# Successive POI Recommendation via Brain-Inspired Spatiotemporal Aware Representation

**Gehua Ma[1,2], He Wang[1,2], Jingyuan Zhao[3,4], Rui Yan[5], Huajin Tang[1,2,6*]**

[1] College of Computer Science and Technology, Zhejiang University;
[2] The State Key Lab of Brain-Machine Intelligence, Zhejiang University
[3] Group Data, The Great Eastern Life Assurance Company Limited
[4] Department of Statistics & Data Science, National University of Singapore
[5] College of Computer Science and Technology, Zhejiang University of Technology
[6] MOE Frontier Science Center for Brain Science and Brain-Machine Integration, Zhejiang University
gehuama@icloud.com, 2449758305@qq.com, zhaojy@nus.edu.sg, Ryan@zjut.edu.cn, htang@zju.edu.cn

## Abstract

Existing approaches usually perform spatiotemporal representation in the spatial and temporal dimensions, respectively, which isolates the spatial and temporal natures of the target and leads to sub-optimal embeddings. Neuroscience research has shown that the mammalian brain entorhinal-hippocampal system provides efficient graph representations for general knowledge. Moreover, entorhinal grid cells present concise spatial representations, while hippocampal place cells represent perception conjunctions effectively. Thus, the entorhinal-hippocampal system provides a novel angle for spatiotemporal representation, which inspires us to propose the SpatioTemporal aware Embedding framework (STE) and apply it to POIs (STEP). STEP considers two types of POI-specific representations: sequential representation and spatiotemporal conjunctive representation, learned using sparse unlabeled data based on the proposed graph-building policies. Notably, STEP jointly represents the spatiotemporal natures of POIs using both observations and contextual information from integrated spatiotemporal dimensions by constructing a spatiotemporal context graph. Furthermore, we introduce a successive POI recommendation method using STEP, which achieves state-of-the-art performance on two benchmarks. In addition, we demonstrate the excellent performance of the STE representation approach in other spatiotemporal representation-centered tasks through a case study of the traffic flow prediction problem. Therefore, this work provides a novel solution to spatiotemporal representation and paves a new way for spatiotemporal modeling-related tasks.

## Introduction

With the rapid growth of location-based web services like Instagram and Yelp, there has been a seismic shift in how people interact with locations around them. Through the exploitation of Points-of-Interest (POIs) and their contexts, successive POI recommendations can benefit users and businesses greatly. As a core of POI information utilization, encoding POIs into vector representation space is of great significance for advanced POI analysis and downstream applications. Existing studies attempt to represent POI from different perspectives and collaborate with user preference modeling to achieve recommendations. Since consecutive check-ins are usually highly correlated, naturally, sequence modeling approaches like the Markov chain model were used to capture the check-in sequential characteristics of POIs (Ye et al. 2011; Liu et al. 2013; Zhang 2014; Feng et al. 2015). Employing the tensor factorization technique, the works (Yang et al. 2017; Wang et al. 2018) modeled target users and POIs separately by interacted features for POI recommendation. More recently, enlightened by neural networks' success, recurrent neural nets were remolded to represent POIs and user preferences implicitly (Liu et al. 2016; Zhao et al. 2019; Zhu et al. 2017). Considering the geographical attributes of POIs, researchers have used power-law distribution, Gaussian distribution, or hierarchical tiling methods to depict the geographical influence over POI distributional features (Ye et al. 2011; Lian et al. 2014; Feng et al. 2017; Chang and Kim 2020; Luo et al. 2020). However, the geographical modeling methods above only provide single-scale or coarse-grained manually designed representations of POI geographical influences, which is deficient in capturing the POI-specific spatial features. Also, arbitrary modeling might even lead to over-parameterization. While temporal dimension offers indeterminate auxiliary information for POI modeling, to utilize the POI temporal information within the check-ins, some works use time interval, time state variables, or temporal transition vectors to promote the POI representing (Zhao et al. 2019, 2016, 2017; Li, Shen, and Zhu 2018; Manotumruksa, Macdonald, and Ounis 2018; Zhao et al. 2020). However, these methods focused on utilizing general temporal patterns among all POIs and failed to exploit the POI-specific visiting time patterns sufficiently. Still, the POI-specific spatiotemporal characteristics were not adequately mined and utilized.

The entorhinal-hippocampal system plays a central role in the mammal cognition architecture. The Nobel Prize-winning neuroscience research (O'keefe and Nadel 1978) demonstrated that entorhinal grid cells provide an effective multi-scale periodic spatial representation (Yuan et al. 2015; Banino et al. 2018; Mai et al. 2020). Moreover, the entorhinal-hippocampal system is also critical for the non-spatial inference that relies on understanding the associations between perceptions from various perspectives

*Author for correspondence.

(Whittington et al. 2018; Stachenfeld, Botvinick, and Gershman 2018; Whittington et al. 2020). Some promising research cast spatial and non-spatial problems as connected graphs and point out that the cells inside the entorhinal-hippocampal structure provide efficient conjunctive representation for those graphs (Stachenfeld, Botvinick, and Gershman 2018; Gustafson and Daw 2011). As the representation mechanism in the entorhinal-hippocampal system was extensively studied, it is widely accepted that conjunctions of representations from different aspects form the hippocampal representation for relational memory (Whittington et al. 2018, 2020; Eichenbaum 2017; MacDonald et al. 2011; Sargolini et al. 2006). For the general spatiotemporal embedding, various contexts can be constructed into affinity graphs for latent representation learning. Furthermore, strategies like conjunctive representing in entorhinal-hippocampal structure can be translated to improve the quality of the representations (see Fig.1 left part).

In this paper, borrowing inspirations from the entorhinal-hippocampal system, we propose the SpatioTemporal aware Embedding framework, namely STE, and apply it to POIs (STEP) for successive POI recommendation. The model architectures are shown in Figure 1. Firstly, we build context graphs to enable unsupervised embedding learning on sparse check-ins. Secondly, we employ a sequential model to represent POIs from the check-in sequence perspective. Most importantly, we introduce a spatiotemporal model consisting of a grid-cell spatial encoder and a visiting time encoder to capture the POI-specific spatiotemporal characteristics. The spatiotemporal model learns to get the POI spatiotemporal latent representations using the spatiotemporal context graph. Finally, we implement successive POI recommendation systems based on the STEP and achieve high performance using simple recurrent neural networks as recommenders.

