A Geographical-Temporal Awareness Hierarchical Attention Network for Next Point-of-Interest Recommendation

Tongcun Liu
[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
[2] EBUPT Information Technology CO., LTD
Beijing, China
tongcun.liu@gmail.com

Jianxin Liao†
[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
[2] EBUPT Information Technology CO., LTD
Beijing, China
liaojx@bupt.edu.cn

Zhigen Wu
Aplustopia Science Research Institute
Calgary, Canada
z.wu@aplustopia.com

Yulong Wang
[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
[2] EBUPT Information Technology CO., LTD
Beijing, China
cpwang@bupt.edu.cn

Jingyu Wang†
[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
[2] EBUPT Information Technology CO., LTD
Beijing, China
wangjingyu@bupt.edu.cn

ABSTRACT

Obtaining insight into user mobility for next point-of-interest (POI) recommendations is a vital yet challenging task in location-based social networking. Information is needed not only to estimate user preferences but to leverage sequence relationships from user check-ins. Existing approaches to understanding user mobility gloss over the check-in sequence, making it difficult to capture the subtle POI–POI connections and distinguish relevant check-ins from the irrelevant. We created a geographically-temporally awareness hierarchical attention network (GT-HAN) to resolve those issues. GT-HAN contains an extended attention network that uses a theory of geographical influence to simultaneously uncover the overall sequence dependence and the subtle POI–POI relationships. We show that the mining of subtle POI–POI relationships significantly improves the quality of next POI recommendations. A context-specific co-attention network was designed to learn changing user preferences by adaptively selecting relevant check-in activities from check-in histories, which enabled GT-HAN to distinguish degrees of user preference for different check-ins. Tests using two large-scale datasets (obtained from Foursquare and Gowalla) demonstrated the superiority of GT-HAN over existing approaches and achieved excellent results.

CCS CONCEPTS

• Information systems ~ Location based services; Recommender systems; Personalization

KEYWORDS

Location-based social networks, Attention mechanism, Next POI recommendation, Geographical-temporal awareness.

ACM Reference format:

Tongcun Liu, Jianxin Liao, Zhigen Wu, Yulong Wang and Jingyu Wang. 2019. A Geographical-Temporal Awareness Hierarchical Attention Network for Next Point-of-Interest Recommendation. In ICMR '19: 2019 International Conference on Multimedia Retrieval, June 10–13, 2019, Ottawa, ON, Canada, 9 pages. https://doi.org/10.1145/3323873.3325024

†Corresponding authors

ICMR '19, June 10–13, 2019, Ottawa, ON, Canada

ACM ISBN 978-1-4503-6765-3/19/06…$15.00
DOI: https://doi.org/10.1145/3323873.3325024

1 INTRODUCTION

To gain greater insight into the user mobility behavior, business and academia could determine which POI recommendations are valuable to users by analyzing the large-scale interaction data in LBSNs. Effective POI recommendation systems in LBSNs can make it easy for users to find places that interest them through their check-in history and have attracted a significant amount of research interest in the development of recommendation

techniques [16,18,19,26]. Creating the next POI recommendation is a much harder task than creating a general POI recommendation as it needs to accurately predict the user's very next move among tens of thousands of POIs [10]. The next POI recommendation must not only estimate the user preference, which is found by a general location prediction but must also consider the sequence of the user's check-ins since the occurrence of human activity is sequence-dependent [28].

Considerable work has been done recently on determining the next POI recommendation using historical user data. Previous works [5,8,9] on next POI recommendation has mainly used Markov chains (MC) and matrix factorization (MF). Examples are the factorizing personalized Markov chain model (FPMC) [5] and the personalized ranking metric embedding model (PRME)) [8]. These models include the assumption that the next user action is strongly determined by the latest check-in. This assumption is wrong because user behavior is continuous over a short time period. Recent literatures [13,16,20] show that the next POI visited by a user is significantly influenced by recently visited POIs (i.e., a set of previously-visited POIs). Several techniques have been suggested to model this sequence dependence. Existing methods are generally governed by one of two paradigms. The first leverages the word2vec framework, in applications such as CBOW or Skip-Gram, to capture the influence of recently visited POIs. However, the sequential nature of the POIs was ignored in these methods, and only the most recent three or five check-ins were modeled. The methods governed by the second paradigm, such as long short-term memory (LSTM) units [24] and gated recurrent units (GRU) [6], were used to capture the long-term sequential information. The user–POI check-in sequences are fed into the recurrent models, and the value of the last hidden state or the average value of the set of hidden states is used to represent a user's dynamic preference.

The recurrent models improve the accuracy of the next POI recommendation over other approaches, but still greater accuracy is needed. There are a number of improvements that can be made. First, only the overall sequence dependence was modeled, so the subtle POI–POI relationships within the check-in sequence could not be explicitly captured. It is crucial to understand these subtle relationships between individual POI pairs instead of simply glossing over them [30]. Thus, it is necessary to model POI–POI relationships within a check-in sequence, and we expect there to be more effort directed to modeling a unified overall sequence dependence and POI–POI connections. Second, although geographical information (e.g., the distance between POIs), which is a unique property of the next POI recommendation, has been used and incorporated into models such as the RNN model [13,17], these models have difficulty in capturing the high variation in geographical influence across POIs. Geographical influence does not consist only in physical distance, so more POI-specific geographical details must be considered [26]. Third, approaches that treat the contributions of POIs in the check-in sequence to the user's next choice as having equal influence are unable to distinguish the differing degrees of influence that user preferences may have on the checked-in POIs. Some POIs represent user preferences better than others [19]. Those POIs that are not particularly relevant to a user preference can overwhelm the influence of a few significantly relevant POIs, leading to a misunderstanding of a user's preference. Thus, it is important to distinguish the degree of a user's preference among checked-in POIs to accurately learn and personalize user preferences.

