Where to Go Next: Modeling Long- and Short-Term User Preferences for Point-of-Interest Recommendation

Ke Sun,[1] Tieyun Qian,[1*] Tong Chen,[2] Yile Liang,[1] Quoc Viet Hung Nguyen,[3] Hongzhi Yin[2*]
[1]Wuhan University, China
[2]The University of Queensland, Australia
[3]Griffith University, Australia
[1]{sunke1995, qty, liangyile}@whu.edu.cn, [2]{tong.chen, h.yin1}@uq.edu.au, [3]henry.nguyen@griffith.edu.au

Abstract

Point-of-Interest (POI) recommendation has been a trending research topic as it generates personalized suggestions on facilities for users from a large number of candidate venues. Since users' check-in records can be viewed as a long sequence, methods based on recurrent neural networks (RNNs) have recently shown promising applicability for this task. However, existing RNN-based methods either neglect users' long-term preferences or overlook the geographical relations among recently visited POIs when modeling users' short-term preferences, thus making the recommendation results unreliable. To address the above limitations, we propose a novel method named Long- and Short-Term Preference Modeling (LSTPM) for next-POI recommendation. In particular, the proposed model consists of a nonlocal network for long-term preference modeling and a geo-dilated RNN for short-term preference learning. Extensive experiments on two real-world datasets demonstrate that our model yields significant improvements over the state-of-the-art methods.

1 Introduction

Location-based social networks (LBSNs) like Foursquare and Yelp are becoming pervasive in our daily lives. Users on LBSNs would like to share with friends their experiences on points of interest (POIs), e.g., restaurants and museums. The huge number of accumulated user check-in data proliferates researches on recommending POIs to unvisited users. POI recommendation is of high value to both the users and service providers, and has thus attracted much attention from researchers in the recent years (Yin et al. 2016; Cheng et al. 2018; Zhao et al. 2019).

The task of next-POI recommendation (Feng et al. 2018) is a natural extension to general POI recommendation. It aims to provide personalized recommendations on POIs to a user based on her/his historical check-in sequence. The user's long sequence can be further partitioned into multiple trajectories, where each trajectory contains a set of check-ins occurring in a specific time window such as one day or one week. By partitioning the sequence into trajectories, the next-POI recommenders facilitate the modeling of users' long- and short-term preferences. The *short-term preferences* indicate that the next POI a user will visit is influenced by her/his recently visited venues in the current trajectory. For example, a user may visit a bar right after having dinner in a restaurant. The *long-term preferences* denote a user's general interests mined from her/his historical trajectories. In summary, the long-term preferences are usually stable, while the short-term preferences tend to change frequently over time.

In next-POI recommendation, with the assumption that a user's next destination is highly correlated with her/his recently visited POIs in the current trajectory, most existing studies mainly focus on modeling the dynamics of short-term user preferences. For example, Cheng et al. (2013) apply a localized region constraint by factorizing personalized Markov Chain (FPMC) (Rendle, Freudenthaler, and Schmidt-Thieme 2010), and Feng et al. (2015) consider the personalized sequential information as well as model the geographical influence by calculating the distance between the destination POI and recently visited ones. More recently, researchers adopt recurrent neural networks (RNNs) and other variants like Gated Recurrent Unit (GRU) or Long Short-Term Memory (LSTM) to characterize users' dynamic short-term preferences, such as ST-RNN (Liu et al. 2016), TMCA (Li, Shen, and Zhu 2018) and CARA (Manotumruksa, Macdonald, and Ounis 2018).

In addition to the short-term preference dynamics reflected by the recent check-in behaviors, human mobility exhibits long-term periodical regularities. For instance, in a student's daily routine, she/he may go to a school cafeteria at lunch time in workdays. Unfortunately, in next-POI recommendation, only a few studies have taken both long- and short-term preferences into account. A successful example is the DRCF (Manotumruksa, Macdonald, and Ounis. 2017) which captures users' general long-term preferences with Collaborative Filtering (CF). DeepMove (Feng et al. 2018) exploits a deep neural network with two attention mechanisms to model the long-term periodicity. For short-term preference modeling, RNN-based approaches are adopted by both DRCF and DeepMove. As a recent attempt,

*Corresponding authors; contributing equally with the first author.

STGN (Zhao et al. 2019) presents a gated mechanism to model both long- and short-term interests, which is also under the LSTM architecture.

Though RNNs have shown inspiring results in characterizing the sequential dependency, they have an inherent limitation as they can only model consecutive activities in the user's check-in sequence. For example, given a user's POI check-in sequence $\{l_1, l_2, l_3, l_4, l_5\}$, suppose l_1 is the user's work place, l_2 is her/his home, l_3 and l_4 are respectively a restaurant and a cinema, and l_5 is a petrol station on her/his way home. Then, RNNs will model the sequence strictly in accordance with the temporal order (Sun et al. 2019). However, in this case, the user's movement from l_4 to l_5 is mainly determined by the short geographical distance between l_5 and l_2, rather than the semantic relationship between l_4 and l_5. In specific, the geographical connections between non-consecutive POIs are a key factor in deciding users' next movements, and such property can hardly be captured via existing RNN-based approaches. Moreover, despite the importance of POIs' mutual geographical relations to the discovery of long- and short-term user interests, the aforementioned methods (Manotumruksa, Macdonald, and Ounis. 2017; Feng et al. 2018) fail to incorporate the important spatial contexts when learning the long-term preferences.