The main contributions of this work are summarized as follows:

(1) Motivated by the graph-representing strategy of structural knowledge in the entorhinal-hippocampal system, we solve the spatiotemporal embedding learning in a graph-based unsupervised learning manner through specific context graph-building policies, especially the spatiotemporal context graph, to fully exploit rich unlabeled data.

(2) Inspired by the conjunctive representation mechanism in the entorhinal-hippocampal complex, we present a spatiotemporal model with a grid-cell spatial encoder and a time pattern encoder to utilize the spatiotemporal information. The conjunctive representing approach based on a unique spatiotemporal context graph addresses the problem of previous spatiotemporal modeling methods in which spatial and temporal information are isolated and represented separately.

(3) We introduce a successive POI recommendation system by incorporating STEP and simple sequence predictors to show the feasibility of implementing specific applications based on the proposed STE framework. We perform experiments on large real-world datasets to demonstrate the effectiveness of STEP, and our method outperforms baselines according to experimental results. Compared with classical recommendation systems, our POI-centered solution can avoid the ethical risks of artificial intelligence, like personal data leakage, as it does not need access to private information such as user preferences. Furthermore, our framework can be applied to more valuable applications like climate forecasting and urban traffic scheduling as a general spatiotemporal modeling method.

| Notation | Definition |
|---|---|
| $t_j^i$ | $j$-th timestamp of $p_i$ |
| $\boldsymbol{t}^i$ | Visiting time pattern matrix of $p_i$ |
| $\boldsymbol{e}_{spa}^i$ | Spatial vector representation of $p_i$ |
| $\boldsymbol{e}_{seq}^i$ | Sequential vector representation of $p_i$ |
| $\boldsymbol{e}_{st}^i$ | Spatiotemporal conjunctive representation of $p_i$ |
| $\boldsymbol{e}_{step}^i$ | STEP vector representation of $p_i$ |
| $\oplus$ | Tensor concatenation operation |

Table 1: Notations and definitions in this work.

## Preliminaries

Given a set of POIs with corresponding coordinates $\mathcal{P} = \{p_i\}, p_i = (x_i, y_i)$, a check-in sequence is one set of continuous check-ins of one user in one day, denoted as $\mathcal{S}_j = \{(p_1, t_1^1), ..., (p_n, t_m^n)\}$. Unlike previous works, we do not regard all check-in records of a user as one sequence since check-ins with relatively long intervals are not very relevant. Although we assign notation to users for generality, the user information is not used in the training phase except to split sequences.

We define context graphs as graphs that encode context information as affinity among POIs. Various contexts in the check-in records can be easily built into graphs $G_p = \{V_p, E_p\}$, where $V_p$ is the set of POIs and $E_p$ is the set of edges between adjacent POIs. The edges in context graphs represent the correlation between neighboring POIs defined by geographical distance, relative position in check-in sequences, or spatiotemporal adjacent criteria. We summarize notations in this paper using Table 1.

**Data description** The Instagram Check-in dataset (Chang et al. 2018) was collected from Instagram in New York, and the data was preprocessed in the same manner as previous works (Zhao et al. 2016, 2017). The Instagram Check-in dataset has been pre-processed when it is made public, it includes 2 216 631 check-in records at 13 187 POIs of 78 233 users. Check-in sequences are sorted by timestamps; the first 70% is used as a training set, and the remaining 30% for validation and testing. The Gowalla dataset is a globally-collected large-scale social media dataset (Cho, Myers, and Leskovec 2011). We eliminate users with fewer than ten check-ins and POIs accessed by fewer than ten users. Then the check-in records are sorted according to timestamps and the first 70% check-ins are used for training and the remaining latest records for testing. We perform vivid data analyses in the Appendix due to space constraints.

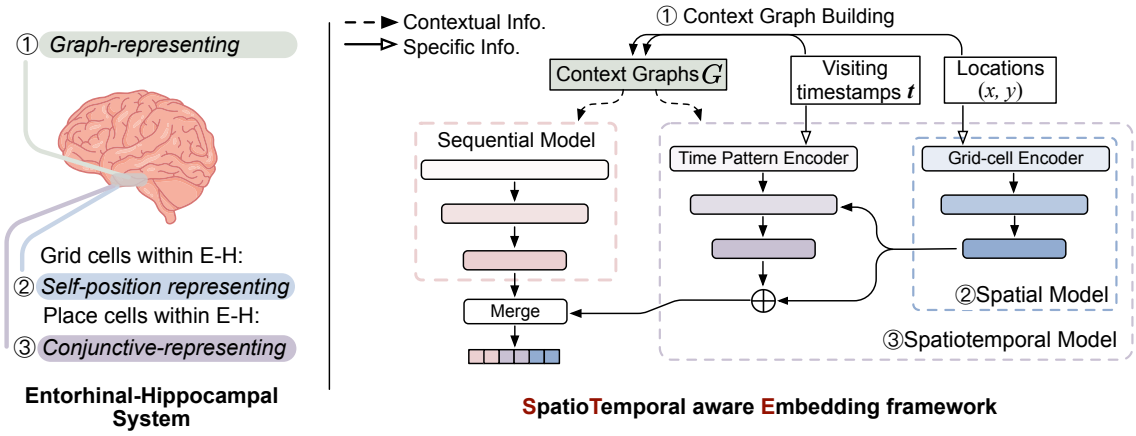


Figure 1: Representing mechanisms in the entorhinal-hippocampal system (*E-H system* for short) and the framework of the spatiotemporal embedding model. The proposed STE framework consists of context graph-building strategies to construct simplified affinity graphs; a spatiotemporal model to extract rich item-specific spatiotemporal features; and a sequential model to extract sequential feature embeddings. The uniqueness of our spatiotemporal information usage is that we represent items from the spatiotemporal perspective (not isolated) using observations and contexts conjunctively.

## SpatioTemporal Embedding Model of POIs

We illustrate the components of STEP in order: the sequential model, the spatiotemporal model, and state the STEP-based successive POI recommendation method. We adopt a *simple-minded (no-parameters) edge weighting policy* for constructing all context graphs. Weight $A_{i,j} = 1$ if vertices $i$ and $j$ are connected; this simplification avoids the necessity of choosing edge-weighting parameters.

