Discovering Subsequence Patterns for Next POI Recommendation

Kangzhi Zhao[1] , **Yong Zhang**[1] , **Hongzhi Yin**[2*] , **Jin Wang**[3] ,
Kai Zheng[4] , **Xiaofang Zhou**[2] and **Chunxiao Xing**[1]
[1]BNRist, RIIT, Department of Computer Science and Technology, Tsinghua University
[2]School of Information Technology and Electrical Engineering, The University of Queensland
[3]Computer Science Department, University of California
[4]University of Electronic Science and Technology of China
zkz15@mails.tsinghua.edu.cn, {zhangyong05, xingcx}@tsinghua.edu.cn,
{h.yin1, zxf}@uq.edu.au, jinwang@cs.ucla.edu, zhengkai@uestc.edu.cn

Abstract

Next Point-of-Interest (POI) recommendation plays an important role in location-based services. State-of-the-art methods learn the POI-level sequential patterns in the user's check-in sequence but ignore the subsequence patterns that often represent the socio-economic activities or coherence of preference of the users. However, it is challenging to integrate the semantic subsequences due to the difficulty to predefine the granularity of the complex but meaningful subsequences. In this paper, we propose Adaptive Sequence Partitioner with Power-law Attention (ASPPA) to automatically identify each semantic subsequence of POIs and discover their sequential patterns. Our model adopts a state-based stacked recurrent neural network to hierarchically learn the latent structures of the user's check-in sequence. We also design a power-law attention mechanism to integrate the domain knowledge in spatial and temporal contexts. Extensive experiments on two real-world datasets demonstrate the effectiveness of our model.

1 Introduction

With the rapid growth of Location-Based Social Networks (LBSNs), such as Yelp and Foursquare, point-of-interest (POI) recommendation has attracted wide attention from both academia and industry [Zhang and Wang, 2015; Yin *et al.*, 2016; Wang *et al.*, 2018]. As a natural extension of general POI recommendation, next POI recommendation aims at predicting the POIs that are most likely to be visited next given the user's check-in history. Compared with the general POI recommendation, next POI recommendation focuses more on exploiting user movement patterns hidden in the historical check-in sequence.

Next POI recommendation has attracted great research interest recently. Early studies model the influence of previously visited POIs on future ones in a static manner. They regard each POI sequence as a Markov chain [Cheng *et al.*, 2013; Feng *et al.*, 2015; He *et al.*, 2016; Zhao *et al.*, 2016] and then integrate the sequential patterns in terms of transition probability between POIs to the conventional matrix factorization models. Recent studies adopt recurrent neural network (RNN) based models to keep track of the sequential dependencies [Kong and Wu, 2018; Liu *et al.*, 2016; Zhao *et al.*, 2019]. However, such sequential dependencies cannot fully account for human mobility [Cho *et al.*, 2011; Kong and Wu, 2018]. The user's check-in behaviors may be influenced by a subsequence of previously visited POIs as a whole rather than single POIs, which is beyond POI-level sequential patterns. Figure 1 shows the subsequence pattern of the user's check-in sequence. Modern cities consist of functional zones such as residential areas, business districts, and educational areas, etc. It is quite possible that people's movement patterns sometimes only exist among functional zones rather than single POIs. Besides, their movements within a specific zone could not show any sequential pattern. Thus, the check-in subsequence generated within a functional zone should be treated as a whole to represent the semantics or function of the corresponding geographical region. More-

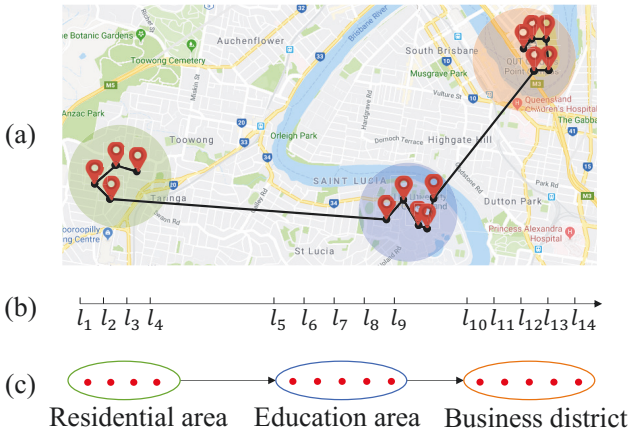


Figure 1: An example of the user's check-in sequence. (a) Different colors represent different functional zones of the city. (b) represents the POI-level sequential pattern. (c) shows check-in subsequences divided by the functional zones.

*Corresponding Author

over, such subsequences are not limited to the functional zones. Actually, they can result from different factors in reality. For example, a subsequence may be generated because all the POIs in the subsequence match the user's personal interests. The set of POIs in the subsequence will perform as a whole when influencing the user's future visit.

Nevertheless, discovering and integrating subsequence-level sequential patterns for the next POI recommendation is challenging. The reason is that it is extremely difficult and even impossible to predefine the length or granularity of effective and meaningful subsequence (i.e., semantic subsequence) due to their variety and complexity [Cho *et al.*, 2011]. In fact, these semantic subsequences have different granularities or lengths. In order to exploit and integrate such multi-granularity sequential patterns in the next POI recommendation, we need to automatically analyze the underlying structures and identify each semantic subsequence in the user's check-in sequence.