The goal of this study is to develop a geographical-temporal awareness hierarchical attention network (GT-HAN) which will improve the next POI recommendation by better learning the relationships between users and POIs from large-scale check-in data. Specifically, we establish a geographical-temporal attention network using attention mechanisms and LSTM units to learn the high-level semantic representation of a sequence. This enables GT-HAN to simultaneously capture both the overall temporal dependence and subtle POI–POI relationships in a check-in sequence. Drawing on [26], we consider each POI to have two propensities: geo-influence, which directs its visitors to other POIs, and geo-susceptibility, which is the receipt of visitors from other POIs. We use three factors (the geo-influence of POIs, the geo-susceptibility of POIs, and the distance between POIs) to model the geographical co-influence between two POIs. GT-HAN can, in consequence, capture the great variation in geographical influence across POIs, rather than being simply limited to the physical distance between them. We further developed a context-specific co-attention network to learn dynamic user preferences to allow GT-HAN to distinguish degrees of user preference in their POI check-in history. Finally, we computed the conditional probability distribution over POIs using the softmax function. GT-HAN, as we developed it, is a simple, effective, and robust model that produces next POI recommendations with few constraints.

2 RELATED WORK

Two aspects of related studies are reviewed: the next POI recommendation and the application of the attention mechanism in recommender systems.

2.1 Next POI Recommendation

Earlier studies mainly used the assumption of first-order Markov transfer learning; the Markov chain (MC) technique has been extended to allow prediction of the next behavior of a user based on their latest check-in [3,4,23,28]. This approach required the creation of a transfer matrix that indicates the probability of a particular behavior based on past behaviors, which was then decomposed by matrix factorization [23] or tensor factorization [9]. For example, Cheng et al. [5] exploited the personalized Markov chain using a location constraint; and Chen et al. [4] developed a general Markov model that took account of both individual and collective movement patterns. Some studies, such as [9], investigated the transfer patterns of POI categories to improve the accuracy of next POI recommendations. These works exploit the influence of a sequence only by using the latest check-in because of computational complexity. They fail to fully consider the influence of any long-term sequence. However, a user's next

choice is highly influenced by the entire set of POIs already visited.

Recently, by considering historic check-ins as a set, word2vec has been extensively used and developed to investigate and learn changing user preferences. For example, Zhao et al. [31] developed a geo-temporal sequential embedding rank model (GT-SEER); Feng et al. [7] created a POI2Vec model that modeled user preferences and transferred POIs determined by sequences; Chang et al. [1] created a content-aware hierarchical POI embedding model (CAPE) that captures the geographical influence of POIs from user check-in sequences and the characteristics of POIs from the semantic content of POI data. However, none of these approaches is capable of capturing sequence dependence because the models isolate POIs from their data sequences.

Recurrent modelling has recently advanced greatly in its ability to deal with sequential data. Some studies have used an RNN model to make the next POI recommendations and performed better than the models based on word2vec. For example, Liu et al. [17] developed the RNN model into a spatial-temporal recurrent neural network (ST-RNN); Kong et al. [13] incorporated spatial-temporal influence into an LSTM through developing a hierarchical spatial-temporal long short-term memory (HST-LSTM) model. However, although existing methods have been successful in modeling the global sequence dependence, they cannot model subtle POI–POI relations. Moreover, current models usually implicitly encode a user's previous records into a latent factor or a hidden state without recognizing the possible influence any check-in may have in predicting the next behavior.

2.2 Attention Mechanism in Recommendations

The great success of the attention mechanism in context learning in computer vision and natural language learning has led to its successful application in next item recommendation [27,29,32]. Chen et al. [2] were the first to incorporate an attention mechanism into collaborative filtering when they modeled implicit feedback at both item level and component level. Wang et al. [27] extended the Skip-Gram model by designing an attention-based transaction embedding model to weight each observed item in a transaction without assuming any order. Li et al. [14] created a temporal and multilevel context attention mechanism to adaptively select relevant check-in activities and contextual factors to predict next POI preferences.

Self-attention has recently become the subject of many studies. Interrelations within the data are used [11,12], and significant performance improvements have been achieved. For example, Zhang et al. [30] utilize a self-attention mechanism to infer the item–item relationship from user interaction history. Zhou et al. [32] projected all types of behaviors into multiple latent semantic spaces and used self-attention within an attention-based user recommendation model. However, these methods cannot be used in the next POI recommendation model because spatial-temporal context information, which is critical for improving recommendation performance, is a unique property. Ma et al. [19] first used self-attention to encode changing user preferences and incorporated neighbor-aware influence into the decoder.

3 PROBLEM DEFINITION

Following conventional symbol notation, we use uppercase bold letters to denote matrices (e.g., $\boldsymbol{U}$), lowercase bold letters to denote vector, and regular typeface letters to represent scalars. We use calligraphic letters to represent sets (e.g., the user set $\mathcal{U}$), and $|\cdot|$ to denote the cardinality of the set. The major notation used in this paper is shown in Table 1.

Table 1: Key mathematical notation.

Variable	Interpretation
$\mathcal{U}, \mathcal{V}, \mathcal{T}$	Sets of users, POIs, and temporal states
$\mathcal{H}$	Set of trajectory sequence for all users
$\mathcal{H}_i$	A set of check-ins for user u_i
$\boldsymbol{P}, \boldsymbol{I}, \boldsymbol{S}$	Matrices of POI preference, POI geo-influence, and POI geo-susceptibility
$\boldsymbol{D}$	Geographical distance matrix between adjacent POI check-ins
$\boldsymbol{U}$	User general preference vector
$\boldsymbol{W}^*$	Weight matrices in the model

We denote $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ to be a set of LBSN users, $\mathcal{V} = \{v_1, v_2, \dots, v_{|\mathcal{V}|}\}$ to be a set of POIs, where each POI is geocoded by {longitude, latitude}, $\mathcal{T} = \{t_1, t_2, \dots, t_{|\mathcal{T}|}\}$ to be a set of temporal states, $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_{|\mathcal{U}|}\}$ to be a trajectory sequence set for all users, and $\mathcal{H}_i = \{v_{i,1}, v_{i,2}, \dots\}$ to be a set of check-ins ordered chronologically by the timestamps of user u_i.

