Knowledge-Based Systems 258 (2022) 109951

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

# Building and exploiting spatial–temporal knowledge graph for next POI recommendation

Wei Chen [a,b], Huaiyu Wan [a,b], Shengnan Guo [a,b,*], Haoyu Huang [a], Shaojie Zheng [a], Jiamu Li [a], Shuohao Lin [a,b], Youfang Lin [a,b]

[a] School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[b] Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing 100044, China

ARTICLE INFO

*Article history:*
Received 6 July 2022
Received in revised form 22 September 2022
Accepted 25 September 2022
Available online 29 September 2022

*Keywords:*
Next POI recommendation
Knowledge graph
Knowledge reasoning
Spatial–temporal transfer relation

ABSTRACT

Next Point-of-Interest (POI) recommendation has shown great value for both users and businesses in the field of location-based services. Many spatial–temporal inferring methods have been developed to perform this task, but the data sparsity of POI trajectories greatly hinders the recommendation performance. Knowledge graphs (KGs) have been demonstrated as an effective way to tackle data sparsity in the general recommendation field by leveraging the valuable information of entities and relations. Yet, few studies have explored applying KGs for the next POI recommendation task because of the following challenges: (1) how to represent the dynamic mobility behaviors of users with the static entities and relations in KGs; and (2) how to utilize the different types of entities and relations in KGs to capture long- and short-term preferences of users. In this work, we investigate building a spatial–temporal KG (STKG) from check-in sequences of users to promote the next POI recommendation, without introducing any external attributes of users and POIs. In STKG, we design a novel spatial–temporal transfer relation to intuitively capture users' transition patterns between neighboring POIs. Then, based on the STKG, we propose an innovative model, named STKGRec, for the next POI recommendation, which explicitly models long- and short-term preferences of users in an end-to-end manner. In particular, STKGRec learns both the spatial–temporal correlation of consecutive and nonconsecutive visits in the current check-in sequence to comprehensively capture the short-term preferences of users. Extensive experiments on four real-world datasets demonstrate the superiority of STKGRec against the state-of-the-art baseline methods. *The code of our proposed model is available at https://github.com/WeiChen3690/STKGRec.*



## 1. Introduction

The prevalence of location-based social networks (LBSNs), such as Foursquare and Yelp, has attracted many people to share their experiences with points-of-interest (POIs) like cafes and museums in daily life. This development has led to a large amount of check-in data accumulated in LSBNs, which provides a great opportunity to carry out research on POI recommendation. The next POI recommendation, as a special kind of POI recommendation, focuses on exploring mobile behavior patterns of users and provides interesting POIs for users who want to go next. Therefore, research on the next POI recommendation is greatly beneficial for both users and businesses, and this research has drawn extensive attention in the research community in recent years [1–4].

Previous studies have developed many models for the next POI recommendation task by leveraging personalized information from different aspects. Early studies on the next POI recommendation apply Markov Chains to model sequential transitions, such as FPMC [5]. More recently, deep learning methods have greatly improved the performance of next POI recommendation. Some works such as STRNN [6] and DCRF [7] capture dynamic short-term preferences of users based on a recurrent neural network (RNN) or its various extensions. To further learn temporal patterns of user preference, some researchers jointly model long- and short-term preferences of users and propose more expressive models, such as DeepMove [2], STGN [8] and PLSPL [9]. Recent state-of-the-art models such as LSTPM [3] and STAN [10], capture underlying mobile behavior patterns of users through nonconsecutive check-ins. These methods have achieved great success, but the sparsity of check-in data still hinders further improvement of the recommendation performance.

In recent years, a popular idea for alleviating data sparsity in the general recommendation field has been to introduce knowledge graphs (KGs) that contain rich semantics about entities and

* Corresponding author at: School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China.
*E-mail addresses:* w_chen@bjtu.edu.cn (W. Chen), shnguo@bjtu.edu.cn (S. Guo).

https://doi.org/10.1016/j.knosys.2022.109951
0950-7051/

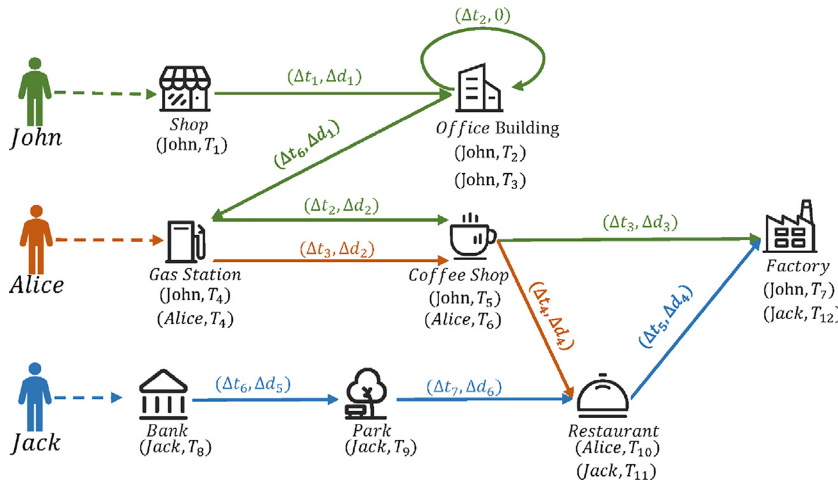



**Fig. 1.** The spatial–temporal changes in movement trajectory of users in STKG.

relations [11–13]. The rich semantic information in knowledge graphs can not only promote the performance of recommendations, but also provide explanations for the recommended results. In the field of POI recommendation, only a few studies have taken KGs as auxiliary information [14–16] to assist in modeling preferences of users. Yet, such KGs built from external POI attributes just learn the static attribute representation of POIs, but cannot effectively represent dynamic movement patterns of users. In addition, we argue that the KGs constructed in these existing methods cannot express the microscopic spatial–temporal movements of users in massive check-in trajectories, which is crucial for mining the dynamic mobility patterns of users among POIs over time.

To address the above challenges, we design a spatial–temporal KG (STKG) that is built from only movement trajectories of users without relying on any external attributes of users and POIs. In STKG, we propose a novel spatial–temporal transfer relation that accurately describes the spatial–temporal changes between successive POIs. A spatial–temporal transfer relation is a combination of the time interval and geographical interval between two neighboring POIs, and the relation with the two neighboring POIs forms a spatial–temporal transfer triplet in STKG. In addition, the check-in records of users can also be regarded as the check-in type triplets in STKG. To this end, the spatial–temporal transfer triplets and check-in triplets, which serve as the basic units of STKG, have strong expressive abilities to represent the continuous movement patterns of users.

An example of an STKG is illustrated in Fig. 1. John is in the Shop at time $T_1$, and reaches the Office Building at time $T_2$ after a geographical distance $\triangle d_1$ and time interval $\triangle t_1$. In this case, the series of behaviors of John can be viewed as three triplets (John, $T_1$, Shop), (Shop, $(\triangle t_1, \triangle d_1)$, Office Building) and (John, $T_2$, Office Building), where John, Shop and Office Building are entities, $T_1$, $(\triangle t_1, \triangle d_1)$ and $T_2$ are relations in STKG. These connected triplets can be combined to build a semantic path that can explicitly capture spatial–temporal movements of users in the trajectory. Such an ability provides great potential to infer accurate user profiles. The construction method of the STKG will be introduced in detail in Section 4. Nevertheless, to apply such STKG for the next POI recommendation task, a non-negligible challenge is how to fully utilize the different types of entities and relations in STKG to capture both the long- and short-term preferences of users. Specifically, long-term preferences that can be learned from historical sequences are relatively stable, while short-term interests depending on the current sequence tend to change dynamically over time. Both long- and short-term preferences play distinct roles in the next POI recommendation, and it is difficult to model these two different preferences of users with the semantic representation provided by STKG.

In this paper, we propose an innovative model based on **S**patial-**T**emporal **K**nowledge **G**raphs for next POI **Rec**-ommendation, namely STKGRec. Our STKGRec is able to jointly train both the STKG and the next POI recommendation model in an end-to-end manner, exploit triplet facts in STKG to model long- and short-term preferences of users, and effectively alleviate the problem of data sparsity. In STKGRec, we employ a historical trajectory encoding module based on different types of entities and relations in STKG and a nonlocal network to learn the long-term preferences of users from the historical sequences. To better learn the short-term preferences, we consider the influence of both consecutive and nonconsecutive visits in the current sequence on modeling movement patterns of users. Specifically, the spatial–temporal changes of continuous movements in the current sequence are captured in a knowledge reasoning manner, where a short-term relation attention network is used to learn the importance of continuous movement patterns. In addition, we present a Time-Geo-dilated GRU to model the spatial–temporal correlations of nonconsecutive check-ins in the current sequence.

To summarize, the main contributions of this paper are as follows:

- To the best of our knowledge, for the first time, STKG without relying on any external attributes of users and POIs is built in the field of the next POI recommendation, and a novel spatial–temporal transfer relation in STKG is designed to explicitly express movement behaviors among POIs of users.
- A STKGRec model is proposed to jointly train both the STKG and the next POI recommendation model in an end-to-end manner, which can capture both long- and short-term preferences of users in STKG and effectively solve the data sparsity issue.
- For long-term preferences, we present a historical trajectory encoding module to capture the importance of different preferences for each POI in historical sequences. For short-term preference, we model the spatial–temporal correlation of both consecutive and nonconsecutive visits in the current sequence.