To this end, we propose a novel Long- and Short-Term Preference Modeling (LSTPM) framework to address the above problems in next-POI recommendation. The main building blocks of LSTPM are inspired by the recent advances of nonlocal operations (Wang et al. 2018) and dilated RNNs (Chang et al. 2017). To thoroughly model users' long-term preferences, we develop a context-aware nonlocal network structure to explore the temporal and spatial correlations between historical and current trajectories. To conquer the limitation of RNNs in short-term user preference modeling, we propose a geo-dilated RNN to fully exploit the geographical relations among non-consecutive POIs. Furthermore, extensive experiments on real-world datasets demonstrate that our model achieves significant improvements over the state-of-the-art methods.

2 Related Work

Early studies in POI recommendation focus mainly on estimating users' preferences using Collaborative Filtering (CF), especially Matrix Factorization (MF) based techniques (Ye et al. 2011; Cheng et al. 2012; Lian et al. 2014; Gao et al. 2015). These methods can only model users' static preferences. For example, when a user living in New York travels to Hawaii for a holiday, these types of recommenders may still recommend POIs located in New York since they are unable to capture the dynamics of user preferences. More recently, deep learning based methods, such as embedding learning (Feng et al. 2015; Shi et al. 2018; Zhang et al. 2019), neural CF (He et al. 2017; Yin et al. 2017), deep latent factor model (Cheng et al. 2018), and metric learning (Tay, Anh Tuan, and Hui 2018) models, achieve promising performance in many recommendation systems.

Researches on next-POI recommendation pay more attention to users' dynamic preference modeling. The pioneering work by Cheng et al. (Cheng et al. 2013) proposes a matrix factorization method to embed the personalized Markov chains and the localized regions. Inspired by the success of RNN in sequential data modeling (Hidasi et al. 2015; Chen et al. 2019a; 2018; Huang et al. 2018), RNN based methods become pervasive in the field of next-POI recommendation (Liu et al. 2016; Manotumruksa, Macdonald, and Ounis. 2017; Feng et al. 2018; Li, Shen, and Zhu 2018). For example, ST-RNN model (Liu et al. 2016) extends RNN to model local temporal and spatial contexts. CARA (Manotumruksa, Macdonald, and Ounis 2018) captures users' dynamic preferences by exploiting GRU's gate mechanism. TMCA (Li, Shen, and Zhu 2018) and STGN (Zhao et al. 2019) adopts the LSTM-based and gated LSTM framework to learn spatial-temporal contexts, respectively. DeepMove (Feng et al. 2018) designs a multi-modal RNN to capture the sequential transition.

Overall, all the RNN/LSTM based approaches for modeling the short-term preferences suffer from the drawback of being unable to model the relations between two non-consecutive POIs. Moreover, only a few methods are presented for modeling the long-term preferences. In contrast, our proposed model considers both the long- and short-term preferences, with a geo-dilated RNN to capture geographical influence from non-consecutive POIs and a context-aware non-local structure to identify spatiotemporally relevant trajectories from the history.

3 Problem Formulation

Let $U = \{u_1, u_2, ..., u_{|U|}\}$ denote a set of LBSN users, and $L = \{l_1, l_2, ..., l_{|L|}\}$ be a set of POIs, where each POI is geocoded by a $(longitude, latitude)$ tuple, i.e., (lon_l, lat_l). For each user $u \in U$, we can obtain a trajectory sequence represented by $S = \{S_1, S_2, ..., S_n\}$, where n is the index of the current trajectory. Each trajectory $S_m \in S$ consists of a sequence of POIs visited by the user u in a consecutive order, i.e., $S_m = \{l_1, l_2, ..., l_{|S_m|}\}$ and $l \in L$.

Given the trajectory sequence S, the next-POI recommendation problem is defined as follows. For a target user $u \in U$, along with the user's historical trajectory sequence $\{S_1, S_2, ..., S_{n-1}\}$ and current trajectory $S_n = \{l_1, l_2, ..., l_{t-1}\}$ where l_{t-1} is the most recent POI that u has visited, the goal is to recommend the top-N preferable POIs to user u at the next timestamp t.

4 The Proposed Model

In this section, we present our proposed model in detail. Figure 1 depicts the architecture of LSTPM. It mainly consists of three modules including the long-term preference modeling, the short-term preference modeling, and the prediction module. Our main contribution lies in the first two modules, where we propose to model the long-term preferences using the nonlocal neural operation and model the short-term preferences using geo-dilated LSTM in a unified way.

Long-Term Preference Modeling

When modeling long-term preferences, an intuitive idea is to selectively gather the most useful information from the

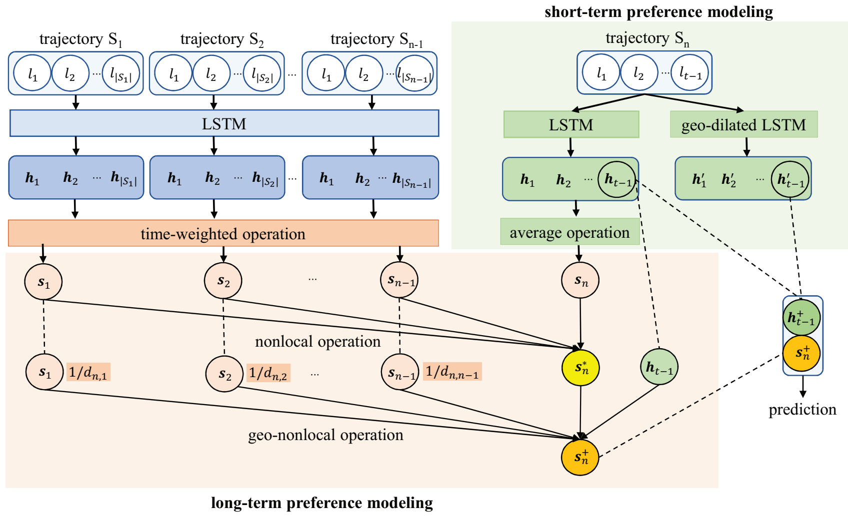


Figure 1: The architecture of our LSTPM model.

