Predicting Activity and Location with Multi-task Context Aware Recurrent Neural Network

Dongliang Liao[1], **Weiqing Liu**[2], **Yuan Zhong**[3], **Jing Li**[1], **Guowei Wang**[1]
[1] University of Science and Technology of China
[2] Microsoft Research Asia ,
[3] Facebook Inc.
liaodl@mail.ustc.edu.cn, weiqing.liu@microsoft.com, yzhong@fb.com,
lj@ustc.edu.cn, weiking@mail.ustc.edu.cn

Abstract

Predicting users' activity and location preferences is of great significance in location based services. Considering that users' activity and location preferences interplay with each other, many scholars tried to figure out the relation between users' activities and locations for improving prediction performance. However, most previous works enforce a rigid human-defined modeling strategy to capture these two factors, either activity purpose controlling location preference or spatial region determining activity preference. Unlike existing methods, we introduce *spatial-activity topics* as the latent factor capturing both users' activity and location preferences. We propose Multi-task Context Aware Recurrent Neural Network to leverage the spatial-activity topic for activity and location prediction. More specifically, a novel Context Aware Recurrent Unit is designed to integrate the sequential dependency and temporal regularity of spatial activity topics. Extensive experimental results demonstrate that the proposed model significantly outperforms state-of-the-art approaches.

1 Introduction

Nowadays, location based services (LBS) have become indispensable in our daily life. The user mobility prediction, as an important role in intelligence and personalization of LBS, sheds light on POI recommendation, travel planning and targeted advertising. Besides, it is also academic valuable and industrial applicable to properly answer the question like "How to utilize large volume mobility data collected in LBS for user activity and location prediction?".

Considering that users' activity and location preferences interplay with each other, many scholars tried to figure out the relationship between users' activities and locations for improving prediction performance. Some researchers believe that users' activity purpose determines where they go [Ye *et al.*, 2013; Liao *et al.*, 2017], so that they firstly modeled users' activities and then predicted locations given activities. Other scholars hold the opposite opinion that users' activities are influenced by where users stay [Yuan *et al.*, 2013; Yang *et al.*, 2015], thus their method inferred users' activity preferences based on users' spatial information.

However, the relationship between user's activities and locations is more complicated than that one decides the other. Actually, people usually begin with a rough idea about what activities and which regions they prefer. After that, activities and spatial regions become interrelated factors for making a decision. For example, Alice wants relaxation near home after a whole day work, so she would prefer a leisure place in the neighborhood. Bob plans to have a party with friends in popular recreation centers on weekends, so he may choose a downtown restaurant or hub. From these inspirations, we introduce the *spatial-activity topic* to describe the interplay of users' activity and location preferences. We believe that users' activity and location preferences are decided by their latent spatial-activity topics. In the activity aspect, the topic represents high-level abstract of the activity purpose, such as work or relaxation. In the spatial aspect, the topic controls which regions users are willing to go. In above examples, "relaxation near home" and "party in recreation centers" are Alice and Bob's spatial-activity topics. What's more, two comprehensive principles of individual mobility [Liao *et al.*, 2017] potentially apply to spatial-activity topics:

- Sequential dependency: Users' spatial-activity topics depend on previous topics. For example, a whole-day work usually follows leisure and recreation topics, rather than any laborious topics. However, the sequential dependency could be vanished, due to the check-in data sparsity [Feng *et al.*, 2015].
- Temporal regularity: Users' spatial-activity topics are influenced by the specific temporal context. For example, party topics are usually performed at weekends instead of work hours.

From these two principles, we develop our modeling techniques. For modeling sequential dependency, gated Recurrent Neural Networks (RNN), such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU), have achieved the best performance in many sequence model applications [Yao *et al.*, 2013; Graves *et al.*, 2013]. However, there are some technical challenges for integrating temporal context into RNNs. When check-ins are very sporadic and sparse, the sequential feature should be "forgotten" due to vanishing of sequential dependency while the temporal context should play an active role for prediction. Thus it is infeasible to feed sequential and temporal contexts together into gated RNN and control them with a single sigmoid gate.