The next POI recommendation is defined thus: given a user u_i and their recent L historical records $\mathcal{H}_i^L = \{v_{i,1}, v_{i,2}, \dots, v_{i,L}\}$, the next POI recommendation is the prediction of where the user will visit at next (future) time step. The next POI recommendation is obtained from the construction and training of a conditional probabilistic distribution $P(v_j | u_i, \mathcal{H}_i^L, t_k)$ over all POIs.

4 PROPOSED MODEL

A pictorial representation of GT-HAN is shown in Figure 1. The arrows represent data flow, $\boldsymbol{E_S}, \boldsymbol{E_I}, and\ \boldsymbol{E_P}$ are query, key, and value; u_i is the target user; v_j is the target POI; t_k is the current temporal state; $\boldsymbol{U_i}$ is the general preference vector of u_i; $\boldsymbol{P}_j$ is the target POI preference vector of v_j; $\boldsymbol{C_k}$ is latent semantic vector of current temporal state t_k; and $\mathcal{H}_i^L$ is the first L check-ins of user u_i.

GT-HAN consists of an embedding layer, a geographical-temporal attention network layer, and a co-attention network

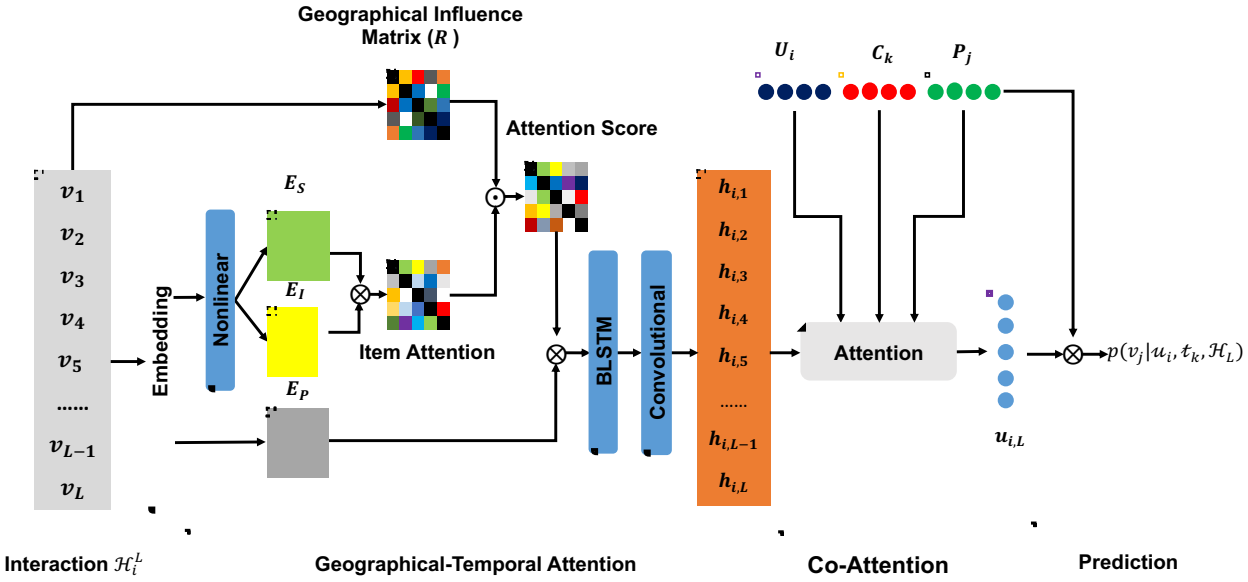

Figure 1:The Proposed GT-HAN model

layer. The geographical-temporal attention network is included to model the geographical relations between POIs and the temporal dependency of a check-in sequence. The co-attention network is used to capture the changing user preferences (dynamic user preferences), given the general preference of the target user and the current time state. Finally, according to [27,32], we use negative sampling and mini-batch stochastic gradient descent (BSGD) algorithms to train GT-HAN. More details of each component of GT-HAN are given in the following sections.

4.1 Embedding Layer

Most previous studies [8,16,17] project users and POIs into the same denser semantic space and make recommendations based on the relations between user and POI in the space. These methods do not reveal the geographical aspects of POIs. Following the work of [26], we project POIs into a different semantic space. Specifically, for each POI v_j, we create a POI preference vector $\boldsymbol{P}_j \in R^{1\times d}$, a POI geo-influence vector $\boldsymbol{I}_j \in R^{1\times d}$, and a POI geo-susceptibility vector $\boldsymbol{S}_j \in R^{1\times d}$, where d is the latent dimensionality. We use the geo-influence vector to capture the tendency of a POI to direct its visitors to other POIs and the geo-susceptibility vector to reflect the tendency of a POI to receive visitors directed from other POIs. All POIs are then transformed into the matrices $\boldsymbol{P} \in R^{|\mathcal{V}|\times d}$, $\boldsymbol{I} \in R^{|\mathcal{V}|\times d}$, and $\boldsymbol{S} \in R^{|\mathcal{V}|\times d}$. We similarly create a general preference matrix $\boldsymbol{U} \in R^{|\mathcal{U}|\times d}$ for all users; this matrix has a constant value over time. We also create a latent semantic matrix $\boldsymbol{C}$ for all temporal states. Notice that $\boldsymbol{P}$, $\boldsymbol{I}$, $\boldsymbol{S}$, $\boldsymbol{U}$, and $\boldsymbol{C}$ can be obtained during model training.

For a given check-in sequence $\mathcal{H}_i$ of a user u_i, we first transform $\mathcal{H}_i$ into a fixed-length sequence with maximum length L. If the sequence length is greater than L, we consider only the most recent L check-in actions; otherwise, we pad $\mathcal{H}_i$ with zeros to the left. We then retrieve the input preference matrix $\boldsymbol{E_P} \in R^{L\times d}$, the geo-influence matrix $\boldsymbol{E_I} \in R^{L\times d}$, and the geo-susceptibility matrix $\boldsymbol{E_S} \in R^{L\times d}$ from $\boldsymbol{P}$, $\boldsymbol{I}$, and $\boldsymbol{S}$ through indexing technology. In these matrices, a zero vector is used for embedding the padded item. For the target user u_i, we retrieve the general preference vector $\boldsymbol{U_i} \in R^{1\times d}$ from $\boldsymbol{U}$; for current time state t_k, we retrieve the latent semantic $\boldsymbol{C_k} \in R^{1\times d}$ from $\boldsymbol{C}$.