- Extensive experiments on four real-world datasets show that our STKGRec model achieves significant performance improvements over the state-of-the-art baseline methods.

The rest of this paper is organized as follows. We first review the related work in Section 2. Then, we describe the definitions and problem statement in Section 3. Next, we introduce the building process of spatial–temporal knowledge graphs and our proposed STKGRec model in Section 4, and report the experimental results in Section 5. Finally, Section 6 concludes our paper.

## 2. Related work

### 2.1. Next POI recommendation

In recent years, next POI recommendation has been a hot topic of research for both academia and industry, which predicts the next destination of users based on their historical check-in records. Early works [5,17] for next POI recommendation applied Markov models to learn sequential patterns of users between successive POIs. For example, FPMC-LR [17] constructs a location transition matrix based on a factoring personalized Markov chain to recommend the next destination for target users. Inspired by natural language processing, some methods based on variant RNNs (e.g., LSTM and GRU) [6,7,18] and self-attention networks [19], are presented to capture sequential dependence of users on POIs by combining various types of contextual information, such as temporal, spatial and social information [20,21]. For example, time-LSTM [18] designs time gates in LSTM to model time intervals between locations. GeoSAN [22] utilizes a self-attention network to enhance geographical relevance among consecutive locations. In addition, some researchers explore that preferences of users that change over time, especially the current check-in sequence, which has a particularly large influence. Therefore, both the long- and short-term preferences of users are considered to improve the recommendation performance [2,8,9, 23]. For example, STGN [8] presents a time and geographical dual gating mechanism based on the LSTM framework to model the spatial–temporal context. PLSPL [9] learns the specific long- and short-term preferences via user-based linear combinations.

Recently, some methods [3,10,24] have studied the influence of geographical context among nonconsecutive POIs on preferences of users, such as LSTPM [3], a state-of-the-art method, presenting a Geo-dilated LSTM to capture geographical influence from nonconsecutive POIs. However, they ignore the influence of time on decisions of users. Therefore, in this paper, we comprehensively consider the influence on both the spatial and temporal dimensions among nonconsecutive POIs.

### 2.2. Knowledge graph for recommendation

Recently, the application of KGs in recommendation has been demonstrated to be effective in dealing with the data sparsity problem [25]. Recommendation methods based on KGs can generally be divided into three types: embedding-based methods, path-based methods and unified methods.

Embedding-based methods [25–27] satisfy the downstream recommendation task by pretraining vector representations of entities and relations in KGs, and the knowledge pretraining technology [28–30] can be adopted to improve the knowledge representation. A classical KG-based recommendation method is collaborative knowledge base embedding (CKE) [25], which leverages TransR [28], a translated strategy to learn the semantic features of items. Such KG embedding techniques can provide better feature representation, but lack path exploration ability. In contrast, the path-based approaches [27,31,32] are able to guide the recommendation through the connectivity between entities in KGs. For example, KPRN [27] generates multiple paths by user–item interactions and leverages LSTM to model path semantic features. The unified methods [33,34] are the combination of the above two kinds of methods, which can not only obtain the semantic information of entities and relations effectively, but also learn the connected path information among triplets. For example, KGAT [33] utilizes a graph neural network (GNN) [35] to aggregate the surrounding neighbor information of entities to learn user and item representations of high-order features.

In the field of POI recommendation, only few studies explore KGs to model movement patterns of users. A recent study ARNN [14] utilizes the external attributes of users and POIs to obtain the static embedding of entities and relations, but fails to learn the dynamic behaviors of users. In this work, we attempt to construct an STKG by exploiting movement trajectories of users, and both long- and short-term preferences based on the STKG are modeled to infer the next POI that users may be interested in.

## 3. Preliminaries

In this section, we will first introduce the basic notations and definitions used in this paper and then present the problem statement for the next POI recommendation task. For clarity, Table 1 summarizes primary the notations and their meanings used in the paper.

Let $U = \left\{u_1, u_2, \ldots, u_{|U|}\right\}$ denote the set of users and $P = \{p_1, p_2 \ldots, p_{|P|}\}$ denote the set of POIs, where $|U|$ and $|P|$ are the number of users and POIs, respectively. Each POI can be uniquely identified by longitude and latitude.

**Definition 3.1** (*Check-in Sequence*)**.** A user $u$'s trajectory sequence can be denoted by an ordered list: $\mathbf{L} = \{L_1, L_2, \ldots, L_n\}$ , where each trajectory $L_i = \left\{p_1, p_2, \ldots, p_{|L_i|}\right\}$ denotes a sequence of POIs visited by $u$, and $|L_i|$ is the length of trajectory $L_i$.

**Definition 3.2** (*Spatial–temporal Knowledge Graph (STKG)*)**.** An STKG $\mathcal{G} = \{(e_i, r, e_j)|e_i, e_j \in \mathcal{E}, r \in \mathcal{R}\}$ is essentially a multi-relational directed graph, where $(e_i, r, e_j)$ is a triple that is constructed by entity-relation-entity. $\mathcal{E}$ and $\mathcal{R}$ are the sets of entities and relations, respectively.

**Definition 3.3** (*Problem Statement*)**.** Given historical trajectory sequences $\{L_1, L_2, \ldots, L_{n-1}\}$ of target user $u$ and the current trajectory $L_n = \left\{p_1, p_2, \ldots, p_{|L_n|}\right\}$ of the list of latest POIs that $u$ has visited, the goal of the next POI recommendation is to predict the top-$K$ most likely POIs that user $u$ will visit at the next time.

## 4. Proposed method

In this section, we elaborate on building the STKG and our proposed STKGRec model that exploits the STKG to promote next POI recommendation in an end-to-end way. As illustrated at the right of Fig. 2, the overall framework of STKGRec is composed of three main components: (1) STKG embedding layer, which converts entities and relations in STKG into vector representations; (2) long- and short-term preference modeling layer, which employs STKG to explore both long- and short-term preferences of users from their complete check-in sequences; and (3) prediction layer, which fuses the latent presentations of users' different preferences, and outputs the prediction scores.

### 4.1. Spatial–temporal knowledge graph building

In this paper, we innovatively build a spatial–temporal knowledge graph (STKG) according to the check-in sequences of users

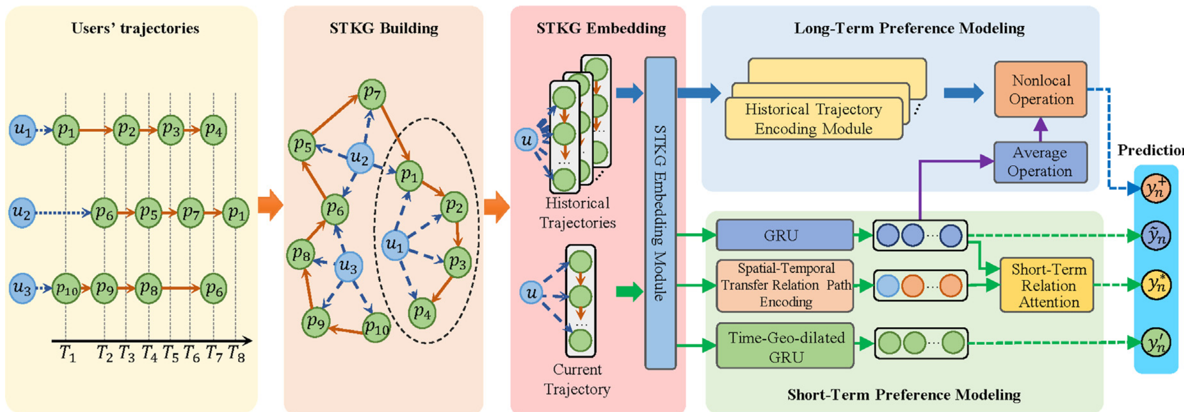



**Fig. 2.** The workflow of STKGRec. The original movement trajectory sequences of each user are on the left side of the figure, where the blue circle represents the user, the green circle represents the POI, the solid red arrow represents the spatial–temporal relationship, and the dotted blue arrow represents the check-in relationship. The STKG is in the middle of the figure, and the framework of STKGRec is on the right.

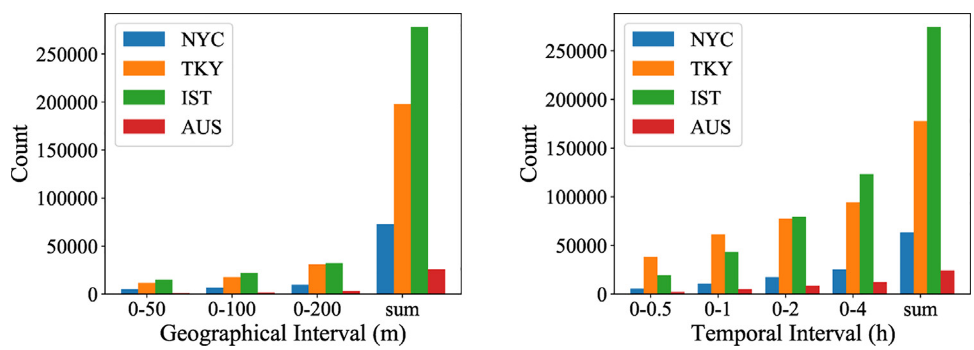



**Fig. 3.** The statistics of the temporal interval and geographical interval according to the movement of the user among POIs on the NYC, TKY, IST and AUS datasets.