### Sequential Model

The sequential model represents POIs using context graph $G_{seq}$. Given one POI and its context in the check-in sequence, entry $A_{i,j}$ in the adjacency matrix of $G_{seq}$ is 1 if $p_i, p_j$ are within the same context window. This is a common way to mine the sequential correlations of tokens like words (Mikolov et al. 2013) or POIs (Lim, Hooi, and Wang 2020). Our sequential model aims to predict true contextual POIs, *i.e.*, connected vertices in $G_{seq}$. Intuitively, minimizing the objective function over all target-neighbor pairs guarantees that POIs sharing similar sequential context will have shorter distances in embedding space (Hadsell, Chopra, and LeCun 2006). To avoid the intractable summation over the whole context space, we follow the noise contrastive sampling approach (Gutmann and Hyvärinen 2012; Mikolov et al. 2013) to get an approximated surrogate loss

$$\mathcal{L}_{seq}(\theta_{seq}) = -\sum_{p_i,p_j\in\mathcal{P}} \big[\mathbb{I}(\gamma=1)\log\sigma(\boldsymbol{e}^i_{seq}\cdot\boldsymbol{e}^j_{seq}) + \mathbb{I}(\gamma=-1)\log\sigma(-\boldsymbol{e}^i_{seq}\cdot\boldsymbol{e}^j_{seq})\big], \tag{1}$$

where $\gamma = 1$ if $(p_i, p_j)$ is a sequential neighboring pair and $\gamma = -1$ if not, indicator $\mathbb{I}$ outputs 1 when the argument condition is true and otherwise 0. This unsupervised loss can also be seen as taking expectation concerning the distribution $\mathbb{P}(p_i, p_j, \gamma)$ over $\mathcal{P}$, which is conditioned on the POI sequential context graph $G_{seq}$.

### Spatiotemporal Model

In this section, we illustrate the POI spatiotemporal conjunctive embedding model in detail. The proposed spatiotemporal model is composed of two key components: a POI spatial model and a POI visiting time encoder; an intuitive illustration can be found in Fig. 1.

**Spatial model** The spatial sub-model takes location observations $(x_i, y_i)$ and spatial context graph $G_{spa}$ to produce spatial representations. Inspired by the multi-scale periodic representation of grid cells in mammals, we formulate our POI spatial contextual encoder to use sinusoidal and cosinusoidal functions of different scales to encode the raw locations of POIs in geographical space following previous works (Gao, Xie, and Zhu 2019; Mai et al. 2020). Given a POI $p_i = (x_i, y_i) \in \mathbb{R}^2$, the grid-cell model-based encoder encodes the coordinates in 2-D Euclidean space into spatial latent representations in $\mathbb{R}^{d_{spa}}$. We denote the grid cell encoder-based spatial embedding of POI $p_i$ as

$$\boldsymbol{e}^i_{spa} = \phi(\psi(x_i, y_i); \theta_{spa}), \tag{2}$$

where

$$\psi(x_i, y_i) = \psi^1(x_i, y_i)\cdots\oplus\psi^s(x_i, y_i)\cdots\oplus\psi^S(x_i, y_i) \tag{3}$$

is concatenated multi-scale representations of $6S$ dimensions, $S$ denotes the number of grid scales and $\phi$ represents fully connected non-linear layers. Considering three unit vectors $\boldsymbol{a}_1$=$[1, 0]^{\mathrm{T}}$, $\boldsymbol{a}_2$=$[-1/2, \sqrt{3}/2]^{\mathrm{T}}$, $\boldsymbol{a}_3$=$[-1/2, -\sqrt{3}/2]^{\mathrm{T}} \in \mathbb{R}^2$, at each frequency, position codes

$$\psi^s(x_i, y_i) = \psi^s_1 \oplus \psi^s_2 \oplus \psi^s_3 \tag{4}$$

are computed via

$$\psi^s_k(x_i, y_i) = \left[\cos(\frac{[x_i, y_i]\cdot\boldsymbol{a}_k}{\rho\lambda_{\min}}), \sin(\frac{[x_i, y_i]\cdot\boldsymbol{a}_k}{\rho\lambda_{\min}})\right], \quad k \in \{1, 2, 3\} \tag{5}$$

and $\rho = (\lambda_{\max}/\lambda_{\min})^{s/(S-1)}$. $\lambda_{\min}$ and $\lambda_{\max}$ are the minimum and maximum scale values, here we use $S = 64$ following the previous work (Mai et al. 2020) and set $\lambda_{\max} = 1km$, $\lambda_{\min} = 0.1km$.

**Spatial-neighboring definition** We project the coordinates in the geographical coordinate system WGS84 to the projection coordinate system NAD27 to get locations of POIs in $\mathbb{R}^2$. For each entry $A_{i,j}$ in the adjacency matrix of spatial context graph $G_{spa}$, we assign $A_{i,j}$ using the geographical distances. Specifically, we computed the geographical distances between POIs and constructed an undirected spatial context graph $G_{spa}$ with *uniform edges* among the *top-ten closest* POIs (nearest neighbors policy). As the grid cell encoder can handle geographical distributions at different scales (Mai et al. 2020), we do not use a specific radius ($\epsilon$-neighborhoods policy) to filter the neighboring POIs to fully exploit the multi-scale representation capability. The spatial graph construction process is related to (Lim, Hooi, and Wang 2020), in which the edges are weighted according to average distances to enable graph attention computations.

Given a target POI $p_i$, neighboring contextual POI set $\mathcal{P}^+_{spa}$ and negative set $\mathcal{P}^-_{spa}$ sampled from $G_{spa}$, the unsupervised embedding learning can simply be maximizing the log-likelihood of observing the true context POIs. We can formulate this target with negative sampling via a general objective function:

$$\begin{aligned}\mathcal{O}(ctx) = -\sum_{p_i \in \mathcal{P}} \sum_{p_j \in \mathcal{P}^+_{ctx}} \big[\log \sigma(\boldsymbol{e}^j_{ctx} \cdot \boldsymbol{e}^i_{ctx}) \\ + \frac{1}{K} \sum_{p_k \in \mathcal{P}^-_{ctx}} \log \sigma(-\boldsymbol{e}^k_{ctx} \cdot \boldsymbol{e}^i_{ctx})\big],\end{aligned} \tag{6}$$

where $ctx$ indicates the context graph type and $ctx \in \{seq, spa, st\}$ in this work, $\sigma$ is the sigmoid function and $K$ denotes the number of samples in negative sample set $\mathcal{P}^-_{ctx}$. Following Eq.6, the loss function for the spatial context embedding model is:

$$\mathcal{L}_{spa}(\theta_{spa}) = \mathcal{O}(spa). \tag{7}$$

**Spatiotemporal context graph construction** For constructing spatiotemporal context graph $G_{st}$, we want to mine the item-specific spatiotemporal conjunctions, so for each entry $A_{i,j}$ of the adjacency matrix of $G_{st}$, we assign $A_{i,j} = 1$ following the hierarchy of *neighboring timestamps* $\rightarrow$ *temporal neighboring* $\rightarrow$ *spatiotemporal neghboring*:

