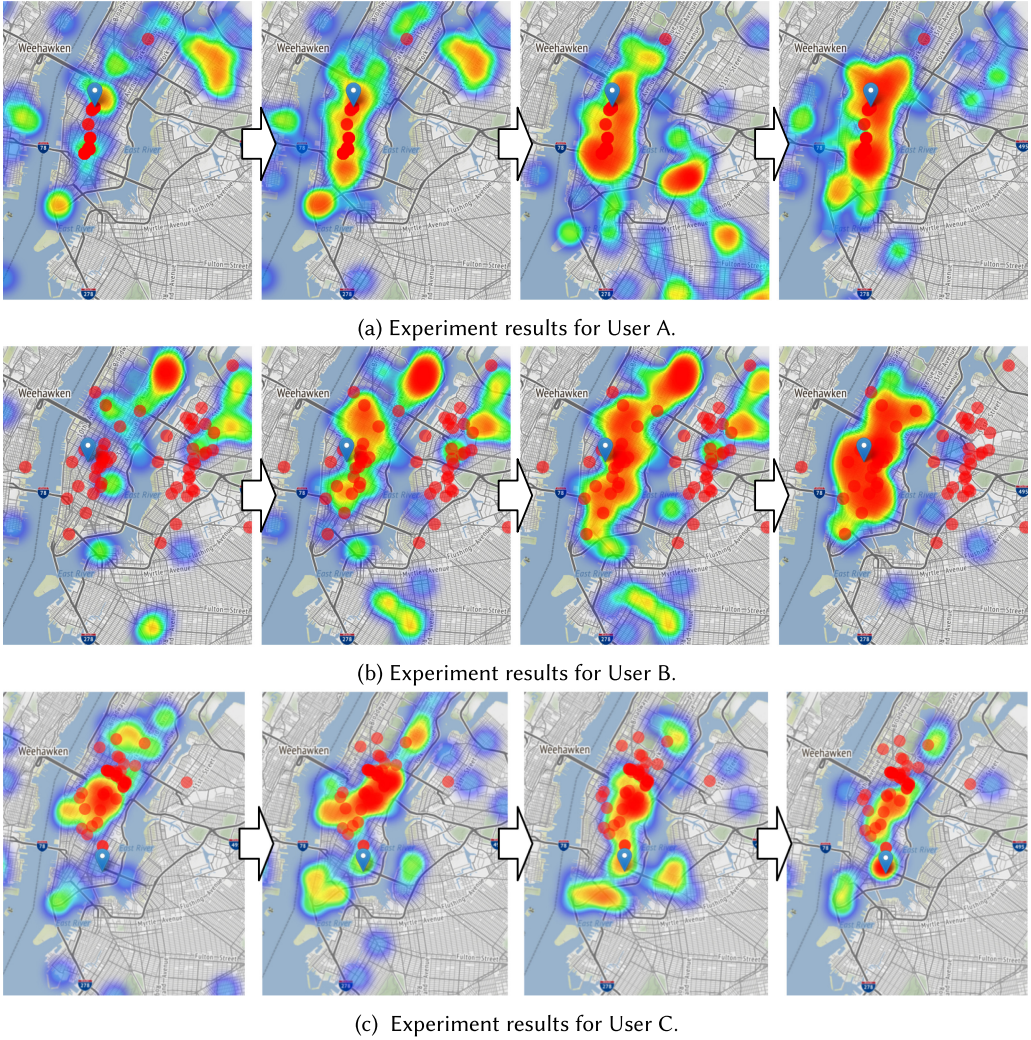(c) Experiment results for User C.

Fig. 9. 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.

sampling process proceeds. This intuitively shows that the diffusion-based sampling process is helpful to figure out the locational preference of specific users.

- Predicting POIs for regular users like User A is easier for the sampling module to depict. As illustrated in Figure 9(a), the historical visits of User 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 User B and locate the target POI through the sampling process.

- For adventurous users like User C, the sampling module plays a crucial role in capturing their future locational preferences. We can observe that even when the target POI is far from User C's historical visits, the sampling process of Diff-POI gradually guides the model

Table 4. Top-5 Recommended POIs Via the Sampled Spatial Preference of Users

	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*	0.00	Caribbean Food*	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*	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*	0.00
4	Medical Center	15.61	Greek Food	3.24	Pharmacy*	0.00	Bakery	1.33
5	Bar	9.62	Medical Center	15.61	Food & Drink	1.03	Deli	0.97
	Recommended POIs for User C w.r.t. Sampling Step							
Top-K	0%		33%		66%		100%	
	Category	Distance	Category	Distance	Category	Distance	Category	Distance
1	Bar	6.59	Apartment	10.01	Gastropub	1.36	Gastropub	1.36
2	Gastropub	1.36	Bar	6.59	Bar	6.59	Bar	6.59
3	Apartment	10.01	Neighborhood	4.07	Spa & Massage*	0.00	Spa & Massage*	0.00
4	Stadium	5.14	Gastropub	1.36	Bridge	1.34	Diner	0.92
5	Sushi Restaurant	3.22	Sushi Restaurant	3.22	Diner	0.92	Bridge	1.34