history based on the current situation. In this light, as shown at the bottom of Figure 1, we design a two-layer nonlocal network structure to learn the latent representation $\mathbf{s}_n^+$ of the target user's long-term preference. First, given a user u, we encode the information of all POIs in each historical trajectory $S_h \in \{S_1, S_2, ..., S_{n-1}\}$ using a LSTM layer that maintains sequential dependencies:

$$\mathbf{h}_t = LSTM(\mathbf{x}_t, \mathbf{h}_{t-1}), t \in \{1, 2, ..., |S_h|\} \quad (1)$$

where $\mathbf{h}_t$ is the hidden state of LSTM, $\mathbf{x}_t \in \mathbb{R}^{d\times 1}$ is the d-dimensional embedding vector for the t-th POI $l_t \in S_h$ which is randomly initialized and will be trained in the network. In this way, the POIs $\{l_1, l_2, ..., l_{|S_h|}\}$ in S_h is encoded into $\{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{|S_h|}\}$. Unlike the time-aware POI recommendation approaches which consider temporal cyclic effects, sequential POI recommendation (Liu et al. 2016; Manotumruksa, Macdonald, and Ounis 2018) focuses mainly on exploiting the sequential patterns of user preferences/interests without considering the associated absolute timestamps. However, the popularity of POIs is changing over time in real-life scenarios. For example, people may seek for restaurants during lunch hours and visit pubs at night. Hence the overall representation $\mathbf{s}_h$ for each trajectory S_h should incorporate such information to capture the time-sensitive property. To achieve this, we develop a time-weighted operation. Specifically, we map one week into 48 time slots (24 slots for hours on weekdays and 24 slots for hours on weekends). For each slot i, we construct a POI set $H_i = \{l_1, l_2, ..., l_{|H_i|}\}$ where $l \in L$ is a POI visited by at least one user in time slot i. Then, we calculate the temporal similarity $\tau_{i,j}$ between the i-th and j-th time slots as follows:

$$\tau_{i,j} = \frac{|H_i \cap H_j|}{|H_i \cup H_j|}. \quad (2)$$

Intuitively, the more overlapping POIs two time slots have, the higher their similarity will be. For a historical trajectory S_h, according to the check-in time of each POI, we can derive a sequence of check-in time slots for $|S_h|$ POIs, represented by $\{p_1, p_2, ..., p_{|S_h|}\}$, where $p \in \{1, 2, ..., 48\}$. With the target user's current time slot c, the trajectory-level representation $\mathbf{s}_h$ is generated by:

$$\mathbf{s}_h = \sum_{t=1}^{|S_h|} w_t \mathbf{h}_t, w_t = \frac{\exp(\tau_{c,p_t})}{\sum_{j=1}^{|S_h|} \exp(\tau_{c,p_j})}, \quad (3)$$

where τ_{c,p_j} is the temporal similarity between the current time slot c and the time slot of the j-th visited POI in S_h.

So far, we can represent the historical $n-1$ trajectories by $\{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_{n-1}\}$. For trajectory S_n, it is worth noting that we deploy a separate LSTM to explicitly model the recently visited POIs, which will later serve the goal of short-term preference modeling. Besides, for S_n, the time-weighted operation is replaced by an average pooling:

$$\mathbf{s}_n = \frac{1}{|S_n|} \sum_{t=1}^{|S_n|} \mathbf{h}_t, \quad (4)$$

where the rationale is that, the freshly visited POIs in trajectory S_n are more representative of users' recent preferences, and the average pooling preserves the information of all POIs in S_n, which can benefit both long- and short-term preference modeling.

After learning the representations $\{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_n\}$ for all trajectories $S = \{S_1, S_2, ..., S_n\}$, to fulfill the goal of identifying relevant trajectories from the history to represent the user's long-term preferences, we derive the *trajectory-based long-term user preference* $\mathbf{s}_n^* \in \mathbb{R}^{d\times 1}$ with respect to the current trajectory S_n using a nonlocal operation.

The basic idea of nonlocal networks is originated from computer vision tasks (Wang et al. 2018). It aims to represent each position of the input signal (e.g., images or sequences) by a weighted sum of the features at all positions to enable the modeling of nonlocal, long-range dependencies. In our case, we tend to capture the influence of each historical trajectory $S_h \in \{S_1, S_2, ..., S_{n-1}\}$ on the current trajectory S_n based on the pairwise affinities. Formally, we develop the following nonlocal operation to calculate $\mathbf{s}_n^*$:

$$\mathbf{s}_n^* = \frac{1}{C(S)} \sum_{h}^{n-1} f(\mathbf{s}_n, \mathbf{s}_h) g(\mathbf{s}_h), \tag{5}$$

where $C(S) = \sum_h^{n-1} f(\mathbf{s}_n, \mathbf{s}_h)$ is the normalization factor, and $g(\cdot)$ generates the representation for $\mathbf{s}_h$. The pairwise function $f(\cdot)$ calculates an affinity score between the current trajectory S_n and the historical trajectory S_h. Specifically, in our solution, $f(\mathbf{s}_n, \mathbf{s}_h)$ and $g(\mathbf{s}_h)$ are defined as:

$$f(\mathbf{s}_n, \mathbf{s}_h) = \exp(\mathbf{s}_n^\top \mathbf{s}_h), \tag{6}$$

$$g(\mathbf{s}_h) = \mathbf{W}_h \mathbf{s}_h, \tag{7}$$

where $\mathbf{s}_n$ and $\mathbf{s}_h$ are the representations of the current and historical trajectory S_n and S_h, respectively, while $\mathbf{W}_h$ is a trainable projection weight matrix. We now reformulate Eq.(5) as:

(8)

With the trajectory-based user preference $\mathbf{s}_n^*$, we further derive the *location-based long-term user preference* by taking the user's most recently visited location into account. Intuitively, for next-POI recommendation, the target user's next move will be highly dependent on her/his current geographic location, and a smart recommender should recommend POIs in New York when a user has just updated a check-in record in New York. So, we build a geo-nonlocal structure to compute a fine-grained long-term preference $\mathbf{s}_n^+$. To geographically measure the similarity between the current location l_{t-1} and historical trajectories $\{S_1, S_2, ..., S_{n-1}\}$, we define a central coordinate for each historical trajectory $S_h \in \{S_1, S_2, ..., S_{n-1}\}$ via the average pooling operation:

$$lon_{S_h} = \frac{lon_{l_1} + lon_{l_2} + ... + lon_{l_{|S_h|}}}{|S_h|}, \tag{9}$$

$$lat_{S_h} = \frac{lat_{l_1} + lat_{l_2} + ... + lat_{l_{|S_h|}}}{|S_h|}, \tag{10}$$

where lat_l and lon_l are the longitude and latitude for POI l in the trajectory. Similar to Eq.(8), to incorporate the spatial impacts for long-term user preference modeling, we perform the following geo-nonlocal operation by considering the distance between l_{t-1} and each historical trajectory:

$$\mathbf{s}_n^+ = \frac{1}{\sum_h^{n-1} \exp(\frac{1}{d_{n,h}} \widetilde{\mathbf{s}}_n^\top \mathbf{s}_h)} \sum_{h}^{n-1} \exp(\frac{1}{d_{n,h}} \widetilde{\mathbf{s}}_n^\top \mathbf{s}_h) \mathbf{W}_{h'} \mathbf{s}_h, \tag{11}$$

where $\widetilde{\mathbf{s}}_n = \mathbf{s}_n^* + \mathbf{h}_{t-1}$ is the aggregation of trajectory-based long-term user preference and the hidden state of the most current POI, and $d_{n,h}$ is the geographical distance between the POI location l_{t-1} and the trajectory S_h, i.e., $d_{n,h} = \sqrt{(lon_{l_{t-1}} - lon_{S_h})^2 + (lat_{l_{t-1}} - lat_{S_h})^2}$. Clearly, the final representation $\mathbf{s}_n^+$ combines temporal information from all trajectories and spatial associations between each historical trajectory and the last check-in location. As a result, with our dedicated nonlocal network structure, the learned long-term user preference is a comprehensive spatiotemporal representation.

Short-Term Preference Modeling

In POI recommendation, existing RNN-based methods treat a user's visited POI sequence as a straight pipeline without considering any spatial relations among the POIs. Given a temporal POI check-in sequence, RNNs can only learn temporal dependencies that strictly follow the sequential order, which is illustrated in Figure 2-(a). Though some RNN-based approaches such as CARA (Manotumruksa, Macdonald, and Ounis 2018) and TMCA (Li, Shen, and Zhu 2018) take time or distance transition as additional information, they can only model relationships among the inputs that are consecutive and well-aligned. However, in real-life scenarios, the visited POIs are often geographically scattered. Following the example in Section 1, within the POI check-in sequence $S_n = \{l_1, l_2, l_3, l_4, l_5\}$, l_5 is geographically much closer to l_2 than l_4, which means l_2 is critical in determining the next move from l_4 to l_5.

To alleviate such incontinuity of model inputs, dilated RNNs (Chang et al. 2017) introduce a fixed skip length on the inputs, e.g., $\{l_1, l_3, l_5\}$ for the skip length of 2. Formally, the LSTM form of dilated RNN is defined as follows:

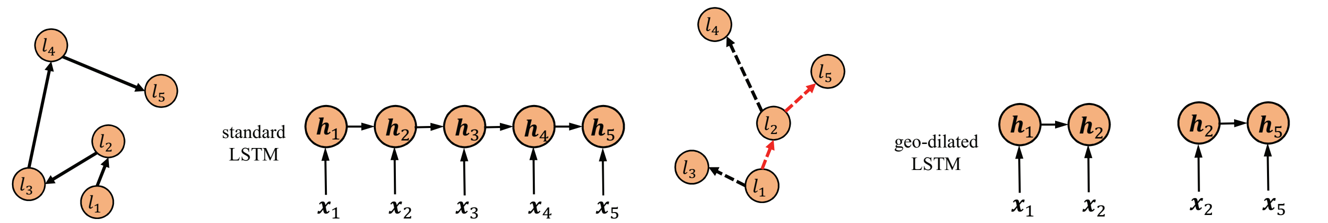
$$\mathbf{h}_t = LSTM(\mathbf{x}_t, \mathbf{h}_{t-\Delta}), \tag{12}$$

where Δ is the skip length. However, the skip length in dilated RNNs is always predefined and fixed, thus making it hard to generalize to POI recommendation tasks. Hence, in what follows, we propose our *geo-dilated LSTM* scheme that automatically determines the relevant inputs to be used based on both geographical and temporal factors.