1. Neighboring timestamps. Given an arbitrary timestamp pair $(t_1, t_2)$, time interval $\Delta t \triangleq |t_1 - t_2|$ and $\Delta wkd \triangleq |wkd(t_1) - wkd(t_2)|$ where $wkd(t) = 1$ if $t$ is weekend else 0. For one time $t$, its temporal-neighboring timestamps are those within the neighborhood window and satisfy $\Delta wkd = 0$, as shown in Fig. 2.A. $\mathtt{h}$ is a hyperparameter indicates temporal context window width and $\mathtt{h} \in (0, 24)$ hours.
2. POI temporal neighboring. POI $p_i$ and $p_j$ with corresponding visiting timestamp sets $\mathcal{T}_i = \{t^i_1, t^i_2, \dots\}$ and $\mathcal{T}_j = \{t^j_1, t^j_2, \dots\}$. The number of neighboring timestamp pairs $(t^i, t^j) > \mathtt{m}$, where $t^i \in \mathcal{T}_i$, $t^j \in \mathcal{T}_j$, $\mathtt{m}$ is a threshold.

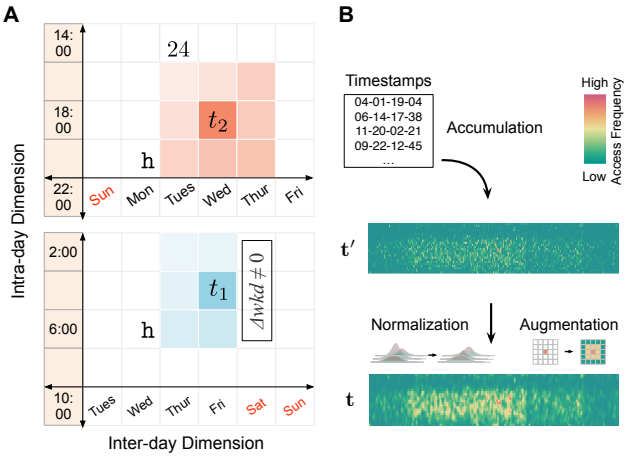


Figure 2: A. Temporal neighboring examples with $\mathtt{h} = 2$. For $t_1$, some time stamps are excluded from the temporal neighborhood window as $\Delta wkd \neq 0$. B. Schematic of the visiting time encoding process.

3. POI spatiotemporal neighboring. If $p_i$, $p_j$ are spatial and temporal neighboring, they are spatiotemporal neighboring.

It is redundant for individual POI-specific temporal modeling since solely relying on time information, we are not able to recommend reasonable candidate POIs (visits may take place contemporarily all over the world). Thus, time is regarded as a supplementary dimension of basic spatial information, our spatiotemporal context graph provides an effective way to combine the POI-specific temporal and geographical information.

**Visiting time pattern encoding** We develop a visiting time pattern encoding method to tensorise the temporal observations (visiting records in timestamps $t$) to provide observation inputs for the spatiotemporal model. Unlike previous works (Liu et al. 2016; Zhao et al. 2019; Zhu et al. 2017; Zhao et al. 2016), we focus on the POI-specific temporal patterns rather than general temporal characteristics among all timestamps. Compared with previously used time interval-based or hard-coded methods, our encoding scheme can tensorise the item-specific temporal information more precisely and be able to provide a reliable decision basis for spatiotemporal modeling. Since the visiting time patterns are relatively stable on the scale of year but fickle on the hour and the date scale, in our encoding scheme, visits are counted by the check-in timestamps to raw matrices $\boldsymbol{t}' \in \mathbb{R}^{24\times366}$. Then, the raw matrices $\boldsymbol{t}'$ are normalized and applied Gaussian kernel for smoothing; this process also reasonably augments the POI accessing time data. The visiting time encoding process is shown schematically in Fig. 2. B. The final matrix representation $\boldsymbol{t} = \mathtt{smooth}(\mathtt{norm}(\boldsymbol{t}'))$ retains the fine-grained check-in time patterns as well as rough item-specific visiting time features.

**Spatiotemporal conjunctive embedding learning** We formulate the spatiotemporal conjunctive representation as:

$$\boldsymbol{e}_{st}^{i} = \phi_{\theta_{st}}(\phi_{\theta_{time}}(\boldsymbol{t}^{i}), \boldsymbol{e}_{spa}^{i}) \oplus \boldsymbol{e}_{spa}^{i}, \tag{8}$$

where $\phi$ indicates fully-connected layers, following the formulation in Eq.6, we implement the spatiotemporal conjunctive representation learning by minimizing:

$$\mathcal{L}(\theta_{time}, \theta_{st}) = \mathcal{O}(st). \tag{9}$$

We sample $\mathcal{P}_{st}^{+}$ and $\mathcal{P}_{st}^{-}$ from $G_{st}$, where $\mathcal{P}_{st}^{+}$ is the set of spatiotemporal-neighboring POIs whereas $\mathcal{P}_{st}^{-}$ is the set of negative POIs.

During the optimization procedure, the spatial model is jointly optimized as a sub-model of the spatiotemporal model; the full objective of the spatiotemporal model is

$$\mathcal{L}_{st} = \mathcal{L}(\theta_{time}, \theta_{st}) + \lambda_{spa}\mathcal{L}_{spa}(\theta_{spa}), \tag{10}$$

$\lambda_{spa}$ is a weighting factor for preserving the spatial context information during the spatiotemporal modeling. We first sample a batch of spatial context $G_{spa}$ to optimize the spatial context loss $\mathcal{L}_{spa}$ to preserve geographical context. Next, we sample a batch of spatiotemporal context $G_{st}$ to optimize the spatiotemporal loss $\mathcal{L}_{st}$ to preserve the spatiotemporal context. We repeat above procedures for $I_0$ and $I_1 = I_0/\lambda_{spa}$ iterations respectively to approximate the balancing factor $\lambda_{spa}$. We update all parameters $\{\theta_{spa}, \theta_{time}, \theta_{st}\}$ of spatiotemporal model in iterations until the overall loss $\mathcal{L}_{st}$ converges.