Motivated by these findings, we propose a Multi-task Context Aware Recurrent Neural Network (MCARNN) to leverage the spatial-activity topic for improving the activity and location prediction. As the activity and location prediction share the same inputs and are both influenced by spatial-activity topics, we adopt a multi-task learning neural network to predict users' activities and locations simultaneously. To integrate the context information and sequential pattern, and elevate sequential and temporal regularity of spatial-activity topics, we propose a novel Context Aware Recurrent Unit (CARU) as hidden layer unit. CARU calculates the sequential hidden state to capture the sequential dependency and takes the temporal context as an extra input. After a non-linear activation function, the temporal context is integrated with sequential hidden state dynamically by a sigmoid gate. Through evaluation on real-world public datasets, the proposed model shows a considerable improvement of prediction performance, demonstrating the validity and superiority of spatial-activity topics and CARU.

Our main contributions are summarized as follows:

- We introduce the spatial-activity topic to capture the interplay of user's activity and location preferences. In addition, a multi-task neural network utilizing the spatial-activity topic is leveraged to improve the activity and location prediction.
- We propose a novel Context Aware Recurrent Unit to integrate context information and sequential pattern dynamically.
- Experimental results demonstrate that proposed model significantly outperforms state-of-the-art methods. Besides, a case study vividly explain the learned spatial activity topics.

2 Related Work

2.1 Mobility Prediction

Human mobility prediction has drawn great attentions for decades. Scholars handled this task with two-broad-category approaches: temporal context model and sequential model. The temporal context model focused on users' behavior under a certain time information. A time-aware Gaussian Mixture model was proposed to combine the periodic behavior and social relationship for mobility prediction [Cho *et al.*, 2011]. RCH (Regularity, Conformity and Heterogeneous) model [Wang *et al.*, 2015] predicted locations at specific time, considering both regularity and conformity. As for sequential modeling, the fallback Markov achieved the best performance on location prediction of WiFi mobility data [Song *et al.*, 2004]. Recently, rather than merely modeling sequential pattern, many extensions have been developed for exploiting additional information, such as Factorizing Personalized Markov Chain [Cheng *et al.*, 2013] and Personalized Ranking Metric Embedding [Feng *et al.*, 2015]. However, the user activity preference, an important part of their mobility that affecting location preference, was ignored in above works.

Viewing that users' activity and location preferences interplay with each other, more and more scholars leveraged both activity and location information of check-ins for mobility prediction. Some researchers believed that activity purpose controls location preference, so that they modeled users' activities at first and then predicted locations given specific activities [Ye *et al.*, 2013; Liao *et al.*, 2017]. However, this kind of assumption is simplistic to model the interplay of user's activity and location preference. W^4 (Who, When, What and Where) model [Yuan *et al.*, 2013] and Periodic & Social Mobility Model (PSMM) [Cho *et al.*, 2011] introduced "spatial temporal topic" and user's "latent state" respectively. In PSMM, user's "latent states" are spatial regions without regard to the activity information. In W^4 model, the "spatial temporal topic" represents user's activity preference under specific spatial and temporal contexts. In contrast, our proposed spatial-activity topic involves both activity and spatial aspects, which is a more reasonable way to capture the complex relationship between user's activities and locations.

2.2 Recurrent Neural Networks

Recurrent Neural Networks have achieved great success in many sequential applications, such as natural language processing [Yao *et al.*, 2013], speech recognition [Graves *et al.*, 2013], and GPS trajectories modeling [Wu *et al.*, 2017]. Most RNNs are designed for the fine-gained sequential data and assumes that elements of nearby time steps are correlated, which are not applicable for the sporadic and sparse check-in sequence. JNTM (Joint Network and Trajectory Model) [Yang *et al.*, 2017b] and ST-RNN (Spatial Temporal RNN) [Liu *et al.*, 2016] applied RNNs as mobility models on check-in data. These two models ignored the impact of the temporal context and failed to integrate activity information. SERM (Semantics-Enriched Recurrent Models) [Yao *et al.*, 2017] captured temporal regularity, activity semantics and user preference in a unified RNN model. SERM took sequential context (i.e. user's last location and acitivity) and temporal information together into LSTM and control them with a single input gate. However, when check-ins are very sporadic and sparse, sequential and temporal contexts should be controlled separately. Thus SERM is not a feasible method for modeling the spatial-activity topic.