The category and distance (in kilometers) between the recommended POI and target POI (i.e., ground truth) is recorded in the table. The category marked with * represents the target POI.

to recommend POIs in the neighborhood around the ground truth. The observation reflects the capability of the sampling module in exploring personalized locational preference.

- Although 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 the target POI. As illustrated, when the sampled process converges, the predicted POIs are gathering around a relatively large area around the 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.

5.4.3 Case Study of the 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. In particular, 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 the following:

- The sampled representations reflect the locational characteristics of POIs. The recalled POIs are more likely to be in the same region, as the distance between the target and recalled POIs is 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 reverse 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 spatio-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 geo-
graphically close to the target POI, which makes them reasonable candidates as well.
- However, for users who are exploring novel areas (e.g., users B and C), simply aggregating
  the representations of historical visits would not be as effective. Meanwhile, as the sampling
  step increases, Diff-POI successfully samples the user's implicit locational preference, and
  there are more and more nearby POIs that appear in the top-5 recommended list. The target
  POIs are eventually found out by the model as well. The result illustrates the necessity to
  iteratively sample the locational preference to locate the accurate spot of the next-to-visit
  POIs.

6 CONCLUSION

In this article, we proposed a novel graph-based POI recommendation model, namely Diff-POI, to
exploit the spatio-temporal transition graph and model the user's locational preference. Diff-POI
is equipped with a tailor-designed sequence graph encoder that depicts the user's spatio-temporal
trajectory, followed by a diffusion-based sampling strategy to sample the locational preference of
the user. Comprehensive experiments on four real-world datasets demonstrated the effectiveness
of the proposed Diff-POI framework and its superiority over the existing state-of-the-art baseline
models. Ablation and case studies further provided an intuitive view of how the sampling process
depicts the user's hidden locational preference.

ACKNOWLEDGMENTS

REFERENCES

[1] Brian D. O. Anderson. 1982. Reverse-time diffusion equation models. *Stochastic Processes and Their Applications* 12,
3 (1982), 313–326.

[2] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. 2020. Structural deep clustering network. In
Proceedings of The Web Conference 2020. 1400–1410.

[3] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. 2013. Where you like to go next: Successive point-of-
interest recommendation. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*.

[4] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat GANs on image synthesis. *Advances in Neural
Information Processing Systems* 34 (2021), 8780–8794.

[5] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. 2018. DeepMove: Predict-
ing human mobility with attentional recurrent networks. In *Proceedings of the 2018 World Wide Web Conference*.
1459–1468.

[6] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, and Yeow Meng Chee. 2015. Personalized ranking metric embedding
for next new POI recommendation. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI
'15)*. ACM, New York, NY, 2069–2075.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and
Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM* 63, 11 (2020), 139–144.

[8] 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.

[9] 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 *Proceedings
of the 2020 IEEE International Conference on Data Mining (ICDM '20)*. IEEE, Los Alamitos, CA, 1052–1057.

[10] Ruining He and Julian McAuley. 2016. Fusing similarity models with Markov chains for sparse sequential recommen-
dation. In *Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM '16)*. IEEE, Los Alamitos,
CA, 191–200.

[11] 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.

[12] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[14] Zheng Huang, Jing Ma, Yushun Dong, Natasha Zhang Foutz, and Jundong Li. 2022. Empowering next POI recommendation with multi-relational modeling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2034–2038.

[15] Wei Ju, Zheng Fang, Yiyang Gu, Zequn Liu, Qingqing Long, Ziyue Qiao, Yifang Qin, Jianhao Shen, Fang Sun, Zhiping Xiao, Junwei Yang, Jingyang Yuan, Yusheng Zhao, Xiao Luo, and Ming Zhang. 2023. A comprehensive survey on deep graph representation learning. *arXiv preprint arXiv:2304.05055* (2023).

[16] Wei Ju, Yiyang Gu, Binqi Chen, Gongbo Sun, Yifang Qin, Xingyuming Liu, Xiao Luo, and Ming Zhang. 2023. GLCC: A general framework for graph-level clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4391–4399.