## Successive POI Recommendation with STEP

We present the STEP (STE of POIs)-based recommendation method in detail in this section. Taking the spatiotemporal data as input, we construct context graphs $G$ and feed the observations (locations and time patterns) into the STEP model to perform embedding learning. The embeddings are smoothed according to corresponding context graphs to preserve contextual information. Then, POI vector representations (embeddings) $\boldsymbol{e}_{seq}, \boldsymbol{e}_{st}$ are merged as spatiotemporal embedding $\boldsymbol{e}_{step}$ and fed into the recommender to generate an estimated embedding $\hat{\boldsymbol{e}}$. Specifically, we use concatenation

$$\boldsymbol{e}_{step}^{i} = \boldsymbol{e}_{seq}^{i} \oplus \boldsymbol{e}_{st}^{i} \tag{11}$$

to merge the two sequential and spatiotemporal embeddings. This merging policy can preserve information from different spaces without extra parameters and not require the embeddings to be in the same dimension (*e.g.* $d_{st} = d_{seq}$ ) thus providing more flexibility. We adopt two-layer recurrent networks as the recommender model.

**Embedding model optimization** Parameters in the STEP embedding model are optimized according to the corresponding loss function $\mathcal{L}(\theta_*) = \mathcal{L}_* + \alpha||\theta_*||_2$, where $\mathcal{L}_* \in \{\mathcal{L}_{seq}, \mathcal{L}_{st}\}$, $\theta_* \in \{\theta_{seq}, \Theta_{st}\}$, $\Theta_{st} = \{\theta_{spa}, \theta_{time}, \theta_{st}\}$. $\alpha$ is the weighting factor of the 2-norm regularizer.

**Recommender model optimization** The predictor is then optimized with the pre-trained STEP embedding model for the next POI recommendation task. During the training phase, given an $n$-length check-in sequence $\mathcal{S}_j$, we can get corresponding STEP embedding series of POIs $\{\boldsymbol{e}_{step}^{(1)}, \ldots, \boldsymbol{e}_{step}^{(gt)}\}$, the last POI is regarded as the recommendation target. The target of the recommender is to predict the representation $\hat{\boldsymbol{e}}_{step}$ similar to the embedding of true successive POI $\boldsymbol{e}_{step}^{(gt)}$, formally described as:

$$\arg\max_{\theta_{pred}} \sum_{\mathcal{S}_j \in \mathcal{S}} \texttt{sim}\left(\hat{\boldsymbol{e}}_{step},\ \boldsymbol{e}_{step}^{(gt)}\right). \tag{12}$$

The objective function of the recommender is:

$$\begin{aligned}\mathcal{L}_{pred}(\theta_{pred}) = -\sum_{\mathcal{S}_j \in \mathcal{S}} \Big[ &\log \sigma' \left(\texttt{sim}(\boldsymbol{e}_{step}^{(gt)}, \hat{\boldsymbol{e}}_{step})\right) \\ &- \log(\sum_{p_i \in \mathcal{P}} \sigma'(\texttt{sim}(\boldsymbol{e}_{step}^{i}, \hat{\boldsymbol{e}}_{step})))\Big],\end{aligned} \tag{13}$$

where $\sigma' = \exp(\texttt{LeakyReLU}(\cdot))$ and $\texttt{sim}(\cdot,\cdot) = \frac{\mathbf{a}\cdot\mathbf{b}}{||\mathbf{a}||\cdot||\mathbf{b}||}$. During the testing phase, we compute the cosine similarity scores to rank the candidate POIs to generate recommendation lists.

# Experiments

We perform the successive POI recommendation task on the Instagram Check-in dataset, the Gowalla dataset, and the traffic flow forecasting task on TaxiBJ15 and TaxiBJ.

## Successive POI Recommendation Task

**Metrics** During the system inferencing phase, the recommendation system recommends a POI list according to the estimated scores of candidate POIs for every trial sequence. We apply widely-used metrics HIT@K (if the ground truth is within the top-k of the list, a score of 1 is awarded, else 0), $k = 1, 5, 10$, and MRR (Mean Reciprocal Rank) for evaluation. These metrics reflect different aspects of the recommendation lists, HIT@K measures the rate of valid recommendation among all trials, whereas MRR scores the quality of the entire recommendation list.

**Hyper-parameter settings** We set the hyper-parameters of our proposed method to the following default values. We set context window size in the POI sequential model to 2 and adopt $\texttt{h} = 2, \texttt{m} = 11$ for building $G_{st}$. We utilize Adam optimizer with batch size $512, \beta_1 = 0.9, \beta_2 = 0.999$ and set the initial learning rate to $0.001$ followed by a reduce-on-plateau decay policy, the decay factor is $0.1$ during the training. Weighting factors $\alpha, \lambda_{spa}$ are set to $1 \times 10^{-4}, 0.2$ and the embedding dimensions $\{d_{seq}, d_{spa}, d_{st}\}$ are set to $\{32, 64, 96\}$.

We compare the STEP-based successive POI recommendation method with representative methods:

- **Two-stage approaches.** We choose six methods consisting of three embedding models and two recommenders. For the embedding model, we use the following embedding models. (1) *Random*, (2) *Skip-Gram* (Liu, Liu, and Li 2016), (3) *CAPE* (Chang et al. 2018) and (4) *Geo* (Mai et al. 2020). For the recommender model, two-layer networks based on (1) *GRU* unit (Merri and Fellow 2014)