3 Problem Formulation

Our objective is to predict future locations and activities of users. Without loss of generality, we normalize the activity information into the keyword representation. Given the activity keyword set $\mathcal{A} = \{a^{(1)}, a^{(2)}, \ldots, a^{(|\mathcal{A}|)}\}$, user set $\mathcal{U} = \{u^{(1)}, u^{(2)}, \ldots, u^{(|\mathcal{U}|)}\}$ and location set $\mathcal{L} = \{l^{(1)}, l^{(2)}, \ldots, l^{(|\mathcal{L}|)}\}$, the check-in data can be defined as a quadruple $r = (u, l, a, t)$, indicating that user u visits location l for activity a at time t. Here, for the ease of calculation, we discretize continuous time t to hour of day and day of week. With these notations, our problem can then be formulated as: Our goal is to predict user u's next location l_n

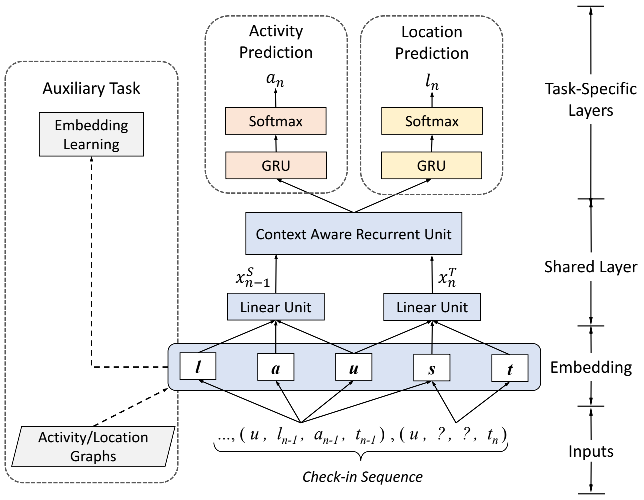


Figure 1: The overall framework of MCARNN

and next activity a_n, given the next check-in time t_n and the historical check-in sequence $\tau_{n-1}^u = \{r_1^u, r_2^u, \ldots, r_{n-1}^u\}$.

4 Model

4.1 Overview

In this section, we introduce the Multi-task Context Aware Recurrent Neural Network (MCARNN). The overall framework is presented in Fig. 1. We employ a multi-task learning framework to capture the spatial-activity topic in shared hidden layers for activity and location prediction.

Considering the sequential dependency and temporal regularity of spatial-activity topics, we extract five features from check-in sequence: the user u, the latest activity a_{n-1}, the latest location l_{n-1}, the current time t_n and the timespan s_n. Here, timespans are measured by hours with a threshold. If the timespan of two check-ins is larger than the threshold, we think these check-ins have no sequential dependency and denote this timespan as "infinity". Then, all these discrete features are mapped into dense continuous vectors as *embedding*, denoted as $\boldsymbol{u}, \boldsymbol{a}_{n-1}, \boldsymbol{l}_{n-1}, \boldsymbol{t}_n, \boldsymbol{s}_n$.

Based on these features, we adopt gated RNN as the shared hidden layer for modeling spatial-activity topics. As we have discussed, it is infeasible to take all five features together as the input of existing gated RNNs. Thus we decompose input features into sequential features $\boldsymbol{x}_{n-1}^S$ (linear combination of $\boldsymbol{u}, \boldsymbol{a}_{n-1}$ and $\boldsymbol{l}_{n-1}$) and temporal context $\boldsymbol{x}_n^T$ (linear combination of $\boldsymbol{u}, \boldsymbol{t}_n$ and $\boldsymbol{s}_n$). Then we propose a novel Context Aware Recurrent Unit (CARU) to integrate sequential feature $\boldsymbol{x}_{n-1}^S$ and temporal context $\boldsymbol{x}_n^T$ dynamically. Details of CARU are shown in Sec. 4.2.

With the CARU layer, we capture the spatial-activity topic, which is then fed into task-specific layers. After task-specific GRU and softmax layers, we get the activity prediction and location prediction results, i.e. a_n and l_n. In addition, to exploit the geo-distance of locations and the correspondence of locations and activity keywords, we employ graph embedding methods [Perozzi *et al.*, 2014; Tang *et al.*, 2015] to learn better embedding of locations and activity keywords. Embedding learning benefits the forward calculation and backward derivation of our model, so that we adopt it as a auxiliary task of the activity and location prediction. Details of the embedding learning will be discussed in Sec.4.3.