4.2 Geographical-Temporal Attention Network

4.2.1 Modeling Geographical Relations.

We developed a geographical attention network to capture the relationships between POIs within a sequence to model geographical relations. The input of the attention network consists of a query, a key, and a value. The output of the attention network is the weighted sum over the value, where the weight matrix is determined by the queries and their corresponding keys. In our examples, the geo-susceptibility matrix $\boldsymbol{E_S}$ is the query, the geo-influence matrix $\boldsymbol{E_I}$ is the key, and the preference matrix $\boldsymbol{E_P}$ is the value. In line with previous works [12,21,29], we first project the query and the key to the same semantic space through a nonlinear transformation with shared parameters and then calculate the weight matrix as:

$$AT(\boldsymbol{E_S}, \boldsymbol{E_I}) = softmax(\frac{\boldsymbol{QK}^T}{\sqrt{d}}) \quad (1)$$

$$\boldsymbol{Q} = \psi(\boldsymbol{E_S}\boldsymbol{W^Q} + B_q) \quad (2)$$

$$\boldsymbol{K} = \psi(\boldsymbol{E_I}\boldsymbol{W^K} + B_k) \quad (3)$$

where $\boldsymbol{W^Q} \in R^{d\times d}$, $\boldsymbol{W^K} \in R^{d\times d}$, $B_q \in R^{L\times d}$, and $B_k \in R^{L\times d}$ are model parameters. The function $\psi(\cdot)$ is the activation function, ReLU was used to increase the nonlinear capability, and $\sqrt{d}$ is used to scale the dot product attention. The output of Eq. (1) is an $L \times L$ weight matrix (or attention map) that represents the geographical influence relations among L POIs.

The dot product indicates how two POIs are related with respect to geographical influences. However, it does not explicitly take account of the geographical distance between two POIs. Tobler's first law of geography is that everything is related to everything else, but near things are more related than distant things. There are several functions that can be used to represent the influence of geographical distance, such as the power law function (PLF), the exponential function (EF), the hyperbolic function (HF), and the Gaussian radial basis function kernel (RBF kernel). We use the RBF kernel to weight the influence of checked-in POIs in favor of nearby POIs because so doing gives better performance than the other functions. The RBF kernel function is defined as:

$$\boldsymbol{R} = exp(-\gamma \|\boldsymbol{D}\|^2) \tag{4}$$

where $\boldsymbol{D} \in R^{L \times L}$ is a geographical distance matrix between adjacent checked-in POIs. Gamma is a hyper-parameter to control the geographical correlation between two given POIs; a larger value of γ will lead to a smaller value of $\boldsymbol{R}$. The value of $\boldsymbol{R}$ is limited to 0 or 1. By introducing the geographical influence between adjacent checked-in POIs, Eq. (1) is rewritten as:

$$\widetilde{AT}(\boldsymbol{E_S}, \boldsymbol{E_I}) = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_1}}) \odot \boldsymbol{R} \tag{5}$$

where $\odot$ is the element-wise product, $\boldsymbol{Q}$ was obtained from Eq. (2), and $\boldsymbol{K}$ from Eq. (3). The output of geographical attention is an $L \times d$ matrix and defined as:

$$\widetilde{GAT}(\boldsymbol{E_S}, \boldsymbol{E_I}, \boldsymbol{E_P}) = \widetilde{AT}(\boldsymbol{E_S}, \boldsymbol{E_I})\boldsymbol{E_P} \tag{6}$$

To allow the model to concurrently attend to information from different representation subspaces at different locations, geographical attention uses l-scale dot product attention. The outputs of all l -scale dot product attention models are concatenated, and then a linear layer is used to project the concatenated feature to a fixed dimensional feature. We formulate the calculation process as:

$$\boldsymbol{F} = concat(\widetilde{GAT}_1, \widetilde{GAT}_2, \dots, \widetilde{GAT}_l) \tag{7}$$

where $\widetilde{GAT}_i$ is an $L \times \frac{d}{l}$ matrix.

Following the work of [15], we also add a residual connection to $\boldsymbol{F}$ to avoid transmission loss and then apply a normalization layer. Thus, the original mapping $\boldsymbol{F}$ is:

$$\widetilde{\boldsymbol{F}} = LayerNorm\ (\boldsymbol{F} + \boldsymbol{E_P}) \tag{8}$$

where $\widetilde{\boldsymbol{F}} \in R^{L \times d}$. The normalization layer is used to normalize the inputs across features (i.e., zero mean and unit variance); this normalization stabilizes and accelerates neural network training.

4.2.2 Modeling Sequence Dependence.

In comparison to traditional RNN networks, such as LSTM, our attention network considers each entry to be independent of others in the sequence and ignores the sequence information in the sequential input. In previous work, such as the Transform [25] model, positional embedding (PE) was commonly used to encode geometric position information about input sequences. Our test shows that such a method produces weak results from input sequences that contain large time variations. An RNN can capture temporal relatedness and give an outstanding performance on natural language processing (NLP) and sequence recommendations. We consider the output of the geographical attention network $\widetilde{\boldsymbol{F}}$ to be sequential input and use Bi-LSTM to capture the temporal relationship between adjacent POIs within a sequence.

$$\begin{aligned} \overrightarrow{\boldsymbol{h_t}} &= \text{LSTM}\big(\boldsymbol{W_t}, \overrightarrow{\boldsymbol{F_{t-1}}}\big) \\ \overleftarrow{\boldsymbol{h_t}} &= \text{LSTM}\big(\boldsymbol{W_t}, \overleftarrow{\boldsymbol{F_{t+1}}}\big) \end{aligned} \tag{9}$$

We concatenate each $\overrightarrow{\boldsymbol{h_t}} \in R^{1 \times d}$ and $\overleftarrow{\boldsymbol{h_t}} \in R^{1 \times d}$ and use a linear mapping to obtain the hidden state $\boldsymbol{h_t} \in R^{1 \times d}$. All $\boldsymbol{h_t}$ form the set $\boldsymbol{H} = \{\boldsymbol{h_1}, \boldsymbol{h_2}, \dots, \boldsymbol{h_L}\}$, which has dimension $L \times d$.

