DisenPOI: Disentangling Sequential and Geographical Influence for Point-of-Interest Recommendation

Yifang Qin*
qinyifang@pku.edu.cn
Peking University
Beijing, China

Yifan Wang*
yifanwang@pku.edu.cn
Peking University
Beijing, China

Fang Sun*
fts@pku.edu.cn
Peking University
Beijing, China

Wei Ju†
juwei@pku.edu.cn
Peking University
Beijing, China

Xuyang Hou
houxuyang@meituan.com
Meituan
Beijing, China

Zhe Wang
wangzhe65@meituan.com
Meituan
Beijing, China

Jia Cheng
jia.cheng.sh@meituan.com
Meituan
Beijing, China

Jun Lei
leijun@meituan.com
Meituan
Beijing, China

Ming Zhang†
mzhang_cs@pku.edu.cn
Peking University
Beijing, China

ABSTRACT

Point-of-Interest (POI) recommendation plays a vital role in various location-aware services. It has been observed that POI recommendation is driven by both sequential and geographical influences. However, since there is no annotated label of the dominant influence during recommendation, existing methods tend to entangle these two influences, which may lead to sub-optimal recommendation performance and poor interpretability. In this paper, we address the above challenge by proposing **DisenPOI**, a novel **Disen**tangled dual-graph framework for **POI** recommendation, which jointly utilizes sequential and geographical relationships on two separate graphs and disentangles the two influences with self-supervision. The key novelty of our model compared with existing approaches is to extract disentangled representations of both sequential and geographical influences with contrastive learning. To be specific, we construct a geographical graph and a sequential graph based on the check-in sequence of a user. We tailor their propagation schemes to become sequence-/geo-aware to better capture the corresponding influences. Preference proxies are extracted from check-in sequence as pseudo labels for the two influences, which supervise the disentanglement via a contrastive loss. Extensive experiments on three datasets demonstrate the superiority of the proposed model.

CCS CONCEPTS

• **Information systems → Recommender systems**.

---

*Authors contributed equally to this research.
†Co-corresponding authors.

KEYWORDS

POI Recommendation, Graph Neural Networks, Disentangled Representation Learning

1 INTRODUCTION

The rapid growth of Location-Based Social Networks (LBSNs) has facilitated the development of Point-of-Interest (POI) recommendation, which lies at the heart of various location-based services, including location-based advertising and online food delivery. POI recommendation is the backbone for numerous popular Apps, such as Foursquare, Gowalla and Meituan. A POI can be a restaurant, a hotel, a scenic spot, or any location-tagged item. Location is much more than an ordinary feature dimension in POI recommendation: it builds up a network structure among POIs, inter-connected by both their interactive history sequence and geographical affinity.

Recent works on POI recommendation endeavor to integrate geographical influences with sequential transitions of user check-in data. Sequence-based methods like LSTPM [24] combine user's long-term interest with geo-dilated short-term interest, using geographical affinity as an auxiliary in modeling sequence behavior. GeoIE [28] explicitly uses physical distance to model user-specific geographical influence and GTAG [40] performs preference propagation based on physical distance and time-slot affinity. Despite the versatility of these POI recommenders, there are rarely approaches that explicitly uncover the collaborative signals. The complex interactions among users and POIs can be readily delineated by graphs, therefore a promising approach is to leverage the rich graph structure to exploit high-order connectivity between POIs.

Graph Neural Networks (GNNs) have been extensively deployed for recommendation to better capture the similarities between high-order neighbors. NGCF [30] and LightGCN [10] collects the collaborative signal on the user-POI bipartite graph, SR-GNN [36] propagates along the user's interactive sequence graph. Although these paradigms provide valuable insight as to organizing interaction data into graph structures for POI recommendation, they do not leverage geographical relationships, which is a prominent characteristic in POI recommendation.

In fact, sequential influence and geographical influence should be regarded as equal counterparts, which are two of the major driving forces in POI recommendation. As the example in Figure 1 shows, a user's visit to v_4 is influenced by its neighboring nodes from geographical and sequential graphs. On the one hand, users are apt to re-visit familiar POIs that have already shown up in their interaction sequences. For example, frequent visits to clothing stores may reveal a user's preference in fashion, which indicates that a similar clothing store has more chance to be the next-to-visit POI. On the other hand, POIs in the vicinity are also more likely to be visited. For instance, a user who has just come out of a downtown shopping mall is more likely to have lunch in a nearby fast-food restaurant, rather than his/her favorite restaurant in the countryside. The two influences affect user behaviors in inherently different ways, which the previous approaches for POI recommendation do not try to distinguish. Since these approaches impose no explicit supervision on the learned user preference, there is no guarantee for the explicit capture of sequential influence and geographical influence in the user preference.

In face of the dual influences on user preference, the key is not simple combination, but rather, **disentanglement**. Specifically, to depict user preference at a finer granularity and enhance POI recommendation quality, we need to disentangle the two latent driving forces, thus distilling the representations for both influences provides better interpretability. Existing disentanglement models for recommendation either use dynamic routing (e.g. DGCF [32], DisenCTR [33]) or self-supervised signals (e.g. MacridVAE [20], CLSR [42]) to disentangle latent interests, yet how to disentangle sequential and geographical influences remains unexplored.

In a nutshell, the ubiquity and uniqueness of the two influences in LBSNs pose two major challenges to POI recommendation: **First**, how to **depict** these two high-order connectivity influences for the user check-in data? **Second**, How to explicitly **disentangle** these two influences for POI recommendation?