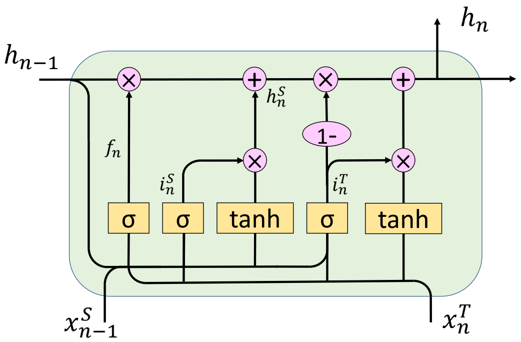


Figure 2: Context Aware Recurrent Unit

4.2 Context Aware Recurrent Unit

As mentioned before, the spatial-activity topics are influenced by not only the sequential feature but also the temporal context. What's more, the sequential feature and temporal context cannot be controlled by a single sigmoid gate. For example, when timespans of consecutive check-ins are as long as several days, the sequential feature should be "forgotten" due to decayed sequential dependency, while the temporal context should play an active role for prediction. Therefore, we propose the Context Aware Recurrent Unit (shown in Fig. 2) to integrate temporal and sequential contexts dynamically.

Specifically, to capture the sequential pattern, we calculate the *sequential hidden state* as follows:

$$i_n^S = \sigma(W^S x_{n-1}^S + U^S x_n^T + V^S h_{n-1} + b^S) \tag{1}$$

$$f_n = \sigma(W^f x_{n-1}^S + U^f x_n^T + V^f h_{n-1} + b^f) \tag{2}$$

$$h_n^S = f_n * h_{n-1} + i_n^S * g(W^h x_{n-1}^S + V^h h_{n-1} + b^h) \tag{3}$$

where n denotes the time step, h is the hidden state of CARU, h^S is called as sequential hidden state, and i, f are sigmoid gates. Here, W, U, V and b represent the weight matrices and bias vectors, $*$ represents the element-wise product of two vectors, $\sigma(\cdot)$ denotes the sigmoid function and $g(\cdot)$ is the activation function (the hyperbolic tangent function in our work). The calculation of the sequential hidden state is similar to the calculation of memory state of LSTM cell. Here, the major difficulty in our problem is not the long-term dependency but the vanishing of sequential dependency. Instead of the output gate in LSTM, we integrate the sequential hidden state and temporal context via a sigmoid gate i_n^T as follows:

$$i_n^T = \sigma(W^T x_{n-1}^S + U^T x_n^T + V^T h_{n-1} + b^T) \tag{4}$$

$$h_n = i_n^T * g(x_n^T) + (1 - i_n^T) * h_n^S \tag{5}$$

The sigmoid unit i_n^T sets the weight of the sequential hidden state and the temporal context to a value between 0 and 1. Especially, the proposed gated RNN cell degenerates to a simplified LSTM cell when $i_n^T = 0$, or a common non-linear neuron with temporal context as the input when $i_n^T = 1$. Note that when calculating spatial activity topic with time step n, the sequential feature x^S only involves time step $n-1$, while the temporal context x^T depends on time step n. That is why we call x^T as "context". Generally, the context can be any information associated with time step n.

4.3 Embedding Learning

Representing similar items as similar embedding vectors is conducive to the forward calculation and the backward derivation of neural network [Mikolov *et al.*, 2013]. To learn the location embedding, we leverage both the geo-spatial distance and activity semantic of locations to represent the location similarity. We build a location-location graph G_l. Two locations are connected if they share same activity keywords or they are closer than a distance threshold. Then we employ the graph embedding method [Perozzi *et al.*, 2014], which generates "sentences" by random walk on the graph and applies Skipgram [Mikolov *et al.*, 2013] to learn embedding.

The similarity of activity keywords is difficult to define. Instead, we map activity keywords to correlative embedding vectors with corresponding locations in the same vector space, which is beneficial to learning shared parameters. We construct a bipartite activity-location graph G_a with the corresponding of activity and location as edges. A bipartite network embedding method [Tang *et al.*, 2015] is adopted for learning activity embedding. This method randomly selects a activity keyword as "input word", and locations linked to this activity as "context" in each step for implementing Skipgram.

5 Alternative Training

The overall loss of MCARNN is the total loss of the activity prediction, location prediction and embedding learning.

$$J = \lambda_1 J_a + \lambda_2 J_l + \lambda_3 (J_{G_l} + J_{G_a}) \tag{6}$$

where J is the overall loss, J_a and J_l are the losses of activity and location prediction, J_{G_l} and J_{G_a} are the losses of the auxiliary task, and $\lambda_1, \lambda_2, \lambda_3$ are the weighting factors. For activity and location prediction tasks, the training goal is minimizing the divergence between the predicted distribution and the true distribution. Thus we adopt J_a and J_l as the cross-entropy loss. Then we can minimize J_a and J_l by mini-batch Stochastic Gradient Descent (SGD). As for the auxiliary embedding learning task, we follow the training process of the article [Yang *et al.*, 2017a], defining the loss of Skipgram models with the negative sampling [Mikolov *et al.*, 2013] and minimizing it with Adam [Kingma and Ba, 2014].