In this paper, we aim to automatically detect and identify each semantic subsequence of POIs and discover their sequential patterns (i.e., sequential dependencies among semantic subsequences). To this end, we propose a novel model Adaptive Sequence Partitioner (ASP). Specifically, we design a hierarchical framework to automatically learn the latent structures of the user's check-in sequence, which is similar to the latent topic discovery from texts [Blei *et al.*, 2003], but much more challenging due to the consideration of POI order structure. Technically, we adopt a stacked recurrent neural network to hierarchically learn the representations from the check-in sequence. In this design, the higher layer learns from the lower in order to perform an abstraction with a larger level of granularity. In order to automatically identify the partition of each layer, we introduce the boundary detector into the RNN cell and adopt the state-based methods to detect boundaries in the input sequence. In this way, successive cells with the same state will formalize a segment and be updated in a lower frequency, while the higher layers can iteratively learn from the lower and formalize the larger segment. As a result, the latent structure can be well captured into the segments with different levels of granularity. In order to aggregate the structures from different stacked layers, we further introduce the layer-wise attention to learn the alignment for each layer.

However, the intrinsic sparsity of the user's check-in sequence may become the bottleneck of our proposed ASP and lead to overfitting. This is because LBSN data has a low sampling rate in both space and time compared with GPS trajectories [Wang *et al.*, 2016; Lv *et al.*, 2018]. We need to incorporate domain knowledge to alleviate the issue of data sparsity in order to leverage the power of deep neural networks. To this end, we propose to integrate the domain knowledge in both spatial and temporal contexts (i.e., the spatial and temporal effects on the user's check-in behaviors) and develop a novel Power-law Attention (PA) mechanism with fewer parameters to replace the conventional vanilla attention mechanism. Specifically, we adopt the power-law decay towards the distance and exponential decay towards the time interval into the standard attention alignments among all the representations of outputs. In this way, PA could express the model that is equivalent to the real-world average willingness-to-visit [Cho *et al.*, 2011; Yuan *et al.*, 2013] under power law.

To sum up, the main contributions of our research are summarized as follows:

- We design and develop a novel recommendation framework ASPPA to identify the semantic subsequences of POIs and discover their sequential patterns. Specifically, this model can learn the latent structures by automatically partitioning the POI sequence without handcrafted features.
- We propose a Power-law Attention mechanism to integrate the domain knowledge in both spatial and temporal contexts for the user's check-in sequence. Our design alleviates the data sparsity and largely reduces down the number of parameters.
- We conduct extensive experiments to evaluate our framework on two widely used real-world datasets. The results show that our model is effective and outperform state-of-the-art techniques.

2 Related Work

POI recommendation has attracted intensive attention due to a wide range of potential applications. Most existing studies focus on how to model user preferences for POIs with the awareness of the joint effect of rich contexts [Liu *et al.*, 2017]. Some recent studies utilize neural network based approaches for POI recommendation [Yin *et al.*, 2017]. Yang et al. [2017] regard user-user and POI-POI graph as the semi-supervised labels and adopt a multi-layer feed-forward network to conduct collaborative filtering.

The next POI recommendation is an emerging task and has attracted great research interest. Zhao et al. [2016] build a pairwise tensor factorization framework with user-POI, POI-time, and POI-POI interactions. He et al. [2016] explore the latent pattern-level preference for each user and propose a tensor-based latent model to capture the successive check-in behavior. Feng et al. [2015] develop a pairwise metric embedding algorithm to jointly model the sequential information and individual preference. Liu et al. [2016] introduce time-specific and distance-specific transition matrix in RNN to model local temporal and spatial contexts. Kong et al. [2018] and Zhao et al. [2019] integrate spatial and temporal gating mechanism into LSTM units to incorporate the necessary information for prediction. Chang et al. [2018] provide a better pre-trained POI embedding model to capture the characteristics of POIs and improve existing POI recommendation models. Wang et al. [2020] develop a next POI recommendation system for mobile devices. Although previous studies exploit the sequential patterns in the user's check-in sequence, they only focus on POI-level and ignore the sequential dependency among semantic subsequences. Kong et al. [2018] explore the area of interest that the user visits, but they need to manually predefine the user's visit sessions. Our ASPPA can detect the semantic subsequences of POIs and further discover their sequential patterns to improve the next POI recommendation.