**Table 1**
Summary of primary notations.

| Notation | Description |
|---|---|
| $u, p$ | a user and a POI |
| $U, P$ | the set of users and POIs |
| $L, \mathbf{L}$ | a sequence and a set of sequences |
| $e, r$ | an entity and a relation |
| $\mathbf{e}, \mathbf{r}$ | the vector representation of entity and relation |
| $\mathcal{E}, \mathcal{R}$ | the set of entities and relations |
| $\mathcal{G}$ | an STKG |
| $(e^u, r^t, e^p)$ | a check-in triplet |
| $(e_i^p, r^{st}, e_j^p)$ | a spatial–temporal transfer triplet |
| $h, m$ | hour and meter |
| $\triangle d, \triangle t$ | geographical interval and time interval |

without bringing in any external attributes of users and POIs. To naturally learn the movement patterns of users with entities and relations in STKG, users $u$ and POIs $p$ are represented as user entities $e^u \in \mathcal{E}$ and POI entities $e^p \in \mathcal{E}$.

As shown at the left of Fig. 2, we extract two types of triple facts from the trajectory sequences of all users, i.e., check-in triplet $(e^u, r^t, e^p)$ and spatial–temporal transfer triplet $(e_i^p, r^{st}, e_j^p)$, where $r^t \in \mathcal{R}$ and $r^{st} \in \mathcal{R}$ are relations. These triplets are used to describe the spatial–temporal semantic relevance of entities and relations in STKG. To be specific, triplet $(e^u, r^t, e^p)$ reflects the check-in behavior of the user $u$ at a specific time. Referring to [3], we divide the check-in relation $r^t$ into 48 time slots, in which 24 slots are different hours on weekdays and the other 24 slots are different hours on weekends, indicating the check-in behaviors of users within a week. Triplet $(e_i^p, r^{st}, e_j^p)$ represents the historical movement behavior of all users between two POIs $p_i$ and $p_j$, where the spatial–temporal transfer relation $r^{st} = (\triangle t, \triangle d)$ is designed as a combination of temporal interval $\triangle t$ and geographical interval $\triangle d$. To more accurately depict the spatial–temporal movement of users between neighboring POIs, we analyze both spatial and temporal factors on four public LBSN datasets, i.e., NYC, TKY, IST and AUS. Detailed description of the LBSN datasets will be introduced in Section 5.1. As shown in Fig. 3, the temporal interval is less than half an hour and the geographical interval is less than 50 meters in a small proportion of the four datasets. To this end, we set the basic unit of $r^{st}$ as a combination of half hour and 50 m. Then we increase the two dimensions in multiples to obtain more spatial–temporal transfer relations on different granularities, such as (1 h, 100 m), (2 h, 200 m), etc. Note that, the spatial–temporal transfer relation is a self-loop relation on POIs if $\triangle d$ is 0, which means a special behavior of users that have checked to the same location twice in a row at different times.

In our work, the trajectory of each user can be viewed as a semantic path consisting of check-in triplets and spatial–temporal transfer triplets, which can explicitly capture spatial–temporal movement behaviors of users. Our constructed STKG can integrate knowledge into the movement patterns of users and provides a structured semantic representation for dynamic trajectory sequences, which is quite beneficial for the downstream next POI recommendation task.

### 4.2. Spatial–temporal knowledge graph embedding

KG embedding is an effective way to convert nodes and relations into vector representations, while retaining the ability of relation reasoning in KGs [36]. Considering that there are many different relations between two identical entities in STKG, we

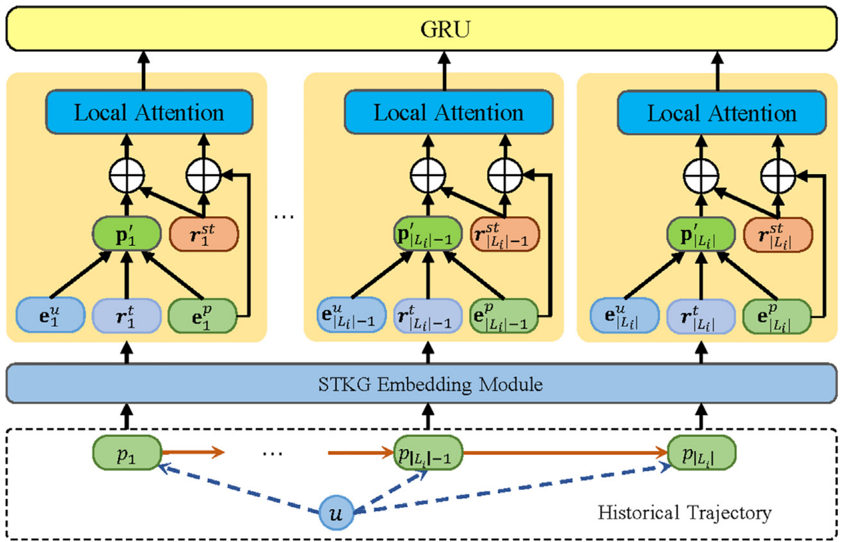



**Fig. 4.** The historical trajectory encoding module.

adopt a widely used method TransR [28] that can effectively distinguish the semantics of different relations between entities. Therefore, we adopt the following scoring function to optimize check-in triplets:

$$f_{\mathrm{KG}}(e^u, r^t, e^p) = \left\| \mathbf{e}^u + \mathbf{r}^t - \mathbf{e}^p \right\|_2^2, \tag{1}$$

where $\mathbf{e}^u$ and $\mathbf{e}^p \in \mathbb{R}^d$ are the projected representations of $e^u$ and $e^p$ in relation $r^t$, respectively; $\mathbf{r}^t \in \mathbb{R}^d$ is the embedding of $r_t$, and the triplet $(e^u, r^t, e^p)$ is valid in STKG. A valid triplet is likely to obtain a lower score of $f_{KG}$, and vice versa. When training triplets, the relative order between valid triplets and broken triplets can be distinguished by a pairwise ranking loss:

$$\mathcal{L}_{\mathrm{KG}}(e^u, r^t, e^p) = -\sum_{\substack{(e^u, r^t, e^{p'}) \notin \mathcal{G} \\ \cup (e^u, r^t, e^p) \in \mathcal{G}}} \ln \sigma(f_{\mathrm{KG}}(e^u, r^t, e^{p'}) - f_{\mathrm{KG}}(e^u, r^t, e^p)), \tag{2}$$

where $\left(e^u, r^t, e^{p'}\right)$ is a broken triplet generated through replacing by randomly one tail entity of a valid triplet. Here, the replacing tail entity should be a POI entity not a user entity, which can enhance the efficiency of negative sampling. $\sigma(\cdot)$ is the sigmoid function. The optimization of the spatial–temporal transfer triplet in STKG is similar to the check-in triplet. This layer views triplets as the base unit to model entities and relations, injecting direct connections into the representation and working as a regularizer to improve the representation ability of the model.

### *4.3. Long-term preference modeling with knowledge*

In general, long-term preference reflects the general taste of check-in behaviors of users. Such preferences are affected not only by historical check-in behaviors of users but also by their situation. Therefore, accurately capturing long-term preferences of users to facilitate the performance of the next POI recommendation is extremely beneficial. To thoroughly learn long-term preferences of users, we develop a historical trajectory encoding module that captures the importance of different preferences for each POI in historical sequences by various entities and relations in STKG, as shown in Fig. 4. We then apply a nonlocal network to learn the correlation between historical and current sequences. The details are shown as follows.

Given all historical trajectory sequences $L_i \in \{L_1, L_2, \ldots, L_{n-1}\}$ of a user $u$, we first apply the check-in triplet to learn the user check-in representation in each sequence:

$$\mathbf{p}' = \mathbf{e}^u + \mathbf{r}^t + \mathbf{e}^p, \tag{3}$$

where $\mathbf{e}^u$, $\mathbf{r}^t$ and $\mathbf{e}^p$ are the vector representations of $u$, $r^t$ and $p$ in the check-in triplet after STKG embedding, respectively. To accurately represent the local spatial–temporal transfer behavior by knowledge, we adopt the translation scheme [37,38] to obtain the local spatial–temporal transfer representation of user $u$ after the check-in POI: $\mathbf{z}^{tr} = \mathbf{p}' + \mathbf{r}^{st}$ and the general spatial–temporal transfer representation: $\mathbf{z}^r = \mathbf{e}^p + \mathbf{r}^{st}$, where $\mathbf{r}^{st}$ is the vector representation of $r^{st}$ after STKG embedding. Next, we apply a local attention mechanism to model the importance of local transfer behavior for users on POIs. Here, the importance of transfer behavior is calculated as the normalized similarity between $\mathbf{z}^r$ and $\mathbf{z}^{tr}$:

$$\mathbf{g} = \sum_{j=1}^{|L_i|} \alpha'_j \mathbf{z}_j{}^{tr}, \tag{4}$$

$$\alpha'_j = \frac{\exp(\mathbf{z}_j^r (\mathbf{z}_j{}^{tr})^\top)}{\sum_{k=1}^{|L_i|} \exp(\mathbf{z}_k{}^r (\mathbf{z}_k{}^{tr})^\top)}, \tag{5}$$

where $\alpha'$ is the importance of each local transfer behavior, and the transfer representation of POIs in each historical sequence is encoded as $\{\mathbf{g_1}, \mathbf{g_2}, \ldots, \mathbf{g}_{|L_i|}\}$. To further study sequential dependencies for the target user, we adopt a GRU method that is better suited than LSTM for less frequent and spare datasets [39]. The historical sequential dependency representation is calculated as follows:

$$\mathbf{h_t} = \mathrm{GRU}(\mathbf{g_t}, \mathbf{h_{t-1}}) \quad 1 \le \mathbf{t} \le |L_i|, \tag{6}$$

where $\mathbf{h_t}$ is the hidden state. In this way, the $n-1$ historical sequences can be represented as $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{n-1}\}$. Similarly, we directly utilize the GRU to obtain the current sequential dependency representation:

$$\tilde{\mathbf{h}}_t = \mathrm{GRU}\left(\mathbf{p}'_\mathbf{t}, \tilde{\mathbf{h}}_{t-1}\right) \quad 1 \le \mathbf{t} \le |L_n|. \tag{7}$$