4.3 Co-Attention Network

Previous models have often used the averaged hidden state of the recurrent model to represent changing user preferences [20,27]. They create recommendations by forming the dot product between the dynamic user preference vector and the item preference vector. This method fails to recognize the significance of individual user behavior in determining the next POI visited, and thus it reduces the value of the recommendation. Whether a user will check-in at a recommended POI is determined also by context information, such as the user's general preferences, the time state, and distance from the POI. We created a context-specific co-attention network to capture dynamic user preferences. Specifically, we project each context into a d-dimensional semantic space and compute the attention weight, using Eq. (1), and the weighted attention value of $\boldsymbol{H}$. We then use a late fusion strategy to concatenate the weighted attention values and a nonlinear connection layer to learn the dynamic user preference.

Formally, given the general preference vector $\boldsymbol{U_i}$ of target user u_i, the preference vector $\boldsymbol{P_j}$ of target POI v_j, and the latent semantic vector $\boldsymbol{C_k}$ of time context t_k, the dynamic preference of user u_i, is calculated as:

$$\boldsymbol{u_{i,L}} = \emptyset(\boldsymbol{\alpha}\boldsymbol{W}^{\alpha} + b)\, \boldsymbol{W}^{U} \tag{10}$$

$$\boldsymbol{\alpha} = concat\ \big([AT(\boldsymbol{U_i}, \boldsymbol{H})\boldsymbol{H}, AT\big(\boldsymbol{P_j}, \boldsymbol{H}\big)\boldsymbol{H}, AT(\boldsymbol{C_k}, \boldsymbol{H})\boldsymbol{H}]\big) \tag{11}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{\boldsymbol{1 \times 3d}}$ is a co-attention score after concatenation, $\boldsymbol{W^{\alpha}} \in \boldsymbol{R^{3d \times d}}$ and $\boldsymbol{W^{U}} \in \boldsymbol{R^{d \times d}}$ are model parameters, and $\emptyset(\cdot)$ is an activity function to increase nonlinearity.

4.4 Learning and Model Optimization

After obtaining the dynamic user preference, the conditional probability distribution $p\big(v_j | u_i, c_k, \mathcal{H}_i^L\big)$ is defined in terms of the softmax function, which is commonly used in neural networks:

$$p\big(v_j | u_i, t_k, \mathcal{H}_i^L\big) = \frac{exp\ (\boldsymbol{u_{i,L}}\boldsymbol{P_j}^T)}{\sum_{\iota=1}^{|\mathcal{V}|} exp\ (\boldsymbol{u_{i,L}}\boldsymbol{P_\iota}^T)} \tag{12}$$

Thus, given a training dataset $\mathcal{X} = \{< \mathcal{H}_i^L, u_i, t_k, v_j >\}$ the joint probability distribution can be obtained:

$$P_\theta(\mathcal{D}) = \prod_{x\in\mathcal{X}} p\left(v_j|u_i, t_k, \mathcal{H}_i^L\right) = \prod_{x\in\mathcal{X}} \frac{exp\left(\boldsymbol{u}_{i,L}\boldsymbol{P}_j^T\right)}{\sum_{l=1}^{|\mathcal{V}|} exp\left(\boldsymbol{u}_{i,L}\boldsymbol{P}_l^T\right)} \quad (13)$$

The model parameters $\boldsymbol{\Theta} = \{\boldsymbol{P}, \boldsymbol{I}, \boldsymbol{S}, \boldsymbol{U}, \boldsymbol{C}, \boldsymbol{W^Q}, \boldsymbol{W^K}, \boldsymbol{W^U}, \boldsymbol{W^\alpha}\}$ can be learned by maximizing the conditional log-likelihood:

$$\mathcal{L} = \sum_{x\in\mathcal{X}} p\left(v_j|u_i, t_k, \mathcal{H}_i\right) - \lambda\|\boldsymbol{\Theta}\| \quad (14)$$

It is impractical to directly optimize the above objective function because the computing cost of the full softmax is proportional to the size of $|\mathcal{V}|$, which is often extremely large. Following the work of [27,32], we use the negative sampling technique for efficient optimization, and approximate the original objective $\mathcal{L}$ with the objective function:

$$\mathcal{L} = \sum_{x\in\mathcal{X}} \left(log\,\sigma\left(\boldsymbol{u}_{i,L}\boldsymbol{P}_j^T\right) + k\cdot\mathbb{E}_{j'\sim P_I}\left[log\sigma\left(-\boldsymbol{u}_{i,L}\boldsymbol{P}_{j'}{}^T\right)\right]\right) - \lambda\|\theta\| \quad (15)$$

where $\sigma = 1/(1+e^{-x})$ is an activity function, k is the number of sampled negative POIs drawn according to the noise distribution P_I, and P_I is a distribution of POI check-in frequency.

It is computationally expensive to directly minimize Eq. (15) due to the summing of the huge dataset D. Thus, a mini-batch stochastic gradient descent (BSGD) algorithm was used to train the model. In our experiment, each batch contained 64 training iterations.

5 EXPERIMENTAL EVALUATION

5.1 Data Description

We used two publicly-available datasets from Foursquare [18] and Gowalla [18] to demonstrate the effectiveness of the proposed model. These two datasets contain abundant check-in data and have been widely used in previous studies.

The Foursquare dataset comprises check-ins from 2012-04 to 2013-09 within the contiguous United States. The Gowalla dataset was generated worldwide from 2009-02 to 2010-10. Each check-in record in the two datasets includes a timestamp, a user ID, a POI ID with the latitude and longitude of the POI. We eliminated users with <10 POI check-ins and POIs with <10 visitors from the Foursquare dataset. The resultant dataset contained 24 941 users, 28 593 POIs and 1 196 248 check-ins. We removed users with <20 POI check-ins and POIs with <20 visitors from the Gowalla dataset. The resultant dataset contained 18 737 users, 32 510 POIs, and 1 278 274 check-ins. We use the first L check-ins of u_i to predict the $(L+1)$th check-in in the training dataset, where L= {1, 2, …, L–2}, and we use the first $(L-1)$th check-ins to predict the Lth check-in in the test dataset, similar to the method described in [32].