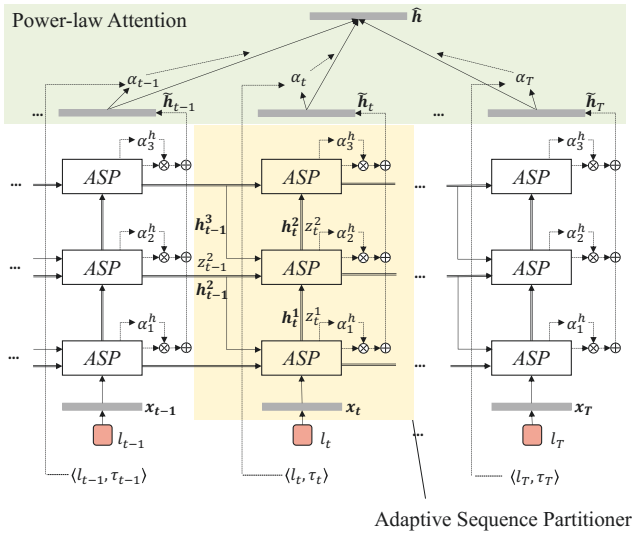


Figure 2: Architecture of ASPPA model

3 Proposed Model

In this section, we introduce the details of our **A**daptive **S**equence **P**artitioner with **P**ower-law **A**ttention (ASPPA) model. Our model identifies the semantic subsequences of POIs and integrates the domain knowledge in spatial and temporal context. To this end, we design an attention-based stacked RNN architecture. As shown in Figure 2, the model consists of two components, i.e., Adaptive Sequence Partitioner (ASP) and Power-law Attention (PA). Next, we give the problem statement and then introduce each component.

3.1 Problem Statement

Let $\mathcal{U}$ be the set of users and $\mathcal{L}$ be the set of POIs. For each user u, her check-in history (in chronology order) is denoted as $\mathcal{V}_u = \{(l_t^u, \tau_t) \mid t = 1, 2, ...\}$, where each tuple (l_t^u, τ_t) is the t^{th} visit of user $u \in \mathcal{U}$, with POI $l_t^u \in \mathcal{L}$ and time stamp τ_t. The problem of next POI recommendation is to predict the POIs for user u at the next time step, i.e., for each user we learn a personalized ranking function $f_u : \mathcal{V} \to \Omega$ that maps the user's check-in history to a probability distribution for the target POI set

$$f_u = \Pr\left(y \in \mathcal{L} \mid \mathcal{V}_u, u\right) \tag{1}$$

The output is the top-k ranked POIs according to the predicted probabilities.

3.2 Adaptive Sequence Partitioner

We propose the Adaptive Sequence Partitioner (ASP) to learn the latent structures of the user's check-in sequence. In order to capture the sequential patterns of semantic subsequences, we need to learn hierarchically from the check-in sequence, which is similar to the latent topic discovery from texts [Blei *et al.*, 2003]. Thus ASP is designed to be a *stacked RNN architecture* [Hihi and Bengio, 1995]. As shown in Figure 2, lower layer is served as the input of the upper layer. It transfers the state output $\boldsymbol{h}$ as well as the boundary detector z. If the input boundaries can form the partition and control the update of the sequence, we can learn the subsequence information in each layer and the sequential patterns among these POI subsequences can also be learned from the hierarchy of the model. We will show in the following parts.

Gating Mechanism. In ASP, the subsequence is identified through the partition of each layer. Inspired by [Chung *et al.*, 2017], ASP adopts a binary boundary detector z to control the pattern of cell update. Specifically, for ℓ layer, ASP has the input, forget, output gates $(\boldsymbol{i}_t^\ell, \boldsymbol{f}_t^\ell, \boldsymbol{o}_t^\ell)$, cell proposal $\tilde{\boldsymbol{c}}_t^\ell$ and the pre-activation of boundary detector $\tilde{z}_t^\ell$, which are obtained by

$$\begin{bmatrix} \boldsymbol{i}_t^\ell \\ \boldsymbol{f}_t^\ell \\ \boldsymbol{o}_t^\ell \\ \tilde{\boldsymbol{c}}_t^\ell \\ \tilde{z}_t^\ell \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \phi \\ \sigma \end{bmatrix} \left(\boldsymbol{W}_\ell^\ell \boldsymbol{h}_{t-1}^\ell + z_{t-1}^\ell \boldsymbol{W}_{\ell+1}^\ell \boldsymbol{h}_{t-1}^{\ell+1} + z_t^{\ell-1} \boldsymbol{W}_{\ell-1}^\ell \boldsymbol{h}_t^{\ell-1} + \boldsymbol{b}^\ell \right) \tag{2}$$