Figure 2-(b) presents the geo-dilated POI sequence based on the original temporal sequence. Given S_n, we propose Algorithm 1 to construct the input set S_n^{geo} at time $t-1$

Algorithm 1 Constructing the input set S_n^{geo} at time $t-1$ for geo-dilated LSTM

Input: the temporal POI sequence in the recent trajectory $S_n = \{l_1, l_2, ..., l_{t-1}\}$
Output: the geo-dilated sequence set S_n^{geo}
initialize with $S_n^{geo} = \varnothing$, $S_n' = \varnothing$, and $i = 2$, where i is an index for generating the dilated sequence;
compute the POI distance matrix $D \in \mathbb{R}^{(t-1)\times(t-1)}$, with entry $d_{y,z} = d_{z,y} = \sqrt{(lon_{l_y} - lon_{l_z})^2 + (lat_{l_y} - lat_{l_z})^2}$;
while $i \le t-1$ **do**
    set $j = i - 1$ and $D' = \varnothing$;
    while $j \ge 1$ **do**
        $d_{l_i, l_j} \mapsto D'$;
        $j - -$;
    end while
    find $d_{l_i, l_k} = \min(D')$;
    $\{l_k, l_i\} \mapsto S_n'$
    $i + +$;
end while
find the path from l_1 to l_{t-1}, denoted by $\{\{l_1, l_\theta\}, ..., \{l_\delta, l_{t-1}\}\}$, where $1 < \theta < ... < \delta < t-1$;
$\{\{\mathbf{x}_1, \mathbf{x}_\theta\}, ..., \{\mathbf{x}_\delta, \mathbf{x}_{t-1}\}\} \mapsto S_n^{geo}$, where each $\mathbf{x} \in \mathbb{R}^{d\times 1}$ is the embedding vector for the corresponding POI l;
end



(a) the temporal POI sequence and the standard LSTM (b) the geo-dilated POI sequence and the geo-dilated LSTM

Figure 2: The comparison between the standard LSTM and the geo-dilated LSTM. In this demonstration, we use both the standard LSTM and the geo-dilated LSTM to learn the representation of the last POI l_5 from the temporal sequence and geo-dilated sequence respectively.

for geo-dilated LSTM. In this case, l_3 has two preceding POIs in the temporal sequence: l_1 and l_2. Since the geographical distance between l_1 and l_3 is shorter than the distance between l_1 and l_2, there exists a geo-dilated sequence $\{l_1, l_3\}$. Similarly, we construct a set of geo-dilated sequences from the temporal sequence, denoted by S'_n. Afterwards, by finding the path from l_1 to l_5, we construct the input set $S_n^{geo} = \{\{\mathbf{x}_1, \mathbf{x}_2\}, \{\mathbf{x}_2, \mathbf{x}_5\}\}$ with the POI embedding vector $\mathbf{x} \in \mathbb{R}^{d \times 1}$. In short, S_n^{geo} contains the two-step paths from l_1 to l_{t-1}. In a mathematical form, the computation of geo-dilated LSTM involves the iteration of a two-step LSTM:

$$\mathbf{h}'_{t-1} = LSTM(\mathbf{x}_{t-1}, \mathbf{h}'_{\delta}), \quad (13)$$

where $\mathbf{h}'_{t-1}$ is computed from the last sequence $\{\mathbf{x}_{\delta}, \mathbf{x}_{t-1}\} \in S_n^{geo}$. Intuitively, our geo-dilated LSTM first picks POIs from the current trajectory as the input with different skip lengths determined by the geographical relevance, and then learns the short-term user preference via the dilated LSTM scheme.

At time $t-1$, with the latent representations $\mathbf{h}_{t-1}$ and $\mathbf{h}'_{t-1}$ respectively learned by the standard and geo-dilated LSTMs, the final representation of the short-term user preference is an average of both vectors:

$$\mathbf{h}^+_{t-1} = \frac{\mathbf{h}_{t-1} + \mathbf{h}'_{t-1}}{2}, \quad (14)$$

where $\mathbf{h}^+_{t-1}$ fuses the temporal dependencies from $\mathbf{h}_{t-1}$ with the spatial dependencies from $\mathbf{h}'_{t-1}$.

Prediction

After obtaining the representations for both long- and short-term user preferences, we compute the probability distribution $\mathbf{p}$ over the $|L|$ POIs via the following:

$$\mathbf{p} = softmax\big(\mathbf{W}_p(\mathbf{s}^+_n \oplus \mathbf{h}^+_{t-1})\big) \quad (15)$$

where $\oplus$ is the concatenation of long- and short-term user preferences, $\mathbf{W}_p \in \mathbb{R}^{|L| \times 2d}$ is a trainable projection matrix for all POIs. Consequently, the most likely POI that will be visited by the target user at the next time step t is the one with the largest probability. If we denote the probability of ground truth POI as $p \in \mathbf{p}$, then the objective function can be formulated as the log likelihood:

Table 1: Statistics of the evaluation datasets.

Datasets	#user	#POI	#check-in	#sessions	sparsity
Foursquare	934	9,296	52,983	12,510	99.38%
Gowalla	5,802	40,868	301,080	75,733	99.87%

$$\mathcal{L} = -\sum_{k=1}^{N} log(p_k) \quad (16)$$

where N is the total number of training samples, p_k represents the probability of the ground truth POI generated by the model regarding the k-th training sample.

5 Experiments

In this section, we evaluate LSTPM by competing against the state-of-the-art methods on two real-world datasets.

Evaluation Datasets

We conduct experiments on two widely-used real LSBN datasets, namely Foursquare (Feng et al. 2018) and Gowalla (Yin et al. 2015). The Foursquare check-in dataset is collected from February 2010 to January 2011 in New York while Gowalla contains world-wide check-ins from February 2009 to October 2010. For both of them, we eliminate unpopular POIs that are visited by less than 10 users. We treat users' all check-ins in one day as a single trajectory and remove trajectories having less than three check-ins. Inactive users with less than 5 trajectories are also filtered out. According to (Feng et al. 2018), the first 80% of each users' trajectories are used for training and the rest are for testing. The statistics of both datasets after preprocessing are shown in Table 1.