Considering the current situation in which the target user is located is correlated with his or her long-term preference. Here, we use a nonlocal network that has been proven effective in learning the influence of both each historical sequence and the current sequence based on pairwise affinities [3]. Formally, we adopt the following operation to calculate the latent representation $\mathbf{y}_n^+$ of long-term preferences of the user $u$:

$$\mathbf{s}_n = \frac{1}{|L_n|}\sum_{\mathbf{t}=1}^{|L_n|}\tilde{\mathbf{h}}_t, \tag{8}$$

$$\mathbf{y}_n^+ = \frac{\sum_{i=1}^{n-1} f\left(\mathbf{s}_n, \mathbf{s}_i\right)\mathbf{W}_i\mathbf{s}_i}{\sum_{i=1}^{n-1} f\left(\mathbf{s}_n, \mathbf{s}_i\right)}, \tag{9}$$

$$f\left(\mathbf{s}_n, \mathbf{s}_i\right) = \exp\left(\mathbf{s}_n^\top \mathbf{s}_i\right), \tag{10}$$

where $\mathbf{s}_n$ is the vector of the current trajectory after an average pooling that can preserve the information of all POIs in $L_n$, and $\mathbf{s}_i$ indicates the vector of a historical trajectory. The pairwise function $f(*)$ calculates an affinity score between the current trajectory $\mathbf{s}_n$ and a historical trajectory $\mathbf{s}_i$, and $\mathbf{W}_i$ is a learnable projection weight matrix.

### 4.4. Short-term preference modeling with knowledge

In the next POI recommendation, assuming that the next destination of a user has a higher correlation to her or his recently visited POIs, existing methods focus more on learning the short-term preferences of the user from the current trajectory. They either explore time or distance transitions among consecutive POIs in the current sequence [8,40], or learn geographical influence from the nonconsecutive visits in the current sequence [3]. However, these methods do not comprehensively consider both consecutive and nonconsecutive visit in the current sequence to model short-term preferences. To thoroughly capture short-term preferences of users, we employ knowledge graph path reasoning method and Time-Geo-dilated GRU respectively to learn from the spatial–temporal correlation of their movement from consecutive and non-consecutive check-ins in the current sequence.

#### 4.4.1. Spatial–temporal correlation learning of consecutive check-ins

Unlike the RNN-based method considering the time and/or geographical intervals among POIs by data-driven ways, we attempt to capture spatial–temporal changes depending on continuous movements of users in the current sequence in a knowledge reasoning manner. Inspired by PtransE [41] which captures the relation dependency of triplets via connected paths in KGs, we encode consecutive spatial–temporal transfer relation path in the current trajectory by the operation of summation for user $u$:

$$\mathbf{h}_j^{st} = \begin{cases} \mathbf{e}^u & j = 1 \\ \mathbf{e}^u + \sum_{k=1}^{j-1}\mathbf{e}_k^{st} & 1 < j \le |L_n|, \end{cases} \tag{11}$$

where $\mathbf{h}^{st}$ is the representation of continuous spatial–temporal transfer behaviors of the target user in the current sequence. To the best of our knowledge, very few approaches apply this technique to infer the continuous mobile behavior of users in the next POI recommendation task. Here, the operation of summation can be substituted by other techniques, such as concatenation and RNN. After that, the short-term relation attention is applied to obtain the importance of the continuous movement behaviors user in the current sequence. Formally, the similarity score is calculated between $\mathbf{h}^{st}$ and $\tilde{\mathbf{h}}$, and then normalized by a softmax function:

$$\mathbf{y}_n^* = \sum_{j=1}^{|L_n|}\beta_j'\tilde{\mathbf{h}}_j, \tag{12}$$

$$\beta_j' = \frac{\exp(\tilde{\mathbf{h}}_j(\mathbf{h}_j^{st})^\top)}{\sum_{k=1}^{|L_n|}\exp(\tilde{\mathbf{h}}_k(\mathbf{h}_k^{st})^\top)}, \tag{13}$$

where $\beta'$ is the attention weight that measures the importance of continuous spatial–temporal transfer relation behaviors in the current sequence, and $\mathbf{y}_n^*$ is the consecutive spatial–temporal movement latent representation of user $u$ in the current consecutive check-in sequence. In addition, the short-term preferences of user $u$ are also affected by the non-consecutive check-ins, which we will elaborate next.

#### 4.4.2. Spatial–temporal correlation learning of nonconsecutive check-ins

Recent studies [3,24] developed Geo-dilated LSTM to capture the geographical influence among nonconsecutive check-ins, but they did not consider the temporal information, which is a crucial factor on users' decisions. In real life, a user often estimates the time cost before she or he goes to the next destination. Based on this observation, we present a Time-Geo-dilated GRU scheme that applies spatial–temporal transfer relations in STKG to explore the influence of spatial–temporal dependencies among nonconsecutive POIs in the current trajectory. The Time-Geo-dilated GRU selects POIs from the check-in sequence as inputs with different skip lengths, and the inputs are determined by spatial–temporal intervals among POIs in the sequence. For example, given a check-in sequence $L_n = \{p_1, p_2, p_3, p_4, p_5\}$ and the skip length fixed as 2, there are two POIs in front of $p_3$ including $p_1$ and $p_2$. We use the following operation to measure the total cost of time and geographical distance:

$$\pi = \lambda * \triangle d + (1-\lambda)\triangle t, \tag{14}$$

where $\triangle d$ and $\triangle t$ are the normalized geographical interval and time interval geographical interval from the spatial–temporal transfer relation, respectively. $\lambda$ is a hyperparameter that controls the different importance of geography and time. When $\lambda = 1$, the Time-Geo-dilated GRU will be transformed into one that only considers geographical information and is similar to the Geo-dilated LSTM. If $\pi(p_1, p_3) < \pi(p_1, p_2)$, then there will be a spatial–temporal dilated sequence $\{p_1, p_3\}$, a similar operation is applied to check-in POIs $p_3$, $p_4$ and $p_5$, and we can build a new input set $L_n^{Time-Geo} = \{\{p_1, p_3\}, \{p_3, p_5\}\}$. Then, the spatial–temporal preferences of the user among nonconsecutive POIs can be obtained through dilated GRU learning with the input $L_n^{Time-Geo}$:

$$\mathbf{h}'_{\mathbf{t-1}} = \mathrm{GRU}(\mathbf{p}'_{\mathbf{t-1}}, \mathbf{h}'_{\mathbf{t-\gamma}}) \quad 1 < \mathbf{t} \le |L_n|, \tag{15}$$

where $\gamma$ is the skip length and can be automatically determined by spatial–temporal factor $g$, $\mathbf{h}'_{\mathbf{t-\gamma}}$ is computed from the last sequence $\{\mathbf{p}'_\gamma, \mathbf{p}'_{\mathbf{t-1}}\} \in L_n^{Time-Geo}$.