5.2 Experiment Setting

We compared GT-HAN with BPR [22], Bi-LSTM [24], Bi-LSTM+Attention, Geo-Teaser [31], and ATRank [32], which are the state-of-the-art models for the next POI recommendation. To identify the individual benefits of including the new technologies incorporated into GT-HAN (co-attention, technique T1, Bi-LSTM, technique T2, and residual connections, technique T3), we designed and built 4 variants of GT-HAN using ablation. For the first variant, GT-HAN-V1, we removed co-attention; the average value of the hidden states in the Bi-LSTM was used to represent dynamic user preferences. For GT-HAN-V2 we removed Bi-LSTM, and use position embedding (PE), which is similar to the transform model [25], to capture temporal dependency. For GT-HAN-V3 we removed Bi-LSTM, and temporal dependency is not considered. That is, the model makes recommendations based only on past user behavior without considering the order. For GT-HAN-V4 we removed the residual connections.

The aim of this experiment is to find the top-N POIs that may interest a user. We use the common evaluation metric, Accuracy@N, and AUC metrices to evaluate the quality of the model. We first use Eq. (12) to compute the probability of a user visiting members of the set containing each target POI v_j and other candidate POIs. We then create a ranked list by ordering these POIs according to their probabilities. We create a top-N recommendation list by selecting the N highest-ranking POIs from the list. If $\text{rank}\left(v_j\right) < \text{N}$, then we have a hit; otherwise, we have a miss. Finally, the overall Accuracy@N is defined by averaging over all test cases:

$$Accuracy@N = \frac{\#hit@N}{|S_{test}|} \quad (16)$$

where $\#hit@N$ is the number of hits in the test set, and $|\text{S}_{test}|$ is the number of all test cases.

We also calculate the area under the curve (AUC) for the average user:

$$\text{AUC} = \frac{1}{|U^{Test}|}\sum_{i\in\mathcal{V}_u^+}\sum_{j\in\mathcal{V}_u^-} I(p(v_i) - p\left(v_j\right)) \quad (17)$$

where $p(v_i)$ is the predicted probability that a user will visit POI i in the test set, and $I(\cdot)$ is the indicator function.

We trained our model on a computer server equipped with four high-performance NVIDIA GPUs, each having 12 GB video memory. Our model was implemented in python with the TensorFlow deep learning library. There are several hyperparameters in our model, and we performed 5-fold cross-validation to find the optimal parameters. The geographical correlation level γ is determined by a grid search and was set to 10 in our experiment. For the gradient descent parameters, the learning rate starts at 1.0, the decay rate is set to 0.1, and the regularization λ was set to 0.000 05. The number of negative samples k was set to 1 for simplicity. Sensitivity analysis for two important hyperparameters, the dimension of latent factors d and the sequence length L, is given in section 5.3.2.

For the comparisons with other models, we used the source code provided by the authors. The optimal hyperparameters for those models were obtained by a grid search algorithm using our datasets.

5.3 Results and Discussion

5.3.1 Comparison of Various Approaches.

The results given by GT-HAN were compared with the results of the other models, with validated parameters, using the Foursquare

and Gowalla datasets. Our goal was to make top-N POI recommendations within a reasonable time that would be acceptable to a user, so we used the Accuracy@N metric to evaluate the performance of our model. The dimension of latent factors d was set to 700, and the sequence length L was set to 20. The detailed parameter tuning process is described in section 5.3.2. Only the results for N = {5, 10, 15, 20, 25, 30} are shown because a greater value of N is usually ignored by users.

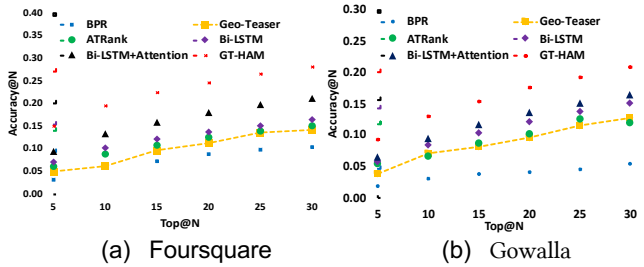

Figure 2: Results of GT-HAN with comparative approaches on Foursquare and Gowalla dataset.

Table 4: AUC Performance of GT-HAN compared to other approaches

Methods	Foursquare	Gowalla
BPR	0.7000	0.5050
Geo-Teaser	0.8124	0.7381
ATRank	0.9513	0.8962
Bi-LSTM	0.9367	0.9542
Bi-LSTM+Attention	0.9536	0.9649
GT-HAN	**0.9661**	**0.9721**

Figure 2 shows the Accuracy@N results obtained from the Foursquare and Gowalla datasets for different values of N. The results show first that our GT-HAN model significantly outperforms all the others for both datasets. For example, compared with Bi-LSTM+Attention, the second-best model, Accuracy@10 shows >46.3% and >37.8% improvement for the Foursquare and Gowalla datasets. Second, Bi-LSTM+Attention performed better than Bi-LSTM and ATRank. The improvement shown by Accuracy@10 is ~30.4% and ~50.8% for the Foursquare dataset and ~12.2% and ~43.2% for the Gowalla dataset. Third, the comparison between the two datasets shows that the results for Gowalla are worse than for Foursquare. Results given by GT-HAN for the Foursquare dataset are ~50% better than for the Gowalla dataset. We also evaluated AUC, as shown in Table 2. The GT-HAN model performed significantly better than any of the other methods for both the Foursquare and the Gowalla datasets.

The better Accuracy@N and AUC values for GT-HAN are due to the following. First, GT-HAN uses an attention mechanism to capture the geographical relationships among POIs and uses Bi-LSTM to capture overall temporal sequence dependence. Bi-LSTM improves performance over the position embedding (PE) technique used in the Transformer model. Second, dynamic user preferences were learned by the context-specific co-attention network, which can identify highly similar user behaviors from great numbers of check-ins.