Baselines and Settings

We compare the performance of LSTPM with the following eight peer methods:

LSTM: This is a variant of RNN model which has shown effective in handling sequential data.

Time-LSTM: (Zhu et al. 2017) This is an extension of LSTM which employs time gates to model the time intervals between continuous inputs.

Table 2: Performance comparison on two datasets w.r.t. $Recall@K$ and $NDCG@K$.

	Foursquare						Gowalla					
	Rec@1	Rec@5	Rec@10	NDCG@1	NDCG@5	NDCG@10	Rec@1	Rec@5	Rec@10	NDCG@1	NDCG@5	NDCG@10
LSTM	0.1024	0.2115	0.2639	0.1024	0.1597	0.1768	0.0443	0.0971	0.1259	0.0443	0.0716	0.0809
ST-RNN	0.0884	0.2291	0.2968	0.0884	0.1615	0.1835	0.0505	0.1245	0.1687	0.0504	0.0885	0.1027
Time-LSTM	0.1307	0.2868	0.3667	0.1307	0.2117	0.2375	0.0639	0.1500	0.1963	0.0639	0.1082	0.1232
CARA	0.1112	0.3002	0.3902	0.1113	0.2093	0.2384	0.0627	0.1563	0.2090	0.0627	0.1112	0.1283
DRCF	0.1118	0.2743	0.3466	0.1118	0.1971	0.2206	0.0751	0.1709	0.2192	0.0751	0.1249	0.1406
DeepMove	0.1451	0.3110	0.3831	0.1451	0.2323	0.2556	0.0681	0.1330	0.1656	0.0681	0.1023	0.1128
TMCA	0.0953	0.2198	0.2722	0.0953	0.1601	0.1771	0.0499	0.1208	0.1588	0.0499	0.0865	0.0988
STGN	0.1118	0.2730	0.3547	0.1118	0.1951	0.2217	0.0745	0.1600	0.2041	0.0745	0.1191	0.1333
LSTPM	**0.1561**	**0.3372**	**0.4091**	**0.1561**	**0.2515**	**0.2749**	**0.0984**	**0.2021**	**0.2510**	**0.0984**	**0.1523**	**0.1681**

ST-RNN: (Liu et al. 2016) This is a RNN-based deep POI recommendation model that incorporates spatial-temporal transition matrices.

TMCA: (Li, Shen, and Zhu 2018) This method fuses multiple types of context including spatiotemporal transitions and POI categories with two attention mechanisms. Note that we remove the POI categorical context for fairness because no other methods make use of it.

CARA: (Manotumruksa, Macdonald, and Ounis 2018) This method extends the GRU gates to respectively model the ordinary and transition context for POI recommendation.

DCRF: (Manotumruksa, Macdonald, and Ounis. 2017) This method aims to capture both users' dynamic and static preferences for POI recommendation. In our case, it can also model long- and short-term preferences.

DeepMove: (Feng et al. 2018) This method learns user's long-term preference from the history with the attention mechanism and learns short-term preference from the current trajectory using a RNN module.

STGN: (Zhao et al. 2019) This method models temporal and spatial contexts by adding time and distance gates to integrate time and distance intervals.

Following (Feng et al. 2018), we set the dimension of embeddings and the hidden states to 500 for all deep learning-based methods. All the parameters in our model are optimized using the gradient descent optimization algorithm Adam with the batch size of 32 and the learning rate of 0.0001. Note that we do not compare our model with non-deep learning methods. The reason is that it has been widely proved that the baselines adopted in our experiments outperform the non-deep learning methods, e.g., DRCF (Manotumruksa, Macdonald, and Ounis. 2017) vs. the traditional MF, and the BPR model (Rendle et al. 2009), and thus we only show the improvements over these deep learning-based baselines.

Evaluation Criteria

To evaluate the performance of each method for next-POI recommendation, we adopt two evaluation metrics that are commonly-applied in previous works (Li, Shen, and Zhu 2018; Liu et al. 2016; Chen et al. 2019b): $Recall@K$ and $Normalized\ Discounted\ Cumulative\ Gain$ ($NDCG@K$). $Recall@K$ measures the presence of the correct POI among the top K recommended POIs, and $NDCG@K$ measures the quality top-K ranking list. In this paper, we choose the popular $K = \{1, 5, 10\}$ for evaluation.

Analysis on Recommendation Effectiveness

The results of different methods for next-POI recommendation are shown in Table 2. In each column, the best result is highlighted in boldface and the second best is underlined. From the statistics, we draw the following observations:

- Our proposed LSTPM consistently and significantly outperforms all baselines in terms of every metric on both Foursquare and Gowalla datasets. For example, on Foursquare, compared with the second best method DeepMove, LSTPM improves the $Recall@5$ and $NDCG@5$ by 8.4% and 8.2% respectively. On Gowalla, our method shows the advantage against others by an obvious margin, where the overall improvement over the best competitor DCRF achieves 22.7% on average. The quantitative evaluation clearly demonstrates the superior effectiveness of our method.
- Among the baseline methods, DeepMove performs the best on Foursquare dataset while DRCF is the best on Gowalla dataset. As both approaches take the long- and short-term preferences into consideration, it strongly illustrates the importance of modeling long-term preferences from historical information apart from short-term preference modeling. However, without the consideration of transition contexts (e.g., time or distance), DRCF and DeepMove can hardly maximize the recommendation accuracy, and this is the key factor that LSTPM can outperform these two methods.
- Though showing promising results on Foursquare, the performance of DeepMove on Gowalla drops greatly. This is because DeepMove does not integrate distance information, but Gowalla contains world-wide POIs that are highly sparse. This further verifies the importance of geographical distance for POI recommendation.
- ST-RNN, Time-LSTM, CARA, TMCA, STGN are better than LSTM. This shows the advantage of modeling the spatiotemporal relations among different POIs. In addition, TMCA performs worse than ST-RNN, Time-LSTM, CARA, and STGN as it simply takes contextual information as the input. In contrast, other methods integrate spatial or temporal information into the RNN model, which can result in the eventual performance improvement.