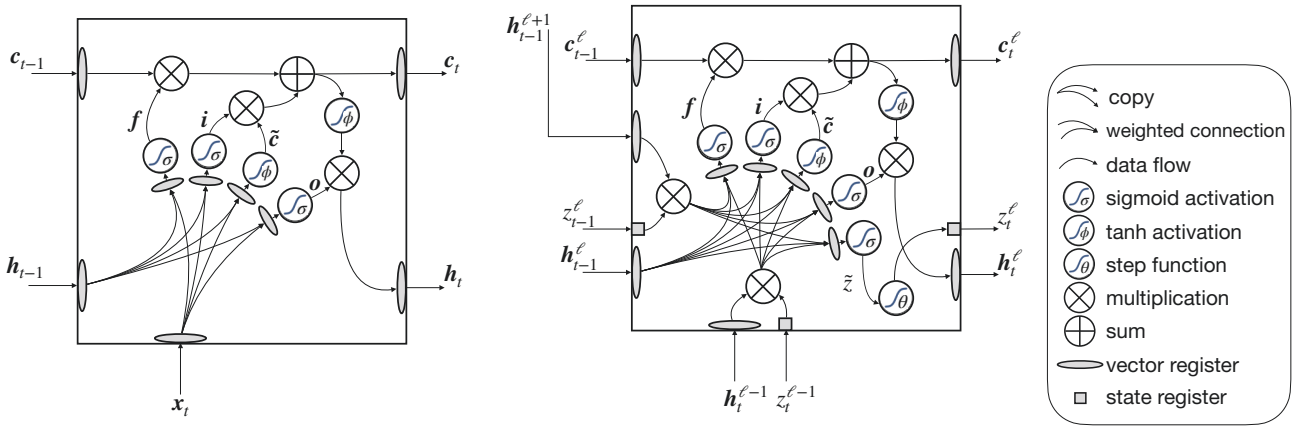
where $\boldsymbol{i}_t^\ell, \boldsymbol{f}_t^\ell, \boldsymbol{o}_t^\ell, \tilde{\boldsymbol{c}}_t^\ell \in \mathbb{R}^{N_h}$, $\tilde{z}_t^\ell \in \mathbb{R}$. σ is the logistic sigmoid function. ϕ is the hyperbolic tangent function. $\boldsymbol{W}_i^j (i > 0)$ is the linear transition matrix from layer i to layer j. $\boldsymbol{h}_t^\ell$ is the hidden output for layer ℓ at time t. For the top layer, we do not include the output from upper layers. For the bottom layer, $\boldsymbol{h}_t^0$ is the POI embeddings $\boldsymbol{x}_t$. The input boundaries $z_t^0 = 1$ make sure that all the inputs are valid.

Figure 3 compares the gating mechanism between LSTM and ASP. We can see that ASP integrates the information from upper and lower layers with a stacked RNN architecture. The gate values are generated the same as LSTM with the previous hidden state $\boldsymbol{h}_{t-1}^\ell$, the upper layer $\boldsymbol{h}_{t-1}^{\ell+1}$ and the lower layer $\boldsymbol{h}_t^{\ell-1}$ which are controlled by the boundary detectors z. The boundary detector z is learned through pre-activation $\tilde{z}$ and served as the outputs of this time step. We will show it in the follow parts.

Cell Update. The cell state $\boldsymbol{c}$ of ASP is updated according to the gates. In order to transfer the cell information in a stable way as well as to form the partition, we identify the recurrent generalization in different depths [Krueger *et al.*, 2017]. Specifically, the cell state updates in three different ways according to the combination of boundary detectors

$$\boldsymbol{c}_t^\ell = \begin{cases} \boldsymbol{f}_t^\ell \odot \boldsymbol{c}_{t-1}^\ell + \boldsymbol{i}_t^\ell \odot \tilde{\boldsymbol{c}}_t^\ell & \text{UPDATE, } z_{t-1}^\ell = 0 \wedge z_t^{\ell-1} = 1 \\ \boldsymbol{c}_{t-1}^\ell & \text{COPY, } z_{t-1}^\ell = 0 \wedge z_t^{\ell-1} = 0 \\ \boldsymbol{i}_t^\ell \odot \tilde{\boldsymbol{c}}_t^\ell & \text{FLUSH, } z_{t-1}^\ell = 1 \end{cases} \tag{3}$$

where $\odot$ denotes the element-wise product. UPDATE operation is the same as LSTM. The current cell state $\boldsymbol{c}_t$ is updated with the previous cell state $\boldsymbol{c}_{t-1}$ and cell proposal $\tilde{\boldsymbol{c}}_t$ after the trade-off from forget gate $\boldsymbol{f}_t$ and input gate $\boldsymbol{i}_t$ respectively. Following the idea of Krueger et al. [2017], we incorporate COPY operation that selects some RNN cells to hold as the previous time steps. However, rather than randomly selection to alleviate the issue of over-fitting, the COPY operation is learned to generate a more coalesce granularity. Furthermore, in order to learn the new geographical region the user begins to visit, we also generate the FLUSH operation to clear the cell state and only preserve the information of this time step. The FLUSH can start a new semantic subsequence.



(a) LSTM (b) ASP

Figure 3: Cell comparison between LSTM and ASP. We only show the UPDATE cell state of ASP for simplicity.

Then the final output of the cell $< \boldsymbol{h}_t^\ell, z_t^\ell >$ is obtained by

$$\boldsymbol{h}_t^\ell = \begin{cases} \boldsymbol{h}_{t-1}^\ell & \text{COPY} \\ \boldsymbol{o}_t^\ell \odot \phi(\boldsymbol{c}_t^\ell) & \text{otherwise} \end{cases} \qquad z_t^\ell = \begin{cases} 1 & \text{if } \tilde{z}_t^\ell > 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $\boldsymbol{h}_t^\ell$ gets updated the same as LSTM except when the cell state is COPY to transfer the state from the last time step. Boundary proposal $\tilde{z}_t^\ell$ is projected to $\{0, 1\}$ with step function in order to generate a binary boundary.