| DATASET | Instagram Check-in | | | | Gowalla | | | |
|---|---|---|---|---|---|---|---|---|
| METHOD\METRIC | HIT@1 | HIT@5 | HIT@10 | MRR | HIT@1 | HIT@5 | HIT@10 | MRR |
| Random+GRU | 0.1197 | 0.2207 | 0.2726 | 0.1792 | 0.0715 | 0.0725 | 0.0732 | 0.0727 |
| Random+LSTM | 0.1207 | 0.2225 | 0.2751 | 0.1805 | 0.0722 | 0.0736 | 0.0749 | 0.0737 |
| Skip-Gram+GRU | 0.1356 | 0.2419 | 0.3040 | 0.1919 | 0.1090 | 0.2111 | 0.2617 | 0.1612 |
| Skip-Gram+LSTM | 0.1318 | 0.2344 | 0.2984 | 0.1875 | 0.1085 | 0.2101 | 0.2585 | 0.1594 |
| CAPE+GRU‡ | 0.1390 | 0.2433 | 0.3079 | 0.1953 | $N/A$ | $N/A$ | $N/A$ | $N/A$ |
| CAPE+LSTM‡ | 0.1381 | 0.2412 | 0.3054 | 0.1939 | $N/A$ | $N/A$ | $N/A$ | $N/A$ |
| Geo+GRU | 0.1619 | 0.2616 | 0.3248 | 0.2093 | 0.1267 | 0.2309 | 0.2834 | 0.1684 |
| Geo+LSTM | 0.1622 | 0.2594 | 0.3128 | 0.1875 | 0.1233 | 0.2296 | 0.2811 | 0.1701 |
| ST-RNN† | 0.1054 | 0.2019 | 0.2426 | 0.1681 | 0.0519 | 0.0953 | 0.1304 | 0.2187 |
| STGN† | – | – | – | – | 0.0256 | 0.0784 | 0.1144 | 0.0590 |
| STGCN† | – | – | – | – | 0.0424 | 0.1134 | 0.1625 | 0.0842 |
| LSTPM† | 0.1261 | 0.2134 | 0.3121 | 0.1957 | 0.1468 | 0.2506 | 0.2983 | 0.1998 |
| STP-DGAT† | – | – | – | – | 0.1344 | 0.2414 | 0.2653 | 0.1856 |
| STP-UDGAT† | – | – | – | – | 0.1475 | 0.2911 | 0.3285 | 0.2130 |
| STEP+RNN | 0.2458 | 0.3170 | 0.3502 | 0.2822 | 0.1495 | 0.2878 | 0.3634 | 0.2222 |
| STEP+GRU | **0.2467** | 0.3057 | 0.3336 | 0.2781 | 0.1490 | 0.2912 | 0.3636 | 0.2233 |
| STEP+LSTM | 0.2454 | **0.3204** | **0.3556** | **0.2835** | **0.1539** | **0.2968** | **0.3728** | **0.2282** |

Table 2: Comparisons with baselines on two datasets, we mark best values with bold fonts and underline the suboptimal ones. CAPE-based methods are not applicable on Gowalla (since no tweets were provided), and we do not report the results of some methods on Instagram Check-in as they cannot be reproduced faithfully. †: requiring user preference information, ‡: requiring additional semantic content information.

| AVERAGE VALUE | Instagram | Gowalla |
|---|---|---|
| Records per POI | 168.1 | 34.2 |
| $E_{seq}$ per POI | 66.2 | 35.5 |
| $E_{spa}$ per POI | 10.0 | 10.0 |
| $E_{st}$ per POI | 0.997 | 0.596 |

Table 3: Context graphs statistics of two datasets, $E$ stands for edge.

and (2) *LSTM* unit (Hochreiter and Schmidhuber 1997) are used.

- **One-stage approaches.** We choose representative one-stage methods as baselines. (1) *ST-RNN* (Liu et al. 2016). (2) *STGN* (Zhao et al. 2019), an LSTM variant that models visit preferences with time and distance considerations, and the improved variant *STGCN*. (3) *LSTPM* (Sun et al. 2020) is a LSTM-based method. (4) *STP-DGAT* and *STP-UDGAT* (Lim, Hooi, and Wang 2020) are spatial-temporal-preference user dimensional graph attention networks.

**Comparison results** According to results in Table 2, our method outperforms the baselines by significant margins on both datasets, and the gains in recommendation accuracy are especially substantial on the Instagram dataset with rich temporal information (according to Table 3, POIs in Gowalla have less visiting timestamps). The one-stage recurrent network-based methods, such as *ST-RNN*, *LSTPM* surpass basic embedding-based methods by more sufficient exploitation of user-preference spatiotemporal properties. However,

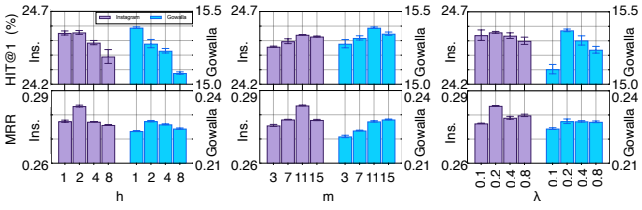


Figure 3: Effect of hyper-parameters h, m, $\lambda_{spa}$ on Instagram Check-in and Gowalla datasets. The means and standard deviations are computed over five runs using different random seeds.

these methods remain inferior to STEP-based ones, although the STEP stands without user preference consideration. The advantageous performance of our method over the competitors can be attributed to its efficient use of the item-specific spatiotemporal nature. We observe significant performance improvements of the STEP-based methods in terms of MRR, indicating the STEP-based methods provide better candidate lists on both datasets and benefit from the efficiency of the proposed spatiotemporal embedding model. The usage of LSTM units in recommender slightly improves the performance compared with basic RNN cells because of their advantages in gate functions of recurrent connections. Moreover, the basic RNN recurrent net equipped with STEP embedding can also outperform one-stage SOTAs. This also proves the effectiveness of our brain-inspired spatiotemporal embedding model.

**Effect of hyper-parameters** We study the effect of newly-introduced hyper-parameters h, m, $\lambda_{spa}$, and report the re-

| | | HIT@1 | HIT@5 | HIT@10 | MRR |
|---|---|---|---|---|---|
| Instagram | FULL | 0.2454 | **0.3204** | **0.3556** | **0.2835** |
| | W/O (1) | 0.2433 | 0.2544 | 0.2607 | 0.2504 |
| | W/O (2) | 0.2452 | 0.3151 | 0.3442 | 0.2798 |
| | W/O (3) | 0.2399 | 0.3037 | 0.3305 | 0.2727 |
| | W/O (4) | 0.2298 | 0.2531 | 0.2674 | 0.2451 |
| | only (5) | **0.2466** | 0.2840 | 0.3021 | 0.2664 |
| Gowalla | FULL | **0.1539** | **0.2968** | **0.3728** | **0.2282** |
| | W/O (1) | 0.1461 | 0.2825 | 0.3540 | 0.2174 |
| | W/O (2) | 0.1509 | 0.2921 | 0.3664 | 0.2248 |
| | W/O (3) | 0.1006 | 0.2169 | 0.2922 | 0.1657 |
| | W/O (4) | 0.0973 | 0.2117 | 0.2866 | 0.1610 |
| | only (5) | 0.1252 | 0.2142 | 0.2519 | 0.1683 |

Table 4: Effectiveness of using (1) temporal information, (2) visiting time encoder, (3) grid-cell encoder, (4) spatial embedding preserving, and (5) the spatiotemporal model. We mark the best ones with bold fonts and underline the suboptimal ones.