Table 3: Performance of different LSTPM variants.

	Foursquare						Gowalla					
	Rec@1	Rec@5	Rec@10	NDCG@1	NDCG@5	NDCG@10	Rec@1	Rec@5	Rec@10	NDCG@1	NDCG@5	NDCG@10
best baseline	0.1451	0.3110	0.3902	0.1451	0.2323	0.2556	0.0751	0.1709	0.2192	0.0751	0.1249	0.1406
LSTPM-L	0.1191	0.3121	0.3912	0.1191	0.2203	0.2460	0.0755	0.1682	0.2100	0.0755	0.1243	0.1378
LSTPM-S	0.1116	0.2187	0.2642	0.1116	0.1673	0.1780	0.0716	0.1539	0.1941	0.0716	0.1145	0.1275
LSTPM	**0.1561**	**0.3372**	**0.4091**	**0.1561**	**0.2515**	**0.2749**	**0.0984**	0.2021	**0.2510**	**0.0984**	**0.1523**	**0.1681**

Analysis on Key Components in LSTPM

To verify the contribution of different components in LSTPM to the performance gain, we further implement two simplified versions of our model to conduct ablation tests:

- **LSTPM-L:** This version removes the short-term component of LSTPM and only engages the long-term one.
- **LSTPM-S:** This version removes the long-term component of LSTPM and only engages the short-term one.

The results of the degraded versions of LSTPM on two datasets are shown in Table 3. Through the ablation tests, we can observe that:

- LSTPM-L always performs better than LSTPM-S, and its results on Gowalla are very close to the strongest baselines. The reason might be that LSTPM-L can better capture users' long-term periodicity based on her/his current state, which is critical for next-POI recommendation. This clearly demonstrates the benefit of our modeling users' long-term preferences.
- While LSTPM-S is less competitive than LSTPM-L, it can still get better prediction performance than many baselines such as LSTM and ST-RNN in Table 2. Remember that these methods also adopt LSTM structures. Hence the effectiveness of LSTPM-S comes from our proposed geo-dilated LSTM which captures geographical relations among non-consecutive POIs.
- The complete model LSTPM, which is the combination of LSTPM-L and LSTPM-S, achieves the best performance on both datasets, showing that both long-term and short-term preferences has positive impacts on the user's choice to the next POI.

Analysis on Impact of History Length

One key contribution of our LSTPM model is that it takes historical information into account to mine the long-term preferences. We investigate the impact of different history lengths on modeling long-term preferences. We split the users' trajectories into 7 groups based on their numbers of historical trajectories. The first group contains users having 4 or less historical trajectories, while users having 10 or more ones are in the last group. We then evaluate the performance separately regarding each group as the model input. The results are shown in Figure 3.

Fig. 3 shows an overall upward trend of model performance on two datasets with the growing history length, as it takes time to reveal people's periodical regularities. The longer the history length is, the more information can be extracted by our long-term preference modeling component.

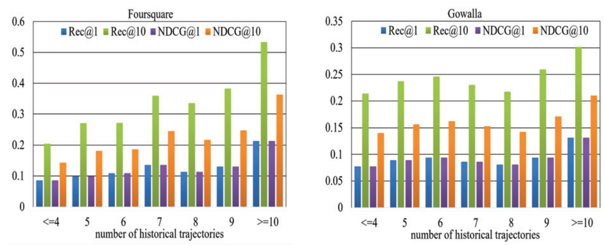


Figure 3: The impact of history length.

Analysis on Effectiveness of Geo-Dilated LSTM

Another contribution of our LSTPM model lies in the geo-dilated LSTM which is used to capture the relations between non-consecutive POIs in the current session. In this section, we examine the effectiveness of geo-dilated LSTM and the LSTM on modeling the short-term preference. Note that to exclude the influence of the long-term component, we conduct this experiment on the short-term component, i.e., we implement two variants of LSTPM-S. One uses the LSTM structure, and the other adopts the combination of LSTM and Geo-Dilated LSTM. The results are shown in Figure 4.

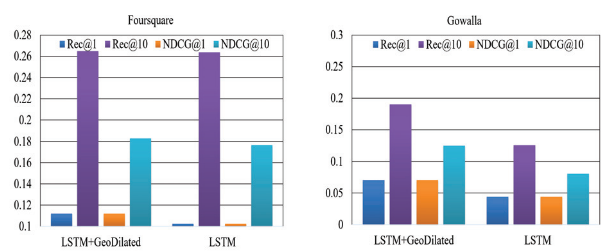


Figure 4: The effectiveness of geo-dilated LSTM.

From Fig. 4, it is clear that our "LSTM+Geo-Dilated" structure yields better performance than the single LSTM on both datasets. This verifies the importance of modeling users activities in visiting non-consecutive POIs. Another interesting finding would be that the superiority of "LSTM+Geo-Dilated" is more significant on Gowalla than Foursquare. Such a phenomenon might be attributed to the property of the dataset. The users in Foursquare exhibits local behaviors whereas those in Gowalla usually have global range. In the global setting, our geo-dilated LSTM can assign higher weights on closer POIs, rather than on consecutive ones used in LSTM. It makes sense when a user lives in New York and has a business trip to Chicago and then back to Yew York.

In this case, the geographical relation play more important role than the temporal visiting order. It therefore links to the positive effect of our proposed geo-dilated LSTM structure.