With ASP, the check-in sequence $\left[\boldsymbol{h}_1^\ell, ..., \boldsymbol{h}_T^\ell\right]$ is hierarchically projected to semantic spaces. Since the stacked layers in ASP capture different granularities, the sequence composition should be determined by all of them. Therefore we propose the layer-wise attention mechanism $\alpha_\ell^h = \sigma\left(\boldsymbol{v}_\ell^T \boldsymbol{h}_t^\ell\right)$ to get the combined output. The layer-wise attention learns the weights of each layer by aggregating all the time steps until now. So we can provide a uniform alignment for each time step by considering the overall effect of the entire sequence. Then we aggregate the hidden states $\tilde{\boldsymbol{h}}_t = \text{ReLU}\left(\boldsymbol{W}_h \sum_\ell \alpha_\ell^h \boldsymbol{h}_t^\ell\right)$ to get the output of each time step.

3.3 Power-law Attention Mechanism

We propose the Power-law Attention (PA) mechanism to integrate spatial and temporal contexts of each check-in into our model. We first introduce spatio-temporal bias

$$b'_{o,t} = f(v_o, v_t) = -\lambda \Delta_{o,t} - k \ln d_{o,t} \tag{5}$$

where $v_o, v_t \in \{(l_i, \tau_i)\}$ are the check-in points, $\Delta_{o,t} = \tau_o - \tau_t$ is the time interval, $d_{o,t} = D(l_o, l_t)$ is the geographical distance, λ and k are the time and distance decays, respectively. We also ensure that it has a negative effect on the bias. With this bias, the weight score is defined as

$$\text{score}_t(\tilde{\boldsymbol{h}}, v) = \boldsymbol{v}^T \phi(\boldsymbol{W}_a \tilde{\boldsymbol{h}}_t + \boldsymbol{b}) + b'_{o,t} \tag{6}$$

where $\tilde{\boldsymbol{h}}_t$ is the output state of ASP. Then we can calculate the attention vector

$$\alpha_t = \frac{\exp\left(\text{score}_t(\tilde{\boldsymbol{h}}, v)\right)}{\sum_i \exp\left(\text{score}_i(\tilde{\boldsymbol{h}}, v)\right)} \tag{7}$$

Now we consider the user's real-world check-in behavior. Previous studies indicate that the spatial items obey the First Law of Geography [Tobler, 1970], which shows the user's willingness-to-visit $\bar{w}$ on a pair of POIs. It can be formulated as a power-law distribution

$$\bar{w}_{ij} = a \cdot D(l_i, l_j)^k \tag{8}$$

where a and k are parameters of the power law, $D(l_i, l_j)$ is the geographical distance of POI l_i and l_j. It has been widely used in POI recommendation problems [Yuan *et al.*, 2013; Cho *et al.*, 2011].

Inspired by the First Law of Geography, we can extend the geographical distribution $\bar{w}$ to the *spatio-temporal willingness*

$$w_{o,t} = a_{o,t} \cdot d_{o,t}^{-k} \tag{9}$$

where $a_{o,t}$ is obtained by

$$a_{o,t} = g(\tilde{\boldsymbol{h}}_t, \Delta_{o,t}) = \exp\left(\boldsymbol{v}^T \phi(\boldsymbol{W}_a \tilde{\boldsymbol{h}}_t + \boldsymbol{b})\right) \cdot \exp(-\lambda \Delta_{o,t}) \tag{10}$$

This coefficient is the non-linear transformation of the user's current state $\tilde{\boldsymbol{h}}_t$ with the exponential decay towards time interval $\Delta_{o,t}$. In this way, the user's willingness incorporates both the spatial and temporal effects. It obeys the power law towards the distance, which is the same as the First Law of Geography. Furthermore, the coefficient considers the representation of current POI as well as the time interval.

With above settings, the attention weight α_t in Equation 7 is equivalent to the average spatio-temporal willingness $w_{o,t}$ because of the following formula transformation

$$\alpha_t = \frac{a_{o,t} \cdot d_{o,t}^{-k}}{\sum_i a_{o,i} \cdot d_{o,i}^{-k}} = \frac{w_{o,t}}{\sum_i w_{o,i}} \tag{11}$$

Then the dynamic representation of the user's check-in history can be represented by $\hat{\boldsymbol{h}} = \sum_{t=1}^T \alpha_t \tilde{\boldsymbol{h}}_t$, which shows that PA actually learns the user power-law willingness to visit the POIs under spatial and temporal contexts when it assigns the attention weights towards different time steps. This can effectively integrate the influence from different time steps.

The output layer of ASPPA consists of two fully connected layers and a drop-out layer. In order to control the negative

samples for each training case and optimize the ranking metric for downstream parameter learning, we solve the following objective function

$$\mathcal{J} = \sum_{Q \in \mathcal{Q}} \sum_{i=1}^{|\mathcal{L}|} \left(y_i \log \left(\Pr \left(\hat{y}_i = 1 \mid \mathcal{V}_u, u \right) \right) + (1 - y_i) \log \left(1 - \Pr \left(\hat{y}_i = 1 \mid \mathcal{V}_u, u \right) \right) \right) + \frac{\lambda'}{2} \|\Theta\|^2 \tag{12}$$

where $\mathcal{Q}$ is the query set, Θ denotes all the model parameters, λ' is the weight decay coefficient of the L2 norm regularization posed on all trainable parameters.