### 4.5. Prediction and optimization

After the user preference modeling layer, we obtain the long-term preference latent representation $\mathbf{y}_n^+$, the current sequence latent representation $\tilde{\mathbf{y}}_n = \tilde{\mathbf{h}}_{|L_n|}$, the current consecutive spatial–temporal movement latent representation $\mathbf{y}_n^*$ and the current nonconsecutive spatial–temporal dependency representation $\mathbf{y}'_n = \mathbf{h}'_{|L_n|}$. Then we obtain the probability distribution $\hat{\mathbf{y}}$ for the prediction POIs in the following way:

$$\hat{\mathbf{y}} = \mathrm{softmax}(\mathbf{W}_p[\tilde{\mathbf{y}}_n \parallel \mathbf{y}_n^* \parallel \mathbf{y}_n^+ \parallel \mathbf{y}'_n]), \tag{16}$$

where $\parallel$ is the concatenation operation, and $\mathbf{W}_p$ is a trainable matrix for all POIs. Therefore, the POI with the highest probability is the one that the user wants to visit at time $t$. In this work, the next POI recommendation task can be regarded as a multi-class classification problem, where each class corresponds to each POI. To optimize the parameters of the recommendation model, the log likelihood function is adopted as follows:

$$\mathcal{L}_{\mathrm{POI}} = -\sum_{k=1}^{N} \log(\hat{\mathbf{y}}_k), \tag{17}$$

where $N$ denotes the total number of training samples, and $\hat{\mathbf{y}}_k$ represents the prediction probability of the ground truth POI calculated on the $k$th training sample. Finally, the objective function is formulated by integrating Eqs. (2) and (17):

$$\mathcal{L}_{\mathrm{loss}} = \mathcal{L}_{\mathrm{KG}} + \mathcal{L}_{\mathrm{POI}}. \tag{18}$$

In the model training process, we alternately optimize $\mathcal{L}_{\mathrm{KG}}$ and $\mathcal{L}_{\mathrm{POI}}$ and batch Adam [42] is adopted for optimizing the STKG embedding loss and recommendation loss. To be specific, we update the embeddings of all entities and relations for randomly sampling a batch of $(e^u, r^t, e^p, e^{p'})$; then, we sample a batch of $\{L_1, L_2, \ldots, L_{n-1}, L_n\}$ of users and input them into the recommendation model, updating the parameters of the model according to the gradients of the recommendation loss. The detailed training procedure of our proposed model is summarized in Algorithm 1.

**Algorithm 1:** Training procedure of STKGRec

```
Input: G:Temporal–spatial knowledge graph
       L: Sessions of all users
       Φ(G) : Parameters of knowledge graph embedding
       Φ(L) : Parameters of recommendation
while not done do
    sample triplets T_triplet from G;
    sample sessions T_session from L;
    for 1 in S_Batch_Triples do
        Calculate triplets scoring function by Eq.1;
        Calculate KG embedding loss by Eq.2 and update
          Φ(G) in T_triplet;
    end
    for 1 in S_Batch_Sessions do
        Obtain the long-term preferences representation of
          users by Eq.3-Eq.10;
        Obtain the current consecutive spatial–temporal
          movement representation by Eq. 11–Eq. 13;
        Obtain the current non-consecutive
          spatial–temporal movement representation by Eq.
          14–Eq. 15;
        Calculate recommendation loss by Eq. 17 and
          update Φ(L) in T_session;
    end
end
```

## 5. Experiments

### 5.1. Datasets and preprocessing

We evaluate our STKGRec model on four real-world city datasets, i.e., NYC (New York City), TKY (Tokyo), IST (Istanbul) and Austin (AUS). NYC, TKY and IST are collected from the public Foursquare[1] check-in data [43,44], where NYC and TKY are both collected from April 2012 to February 2013, IST is extracted from April 2012 to January 2014. AUS is generated from public dataset Gowalla[2] that contains check-in data February from 2009 to October 2010. In our experiments, we eliminate POIs that are visited less than 10 times in NYC, TKY and IST three datasets, while filtering POIs that are visited less than 5 times in AUS.

**Table 2**
Statistics of datasets.

| Dataset | NYC | TKY | IST | AUS |
|---|---|---|---|---|
| # of users | 1020 | 2232 | 5377 | 1352 |
| # of POIs | 14,085 | 21,139 | 26,329 | 8220 |
| # of check-ins | 89,897 | 255,827 | 329,792 | 113,866 |
| # of sessions | 18,459 | 50,848 | 87,205 | 21,390 |
| # of entities | 15,105 | 25,735 | 31,706 | 9572 |
| # of relations | 18,762 | 29,786 | 50,084 | 17,655 |
| # of triplets | 153,098 | 396,565 | 634,241 | 217,520 |

For all datasets, the check-in records of each user are divided into multiple sessions according to a 24-hour time window, each session contains at least 3 check-ins, and the users who have less than 5 sessions are filtered out. Then, we need to construct the STKG according to the processed session sequences. For four datasets, we select the first 80% sessions of each user into the training set and take the most recent 10% sessions as the test set. The remaining 10% are used as the validation set to tune the hyperparameters. The statistics of the datasets are summarized in Table 2.

### 5.2. Experimental settings

#### 5.2.1. Baselines

To demonstrate the effectiveness of our proposed model, we compare STKGRec with the following baselines:

- **GRU** [45] is a variant of RNN and is able to effectively capture the long-term dependence of sequences;
- **STRNN** [6] integrates the spatial–temporal context features into the RNN framework to model the movement behaviors of users among POIs;
- **DeepMove** [2] utilizes the attention mechanism to obtain the regularity of users' movements and is a very competitive method;
- **STGN** [8] extends LSTM to incorporate the time and geographical intervals by two gate units;
- **LSTPM** [3] is a state-of-the-art POI recommendation method that learns long- and short-term preferences of users, where the Geo-dilated LSTM is presented to achieve the non-consecutive inputs in the current sequence;
- **PLSPL** [9] captures the specific long- and short-term preferences via a user-based linear combination unit that is able to obtain the personalized weights; and
- **STAN** [10] models the temporal and spatial correlations between non-adjacent locations by a self-attention network.

#### 5.2.2. Parameter settings and evaluation metrics

We implement our STKGRec model in Torch. The embedding size is set to 100 for all deep learning-based models. Following the work in [3], we optimize all the parameters in our model with the gradient descent optimization algorithm Adam and the batch size is set to 128. A grid search is performed on a set of learning rates of {0.00005, 0.0001, 0.0005, 0.001} and the hyperparameter $\lambda$ of {0.2, 0.4, 0.6, 0.8}. The temporal interval and geographical interval of the spatial–temporal transfer relation $r_{trans}$ are (1 **h**, 100 **m**) for all datasets. For the different parameters of other baselines, we follow the best settings in their papers. Note that the non-deep learning methods are not compared, the reason is that the baselines [2,8] applied in our experiments have proven that they are much superior to non-deep learning methods.

To evaluate the performance of STKGRec, we adopt two widely used evaluation metrics [3,40,46]: **Recall** (Rec)@$k$ and **Normalized Discounted Cumulative Gain** (NDCG)@$k$. Rec@$k$ focuses on

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

**Table 3**
Overall performance comparison on the NYC and TKY datasets.

| Model | NYC | | | | | | TKY | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Rec@1 | Rec@5 | Rec@10 | NDCG@1 | NDCG@5 | NDCG@10 | Rec@1 | Rec@5 | Rec@10 | NDCG@1 | NDCG@5 | NDCG@10 |
| GRU | 0.0933 | 0.1959 | 0.2378 | 0.0933 | 0.1469 | 0.1604 | 0.122 | 0.2607 | 0.3177 | 0.122 | 0.1951 | 0.2136 |
| STRNN | 0.0921 | 0.1935 | 0.235 | 0.0921 | 0.1459 | 0.1593 | 0.1229 | 0.2620 | 0.316 | 0.1229 | 0.1963 | 0.2137 |
| DeepMove | 0.1186 | 0.2511 | 0.3062 | 0.1186 | 0.189 | 0.206 | 0.147 | 0.3066 | 0.3691 | 0.147 | 0.2314 | 0.2517 |
| STGN | 0.0868 | 0.1495 | 0.1758 | 0.0868 | 0.1204 | 0.1289 | 0.1222 | 0.2186 | 0.2612 | 0.1229 | 0.1732 | 0.1869 |
| PLSPL | 0.1171 | 0.2916 | 0.3558 | 0.1171 | 0.2029 | 0.23 | 0.1278 | 0.3105 | 0.3808 | 0.1278 | 0.223 | 0.2456 |
| LSTPM | <u>0.155</u> | <u>0.3683</u> | 0.4545 | <u>0.1555</u> | <u>0.2665</u> | <u>0.2946</u> | <u>0.1612</u> | <u>0.3608</u> | 0.4391 | <u>0.1612</u> | <u>0.2660</u> | <u>0.2914</u> |
| STAN | 0.1123 | 0.3318 | <u>0.4726</u> | 0.1123 | 0.2247 | 0.2695 | 0.1123 | 0.3418 | <u>0.4512</u> | 0.1123 | 0.2222 | 0.2576 |
| STKGRec | **0.1764** | **0.3927** | **0.4773** | **0.1764** | **0.2901** | **0.3176** | **0.224** | **0.4142** | **0.4824** | **0.224** | **0.325** | **0.3471** |