To address the aforementioned challenges, we propose **Disen-POI**, a novel **Disen**tangled dual-graph framework for **POI** Recommendation, which explicitly models sequential and geographical influences on user check-in data. DisenPOI constructs two disentangled POI graphs: a geographical graph based on spatial affinity, and a sequential graph based on interaction history. To better capture high-order connectivity on different graph topologies, a distance-aware and a sequence-aware GNN are proposed to propagate on the corresponding geographical and sequential graph. We adopt contrastive learning methods to disentangle the graph representations in a self-supervised way.With the help of disentangled representations, DisenPOI can make recommendations that reflect both sequential and geographical influence.
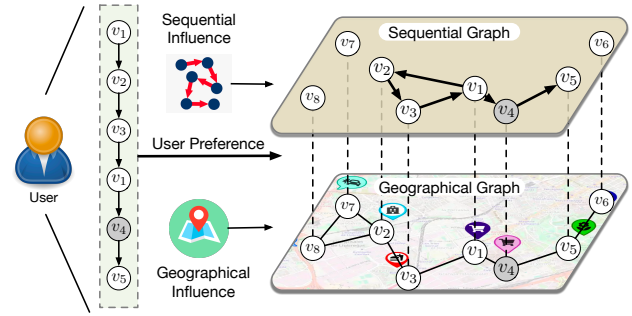
We summarize our main contributions as follows:

Figure 1: An illustration of disentangled influences behind check-ins. A visit to v_4 is influenced by its neighbouring nodes from geographical and sequential graph.

- We construct dual graphs from user's visiting interactions to jointly utilize both sequential and geographical relationships, devising sequence- and geo-aware propagation schemes for the two graphs respectively to improve embedding quality.
- We propose to extract disentangled representations of sequential and geographical influences. Sequential and geographical proxies are obtained as self-supervised signals, which are specially tailored for the LBSN setting. To the best of our knowledge, this is the first graph-based POI recommendation framework with disentangled sequential and geographical representations.
- We conduct extensive experiments on three real-world benchmarks for POI recommendation. Our model consistently outperforms strong baselines, achieving state-of-the-art performance on all three benchmark datasets.

2 RELATED WORK

2.1 Location-based POI Recommendation

Location-based POI recommendation puts special emphasis on the geographical features of POIS. GeoIE [28] models geographical influence between two POIs by the inner product of geo-influence and geo-susceptibility vectors. PACE [38] combines Matrix Factorization (MF) [23] approach with self-supervised learning to better exploit location information. There are also works [4, 17, 22] that leverage Markov Chain for next-POI prediction. This line of work [6, 18, 24, 27, 41] is characterized by their wide use of Recurrent Neural Networks (RNNs) for modeling sequential data. For example, LSTPM [24] combines geographical information with sequential method by estimating the spatial-temporal similarities and finding shortcuts between POIs. There are also works [3, 7, 8, 14] that consider the social influence between users.

2.2 GNN-based Recommendation

Graph Neural Networks (GNNs) like Graph Convolutional Network (GCN) [13], Gated Graph Neural Network (GGNN) [15] and Graph Attention Network (GAT) [26] have been consistently showing their potency with graph-structured data. GNNs learn the hidden representation of a graph node by passing and aggregating messages collected from its neighboring nodes. By stacking GNN layers, models[12, 19] are empowered to exploit high-order graph connectivity. Graph structures naturally exist among users and

items, which makes GNNs ideal for modeling recommendation tasks. NGCF [30] and its variant LightGCN [10] consider the collaborative signals on a user-item bipartite graph. DSGL [5] uses dynamic sequential graphs to model the dynamics of user's interaction history. Recently, works like KBGNN [11], GSTN [35], GEAPR [16] and IMUP [29] propose to integrate geographical factors into GNN-based recommendation to improve performance under location-based settings. However, every observed interaction can be driven by various ambient influences, as well as diverse user interests. Representations learned by GNNs cannot readily uncover these rich semantics.

2.3 Disentangled Learning in Recommendation

Disentangled Representation Learning [1] aims to learn separate explanatory representations from different latent factors behind data. MacridVAE [20] employs Variational Auto-Encoders (VAEs) to encode items via disentangled prototypes. CLSR [42] attempts to disentangle users' long and short-term interests from their clicking sequences. GNNs are also used in disentangled recommenders, which propose to learn disentangled representation from various graph structures. Such models include DGCF [32] (user-item graph), DisenHAN [34] (heterogeneous graph) and KGIN [31] (knowledge graph). Yet, the aforementioned works only perform disentangled learning on one type of graph topology. They are not intended to capture influences from POI locations, either. These short-comings shall be addressed by our proposed DisenPOI model, which uses two disentangled graphs to utilize sequential and geographical relationships, and extracts the rich semantics behind observed interactions by disentangling sequential and geographical influences.

3 PRELIMINARY

3.1 Problem Formulation

Location-based Click Through Rate (CTR) prediction aims to leverage geographical information in user's interactive history with POIs. We denote the set of users as $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$, and the set of POIs as $\mathcal{V} = \{v_1, v_2, ..., v_{|\mathcal{V}|}\}$, where each POI $v \in \mathcal{V}$ is geo-encoded by its coordinates $(longitude, latitude)$ tuple, i.e. (lon_v, lat_v). For each user $u \in \mathcal{U}$, his/her check-in sequence is organized into a list $s_u = [v_{u,1}, v_{u,2}, ..., v_{u,t-1}]$, containing the $t-1$ POIs he/she has checked in, sorted by timestamp. Given user u and target POI v, the goal of location-based CTR prediction is to predict the probability for u to visit v, formulated as $\hat{y}_{uv} = f(u, v|s_u; \theta)$, where f is a learnable function parameterized by θ.

3.2 Construction of Two Disentangled Graphs

As we aim to make predictions based on both the geographical relationship between POIs and the sequential dependency in users' check-in history, two POI graphs with different propagation schemes are thus constructed to meet the two ends respectively.

3.2.1 Geographical Graph. The undirected geographical graph $\mathcal{G}_g = \{\mathcal{V}, \mathcal{E}_g, A_g\}$ is built upon the location of POIs. An undirected edge $e_g = (v_i, v_j) \in \mathcal{E}_g$ indicates that the distance between v_i and v_j is within a specific distance threshold Δd. The edge weight matrix $A_g(i, j)$ is the geographical distance between POI v_i and v_j. The

construction of $\mathcal{G}_g$ is based on the fact that a user would usually prefer a nearby POI for the next visit.

3.2.2 Sequential Graph. Inspired by previous work on session-based recommendation [36], we propose to construct sequential graphs for visiting sequences. Given a user u and his/her check-in history s_u, a directed sequential graph $\mathcal{G}_{s,u} = \{\mathcal{V}_{s,u}, \mathcal{E}_{s,u}\}$ can then be constructed. For each edge $e_s = \langle v_i, v_j \rangle \in \mathcal{G}_s$, it represents that the user has successively check-in to v_j after v_i in s_u. Thus, $\mathcal{G}_{s,u}$ gleans sequential information from the check-in history.