sults using HIT@1 and MRR in Fig. 3. The parameters h and m control the sparsity of the spatiotemporal context graph and $\lambda_{spa}$ regulates the importance of the spatial context smoothness term in the spatiotemporal model objective function. We alter h to build spatiotemporal context graphs with decreasing sparsity as larger h corresponds to coarse temporal-neighboring condition. The increasing h within a certain range results in performance improvements but appears detrimental to the precise top-1 recommendation. The best performance (determined by MRR) is obtained when $\mathtt{h} = 2$. The choice of h is task-related; according to our results on two datasets, $\mathtt{h} = 2$ can be a good initial value for POI recommendation. This value can be further adjusted for different application scenarios or datasets to build spatiotemporal context graphs with desired sparsity. We set m from 3 to 15, and larger m leads to a sparser spatiotemporal context graph. We observe the performance slightly changes after increasing the threshold m, when $\mathtt{m} = 11$, the best performance is obtained. We also investigate the effect of the balancing factor $\lambda_{spa}$ for spatiotemporal model training, the recommendation system achieves the best performance when $\lambda_{spa} = 0.2$ and further increases only bring minor improvements. Thus we select 0.2 as a default value in this work, this also helps reduce unnecessary iterations during the model training.

**Ablation study** We study the effectiveness of STEP modules by performing successive POI recommendation tasks with LSTM recommender, the method using the standard STEP embedding model is referred to as FULL. After the removal of (1) POI-specific time information processing module, the spatiotemporal model degenerates into a spatial model. According to Table 4, the method performs worse without (1) on both datasets. According to Table 3, POIs in Gowalla have sparser specific observation $\boldsymbol{t}$s and contextual information from $G_{st}$. This results in more significant performance improvements on the Instagram set than on Gowalla. We replace POI visiting time pattern matrices $\boldsymbol{t}$ with random-initialized matrices in $\mathbb{R}^{24\times366}$. We note that the use of (2) POI visiting time encoder improves the recommendation performance on both datasets, and the improvement is positively correlated with the temporal information abundance of the dataset. We use a one-layer neural network location encoder $\psi'(x, y)$ to replace (3) grid-cell encoder in the STEP to demonstrate its effectiveness. Results in Table 4 demonstrate that the grid-cell encoder improves the quality of STEP representation and leads to better successive POI recommendation performances on both datasets. We observe noticeable performance drops after the removal of (4) spatial embedding preserving in Table 4 since the spatiotemporal conjunctive representation integrates spatial and temporal attributes at the cost of spatial information loss (due to the dimensional reduction). The use of (4) spatial embedding information preserving alleviates this problem to a certain extent.

A simple LSTM recommender can achieve competitive recommendation performance *without* check-in sequential information consideration (Table 4 only (5) rows), demonstrating the effectiveness of the spatiotemporal model in STEP. As visiting sequential information provides relatively coarse-grained POI depictions, the removal of the sequential model even leads to HIT@1 improvement on the Instagram dataset. Also, the metric fallen on Gowalla (-18.6%, -27.8%, -32.4%, -26.2%) is more significant than those on Instagram set (+3.1%, -11.4%, -15.0%, -6.0%), exactly opposed to their temporal information abundance, quantitatively evaluated by average timestamps per POI.

### Traffic Flow Forecasting with STE

To demonstrate the generalizability of STE, we perform the traffic flow forecasting task with the proposed spatiotemporal embedding methods. The relevant content is presented in detail in the Appendix.

## Conclusion and Discussion

In this paper, we propose the spatiotemporal embedding framework STE and apply it to POIs (STEP). To the best of our knowledge, this is the first work that translates entorhinal-hippocampal representing mechanisms to the spatiotemporal embedding. Inspired by the graph-representing policy in the brain entorhinal-hippocampal system, STEP captures sequential and spatiotemporal representation from unlabeled sparse data through context graph building and graph-based embedding learning. Moreover, STEP provides a highly efficient spatiotemporal model motivated by grid cells' multi-scale spatial representation and place cells' conjunctive representation, which overcomes the problem caused by frequently used separate representing. STEP-based successive POI recommendation method outperforms baselines and SOTAs on two real datasets without user preference invasion. Furthermore, this work presents a practical framework for effective spatiotemporal modeling of general items, enabling more valuable spatiotemporal tasks such as climate forecasting and urban traffic management.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 62236007 and Grant 62276235, and the Key R&D Program of Zhejiang (2023C03001).

The authors would also like to acknowledge anonymous reviewers and chairs for providing insightful comments to help improve this work.

## References

Banino, A.; Barry, C.; Uria, B.; Blundell, C.; Lillicrap, T.; Mirowski, P.; Pritzel, A.; Chadwick, M. J.; Degris, T.; Modayil, J.; Wayne, G.; Soyer, H.; Viola, F.; Zhang, B.; Goroshin, R.; Rabinowitz, N.; Pascanu, R.; Beattie, C.; Petersen, S.; Sadik, A.; Gaffney, S.; King, H.; Kavukcuoglu, K.; Hassabis, D.; Hadsell, R.; and Kumaran, D. 2018. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705): 429–433.

Chang, B.; and Kim, S. 2020. Learning Graph-Based Geographical Latent Representation for Point-of-Interest Recommendation. In *Proceedings of the ACM International Conference on Information & Knowledge Management*, 135–144.

Chang, B.; Park, Y.; Park, D.; Kim, S.; and Kang, J. 2018. Content-aware hierarchical point-of-interest embedding model for successive POI recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 3301–3307.

Cho, E.; Myers, S. A.; and Leskovec, J. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1082–1090.

Eichenbaum, H. 2017. On the integration of space, time, and memory. *Neuron*, 95(5): 1007–1018.

Feng, S.; Cong, G.; An, B.; and Chee, Y. M. 2017. POI2Vec : Geographical Latent Representation for Predicting Future Visitors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 102–108.

Feng, S.; Li, X.; Zeng, Y.; Cong, G.; Meng, Y.; and Quan, C. 2015. Personalized Ranking Metric Embedding for Next New POI Recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

Gao, R.; Xie, J.; and Zhu, Y., Songchunand Wu. 2019. Learning Grid Cells as Vector Representation of Self-Position Coupled with Matrix Representation of Self-Motion. In *Proceedings of the International Conference on Learning Representations*.