**Table 4**
Overall performance comparison on the IST and AUS datasets.

| Model | IST | | | | | | AUS | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Rec@1 | Rec@5 | Rec@10 | NDCG@1 | NDCG@5 | NDCG@10 | Rec@1 | Rec@5 | Rec@10 | NDCG@1 | NDCG@5 | NDCG@10 |
| GRU | 0.0584 | 0.1297 | 0.1668 | 0.0584 | 0.0943 | 0.1068 | 0.0473 | 0.0994 | 0.1289 | 0.0473 | 0.0745 | 0.084 |
| STRNN | 0.0585 | 0.1293 | 0.168 | 0.0585 | 0.0956 | 0.1081 | 0.0453 | 0.0942 | 0.1236 | 0.0453 | 0.071 | 0.08 |
| DeepMove | 0.0761 | 0.164 | 0.2071 | 0.0761 | 0.1221 | 0.136 | 0.0653 | 0.1248 | 0.1549 | 0.0653 | 0.0967 | 0.1064 |
| STGN | 0.0614 | 0.123 | 0.1574 | 0.0614 | 0.0936 | 0.1046 | 0.0497 | 0.0898 | 0.1125 | 0.0497 | 0.0705 | 0.0778 |
| PLSPL | 0.0896 | 0.1969 | 0.2433 | 0.0896 | 0.1459 | 0.1609 | 0.0629 | 0.1611 | 0.1879 | 0.0629 | 0.1133 | 0.1219 |
| LSTPM | <u>0.0936</u> | 0.2193 | 0.2781 | <u>0.0936</u> | 0.1595 | 0.1786 | 0.0778 | 0.1629 | 0.2039 | 0.0788 | 0.123 | 0.1362 |
| STAN | 0.082 | <u>0.2517</u> | <u>0.3221</u> | 0.082 | <u>0.1662</u> | <u>0.1868</u> | <u>0.08</u> | <u>0.18</u> | <u>0.223</u> | <u>0.08</u> | <u>0.1314</u> | <u>0.1456</u> |
| STKGRec | **0.2196** | **0.3490** | **0.4015** | **0.2196** | **0.2885** | **0.3055** | **0.1046** | **0.2059** | **0.2551** | **0.1046** | **0.1575** | **0.1734** |

the presence of the ground truth among the top-$k$ recommended list, while NDCG@$k$ evaluates the quality of the ranking. The higher the Rec@$k$ and NDCG@$k$ are, the better the performance is. Suppose a training set with $\mathcal{M}$ samples, Rec@$k$ is defined as:

$$\text{Rec@}k = \frac{1}{\mathcal{M}} \sum_{i \in \mathcal{M}} \frac{\left| Q_{\text{rec}}^{i} \cap Q_{\text{visited}}^{i} \right|}{\left| Q_{\text{visited}}^{i} \right|}. \tag{19}$$

where $Q_{\text{rec}}^{i}$ represents the top-$k$ recommended rank list, and $Q_{\text{visited}}^{i}$ represents the ground truth POIs that users have visited. Note that, the number of $Q_{\text{visited}}^{i}$ is 1 in our next POI recommendation task. The NDCG@$k$ is defined as:

$$\text{NDCG@}k = \frac{1}{\mathcal{M}} \sum_{i \in \mathcal{M}} \frac{\text{DCG@}k}{\text{IDCG@}k}, \tag{20}$$

where

$$\text{DCG@}k = \sum_{j=1}^{k} \frac{2^{rel_j} - 1}{\log_2(j+1)}, \tag{21}$$

and $rel_i$ indicates the graded correlation of the prediction result ranked at position $i$. The value of $rel_i$ is obtained by the binary relevance in our work, i.e., $rel_i = 1$ if $Q_{\text{visited}}^{i}$ is in rank list $Q_{\text{rec}}^{i}$, and 0, otherwise. IDCG@$k$ is the DCG@$k$ value at the ideal rank in the recommendation list. In the paper, we set $K = \{1, 5, 10\}$ and calculate the average performance of all users in the test set.

### 5.3. Overall performance

The overall performance comparisons of different models on the NYC, TKY, IST and AUS datasets are shown in Tables 3 and 4. In each column, we employ the bold font to highlight the best result and underline to show the second place result. From the experiment results, we have the following observations:

- The STKGRec model consistently yields the best performances in terms of every metric on all datasets. For example, compared with the strongest baselines, our model improves the performance w.r.t. the Rec@5 by 6.63%, NDCG@5 by 8.86% on the NYC dataset and the Rec@5 by 14.8%, NDCG@5 by 22.18% on the TKY dataset. In particular, the overall improvement of STKGRec against the best competitors reaches an average of 78.28% on different metrics on the IST dataset. The quantitative evaluation results demonstrate the advantages of our model over the baseline methods. From the perspective of ranking, STKGRec has a more significant performance improvement in top-1 than top-5 and top-10 compared to the strongest baseline, which indicates the ability of STKGRec to generate high-quality ranked lists.
- All over the models, the results on the AUS dataset are worst. The reason for this phenomenon is that most users in the AUS dataset visit the same location less frequently, which makes it difficult for the model to learn the preferences of users. Moreover, the results on the IST dataset are worse than those on the NYC and TKY datasets. We attribute this result to the fact that IST has larger time span and is more sparse. Compared with baselines, the performance degradation of STKGRec drops relatively little, indicating the powerful ability of STKG in alleviating the data sparsity problem.
- All over the baselines, LSTPM has better performance on the NYC and TKY datasets, while STAN performs better on the IST and AUS datasets. Both the LSTPM and STAN methods strongly illustrate the importance of modeling nonconsecutive check-in POIs. Yet, because of the overlooked importance of continuous changes in context (e.g., time and distance), the performance of LSTPM and STAN is hindered, and this is a key factor why STKGRec is superior over these two methods. PLSPL and DeepMove perform better than STGN, STRNN and GRU, the reason for this is that DeepMove takes into account the periodic behavior of users and PLSPL is able to model the personalized weights of users at different locations.

### 5.4. Effect of different components

In this section, we will report the contribution of each module in STKGRec. We will conduct the experiment on three variants of STKGRec that are listed as follows:

- **STKGRec-L:** This variant contains a historical trajectory encoding module and a nonlocal network, which is used to verify the effectiveness of modeling the long-term preferences of users.

**Table 5**
Performance of different STKGRec variants.

| Model | NYC | | TKY | | IST | | AUS | |
|---|---|---|---|---|---|---|---|---|
| | Rec@5 | NDCG@5 | Rec@5 | NDCG@5 | Rec@5 | NDCG@5 | Rec@5 | NDCG@5 |
| Best Baseline | 0.3683 | 0.2665 | 0.3608 | 0.2660 | 0.2193 | 0.1595 | 0.18 | 0.1314 |
| STKGRec-L | 0.3733 | 0.2743 | 0.3704 | 0.2729 | 0.2154 | 0.1673 | 0.1753 | 0.132 |
| STKGRec-P | 0.3873 | 0.2793 | 0.3973 | 0.3133 | 0.3249 | 0.2678 | 0.1856 | 0.1456 |
| STKGRec-TG | 0.3761 | 0.2753 | 0.3577 | 0.2664 | 0.1945 | 0.1439 | 0.1649 | 0.1249 |
| STKGRec | **0.3927** | **0.2901** | **0.4142** | **0.3250** | **0.3490** | **0.2885** | **0.2059** | **0.1575** |

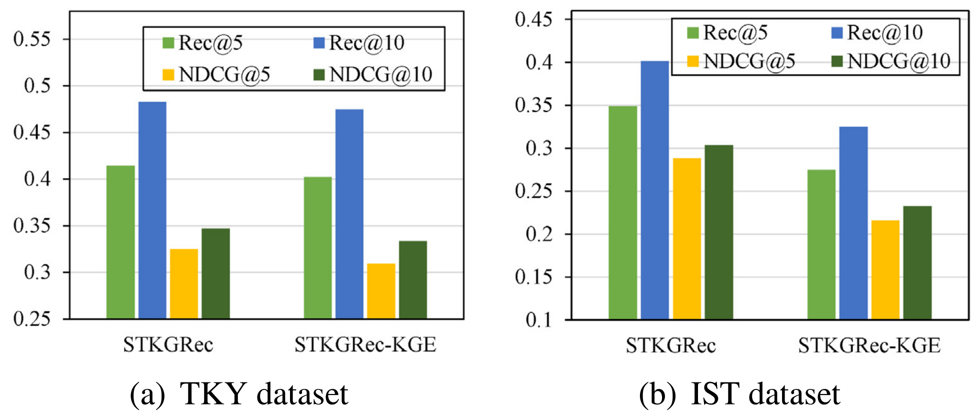



**Fig. 5.** The performance of both STKGRec and STKGRec-KGE.

- **STKGRec-P:** This variant keeps PtransE and the short spatial–temporal transfer relation attention module, which is used to learn the spatial–temporal changes of continuous movements of users in the current sequence in a knowledge reasoning manner.
- **STKGRec-TG:** This variant only keeps the Time-Geo-Dilated GRU module that is used to explore the effectiveness of spatial–temporal correlation among nonconsecutive check-ins in the current sequence.

To exclude the influence of different modules, we gradually add a single module based on the GRU model to conduct our experiments. These STKGRec variants are unified training with STKG embedding since each module involves the entities and relations in the STKG. The results w.r.t Rec@5 and NDCG@5 of different STKGRec variants are presented in Table 5, and we can observe the following:

- STKGRec-P achieves better performance than both STKGRec-L and STKGRec-TG on four datasets, which demonstrates that the knowledge graph reasoning method can effectively capture the movement trend of users in each POI in the current sequence. We argue that such a method directly modeling the spatial–temporal transfer relation in the current sequence is beneficial to infer the movement pattern of the user, and it is crucial for improving the recommendation accuracy.
- The performances of STKGRec-L and STKGRec-TG are relatively close. The results of STKGRec-L and STKGRec-TG are slightly better than the results of strongest baselines on both the NYC and TKY datasets, and slightly worse than the results of strongest baselines on the IST and AUS datasets. The results demonstrate that STKGRec-L and STKGRec-TG based on STKG can better model long-term periodicity and spatial–temporal correlations of users among nonconsecutive check-ins, respectively.
- The complete STKGRec model can achieve the best performance on three datasets, indicating that considering both long-term and short-term preferences in STKG has a positive impact on the next destination of users.