We conducted tests to compare GT-HAN with the four variants described in Section 5.2. The results for Accuracy@5, Accuracy@10, and Accuracy@20 using the Foursquare and Gowalla datasets are shown in Figure 3. As expected, GT-HAN consistently outperforms the four variant versions for both datasets, demonstrating the benefits produced by each additional technique. The following important observations can be made from the results. First, without the co-attention network, GT-HAN-V1 shows the worst performance for both datasets. This is because, when using the average value of hidden states, the advantage of the co-attention network learning dynamic user preferences by distinguishing relevant check-ins from irrelevant ones is lost. More importantly, the co-attention network considers a user's general preferences, target POIs, and the temporal context. Second, Bi-LSTM is more capable of capturing the temporal dependence than PE, a technique that is commonly used in a Transformer network. PE fails to work in this scenario because the user's check-in actions have strict order relations. Third, GT-HAN-V4 performs worst because it has no residual connections. Presumably, this is because information in lower

Table 2: Impact of latent dimension d in GT-HAN on Foursquare and Gowalla datasets.

d	Foursquare			Gowalla		
	Accuracy@5	Accuracy@10	Accuracy@20	Accuracy@5	Accuracy@10	Accuracy@20
100	0.1298	0.1730	0.2228	0.0690	0.0995	0.1393
300	0.1383	0.1822	0.2324	0.0810	0.1148	0.1587
500	0.1423	0.1871	0.2377	0.0859	0.1213	0.1638
700	**0.1516**	**0.1958**	**0.2470**	**0.0930**	**0.1305**	**0.1760**
900	0.1446	0.1888	0.2382	0.0877	0.1248	0.1689
1100	0.1387	0.1826	0.2366	0.0838	0.1189	0.1621

Table 3: Impact of the sequence length L in GT-HAN on Foursquare and Gowalla datasets.

L	Foursquare			Gowalla		
	Accuracy@5	Accuracy@10	Accuracy@20	Accuracy@5	Accuracy@10	Accuracy@20
5	0.1276	0.1694	0.2200	0.0913	0.1293	0.1764
10	0.1449	0.1897	0.2389	0.0905	0.1259	0.1704
15	0.1330	0.1798	0.2322	0.0952	0.1301	0.1794
20	**0.1516**	**0.1958**	**0.2470**	**0.0930**	**0.1305**	**0.1760**
25	0.1451	0.1905	0.2424	0.0958	0.1334	0.1796
30	0.1403	0.1853	0.2384	0.0910	0.1258	0.1706

layers cannot easily be propagated to the final layer, and lower-layer information is extremely useful in making recommendations. We also found that the three different techniques have different impacts. In order of importance, the techniques are T1>T2>T3.

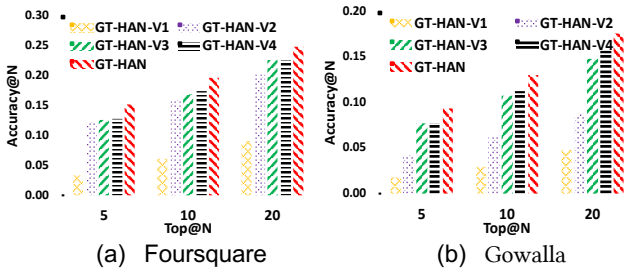

Figure 3: Results for different GT-HAN variants using Foursquare and Gowalla datasets.

5.3.2 Parameter Sensitivity.

We investigated the sensitivity of hyperparameters, specifically the dimension of latent factors d and the sequence length L, in GT-HAN using the Foursquare and Gowalla datasets.

We first set L to 20 and investigated the effect of the dimension on recommendation accuracy. We tested model performance by varying d from 100 to 1100 in increments of 200. We conducted the test 5 times. The average values are reported as the final results for the two datasets. The results for Accuracy@5, Accuracy@10, and Accuracy@20 are shown in Table 3. The results show that the recommendation accuracy of GT-HAN first slightly increases as d increases and begins to sharply decrease as d continues to increase. The parameter d represents model complexity. When d is small, GT-HAN fails to describe user preferences. However, when d exceeds the threshold (d=700, the model is complex enough to handle the data but with less accuracy. At this point, increasing d will undoubtedly improve model performance, but it will also increase the time taken for model training, leading to diminishing returns. Thus, the best results for both datasets are when d=700.

We also investigated the influence of sequence length L on recommendation accuracy. We set d to 700 and tested model performance by varying L from 5 to 30 in increments of 5. We conducted 5 tests and used the average values as the final results for the two datasets. Table 4 shows the recommendation accuracy for Accuracy@5, Accuracy@10, and Accuracy@20. Recommendation accuracy first increases then slightly decreases as L continues to increase. The best results are given for both datasets when L is 20. Early increases are seen because the greater sequence length enables GT-HAN to capture the effects of historical check-ins. Model overfitting occurs as the sequence length continues to increase. Additionally, larger values also result in time complexity. The results also show that next check-in behavior is highly influenced by the set of POIs visited previously.

6 CONCLUSIONS

GT-HAN improved next POI recommendations on both the Accuracy@N and AUC criteria. The major reason for this improvement is that the model exploits POI–POI relationships by using an attention mechanism to include the geographical influence and captures overall sequence dependence with a Bi-LSTM network. As a result, GT-HAN can learn high-level semantic representations from huge check-in sequence datasets. GT-HAN also distinguishes the degree of user preference from check-in history from the context-specific co-attention network. Our tests with Foursquare and Gowalla datasets demonstrated the significant improvement given by GT-HAN over state-of-the-art approaches. Accuracy@10 is >46.3% better for the Foursquare dataset and >37.8% better for the Gowalla dataset compared with Bi-LSTM+Attention model. Although our model outperforms other state-of-the-art approaches, content information that reveals user semantic preferences was not investigated in this study; we will address this in our future work.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61671079, Grant 61771068, and Grant 61471063, and in part by the Beijing Municipal Natural Science Foundation under Grant 4182041.