4 THE PROPOSED MODEL

4.1 Overview

The proposed model has four modules as illustrated in Figure 2. Given a user's check-in history and a target POI, the model first maps IDs of the involved POIs into a hidden space, i.e. $X \in \mathbb{R}^{D \times D}$, to get their initial embeddings. After the POIs are mapped, the model constructs two graphs, G_g and G_s, which reflect the geographical and sequential features of the given visiting history respectively. Two graph propagation modules are used upon the drawn graphs to generate representations from a disentangled view. A soft-attention mechanism is applied to extract the user preference behind visits on both modules, in order to dynamically measure the influence of visited and nearby POIs on the target POI. To explicitly disentangle this hybrid influence into two independent fields, we introduce a contrastive auxiliary loss to ensure that each module is able to extract disentangled information from the corresponding POI graph. Finally, we use an MLP layer to aggregate these disentangled representations and generate CTR prediction for the target POI.

During the training process, we adopt a curriculum learning scheme to integrate the contrastive loss with the target function. We control the curriculum weight to train the model in an easy-to-difficult manner so that our model can be trained adequately to give valid predictions while maintaining effective disentanglement.

4.2 Propagation on Disentangled Graphs

The graph propagation layer encodes the constructed graphs and outputs two sets of POI representations with geographical and sequential information. Since the two disentangled graphs are organized according to sequential/geographical relationships, we apply sequence-/geo-aware graph propagation schemes on them respectively, to better exploit the intrinsic characteristics of each graph.

4.2.1 Geographical Graph Propagation Layer. Given the location-based graph $\mathcal{G}_g$, we build our geographical graph propagation layer using the message-passing scheme of GNNs [9, 37].

Message Construction. For a pair of neighboring POIs v_i, v_j, the message on the l-th layer is defined as:

$$m_{j \leftarrow i}^{(l)} = f_d(h_i^{(l-1)}, h_j^{(l-1)}), \qquad (1)$$

where $f_d(\cdot)$ is the message encoding function that takes item representation $h^{(l-1)}$ from the previous GNN layer. Specifically, the input of the first layer is initialized with POI's embedding X:

$$h_i^{(0)} = x_i, \forall v_i \in \mathcal{V}. \qquad (2)$$

In order to better leverage the geographical information in $\mathcal{G}_g$, the message function $f_d(\cdot)$ is expected to reflect the distance influence
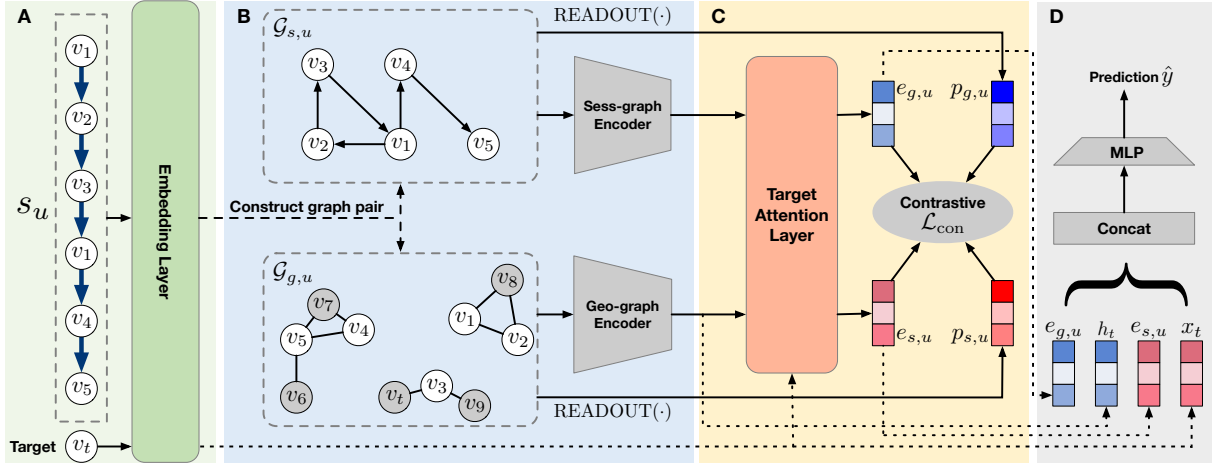
Figure 2: Illustration of DisenPOI. A) Embedding Layer for POIs; B) graph propagation layer that generates disentangled representations $e_{g,u}, e_{s,u}$ and their proxies $p_{g,u}, p_{s,u}$; C) soft-attention layer that aggregates POI embeddings for contrastive loss and model prediction; D) prediction layer to make predictions.

between POIs. Specifically, POIs that are closer tend to share more similarities than distant ones. Formally, $f_d(\cdot)$ is implemented as:

$$m_{j \leftarrow i}^{(l)} = \frac{1}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}}(W_1^{(l)} h_i^{(l-1)} + w(d_{ij})W_2^{(l)} h_i^{(l-1)} \odot h_j^{(l-1)}), \quad (3)$$

where $W_1, W_2 \in \mathbb{R}^{D \times D}$ are trainable weight matrices as linear transformations for lower layer representations. To take the effect of distance into consideration, an element-wise production item is included. Distance kernel $w(d_{ij}) = e^{-d_{ij}^2}$ decays exponentially as d_{ij}, the distance between v_i and v_j grows. We use the graph Laplacian norm like GCNs [13], where $\mathcal{N}_i$ and $\mathcal{N}_j$ denote the first-hop neighbors for v_i and v_j on $\mathcal{G}_g$ respectively.

Message Aggregation. To generate node representations on each GNN layer, the message aggregation function is defined as:

$$h_j^{(l)} = \text{LeakyReLU}(m_{j \leftarrow j} + \sum_{i \in \mathcal{N}_u} m_{j \leftarrow i}), \quad (4)$$

where $m_{j \leftarrow j} = W_1 h_j^{(l-1)}$ is induced by the representation of v_j.