6 Conclusion

In this paper, we propose a novel model, namely LSTPM for next-POI recommendation. Specifically, we develop a context-aware non-local network to model long-term preferences and a geo-dilated LSTM to model short-term preferences. The experimental results demonstrate that our proposed approach substantially improves the recommendation accuracy compared with the state-of-the-art methods. Notably, while our proposed method outperforms all baselines, we are aware of the common problem in POI recommendation, i.e., the low recall and NDCG scores. This is reasonable due to the high data sparsity. In the future, we plan to address this issue by introducing user/POI side information.

Acknowledgments. This work has been supported in part by the NSFC Projects (61572376, 91646206) and ARC Discovery Project (Grant No. DP190101985).

References

Chang, S.; Zhang, Y.; Han, W.; Yu, M.; Guo, X.; Tan, W.; Cui, X.; Witbrock, M.; Hasegawa-Johnson, M. A.; and Huang, T. S. 2017. Dilated recurrent neural networks. In *NIPS*, 77–87.

Chen, T.; Yin, H.; Chen, H.; Wu, L.; Wang, H.; Zhou, X.; and Li, X. 2018. Tada: trend alignment with dual-attention multi-task recurrent neural networks for sales prediction. In *ICDM*, 49–58.

Chen, T.; Yin, H.; Chen, H.; Yan, R.; Nguyen, Q. V. H.; and Li, X. 2019a. Air: Attentional intention-aware recommender systems. In *ICDE*, 304–315.

Chen, T.; Yin, H.; Nguyen, Q. V. H.; Peng, W.-C.; Li, X.; and Zhou, X. 2019b. Sequence-aware factorization machines for temporal predictive analytics. *arXiv preprint arXiv:1911.02752*.

Cheng, C.; Yang, H.; King, I.; and Lyu, M. R. 2012. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 17–23.

Cheng, C.; Yang, H.; Lyu, M. R.; and King., I. 2013. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, 2605–2611.

Cheng, W.; Shen, Y.; Zhu, Y.; and Huang, L. 2018. Delf: A dual-embedding based deep latent factor model for recommendation. In *IJCAI*, 3329–3335.

Feng, S.; Li, X.; Zeng, Y.; Cong, G.; Chee, Y. M.; and Yuan, Q. 2015. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI*, 2069–2075.

Feng, J.; Li, Y.; Zhang, C.; Sun, F.; Meng, F.; Guo, A.; and Jin, D. 2018. Deepmove: Predicting human mobility with attentional recurrent networks. In *WWW*.

Gao, H.; Tang, J.; Hu, X.; ; and Liu, H. 2015. Content-aware point of interest recommendation on location-based social networks. In *AAAI*, 1721–1727.

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering. In *WWW*, 173–182.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Huang, J.; Zhao, W. X.; Dou, H.; Wen, J.-R.; and Chang, E. Y. 2018. Improving sequential recommendation with knowledge-enhanced memory networks. In *SIGIR*, 505–514.

Li, R.; Shen, Y.; and Zhu, Y. 2018. Next point-of-interest recommendation with temporal and multi-level context attention. In *ICDM*, 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 *KDD*, 831–840.

Liu, Q.; Wu, S.; Wang, L.; and Tan, T. 2016. Predicting the next location: a recurrent model with spatial and temporal contexts. In *AAAI*, 194–200.

Manotumruksa, J.; Macdonald, C.; and Ounis., I. 2017. A deep recurrent collaborative filtering framework for venue recommendation. In *CIKM*, 1429–1438.

Manotumruksa, J.; Macdonald, C.; and Ounis, I. 2018. A contextual attention recurrent architecture for context-aware venue recommendation. In *SIGIR*, 555–564.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2009. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, 452–461.

Rendle, S.; Freudenthaler, C.; and Schmidt-Thieme, L. 2010. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, 811–820.

Shi, C.; Hu, B.; Zhao, W. X.; and Philip, S. Y. 2018. Heterogeneous information network embedding for recommendation. *TKDE* 357–370.

Sun, K.; Qian, T.; Yin, H.; Chen, T.; Chen, Y.; and Chen, L. 2019. What can history tell us? identifying relevant sessions for next-item recommendation. In *CIKM*, 1593–1602.

Tay, Y.; Anh Tuan, L.; and Hui, S. C. 2018. Latent relational metric learning via memory-based attention for collaborative ranking. In *WWW*, 729–739.

Wang, X.; Girshick, R.; Gupta, A.; and He, K. 2018. Non-local neural networks. In *CVPR*, 7794–7803.

Ye, M.; Yin, P.; Lee, W.-C.; and Lee, D.-L. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, 325–334.

Yin, H.; Cui, B.; Chen, L.; Hu, Z.; and Zhang, C. 2015. Modeling location-based user rating profiles for personalized recommendation. *TKDD* 9(3):19.

Yin, H.; Cui, B.; Zhou, X.; Wang, W.; Huang, Z.; and Sadiq, S. 2016. Joint modeling of user check-in behaviors for real-time point-of-interest recommendation. *TOIS* 35(2):11.

Yin, H.; Wang, W.; Wang, H.; Chen, L.; and Zhou, X. 2017. Spatial-aware hierarchical collaborative deep learning for poi recommendation. *IEEE Transactions on Knowledge and Data Engineering* 29(11):2537–2551.

Zhang, S.; Yin, H.; Wang, Q.; Chen, T.; Chen, H.; and Nguyen, Q. V. H. 2019. Inferring substitutable products with deep network embedding. *IJCAI* 4306–4312.

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 *AAAI*, 5877–5884.

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 *IJCAI*, 3602–3608.