[17] Wei Ju, Zequn Liu, Yifang Qin, Bin Feng, Chen Wang, Zhihui Guo, Xiao Luo, and Ming Zhang. 2023. Few-shot molecular property prediction via hierarchically structured learning on relation graphs. *Neural Networks* 163 (2023), 122–131.

[18] Wei Ju, Xiao Luo, Meng Qu, Yifan Wang, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. 2023. TGNN: A joint semi-supervised framework for graph-level classification. *arXiv preprint arXiv:2304.11688* (2023).

[19] 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).

[20] Diederik P. Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[21] Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[22] Dejiang Kong and Fei Wu. 2018. HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI '18)*. 2341–2347.

[23] Zhifeng 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).

[24] 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).

[25] 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).

[26] 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 and Data Mining*. 2009–2019.

[27] 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.

[28] 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.

[29] 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*.

[30] Nicholas Lim, Bryan Hooi, See-Kiong Ng, Xueou Wang, Yong Liang Goh, Renrong Weng, and Jagannadan Varadarajan. 2020. STP-UDGAT: Spatial-temporal-preference user dimensional graph attention network for next POI recommendation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. 845–854.

[31] 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.

[32] Xingchao Liu, Chengyue Gong, and Qiang Liu. 2022. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003* (2022).

[33] Xiao Luo, Wei Ju, Meng Qu, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. 2022. DualGraph: Improving semi-supervised graph classification via dual contrastive learning. In *Proceedings of the 2022 IEEE 38th International Conference on Data Engineering (ICDE '22)*. IEEE, Los Alamitos, CA, 699–712.

[34] Xiao Luo, Yusheng Zhao, Yifang Qin, Wei Ju, and Ming Zhang. 2023. Towards semi-supervised universal graph classification. *IEEE Transactions on Knowledge and Data Engineering*. Published online May 31, 2023.

[35] 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.

[36] 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).

[37] Andriy Mnih and Russ R. Salakhutdinov. 2007. Probabilistic matrix factorization. *Advances in Neural Information Processing Systems* 20 (2007), 1–8.

[38] Yanjun Qin, Yuchen Fang, Haiyong Luo, Fang Zhao, and Chenxing Wang. 2022. Next point-of-interest recommendation with auto-correlation enhanced multi-modal transformer network. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2612–2616.

[39] Yifang Qin, Wei Ju, Hongjun Wu, Xiao Luo, and Ming Zhang. 2023. Learning graph ODE for continuous-time sequential recommendation. *arXiv preprint arXiv:2304.07042* (2023).

[40] Yifang 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 16th ACM International Conference on Web Search and Data Mining*. 508–516.

[41] 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.

[42] Xuan Rao, Lisi Chen, Yong Liu, Shuo Shang, Bin Yao, and Peng Han. 2022. Graph-flashback network for next location recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1463–1471.

[43] 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.

[44] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the International Conference on Machine Learning*. 1278–1286.

[45] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the International Conference on Machine Learning*. 2256–2265.

[46] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. 2023. Consistency models. *arXiv preprint arXiv:2303.01469* (2023).

[47] Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems* 32 (2019), 1–13.

[48] 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).

[49] 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.

[50] 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.

[51] 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).

[52] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-specific geographical influence for point-of-interest recommendation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI '18)*. 3877–3883.

[53] 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.

[54] Yifan Wang, Yifang Qin, Fang Sun, Bo Zhang, Xuyang Hou, Ke Hu, Jia Cheng, Jun Lei, and Ming Zhang. 2022. DisenCTR: Dynamic graph-based disentangled representation for click-through rate prediction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2314–2318.

[55] 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.

[56] 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* 16, 6 (2022), 1–21.

[57] 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.

[58] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 346–353.

[59] 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 on Conference on Information and Knowledge Management*. 15–24.

[60] 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).

[61] Song Yang, Jiamou Liu, and Kaiqi Zhao. 2022. GETNext: Trajectory flow map enhanced transformer for next POI recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1144–1153.

[62] Jingyang Yuan, Xiao Luo, Yifang Qin, Yusheng Zhao, Wei Ju, and Ming Zhang. 2023. Learning on graphs under label noise. In *Proceedings of the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '23)*. IEEE, Los Alamitos, CA, 1–5.

[63] 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 Information and Knowledge Management*. 659–668.

[64] 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).

[65] 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.

[66] 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.

[67] Yusheng Zhao, Xiao Luo, Wei Ju, Chong Chen, Xian-Sheng Hua, and Ming Zhang. 2023. Dynamic hypergraph structure learning for traffic flow forecasting. In *Proceedings of the 2023 IEEE International Conference on Data Engineering (ICDE '23)*.