After propagating along L GNN layers, we adopt the hidden representation of the L-th layer as the geographical encoding of all the POIs in $\mathcal{V}$:

$$H_g = [h_1^{(L)}; h_2^{(L)}; ...; h_{|\mathcal{V}|}^{(L)}]. \quad (5)$$

As for user u with check-in history s_u, we assume that his/her geographical preference can be captured by aggregating the high-order geographical neighbors of POIs in his/her check-in history. In practice, the user's geographical encoding $H_{g,u}$ is the list of the geographical embeddings of all the POIs that appear in s_u.

4.2.2 Sequential Graph Propagation Layer. Given a target user u and the constructed sequential graph $\mathcal{G}_{s,u}$, we obtain the hidden representation of each item in the user's visiting history to exploit sequential influence. A user's visit sequence contains not only information about his preferences on POIs, but also the evolving history of his interests, which implies his tendency for the next POI to visit. Since plain GCNs are sequential-invariant, we consider

adopting a graph propagation method that can fully leverage sequential information in the session graph. For the given reasons, we apply Gated Graph Neural Networks (GGNNs) [15] on $\mathcal{G}_{s,u}$ to better leverage the sequential information. For each node $v_i \in \mathcal{G}_{s,u}$ and its embedding x_i, the hidden representation of v is updated as:

$$h_v^{(1)} = [x_i^\top, 0]^\top \quad (6)$$

$$a_v^t = A_{v:}^\top [h_1^{(t-1)\top}, ..., h_{|\mathcal{V}|}^{(t-1)\top}]^\top + b \quad (7)$$

$$z_v^t = \sigma(W^z a_v^{(t)} + U^z h_v^{(t-1)}) \quad (8)$$

$$r_v^t = \sigma(W^r a_v^{(t)} + U^r h_v^{(t-1)}) \quad (9)$$

$$\widetilde{h_v^{(t)}} = \tanh(W_o a_v^{(t)} + U_o(r_v^t \odot h_v^{(t-1)})) \quad (10)$$

$$h_v^{(t)} = (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \widetilde{h_v^{(t)}}, \quad (11)$$

where connection matrix $A_v \in \mathbb{R}^{|\mathcal{V}| \times 2|\mathcal{V}|}$ determines the connectivity of nodes in $\mathcal{G}_{s,u}$, Ws and Us are trainable parameters. Inspired by Gate Recurrent Unit (GRU), GGNN proceeds on all nodes of the sequential graph sequentially. The propagation of hidden states is controlled by an update gate z_v^t and a reset gate r_v^t. The gated controlled propagation on $\mathcal{G}_{s,u}$ models the sequential effect behind user behavior. We obtain the output of the last hidden layer as the sequential encoding of POIs:

$$H_{s,u} = [h_{u,1}^{(t)}, h_{u,2}^{(t)}, ..., h_{u,|s_u|}^{(t)}] \quad (12)$$

4.3 Soft-attention Mechanism

After encoding the two constructed graphs from user's previous visiting history, we introduce a soft-attention mechanism to better aggregate the encodings according to current target POI v_t. For geographical graph encoding $H_{g,u} = [h_1, h_2, ..., h_{|s_u|}]$, we denote v_t's hidden representation h_t as the query, the soft-attention mechanism

to generate geographical embedding $e_{g,u}$ is formulated as:

$$w_i = \alpha_g^\top \sigma(Q_g h_t + K_g h_i) \tag{13}$$

$$e_{g,u} = \sum_{i=1}^{|s_u|} w_i h_i, \tag{14}$$

where σ denotes sigmoid function, the attention vector $\alpha_g \in \mathbb{R}^D$, query and key matrix $Q_g, K_g \in \mathbb{R}^{D \times D}$ are trainable parameters. Similarly, for $H_{s,u} = [h_1', h_2', ..., h_{|s_u|}']$, we take v_t's initial embedding x_t as query vector to generate sequential embedding $e_{s,u}$:

$$w_i' = \alpha_s^\top \sigma(Q_s x_t + K_s h_i') \tag{15}$$

$$e_{s,u} = \sum_{i=1}^{|s_u|} w_i' h_i', \tag{16}$$

4.4 Self-supervised Disentanglement

Since the sequential and geographical effects have separate influences on next-to-visit POI, it is essential to disentangle the two representations, $e_{g,u}$ and $e_{s,u}$, from each other so that model could fully utilize the information for CTR prediction. Inspired by previous work on self-supervised disentanglement learning [42], we suggest obtaining proxies for both embeddings. Specifically, two readout functions are applied to the outputs of the graph propagation layer. We choose mean pooling as readout function to apply to both geographical neighbors and user's previous visited POIs to generate two proxies for $e_{g,u}$ and $e_{s,u}$. Formally, we have:

$$p_{g,u} = \text{READOUT}(\{x_j | v_j \in \mathcal{N}_{s_u}\}) = \frac{1}{\sum_{v_i \in s_u} |\mathcal{N}_i|} \sum_{v_i \in s_u} \sum_{v_j \in \mathcal{N}_i} x_j \tag{17}$$

$$p_{s,u} = \text{READOUT}(\{x_i | v_i \in s_u\}) = \frac{1}{|s_u|} \sum_{v_i \in s_u} x_i, \tag{18}$$

which conduct mean pooling on s_u's one-hop geographical neighbors $\mathcal{N}_{s_u}$ and s_u respectively. Since $p_{g,u}$ and $p_{s,u}$ are readout from two different graphs, it's reasonable to assume that each proxy carries representative information of its corresponding graph features. To be specific, we assume that $p_{g,u}$ carries abundant information about the user's geographical preference, while $p_{s,u}$ carries information about the user's historical interests.

Before the embeddings and their proxies were sent to generate self-supervising signals, we design projection heads for each disentangled representations following previous work [2]. To formulate, the projected representations are calculated as:

$$e_{g,u}' = \text{proj}_g(e_{g,u}), \ p_{g,u}' = \text{proj}_g(p_{g,u}) \tag{19}$$

$$e_{s,u}' = \text{proj}_s(e_{s,u}), \ p_{s,u}' = \text{proj}_s(p_{s,u}), \tag{20}$$

where $\text{proj}_g, \text{proj}_s : \mathbb{R}^D \to \mathbb{R}^D$ are linear transformations with trainable parameters. The projection heads project the embeddings into another latent space where information from different aspects can be easily clarified.