REFERENCES

[1] Buru Chang, Yonggyu Park, Donghyeon Park, Seongsoon Kim, and Jaewoo Kang. 2018. Content-Aware Hierarchical Point-of-Interest Embedding Model for Successive POI Recommendation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3301–3307. DOI: https://doi.org/10.24963/ijcai.2018/458

[2] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 335–344. DOI: https://doi.org/10.1145/3077136.3080797

[3] Jun Chen, Chaokun Wang, and Jianmin Wang. A Personalized Interest-Forgetting Markov Model for Recommendations. 7.

[4] Meng Chen, Yang Liu, and Xiaohui Yu. 2014. NLPMM: A Next Location Predictor with Markov Modeling. *Advances in Knowledge Discovery and Data Mining* 8444, (2014), 186–197. DOI: https://doi.org/10.1007/978-3-319-06605-9_16

[5] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. 2013. Where You Like to Go Next: Successive Point-of-Interest Recommendation. In *Twenty-Third international joint conference on Artificial Intelligence*, 2605–2611.

[6] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2015. Gated Feedback Recurrent Neural Networks. In *International Conference on International Conference on Machine Learning*, 2067–2075.

[7] Shanshan Feng, Gao Cong, Bo An, and Yeow Meng Chee. 2017. POI2Vec: Geographical Latent Representation for Predicting Future Visitors. In *Process of the thirty-First AAAI Conference on Artificial Intelligence*, 102–108.

[8] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. Personalized Ranking Metric Embedding for Next New POI Recommendation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 2069–2075.

[9] Jing He, Xin Li, and Lejian Liao. 2017. Category-aware Next Point-of-Interest Recommendation via Listwise Bayesian Personalized Ranking. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 1837–1843. DOI: https://doi.org/10.24963/ijcai.2017/255

[10] Jing He, Xin Li, Lejian Liao, Dandan Song, and William K Cheung. 2016. Inferring A Personalized Next Point-of-Interest Recommendation Model with Latent Behavior Patterns. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 137–143.

[11] Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. 2018. CSAN: Contextual Self-Attention Network for User Sequential Recommendation. In *Proceedings of the 26th ACM international conference on Multimedia*, 447–455.DOI: https://doi.org/10.1145/3240508.3240609

[12] Wang-Cheng Kang and Julian McAuley. 2018. Self-Attentive Sequential Recommendation. *arXiv:1808.09781 [cs]* (August 2018). Retrieved September 27, 2018 from http://arxiv.org/abs/1808.09781

[13] Dejiang Kong and Fei Wu. 2018. 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, 2341–2347. DOI: https://doi.org/10.24963/ijcai.2018/324

[14] Ranzhen Li, Yanyan Shen, and Yanmin Zhu. 2018. Next Point-of-Interest Recommendation with Temporal and Multi-level Context Attention. In *2018 IEEE International Conference on Data Mining (ICDM)*, 1110–1115. DOI: https://doi.org/10.1109/ICDM.2018.00144

[15] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. 2019. Beyond RNNs: Positional Self-Attention with Co-Attention for Video Question Answering. In *The 33rd AAAI Conference on Artificial Intelligence*, 8.

[16] Jianxin Liao, Tongcun Liu, Meilian Liu, Jingyu Wang, Yulong Wang, and Haifeng Sun. 2018. Multi-Context Integrated Deep Neural Network Model for Next Location Prediction. *IEEE Access* 6, (2018), 21980–21990. DOI: https://doi.org/10.1109/ACCESS.2018.2827422

[17] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 194–200.

[18] Yiding Liu, Tuan-Anh Nguyen Pham, Gao Cong, and Quan Yuan. 2017. An experimental evaluation of point-of-interest recommendation in location-based social networks. In *Proceedings of the VLDB Endowment*, 1010–1021. DOI: https://doi.org/10.14778/3115404.3115407

[19] Chen Ma, Yingxue Zhang, Qinglong Wang, and Xue Liu. 2018. Point-of-Interest Recommendation: Exploiting Self-Attentive Autoencoders with Neighbor-Aware Influence. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 697–706. Retrieved November 6, 2018 from http://arxiv.org/abs/1809.10770

[20] Jarana Manotumruksa, Craig Macdonald, and Iadh Ounis. 2018. A Contextual Attention Recurrent Architecture for Context-Aware Venue Recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval - SIGIR '18*, 555–564. DOI: https://doi.org/10.1145/3209978.3210042

[21] Yao Qin, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison Cottrell. 2017. A Dual-Stage Attention-Based Recurrent Neural Network for Time Series Prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2627–2633. Retrieved September 27, 2018 from http://arxiv.org/abs/1704.02971

[22] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, 452–461.

[23] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World Wide Web - WWW '10*, 811. DOI: https://doi.org/10.1145/1772690.1772773

[24] Hasim Sak, Andrew Senior, and Francoise Beaufays. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling. 5.

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, 5998–6008.

[26] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3877–3883. DOI: https://doi.org/10.24963/ijcai.2018/539

[27] Shoujin Wang, Liang Hu, Longbing Cao, Xiaoshui Huang, Defu Lian, and Wei Liu. 2018. Attention-based Transactional Context Embedding for Next-Item Recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 8.

[28] Jihang Ye, Zhe Zhu, and Hong Cheng. 2013. What's Your Next Move: User Activity Prediction in Location-based Social Networks. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, 171–179. DOI: https://doi.org/10.1137/1.9781611972832.19

[29] Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, Yanchi Liu, Guandong Xu, Xing Xie, Hui Xiong, and Jian Wu. 2018. Sequential Recommender System based on Hierarchical Attention Networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 3926–3932. DOI: https://doi.org/10.24963/ijcai.2018/546

[30] Shuai Zhang, Yi Tay, Lina Yao, Aixin Sun, and Jake An. 2019. Next Item Recommendation with Self-Attentive Metric Learning. In *Thirty-Third AAAI Conference on Artificial Intelligence*, 9.

[31] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R. Lyu. 2017. Geo-Teaser: Geo-Temporal Sequential Embedding Rank for Point-of-interest Recommendation. In *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*, 153–162. DOI: https://doi.org/10.1145/3041021.3054138

[32] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiusi Chen, and Jun Gao. 2018. ATRank: An Attention-Based User Behavior Modeling Framework for Recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 4564–4571.