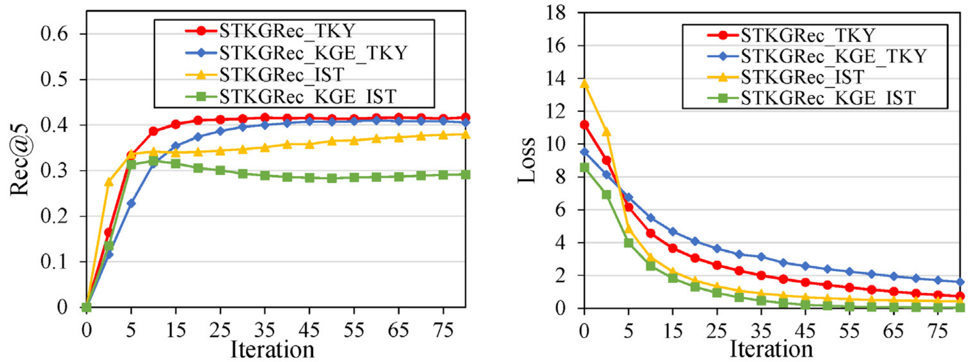
### 5.5. Effect of STKG embedding

As the STKG embedding provides a rich semantic representation, it is necessary to investigate the impact of STKG embedding. We disable the TransR embedding and train a variant model STKGRec-KGE with random initialization on the TKY and IST datasets. The results are shown in Figs. 5 and 6. Removing the STKG embedding obviously degrades the performance of the model on both datasets, verifying the importance of jointly training both the STKG and the next POI recommendation. Another interesting finding would be that the prediction performance reduction of STKG-KGE is more significant on IST than TKY. Such a phenomenon might be attributed to the much sparse of the IST dataset and STKGRec-KGE cannot obtain rich semantic representation. In addition, to understand the convergence of both STKGRec and SRKGRec-KGE, we report the change of the performance and loss with respect to the number of iterations. The loss curves and Rec@5 curves are shown in Fig. 6. Fig. 6(b) clearly shows that STKGRec converges much better than STKGRec-KGE, which proves that STKGRec has better abilities to regularize the representation of entities and relations. Similar results were obtained with the evaluation criteria from Fig. 6(a).

### 5.6. Impact of hyper-parameter setting

**Impact of Spatial–temporal Relation** One key contribution of STKGRec is proposing the spatial–temporal transfer relation that reflects the transition patterns of users between neighboring POIs. We conduct experiments to explore the sensitivity of different time interval units and geographical interval units. Specifically, by setting the time interval units to {0.5 **h**, 1 **h**, 1.5 **h**, 2 **h**}, the geographical interval units to {50 **m**, 100 **m**, 200 **m**, 300 **m**, 400 **m**}, we construct different spatial–temporal transfer relations by pairwise combination.

The results show the performance w.r.t Rec@5 on the NYC dataset, and the trends of the TKY, IST and AUS datasets are similar. As shown in Fig. 7, STKGRec can achieve the best performance when the time interval unit is 0.5 h and the geographical




(a) Rec@5 curves (b) Loss curves

**Fig. 6.** The Rec@5 curves and loss curves of both STKGRec and STKGRec-KGE.

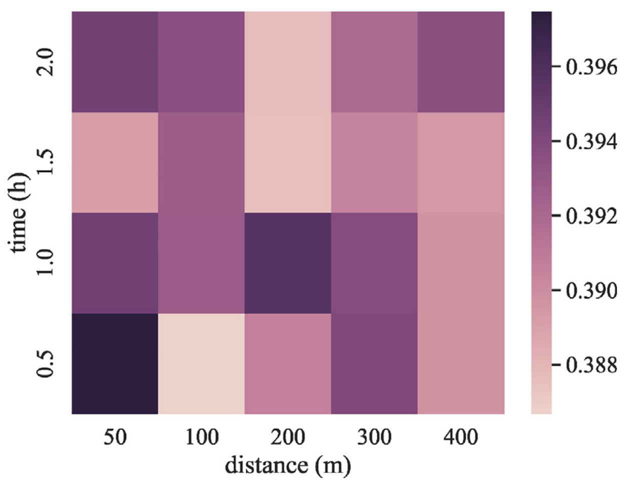



**Fig. 7.** The performance w.r.t Rec@5 with different spatial–temporal transfer relations.

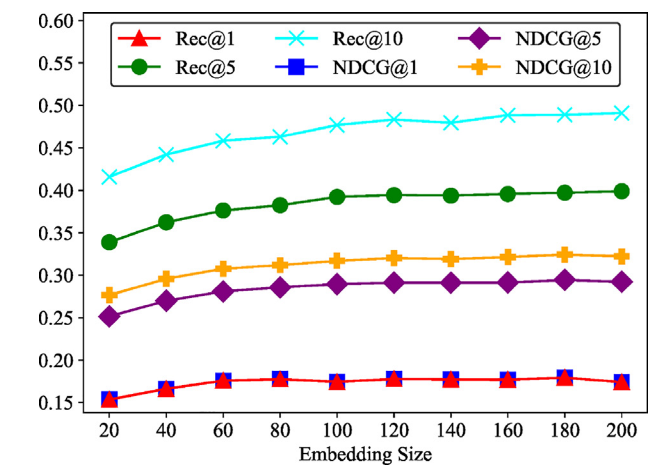



**Fig. 8.** The performance with different embedding size.

interval unit is 50 meter, proving that a fine-grained unit of the spatial–temporal transfer relation can better describe movements of users. However, a relation with a smaller granularity will increase the number of relations, which will lead to a higher time cost of model training. Therefore, the relations composed of appropriate spatial–temporal intervals will benefit the model.

**Impact of Embedding Size** We perform a batch of experiments with different embedding size settings with other optimal hyperparameters fixed on the NYC dataset, and the Recall and NDCG results are shown in Fig. 8. Note similar trends can be observed for TKY, IST and AUS datasets. We can see that when the embedding size is set higher, better prediction performance is achieved. The performance becomes stable when the dimension increases to a certain level.

### *5.7. Impact of data sparsity*

Data sparsity is an inevitable challenge for next POI recommendation. To gain insight into the data sparsity affecting the prediction performance of the STKGRec, we conduct experiments that change the sparsity of the data by reducing check-ins for all users. Considering that the next POI recommendation task has a strong sequential order, we retain the original validation set and test set and generate new training sets with different sparsity by filtering the too old sessions of training data for each user. After data processing, we can obtain four different recent training sets as [20%, 40%, 60%, 80%]. The experimental results of both STKGRec and LSTPM on the NYC, TKY, IST and AUS datasets are shown in Fig. 9.

The performance of both STKGRec and LSTPM intuitively decreases as the sparsity increases on four datasets. Our proposed STKGRec model is superior to LSPTM in terms of Rec@5 and NDCG@5 on four datasets with different sparsity. The performance of STKGRec with the largest sparsity is close to the NDCG@5 of the LSTPM with the least sparsity in Fig. 9(b), (c) and (d), possibly because LSTPM relies too much on the historical check-in data of the user during the training process, and the sparse data cannot make the model fit well. In our STKGRec, the spatial–temporal relation semantic information of the STKG just makes up for this deficiency and greatly alleviates the problem of data sparsity.

## 6. Conclusion and future work

In this paper, we investigate building a spatial–temporal knowledge graph (STKG) by directly using check-in sequences of the users, without considering any external attributes of users and POIs. In STKG, we design a novel spatial–temporal transfer relation that can intuitively capture the transition patterns of the users in trajectories. Based on STKG, we further propose a model for the next POI recommendation in an end-to-end way, which is able to capture both long- and short-term preferences of the users. In the STKGRec model, we apply a historical trajectory encoding module and a nonlocal network to learn the

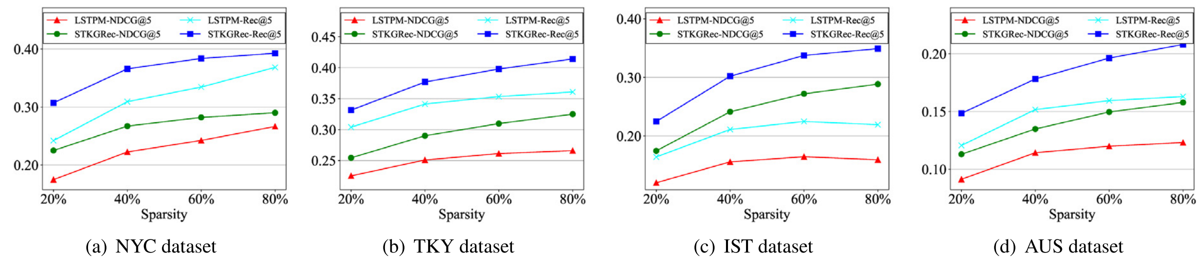



**Fig. 9.** The effectiveness of different sparsity on four datasets.

long-term preferences. To learn short-term preferences of the users in the current sequence, we employ knowledge graph path reasoning to learn the spatial–temporal correlation of consecutive check-ins and present a Time-Geo-dilated GRU to model the spatial–temporal correlation of nonconsecutive check-ins. Extensive experiments on four real-world datasets verify the effectiveness of our STKGRec model. In the future, we plan to expand the STKG with more POI semantic information such as categories and functional areas and multi-source scenario information such as weather and traffic information. Another future direction is to explore the recent hot self-supervised contrastive learning [47] for the underlying motivation representation of user movement.