We treat the projected pooling representations, $p_{g,u}'$ and $p_{s,u'}$ as hidden proxies for $e_{g,u}'$ and $e_{s,u}'$, which brings supervising signals for self-supervised disentanglement. In other words, the two proxies serve as positive samples for the corresponding embedding,

while serving as negative samples for each other. With the above assumption, we introduce the following contrastive loss:

$$\mathcal{L}_{\text{con}} = f(e_{g,u}', p_{g,u}', p_{s,u}') + f(e_{s,u}', p_{s,u}', p_{g,u}'), \tag{21}$$

where $f(\cdot)$ denotes the Bayesian Personalized Ranking loss [21]. Formally, we have:

$$f(a, p, q) = \text{Softplus}(\langle a, q \rangle - \langle a, p \rangle), \tag{22}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of two given representations.

4.5 Location-based CTR Prediction Layer

4.5.1 Model Prediction. After obtaining the disentangled embeddings of the user's visiting history $e_{g,u}$ and $e_{s,u}$, we concatenate the embeddings with target v_t's embedding x_t and geographical representation h_t. A 2-layer MLP is applied to predict the click-through rate of v_i, i.e. $\hat{y}$, which is a common practice in CTR prediction [43, 44]. Formally, $\hat{y}$ is calculated by:

$$\hat{y} = \sigma(\text{MLP}(\text{CONCAT}(e_{g,u}, e_{s,u}, x_t, h_t))) \tag{23}$$

4.5.2 Target Function and Curriculum Learning. Given the label y, we adopt binary cross-entropy loss for supervised CTR prediction, which is formulated as:

$$\mathcal{L}_{\text{rec}} = - \sum_{(u,v)} y \log \hat{y} + (1-y) \log(1-\hat{y}). \tag{24}$$

After adding the disentangled constraint to the model, we have the overall model loss:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \beta * \mathcal{L}_{\text{con}}, \tag{25}$$

where β is the weight of disentanglement. To optimize the model, we propose a curriculum learning method to have the training process follow an easy-to-difficult process. To be specific, we train the model with a warm-up procedure by dynamically increasing the weight of the contrastive loss:

$$\beta = \max\{\alpha, \gamma * k\}, \tag{26}$$

where α and γ are hyper-parameters, k denotes the current curriculum number. Under curriculum training, the strength of disentanglement would increase to the maximum.

5 EXPERIMENT

In this section, we conduct comprehensive experiments on three real-world location-based recommendation datasets to answer the following research questions:

RQ1: How does the proposed DisenPOI perform compared with the current state-of-art baseline methods on location-based CTR predictions? How can DisenPOI alleviate the data sparsity issue with the help of disentangled information?

RQ2: Is the idea of disentangling geographical and sequential influence necessary to achieve a promising model performance? What are the influences of the different hyper-parameters?

RQ3: How effectively does DisenPOI disentangle POI representations? Can DisenPOI recommend POIs with marked geographical structure as expected?

Table 1: Descriptive statistics of our three datasets.

Dataset	#User	#POI	Interactions	Avg.SeqLen
Tokyo	2,293	61,858	573,703	250.20
New York	1,083	38,333	227,428	210.00
Meituan	164,855	5,077	470,095	2.85

5.1 Experimental Setup

5.1.1 Evaluation Datasets. We conduct experiments on three real-world datasets collected from two platforms, namely **Foursquare** [39] and **Meituan** to evaluate the proposed model. The detailed statistics of the three datasets are presented in table 1.

(1) **Foursquare**[1]: It contains two subsets, which are users' check-in records collected in Tokyo and New York City respectively from 12 April 2012 to 16 February 2013. There are 573,703 check-ins in Tokyo dataset and 227,428 check-ins in New York City (NYC).

(2) **Meituan**: It is a dataset drawn from Meituan users' check-ins to restaurants from 7 July 2021 to 20 July 2021. We collect the records in Shanghai, which has 470,095 check-ins, with 164,855 users and 5,077 POIs included.

We sort the recorded user interactions in each dataset in chronological order. All visited POIs and the corresponding previous visiting subsequences make up the positive samples. For each positive sample, we randomly select one of the user's unvisited POIs as the negative sample. The last interaction of each user is reserved for evaluation, while the remaining part is used for training. The evaluation set is then randomly split into two equal-sized subsets as test and validation set respectively.

5.1.2 Baselines. To demonstrate the effectiveness of the proposed DisenPOI, we compare it with three classes of baselines: (A) Traditional sequence-based methods; (B) Graph-based methods; and (C) Location-based methods which leverage geographical information.

- **DIN**[44] (A): a classical method for sequence-based CTR prediction that uses target attention for user's context embedding.
- **DIEN**[43] (A): a variant of DIN which leverages GRUs to capture the evolution of user interest.
- **SR-GNN**[36] (B): a graph-based method that models user's interaction history as session graphs.
- **NGCF**[30] (B): a graph-based method that applies GNN on the user-item interaction graph.
- **LightGCN**[10] (B): a variant of NGCF. It's one of the state-of-art methods for graph-based recommendation.
- **GeoIE**[28] (C): one of the state-of-art location-based recommendation methods, which takes the influence of geographical location and distance into consideration.
- **LSTPM**[24] (C): an LSTM-based method that proposes the geo-dilated LSTM to model geographical effect.
- **GSTN**[35] (C): a location-based graph recommendation model that optimizes the spatial-temporal similarities between POIs.

Since we have constructed training and evaluation set with positive and negative samples, we adopt AUC and Logloss as evaluation

[1]https://sites.google.com/site/yangdingqi/home/foursquare-dataset

Table 2: The performance of DisenPOI (Ours) and other baseline methods over three datasets. ⋆ indicates DisenPOI outperforms the best POI at p-value<0.05 level, unpaired t-test.