Gustafson, N. J.; and Daw, N. D. 2011. Grid Cells , Place Cells , and Geodesic Generalization for Spatial Reinforcement Learning. *PLoS Computational Biology*, 7(10): 1–14.

Gutmann, M. U.; and Hyvärinen, A. 2012. Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics. *Journal of Machine Learning Research*, 13(2).

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 1735–1742.

Hoang, M. X.; Zheng, Y.; and Singh, A. K. 2016. FCCF: forecasting citywide crowd flows based on big data. In *ACM SIGSPATIAL*.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Li, R.; Shen, Y.; and Zhu, Y. 2018. Next Point-of-Interest Recommendation with Temporal and Multi-level Context Attention. *Proceedings of the IEEE International Conference on Data Mining*, 1110–1115.

Lian, D.; Zhao, C.; Xie, X.; Sun, G.; Chen, E.; and Rui, Y. 2014. GeoMF : Joint Geographical Modeling and Matrix Factorization for Point-of-Interest Recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 831–840.

Lim, N.; Hooi, B.; and Wang, X. 2020. STP-UDGAT : Spatial-Temporal-Preference User Dimensional Graph Attention Network for Next POI Recommendation. In *Proceedings of the ACM International Conference on Information & Knowledge Management*, 845–854.

Liu, Q.; Wu, S.; Wang, L.; and Tan, T. 2016. Predicting the next location: A recurrent model with spatial and temporal contexts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 194–200.

Liu, X.; Liu, Y.; Aberer, K.; and Miao, C. 2013. Personalized Point-of-Interest Recommendation by Mining Users ' Preference Transition. In *Proceedings of the ACM International Conference on Information & Knowledge Management*, 733–738.

Liu, X.; Liu, Y.; and Li, X. 2016. Exploring the Context of Locations for Personalized Location Recommendations. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1188–1194.

Luo, H.; Zhou, J.; Bao, Z.; and Li, S. 2020. Spatial Object Recommendation with Hints : When Spatial Granularity Matters. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 781–790.

MacDonald, C. J.; Lepage, K. Q.; Eden, U. T.; and Eichenbaum, H. 2011. Hippocampal ?time cells? bridge the gap in memory for discontiguous events. *Neuron*, 71(4): 737–749.

Mai, G.; Krzysztof, J.; Bo, Y.; Rui, Z.; Ling, C.; and Lao, N. 2020. Multi-Scale Representation Learning for Spatial Feature Distributions using Grid Cells. In *Proceedings of the International Conference on Learning Representations*.

Manotumruksa, J.; Macdonald, C.; and Ounis, I. 2018. A Contextual Attention Recurrent Architecture for Context-Aware Venue Recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 555–564.

Merri, B. V.; and Fellow, C. S. 2014. Learning Phrase Representations using RNN Encoder – Decoder for Statistical

Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1724–1734.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the International Conference on Neural Information Processing Systems*.

O'keefe, J.; and Nadel, L. 1978. *The hippocampus as a cognitive map*. Oxford: Clarendon Press.

Sargolini, F.; Fyhn, M.; Hafting, T.; McNaughton, B. L.; Witter, M. P.; Moser, M.-B.; and Moser, E. I. 2006. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774): 758–762.

Stachenfeld, K. L.; Botvinick, M. M.; and Gershman, S. J. 2018. The hippocampus as a predictive map. *Nature Neuroscience*, 20(11): 1643–1653.

Sun, K.; Qian, T.; Chen, T.; Liang, Y.; Nguyen, Q. V. H.; and Yin, H. 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*, 214–221.

Tong, Y.; Chen, Y.; Zhou, Z.; Chen, L.; Wang, J.; Yang, Q.; Ye, J.; and Lv, W. 2017. The simpler the better: a unified approach to predicting original taxi demands based on large-scale online platforms. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 1653–1662.

Wang, H.; Shen, H.; Ouyang, W.; and Cheng, X. 2018. Exploiting POI-specific geographical influence for point-of-interest recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 3877–3883.

Whittington, J. C.; Muller, T. H.; Barry, C.; Mark, S.; and Behrens, T. E. 2018. Generalisation of structural knowledge in the hippocampal-entorhinal system. In *Proceedings of the International Conference on Neural Information Processing Systems*.

Whittington, J. C.; Muller, T. H.; Mark, S.; Chen, G.; Barry, C.; Burgess, N.; and Behrens, T. E. 2020. The Tolman-Eichenbaum Machine: Unifying Space and Relational Memory through Generalization in the Hippocampal Formation. *Cell*, 183(5): 1249–1263.e23.

Xingjian, S.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, 802–810.

Yang, C.; Bai, L.; Zhang, C.; Yuan, Q.; and Han, J. 2017. Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1245–1254.

Ye, M.; Yin, P.; Lee, W.-c.; and Lee, D.-l. 2011. Exploiting Geographical Influence for Collaborative Point-of-Interest Recommendation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 325–334.

Yuan, M.; Tian, B.; Shim, V. A.; Tang, H.; and Li, H. 2015. An entorhinal-hippocampal model for simultaneous cognitive map building. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 582–596.

Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *AAAI*.

Zhang, J.; Zheng, Y.; Qi, D.; Li, R.; and Yi, X. 2016. DNN-based prediction model for spatio-temporal data. In *ACM SIGSPATIAL*.

Zhang, J.-d. 2014. LORE : Exploiting Sequential Influence for Location Recommendations. In *Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 103–112.

Zhao, P.; Zhu, H.; Liu, Y.; Li, Z.; Xu, J.; and Sheng, V. S. 2020. Where to Go Next: A Spatio-Temporal Gated Network for Next POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 4347(c): 1–13.

Zhao, P.; Zhu, H.; Liu, Y.; Xu, J.; Li, Z.; Zhuang, F.; Sheng, V. S.; and Zhou, X. 2019. Where to Go Next : A Spatio-Temporal Gated Network for Next POI Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Zhao, S.; Zhao, T.; King, I.; and Michael R. Lyu. 2017. Geo-Temporal Sequential Embedding Rank for Point-of-interest Recommendation. In *Proceedings of the International World Wide Web Conference*, 153–162.

Zhao, S.; Zhao, T.; Yang, H.; Lyu, M. R.; and King, I. 2016. STELLAR : Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 315–321.

Zhu, Y.; Li, H.; Liao, Y.; Wang, B.; Guan, Z.; Liu, H.; and Cai, D. 2017. What to Do Next: Modeling User Behaviors by Time-LSTM. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 3602–3608.