## CRediT authorship contribution statement

**Wei Chen:** Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing. **Huaiyu Wan:** Resources, Writing – review & editing, Supervision, Funding acquisition. **Shengnan Guo:** Writing – review & editing, Supervision. **Haoyu Huang:** Validation, Resources, Software. **Shaojie Zheng:** Validation, Data curation. **Jiamu Li:** Visualization, Resources. **Shuohao Lin:** Visualization, Data curation. **Youfang Lin:** Writing – review & editing, Supervision.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

I have shared with the code/data in the paper.

## Acknowledgments

This work was supported by the National Key R&D Program of China (Grant No. 2021QY1502).

## References


[1] D. Yao, C. Zhang, J. Huang, J. Bi, SERM: a recurrent model for next location prediction in semantic trajectories, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, ACM, 2017, pp. 2411–2414.

[2] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, D. Jin, DeepMove: Predicting human mobility with attentional recurrent networks, in: Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018, ACM, 2018, pp. 1459–1468.

[3] K. Sun, T. Qian, T. Chen, Y. Liang, Q.V.H. Nguyen, H. Yin, Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation, in: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press, 2020, pp. 214–221.

[4] Y. Lin, H. Wan, S. Guo, Y. Lin, Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction, in: Thirty-Fifth AAAI Conference on Artificial Intelligence, February 2-9, 2021, AAAI Press, 2021, pp. 4241–4248.

[5] S. Rendle, C. Freudenthaler, L. Schmidt-Thieme, Factorizing personalized Markov chains for next-basket recommendation, in: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010, ACM, 2010, pp. 811–820.

[6] Q. Liu, S. Wu, L. Wang, T. Tan, Predicting the next location: A recurrent model with spatial and temporal contexts, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, AAAI Press, 2016, pp. 194–200.

[7] J. Manotumruksa, C. Macdonald, I. Ounis, A deep recurrent collaborative filtering framework for venue recommendation, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, ACM, 2017, pp. 1429–1438.

[8] P. Zhao, H. Zhu, Y. Liu, J. Xu, Z. Li, F. Zhuang, V.S. Sheng, X. Zhou, Where to go next: A spatio-temporal gated network for next POI recommendation, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, AAAI Press, 2019, pp. 5877–5884.

[9] Y. Wu, K. Li, G. Zhao, X. Qian, Personalized long- and short-term preference learning for next POI recommendation, IEEE Trans. Knowl. Data Eng. 34 (4) (2022) 1944–1957.

[10] Y. Luo, Q. Liu, Z. Liu, STAN: spatio-temporal attention network for next location recommendation, in: WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021, ACM / IW3C2, 2021, pp. 2177–2185.

[11] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, M. Guo, RippleNet: Propagating user preferences on the knowledge graph for recommender systems, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, ACM, 2018, pp. 417–426.

[12] X. Tang, T. Wang, H. Yang, H. Song, AKUPM: attention-enhanced knowledge-aware user preference model for recommendation, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, ACM, 2019, pp. 1891–1899.

[13] X. Wang, T. Huang, D. Wang, Y. Yuan, Z. Liu, X. He, T. Chua, Learning intents behind interactions with knowledge graph for recommendation, in: WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021, ACM / IW3C2, 2021, pp. 878–887.

[14] Q. Guo, Z. Sun, J. Zhang, Y. Theng, An attentional recurrent neural network for personalized next location recommendation, in: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press, 2020, pp. 83–90.

[15] H. Wang, Q. Yu, Y. Liu, D. Jin, Y. Li, Spatio-temporal urban knowledge graph enabled mobility prediction, 2021, CoRR abs/2111.03465.

[16] T. Qian, B. Liu, Q.V.H. Nguyen, H. Yin, Spatiotemporal representation learning for translation-based POI recommendation, ACM Trans. Inf. Syst. 37 (2) (2019) 18:1–18:24.

[17] C. Cheng, H. Yang, M.R. Lyu, I. King, Where you like to go next: Successive point-of-interest recommendation, in: IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013, IJCAI/AAAI, 2013, pp. 2605–2611.

[18] Y. Zhu, H. Li, Y. Liao, B. Wang, Z. Guan, H. Liu, D. Cai, What to do next: Modeling user behaviors by time-LSTM, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017, ijcai.org, 2017, pp. 3602–3608.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural

Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 5998–6008.
[20] N.A.H. Haldar, J. Li, M.E. Ali, T. Cai, Y. Chen, T. Sellis, M. Reynolds, Top-k socio-spatial co-engaged location selection for social users, IEEE Trans. Knowl. Data Eng. (2022).
[21] N.A.H. Haldar, M. Reynolds, Q. Shao, C. Paris, J. Li, Y. Chen, Activity location inference of users based on social relationship, World Wide Web 24 (4) (2021) 1165–1183.
[22] D. Lian, Y. Wu, Y. Ge, X. Xie, E. Chen, Geography-aware sequential location recommendation, in: KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020, ACM, 2020, pp. 2009–2019.
[23] D. Kong, F. Wu, HST-LSTM: a hierarchical spatial-temporal long-short term memory network for location prediction, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden, ijcai.org, 2018, pp. 2341–2347.
[24] H. Sun, J. Xu, K. Zheng, P. Zhao, P. Chao, X. Zhou, MFNP: a meta-optimized model for few-shot next POI recommendation, in: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021, ijcai.org, 2021, pp. 3017–3023.
[25] F. Zhang, N.J. Yuan, D. Lian, X. Xie, W. Ma, Collaborative knowledge base embedding for recommender systems, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, ACM, 2016, pp. 353–362.
[26] H. Wang, F. Zhang, X. Xie, M. Guo, DKN: deep knowledge-aware network for news recommendation, in: Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018, ACM, 2018, pp. 1835–1844.
[27] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, T. Chua, Explainable reasoning over knowledge graphs for recommendation, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, AAAI Press, 2019, pp. 5329–5336.
[28] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA, AAAI Press, 2015, pp. 2181–2187.
[29] X. Song, J. Li, Q. Lei, W. Zhao, Y. Chen, A. Mian, Bi-CLKT: Bi-graph contrastive learning based knowledge tracing, Knowl. Based Syst. 241 (2022) 108274.
[30] S. Yang, S. Verma, B. Cai, J. Jiang, K. Yu, F. Chen, S. Yu, Variational co-embedding learning for attributed network clustering, 2021, CoRR abs/2104.07295.
[31] X. Yu, X. Ren, Y. Sun, Q. Gu, B. Sturt, U. Khandelwal, B. Norick, J. Han, Personalized entity recommendation: a heterogeneous information network approach, in: Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014, ACM, 2014, pp. 283–292.
[32] H. Zhao, Q. Yao, J. Li, Y. Song, D.L. Lee, Meta-graph based recommendation fusion over heterogeneous information networks, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017, ACM, 2017, pp. 635–644.
[33] X. Wang, X. He, Y. Cao, M. Liu, T. Chua, KGAT: knowledge graph attention network for recommendation, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, ACM, 2019, pp. 950–958.
[34] H. Wang, M. Zhao, X. Xie, W. Li, M. Guo, Knowledge graph convolutional networks for recommender systems, in: The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, ACM, 2019, pp. 3307–3313.
[35] Y. Yang, Z. Guan, J. Li, W. Zhao, J. Cui, Q. Wang, Interpretable and efficient heterogeneous graph convolutional network, 2020, CoRR abs/2005.13183.
[36] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, 2017, pp. 2724–2743.
[37] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: The 27th Annual Conference on Neural Information Processing Systems 2013, December 5-8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 2787–2795.
[38] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, QuÉBec City, QuÉBec, Canada, AAAI Press, 2014, pp. 1112–1119.
[39] J. Chung, Ç. Gülçehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.
[40] R. Li, Y. Shen, Y. Zhu, Next point-of-interest recommendation with temporal and multi-level context attention, in: IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018, IEEE Computer Society, 2018, pp. 1110–1115.
[41] Y. Lin, Z. Liu, H. Luan, M. Sun, S. Rao, S. Liu, Modeling relation paths for representation learning of knowledge bases, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, The Association for Computational Linguistics, 2015, pp. 705–714.
[42] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.
[43] D. Yang, D. Zhang, V.W. Zheng, Z. Yu, Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs, IEEE Trans. Syst. Man Cybern. Syst. 45 (1) (2015) 129–142.
[44] D. Yang, B. Qu, J. Yang, P. Cudré-Mauroux, Revisiting user mobility and social relationships in LBSNs: A hypergraph embedding approach, in: The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, ACM, 2019, pp. 2147–2157.
[45] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, a Meeting of SIGDAT, a Special Interest Group of the ACL, ACL, 2014, pp. 1724–1734.
[46] L. Huang, Y. Ma, Y. Liu, K. He, DAN-SNR: a deep attentive network for social-aware next point-of-interest recommendation, ACM Trans. Internet Techn. 21 (1) (2021) 2:1–2:27.
[47] K. Hassani, A.H.K. Ahmadi, Contrastive multi-view representation learning on graphs, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, PMLR, 2020, pp. 4116–4126.