Model	Tokyo		NYC		Meituan	
	AUC↑	Logloss↓	AUC	Logloss	AUC	Logloss
DIN	0.8623	0.4174	0.8015	0.5203	0.5596	0.6920
DIEN	0.8643	0.4544	0.8062	0.5582	0.5591	0.6891
SR-GNN	0.8867	0.4481	0.8491	0.5262	0.5751	0.6844
NGCF	0.8899	0.4436	0.8461	0.4788	0.5698	0.6857
LightGCN	0.8998	0.4348	0.8729	0.4815	0.5682	0.6859
GeoIE	0.9083	0.4203	0.8769	0.4810	0.5431	0.6926
LSTPM	0.8745	0.4374	0.8568	0.4821	0.5637	0.6867
GSTN	0.8906	0.4312	0.8521	0.5323	0.5807	0.6845
Ours	**0.9246⋆**	**0.3608⋆**	**0.8941⋆**	**0.4248⋆**	**0.5823⋆**	**0.6836⋆**

metrics, which is a common practice in CTR prediction [43, 44]. The experiment results on three datasets are shown in table 2.

5.1.3 Implementation Detail. We implement our DisenPOI and all baselines in Pytorch. The embedding size is fixed to 64. While some of the above baseline models are not intended for CTR prediction, we apply the same-sized 2-layer MLP on the backend of each model to compare their performances on CTR prediction on an equal basis. For DisenPOI, we set hyper-parameters $\alpha = 0.2, \gamma = 0.004$ for training and evaluation and we use 2 layers of GNNs on both graph modules. The distance threshold Δd is set to 1km when constructing geographical graphs. All models are optimized with Adam optimizer with the learning rate $lr = 0.001$. Our implementation is publicly available at https://github.com/Fang6ang/DisenPOI

5.2 Performance Comparison (RQ1)

5.2.1 Overall Comparison. As shown in table 2, methods that leverage geographical information (GeoIE, LSTPM, GSTN, and DisenPOI) generally outperform other models, which is a reasonable result since the high relativity between user's check-in history and POIs' geographical locations in location-based recommendation tasks. The proposed DisenPOI outperforms all of the state-of-art baselines with a significant improvement. In particular, the testing AUC is improved over the strongest baseline w.r.t AUC by 1.7%, 2.0%, 0.3%; Logloss by 1.3%, 1.1%, 0.2% in **Foursquare Tokyo**, **Foursquare NYC** and **Meituan** respectively. The result shows that DisenPOI achieves the best performance on CTR prediction task.

5.2.2 Performance on Cold-start Recommendation. To investigate whether the idea of disentangling sequential and geographical influence behind visitings can help to alleviate the cold-start issue, we randomly divide each user's visiting sequences into five folds, corresponding to 20%, 40%, 60%, 80%, and full train set. The experiment results are illustrated in Figure 3. We can observe that:

- Generally, location-based methods (DisenPOI, GeoIE) can better alleviate the cold-start issue, which indicates that geographical information is essential when dealing with sparse interactions.
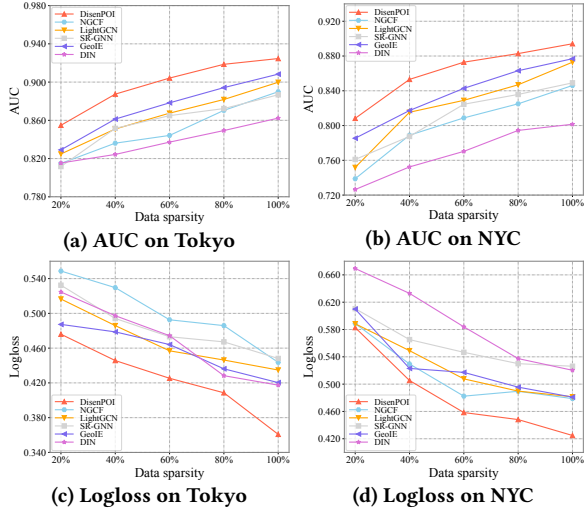
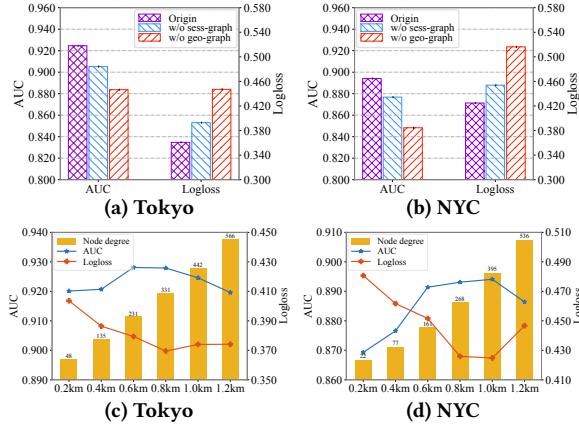**Figure 3: Model performance under cold-start settings.**

Figure 4: Analysis of the dual graph module of DisenPOI.

- DisenPOI outperforms all baselines under each setting, showing its capability of learning effective representations on sparse data. Specifically, DisenPOI gains more advantages when the training data is rather sparse (under 60%).

5.3 Analysis of DisenPOI (RQ2)

We conduct ablation studies to figure out the functionality and effectiveness of each part in DisenPOI. The following ablation studies are carried out on two **Foursquare** datasets.

5.3.1 Effectiveness of Dual Graphs. To analyze how DisenPOI utilizes the graph data from both geographical and sequential perspectives, we evaluate the model performance under three different settings: the original model, the model without geographical graph (w/o geo-graph), and the model without sequential graph (w/o sess-graph). Since the POI graph $\mathcal{G}_g$ is built based on a specified distance threshold Δd, we adjust its value between 0.2 km and 1.2 km to analyze the model's sensitivity to $\mathcal{G}_g$. Figure 4 shows:

- The outputs of both graph modules play different roles to make recommendations. Whenever removing a type of graph, the model performance declines significantly.
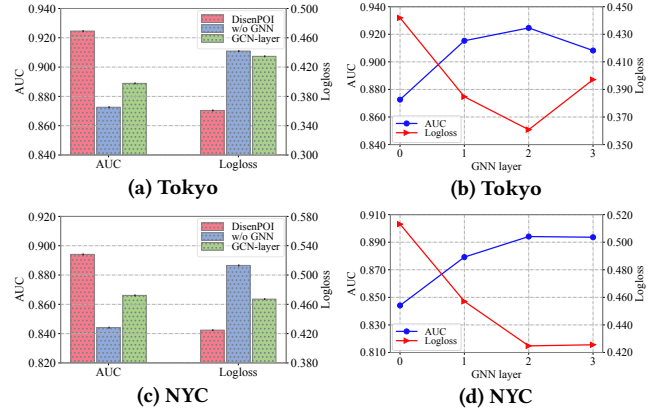

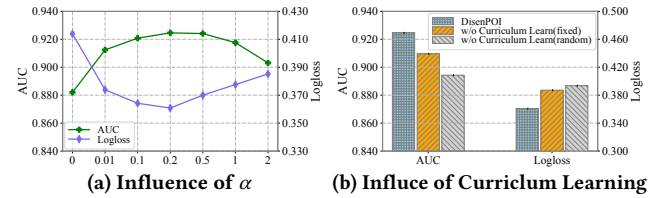**Figure 5: Ablation study of GNN layer on Foursquare Tokyo.**

Figure 6: Performance under different contrastive settings.

- When the distance threshold Δd increases, the density of $\mathcal{G}_g$ (average node degree) increases correspondingly, the model performance increase to the optimal and then suffer from a decline. A reasonable Δd is 1km, which is close to real world situations.

5.3.2 Difference Between Graph Propagation Methods. As we have proposed two different graph propagation methods for different types of graphs respectively, we investigate the effectiveness of the proposed graph propagation layers. To be specific, we compare model performance when the number of propagation layers is set from 0 (w/o GNN) to 3. We also compare the performance when the GNN layers are replaced by graph convolution networks (i.e. GCN[13]). Based on the results shown in Figure 5, we have:

- It's important to design specific GNN structures for different graph topologies. Since we adapt two different types of GNNs to better capture the geographical and sequential influence, the model performance would decline once the graph propagation layers are replaced with plain GCNs.
- The model achieves its best performance when 2 GNN layers are used, under which setting the model could leverage high-order connectivity of POI nodes on the graph. When there is no GNN layers or the layer amount larger than two, the model would suffer from degenerating or over-smoothing issue of GNNs.

5.3.3 Influence of Disentanglement And Curriculum Learning. The contrastive loss $\mathcal{L}_{con}$ in the proposed model serves as a constraint on the model to ensure the similarity between the corresponding graph embeddings and the disentanglement between different types of graph embeddings. To figure out the effectiveness of $\mathcal{L}_{con}$, we compare the model performance with different contrastive weight α from 0 (w/o contrastive loss) to 2.0. Since the curriculum learning method is applied so that the model converges to a better solution, we compare the model performance under three curriculum
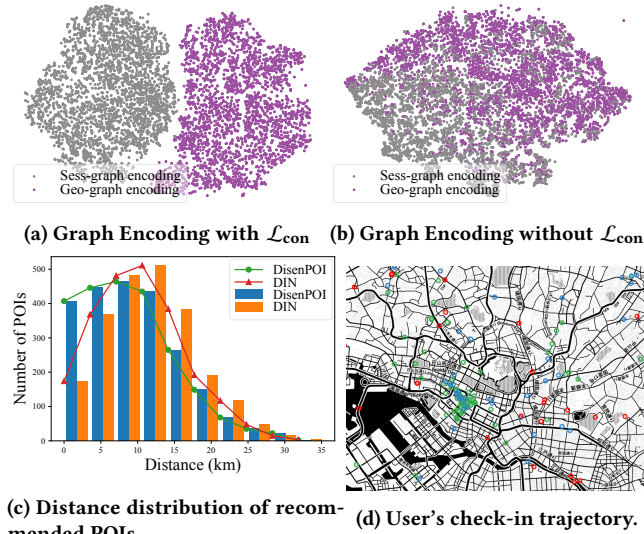
(a) Graph Encoding with $\mathcal{L}_{\mathbf{con}}$ (b) Graph Encoding without $\mathcal{L}_{\mathbf{con}}$

(c) Distance distribution of recommended POIs.

(d) User's check-in trajectory.

Figure 7: Visualizations of graph encodings and geographical distribution of recommended POIs.

Table 3: Difference of top recommended POIs.

DisenPOI		DIN	
Distance(km)	Category	Distance(km)	Category
0.310	Road	4.727	Train Station
1.331	Café	3.908	Subway
0.780	Smoke Shop	0.981	Art Museum
2.971	Subway	5.800	Bridge
0.462	Gift Shop	2.544	Train Station
Average	**ACC**	**Average**	**ACC**
2.41	29.6%	5.97	28.3%

settings: our original model, DisenPOI without curriculum learning (with fixed α), and DisenPOI with α randomly set under each curriculum step. We conduct the ablation experiments on **Tokyo** dataset and the results are illustrated in Figure 6. We observe that:

- The contrastive loss improves model performance since the disentangled influence of location and history sequence can be fully utilized to make a prediction. Entangled representations would influence the CTR prediction layer to make a biased or sub-optimal prediction. Generally models with larger α perform better, before suffering with an extreme α (greater than 1.0).
- It is necessary to train the model in an easy-to-difficult process to achieve better performance. The models trained with fixed or random α are both likely to converge to a sub-optimal solution instead according to Figure 6b.

5.4 Visualization and Case Study (RQ3)

5.4.1 Visualization of Model Output. To further investigate how the proposed DisenPOI disentangles the hidden representation of POIs, we visualize the representations of user's check-in sequences on the evaluation set. The encodings of visiting sequences are transformed via the t-SNE [25] method. The result shown in Figure 7a and 7b illustrates the effect of disentanglement. To be specific, encodings of sequential and geographical graphs form rather clearer cluster structures compared with the model without $\mathcal{L}_{con}$. The comparison indicates that with the contrastive constraint brought by $\mathcal{L}_{con}$, DisenPOI successfully clarifies different types of embeddings and encodes two graphs into disentangled latent spaces.