4 Experiments

4.1 Experiment Setup

Dataset

We evaluate our proposed techniques on two real-world LBSN datasets, namely Gowalla and Foursquare, which have been widely used by previous studies on next POI recommendation [Feng *et al.*, 2015; Zhao *et al.*, 2019]. **Gowalla**[1] dataset includes the world-wide check-in data from February 2009 to October 2010. **Foursquare**[2] dataset includes check-in data from April 2012 to September 2013 within the United States (except Alaska and Hawaii). We filter out those users with fewer than 10 check-in POIs and POIs with fewer than 10 visitors as previous studies [Feng *et al.*, 2015; Zhao *et al.*, 2019] did.

In order to simulate the real-world next POI recommendation scenario, we rank the check-in history of each user in ascending order of the time step and split the dataset $\mathcal{D}$ into $\mathcal{D}^{train} : \mathcal{D}^{validation} : \mathcal{D}^{test}$ as 8:1:1 according to previous studies [Liu *et al.*, 2016; Chang *et al.*, 2018].

Evaluation Metrics

To evaluate the performance of ASPPA, we employ $Hits@k$ and MRR. For each test case, we recommend the POIs with top-k highest scores among all possible candidates in $\mathcal{L}$. If the POI in the recommended list is actually visited by the user, the value of $hit@k$ of a test case is 1; otherwise it is 0. The overall $Hits@k$ is defined as $Hits@k = \frac{\#hit@k}{||\mathcal{D}^{test}||}$ where $\#hit@k$ is the number of hits in the test set and $||\mathcal{D}^{test}||$ denotes the number of test cases. MRR is the average reciprocal rank of positive examples. This metric reflects the overall ranking ability of the model.

Comparison Methods

We compare our ASPPA with following methods for next POI recommendation.

- MF: Most Frequented Location Model is a statistical based model. It calculates the POI distribution of a user based on the statistics of her previous check-ins. It captures the cyclical check-in habit of the user. This is a widely-adopted baseline method in some real-world LBSN services [Cho *et al.*, 2011].
- PRME: Personalized Ranking Metric Embedding [Feng *et al.*, 2015] develops a pair-wise metric embedding method to capture the interaction between the candidate and previous POIs.
- PRME-G: PRME-G [Feng *et al.*, 2015] embeds users and POIs into the geographical context space. It extends the metric similarity scores under the real-world spatial influence.
- LSTM: Long Short-Term Memory [Hochreiter and Schmidhuber, 1997] is a widely used neural network to predict future behavioral trajectories. Compared with conventional RNN model, LSTM models the short-term and long-term sequential patterns by the gating mechanisms.
- Time-LSTM: Time-LSTM [Zhu *et al.*, 2017] is a state-of-the-art variant of LSTM model in recommender systems. The authors develop three versions of Time-LSTM in the paper and we adopt the third version because it achieves the best performance on our datasets. Time-LSTM improves the modeling of sequential patterns by explicitly capturing the multiple time structures in the check-in sequence.
- ST-RNN: Spatial Temporal Recurrent Neural Network [Liu *et al.*, 2016] adopts RNN to model the user's previous check-in sequence. It captures spatial and temporal contexts with time and distance transition matrix.
- STGN: Spatio-temporal Gated Network [Zhao *et al.*, 2019] achieves the state-of-the-art performance in the problem of next POI recommendation. It extends the LSTM gating mechanism with the spatial and temporal gates to capture the user's space and time preference.

Implementation and Training

In ASPPA, we adopt three ASP layers with the dimension of hidden states N_h in order to keep balance of the model effectiveness and efficiency. For consistency, we apply the dimension N_d of the output vector and the dimension N_a of the attention projection vector the same as N_h. The dimension of POI embeddings is N_e. We also embed the weekday $w \in [0, 6]$ and hour $h \in [0, 23]$ of the query time and concatenate with the user's check-in history $\hat{\boldsymbol{h}}$. Besides, in order to determine the time and distance decay λ and k, we apply the grid search over $\{0.0001, 0.001, 0.01, 0.1, 1\}$. The following experimental results of ASPPA are based on the best tuning performance on the validation set ($\lambda = k = 0.001$, $N_e = 128$, $N_h = 256$, learning rate 0.01 and dropout 0.5 for both datasets).

In the training procedure, we leverage the mini-batch SGD algorithm. Specifically, we set the batch size to 1024 according to device capacity. Note that the step function of z_t^ℓ in Equation 4 is not differentiable, we manually apply the unchanged gradient when calculating gradients. We adopt Xavier initialization to initialize the weights of our network. For the implementation of comparison methods, we obtain the source code from the authors and report the best results after tuning all the hyper-parameters on the validation set.