To demonstrate the effectiveness of the proposed location-based idea, we illustrate in Figure 7c by comparing the distance between target POIs in the evaluation set and the predictions made by DisenPOI and DIN. It shows that DisenPOI does a better job in recommending POIs that are close to the target than traditional methods.

5.4.2 Case Study. We randomly choose a user (488) who has a visiting sequence with 148 POIs for detailed investigation. We compare the top-100 recommended POIs of DisenPOI and DIN for the selected user (488) in Figure 7d. POIs visited by the user are marked with green circles, while POIs recommended by DisenPOI and DIN are marked with blue and red circles respectively. It can be observed that the POIs recommended by DisenPOI are generally closer to the selected user's visiting trajectory than POIs recommended by DIN, which shows the effectiveness of modeling location-based influence with a geographical POI graph.

To make an intuitive comparison, we compare the top scored POIs recommended by DisenPOI and DIN in Table 3. We first list the distance and categories of top-5 recommended POIs, then the average distance and accuracy on the category of top-50 recommended POIs. It can be observed that DIN pays more attention to other features of the target POI, such as its category (e.g. train station) and functionality (e.g. transportation), while DisenPOI considers both geographical influence and POI's functionality when making recommendations. The results shows that the ignorance of geographical effects will cause models to recommend POIs that share similarities with user interests but at a rather far distance, which is less likely to be visited next. By contrast, DisenPOI would prefer a similar POI within a short distance from user's current location, which helps it to make more reasonable recommendations.

6 CONCLUSION

In this paper, we propose DisenPOI, a dual-graph model that disentangles sequential and geographical influence for Point-of-Interest recommendations. We construct a pair of a sequence-based graph $\mathcal{G}_{s,u}$ and a location-based graph $\mathcal{G}_{g,u}$ to model the disentangled sequential and geographical factor behind a visit. We use customized graph propagation schemes for accurate delineation of different semantics on both graphs and leverage contrastive loss to achieve disentanglement. Experiments on three datasets demonstrate the effectiveness and expressiveness of DisenPOI to disentangle sequential and geographical influence on location-based CTR prediction.

ACKNOWLEDGMENT

REFERENCES

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2013), 1798–1828.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[3] Chen Cheng, Haiqin Yang, Irwin King, and Michael Lyu. 2012. Fused matrix factorization with geographical and social influence in location-based social networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 26. 17–23.

[4] 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*.

[5] Yunfei Chu, Xiaofu Chang, Kunyang Jia, Jingzhen Zhou, and Hongxia Yang. 2021. Dynamic Sequential Graph Learning for Click-Through Rate Prediction. *arXiv preprint arXiv:2109.12541* (2021).

[6] 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.

[7] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2015. Content-aware point of interest recommendation on location-based social networks. In *Twenty-ninth AAAI conference on Artificial Intelligence*.

[8] Huiji Gao, Jiliang Tang, and Huan Liu. 2012. gSCorr: Modeling geo-social correlations for new check-ins on location-based social networks. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. 1582–1586.

[9] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[10] 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.

[11] 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).

[12] Wei Ju, Junwei Yang, Meng Qu, Weiping Song, Jianhao Shen, and Ming Zhang. 2022. KGNN: Harnessing Kernel-based Networks for Semi-supervised Graph Classification. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 421–429.

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

[14] Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-interest recommendations: Learning potential check-ins from friends. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 975–984.

[15] Yujia Li, Richard Zemel, Marc Brockschmidt, and Daniel Tarlow. 2016. Gated Graph Sequence Neural Networks. In *Proceedings of ICLR'16*.

[16] Zeyu Li, Wei Cheng, Haiqi Xiao, Wenchao Yu, Haifeng Chen, and Wei Wang. 2021. You Are What and Where You Are: Graph Enhanced Attention Network for Explainable POI Recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 3945–3954.

[17] 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.

[18] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Thirtieth AAAI conference on artificial intelligence*.

[19] 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 *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 699–712.

[20] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019. Learning disentangled representations for recommendation. *Advances in neural information processing systems* 32 (2019).

[21] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[22] 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.

[23] Ruslan Salakhutdinov and Andriy Mnih. 2008. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *Proceedings of the 25th international conference on Machine learning*. 880–887.

[24] 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.

[25] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[26] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[27] Dongjie Wang, Pengyang Wang, Kunpeng Liu, Yuanchun Zhou, Charles E Hughes, and Yanjie Fu. 2021. Reinforced imitative graph representation learning for mobile user profiling: An adversarial training perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4410–4417.

[28] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation.. In *IJCAI*. 3877–3883.

[29] Pengyang Wang, Kunpeng Liu, Lu Jiang, Xiaolin Li, and Yanjie Fu. 2020. Incremental mobile user profiling: Reinforcement learning with spatial knowledge graph for modeling event streams. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 853–861.

[30] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.

[31] Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference 2021*. 878–887.

[32] 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.

[33] 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*.

[34] Yifan Wang, Suyao Tang, Yuntong Lei, Weiping Song, Sheng Wang, and Ming Zhang. 2020. Disenhan: Disentangled heterogeneous graph attention network for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1605–1614.

[35] Zhaobo Wang, Yanmin Zhu, Qiaomei Zhang, Haobin Liu, Chunyang Wang, and Tong Liu. 2022. Graph-enhanced Spatial-temporal Network for Next POI Recommendation. *ACM Transactions on Knowledge Discovery from Data (TKDD)* (2022).

[36] 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.

[37] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. 2018. Representation learning on graphs with jumping knowledge networks. In *International Conference on Machine Learning*. PMLR, 5453–5462.

[38] Carl Yang, Lanxiao Bai, Chao Zhang, Quan Yuan, and Jiawei Han. 2017. Bridging collaborative filtering and semi-supervised learning: a neural approach for poi recommendation. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1245–1254.

[39] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2014. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45, 1 (2014), 129–142.

[40] 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.

[41] Pengpeng Zhao, Anjing Luo, Yanchi Liu, Fuzhen Zhuang, Jiajie Xu, Zhixu Li, 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* (2020).

[42] Yu Zheng, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2022. Disentangling Long and Short-Term Interests for Recommendation. *arXiv preprint arXiv:2202.13090* (2022).

[43] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5941–5948.

[44] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.