4.2 Results and Discussion

Overall Performance

We evaluate the top-k recommendation performance ($Hits@k$) and overall ranking ability (MRR) in Table 1. All differences between our model and others are statistically significant ($p < 0.01$). We can see that our model consistently and significantly outperforms all compared models. Take Gowalla dataset as an example. When we score all the 32510 POIs and recommend the highest scored POI to

[1] http://snap.stanford.edu/data/loc-gowalla.html

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

Datasets	Metrics	MF	PRME	PRME-G	LSTM	Time-LSTM	ST-RNN	STGN	**ASPPA**
Gowalla	Hit@1	7.19	8.23	8.68	10.73	10.98	10.51	10.95	**12.12**
	Hit@5	11.63	12.33	13.13	15.39	17.06	18.57	19.47	**23.63**
	Hit@10	12.59	16.07	16.56	19.87	19.90	22.40	23.67	**29.47**
	Hit@20	13.05	17.60	17.97	22.59	25.38	25.17	29.15	**35.73**
	MRR	9.12	10.41	10.97	12.97	13.66	14.37	15.47	**18.07**
Foursquare	Hit@1	9.55	12.43	12.89	10.84	13.67	13.20	13.98	**15.10**
	Hit@5	13.90	14.94	15.17	14.48	19.34	18.69	20.00	**27.96**
	Hit@10	14.34	19.53	20.17	20.64	23.08	23.89	25.92	**33.71**
	Hit@20	14.44	20.56	22.94	23.68	25.47	28.26	29.39	**39.50**
	MRR	11.01	14.06	14.52	13.01	16.05	16.23	17.10	**21.45**

Table 1: Recommendation results w.r.t. Gowalla and Foursquare datasets. The results are reported in percentage (%).

the user, 12.12% of the test cases in $\mathcal{D}^{test}$ are predicted correctly. When we recommend the top-20 highest scored POIs, 35.73% of the test cases have the user's actually visited POI in the recommended list. In addition, we have the following observations based on the experimental results.

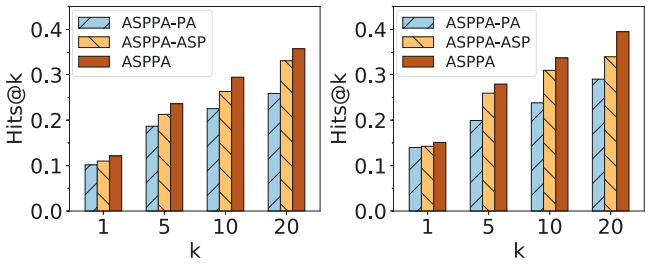
Firstly, the excess performance of MF compared with other baselines shows the prediction ability beyond periodic habit. ASPPA makes significant improvement compared with MF as k increases. This shows that ASPPA can capture more subtle temporal dynamics to make a longer candidate list while the periodical guess of MF only works for the most frequent check-ins.

Secondly, the overall performance of all models on Foursquare is slightly better than that on Gowalla. The reason is that locations in Foursquare are limited in the United States. So the underlying movement patterns for users in Foursquare datasets are more uniform, where MF also shows a better periodic performance in Foursquare. Even Foursquare dataset is more sparse than Gowalla with more users but less user-POI interactions, each model still benefits from the uniform pattern in a limited manner. The superiority of our proposed ASPPA is more obvious in Foursquare, showing that ASPPA can learn more from sparse data compared with other alternatives. This is because ASPPA can capture the semantic subsequence patterns and incorporates the domain knowledge in both spatial and temporal contexts.

Thirdly, the RNN based models (LSTM, Time-LSTM, ST-RNN, STGN, ASPPA) achieve better performance than the metric embedding ones. The sequential recommendation can not only capture the neighboring constraints as the metric embedding models but also learn the user's sequential patterns. This would lead to better potential performance gain when a large number of sequential user interactions are available in real world scenarios. Since learning from the user's sequential patterns is the key to improve the sequential recommendation, ASPPA recognizes the latent structure to identify the sequential pattern among semantic subsequences beyond POI-level sequential patterns and further improves the overall performance.

Ablation Analysis

We conduct ablation tests with two variants that remove one component at a time. The first variant, ASPPA-PA removes the hierarchical framework and boundary detectors of each cell, which is equivalent to the power-law attention based LSTM model. The second variant, ASPPA-ASP replaces the power-law attention with vanilla attention by regarding the spatio-temporal bias $b'_{o,t}$ in Equation 6 as the normal bias variable. Due to the space limitation, we only show the $Hit@k$ on both datasets. And regarding MRR we achieve results with similar trends. Figure 4 shows that the performance gain on Gowalla dataset of adopting subsequence information is more than 25% for all the metrics and while that in Foursquare is 35%. It demonstrates that recognizing semantic subsequences with multi-granularity latent structure is necessary for improving next POI recommendation.



(a) Gowalla (b) Foursquare

Figure 4: Hits@k on different parts of our model

5 Conclusion

In this work, we propose a novel model ASPPA for the problem of next POI recommendation. We adopt a stacked recurrent neural network framework Adaptive Sequence Partitioner to identify the sequential patterns among semantic subsequences by automatically learning the latent structure in the user's check-in sequence. To alleviate the data sparsity, we propose the Power-law Attention mechanism to integrate the domain knowledge in both spatial and temporal contexts. The experimental results on two real-world datasets show that our proposed framework is effective and significantly outperforms state-of-the-art methods.

Acknowledgments

This work was supported by NSFC (91646202, 61972069, 61836007, 61832017, 61532018), National Key R&D Program of China (2018YFB1404401,2018YFB1402701), ARC Discovery Project (Grant No. DP190101985 and DP170103954).

References

[Blei *et al.*, 2003] David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *NIPS*, pages 17–24. MIT Press, 2003.

[Chang *et al.*, 2018] Buru Chang, Yonggyu Park, Donghyeon Park, Seongsoon Kim, and Jaewoo Kang. Content-aware hierarchical point-of-interest embedding model for successive POI recommendation. In *IJCAI*, pages 3301–3307, 2018.

[Cheng *et al.*, 2013] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, pages 2605–2611, 2013.

[Cho *et al.*, 2011] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD*, pages 1082–1090, 2011.

[Chung *et al.*, 2017] Junyoung Chung, Sungjin Ahn, and Yoshua Bengio. Hierarchical multiscale recurrent neural networks. In *ICLR*, 2017.

[Feng *et al.*, 2015] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. Personalized ranking metric embedding for next new POI recommendation. In *IJCAI*, pages 2069–2075, 2015.

[He *et al.*, 2016] Jing He, Xin Li, Lejian Liao, Dandan Song, and William K. Cheung. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In *AAAI*, pages 137–143, 2016.

[Hihi and Bengio, 1995] Salah El Hihi and Yoshua Bengio. Hierarchical recurrent neural networks for long-term dependencies. In *NIPS*, pages 493–499, 1995.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[Kong and Wu, 2018] Dejiang Kong and Fei Wu. HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction. In *IJCAI*, pages 2341–2347, 2018.

[Krueger *et al.*, 2017] David Krueger, Tegan Maharaj, János Kramár, Mohammad Pezeshki, Nicolas Ballas, Nan Rosemary Ke, Anirudh Goyal, Yoshua Bengio, Hugo Larochelle, Aaron C. Courville, and Chris Pal. Zoneout: Regularizing rnns by randomly preserving hidden activations. In *ICLR*, 2017.

[Liu *et al.*, 2016] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*, pages 194–200, 2016.

[Liu *et al.*, 2017] Yiding Liu, Tuan-Anh Pham, Gao Cong, and Quan Yuan. An experimental evaluation of point-of-interest recommendation in location-based social networks. *PVLDB*, 10(10):1010–1021, 2017.

[Lv *et al.*, 2018] Zhongjian Lv, Jiajie Xu, Kai Zheng, Hongzhi Yin, Pengpeng Zhao, and Xiaofang Zhou. LC-RNN: A deep learning model for traffic speed prediction. In *IJCAI*, pages 3470–3476, 2018.

[Tobler, 1970] Waldo R Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, 46(sup1):234–240, 1970.

[Wang *et al.*, 2016] Weiqing Wang, Hongzhi Yin, Shazia Wasim Sadiq, Ling Chen, Min Xie, and Xiaofang Zhou. SPORE: A sequential personalized spatial item recommender system. In *ICDE*, pages 954–965, 2016.

[Wang *et al.*, 2018] Weiqing Wang, Hongzhi Yin, Xingzhong Du, Quoc Viet Hung Nguyen, and Xiaofang Zhou. TPM: A temporal personalized model for spatial item recommendation. *ACM TIST*, 9(6):61:1–61:25, 2018.

[Wang *et al.*, 2020] Qinyong Wang, Hongzhi Yin, Tong Chen, Zi Huang, Hao Wang, Yanchang Zhao, and Nguyen Quoc Viet Hung. Next point-of-interest recommendation on resource-constrained mobile devices. In *WWW*, page 906–916. ACM, 2020.

[Yang *et al.*, 2017] Carl Yang, Lanxiao Bai, Chao Zhang, Quan Yuan, and Jiawei Han. Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation. In *KDD*, pages 1245–1254, 2017.

[Yin *et al.*, 2016] Hongzhi Yin, Bin Cui, Xiaofang Zhou, Weiqing Wang, Zi Huang, and Shazia W. Sadiq. Joint modeling of user check-in behaviors for real-time point-of-interest recommendation. *ACM Trans. Inf. Syst.*, 35(2):11:1–11:44, 2016.

[Yin *et al.*, 2017] Hongzhi Yin, Weiqing Wang, Hao Wang, Ling Chen, and Xiaofang Zhou. Spatial-aware hierarchical collaborative deep learning for POI recommendation. *IEEE Trans. Knowl. Data Eng.*, 29(11):2537–2551, 2017.

[Yuan *et al.*, 2013] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372, 2013.

[Zhang and Wang, 2015] Wei Zhang and Jianyong Wang. Location and time aware social collaborative retrieval for new successive point-of-interest recommendation. In *CIKM*, pages 1221–1230, 2015.

[Zhao *et al.*, 2016] Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael R. Lyu, and Irwin King. STELLAR: spatial-temporal latent ranking for successive point-of-interest recommendation. In *AAAI*, pages 315–322, 2016.

[Zhao *et al.*, 2019] Pengpeng Zhao, Haifeng Zhu, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S. Sheng, and Xiaofang Zhou. Where to go next: A spatio-temporal gated network for next POI recommendation. In *AAAI*, pages 5877–5884, 2019.

[Zhu *et al.*, 2017] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. What to do next: Modeling user behaviors by time-lstm. In *IJCAI*, pages 3602–3608, 2017.