Because the input, loss function and optimizer of the auxiliary task are different with those of major tasks, we perform an alternative training algorithm shown in Alg. 1. The weighting factors λ_1, λ_2 and λ_3 are approximated by sampling one loss function to update from a categorical distribution with probability p_1, p_2 and p_3 in each step, where $p_i = \lambda_i / \sum_j \lambda_j$. When optimizing J_a and J_l, we generate a batch of check-in sequences and adopt the SGD with decayed learning rate α. When optimizing J_{G_l} and J_{G_a}, we sample a batch of (w, c) for Skipgram model from location-location graph and activity-location graph, and update parameters with Adam algorithm. Here, w and c are both locations in location-location graph G_l, while w is activity and c is location in activity-location graph G_a.

Algorithm 1 Alternative Training of MCARNN

```
Input: $B_a, B_l, B_G, p_1, p_2, p_3, \alpha$
repeat
    Sample $op$ from Cat$\{a : p_1, l : p_2, \text{G} : p_3\}$
    if $op = a$ then
        Generate a check-in batch $\mathcal{B}_a$ with size $B_a$
        for $\vartheta$ in $\{\theta^a, \theta, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{l}, \boldsymbol{t}, \boldsymbol{s}\}$ do
            $\vartheta \leftarrow \vartheta - \alpha * \frac{\partial J_a}{\partial \vartheta}$
        end for
    else if $op = l$ then
        Generate a check-in batch $\mathcal{B}_l$ of with size $B_l$
        for $\vartheta$ in $\{\theta^l, \theta, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{l}, \boldsymbol{t}, \boldsymbol{s}\}$ do
            $\vartheta \leftarrow \vartheta - \alpha * \frac{\partial J_l}{\partial \vartheta}$
        end for
    else
        Sample a batch $\mathcal{B}_G$ for Skipgram with size $B_G$
        for $\vartheta$ in $\{\boldsymbol{a}, \boldsymbol{l}, \phi^N\}$ do
            Compute $\Delta\vartheta$ by Adam step
            $\vartheta \leftarrow \vartheta + \Delta\vartheta$
        end for
    end if
until $J$ converges or is sufficiently small
```

	#user	#location	#check-in	#location per user	#category per user
NYC	1,083	38,333	227,420	84.04	40.22
TKY	2,293	61,858	573,703	92.43	32.40

Table 1: Datasets Statistics

6 Experiments

6.1 Experiment Settings

Datasets

We evaluate our model on public Foursquare check-in datasets collected from two big cities, New York (NYC) and Tokyo (TKY) [Yang *et al.*, 2015]. The check-ins last from April 2012 to February 2013. We take the 400 categories of POIs in Foursquare[1] as activity keywords. The overall statistics is shown in Table 1. In following experiments, for each user, we take the first 80% check-ins as the training set, the latter 10% as the evaluation set, and the last 10% as the validation set for the hyper-parameters study.

Metrics

We adopt three metrics to evaluate the performance of our model. Accuracy of top K (Acc@K) is the percentage of accurate predictions for a list of predictions with length K. Mean Average Precision of top K (MAP@K) is a widely used global evaluation for ranking task. Negative Log-Likelihood (NLL) measures the likelihood between predicted distribution and the true distribution. To make results more convincing, we repeat each experiment 10 times and take the average value of metrics into comparison.

Baselines

We compare the MCARNN model with following baselines:

[1] http://developer.foursquare.com/categorytree

		Location Prediction					Activity Prediction				
Dataset	Method	Acc@1	Acc@10	Acc@20	NLL	MAP@100	Acc@1	Acc@3	Acc@5	NLL	MAP@20
NYC	MF	0.0774	0.1788	0.1957	8.3399	0.4005	0.1739	0.3008	0.3980	4.0183	0.6393
	CAH	0.1328	0.3521	0.4179	7.0680	0.5331	0.2338	0.3825	0.4481	3.7882	0.6716
	ST-RNN	0.1663	0.4126	0.4671	6.7701	0.5632	-	-	-	-	-
	SERM	0.1480	0.3731	0.4386	7.0104	0.5395	0.2240	0.3905	0.4624	3.7406	0.6788
	MTGRU	0.1640	0.4128	0.4688	6.6957	0.5722	0.2310	0.3912	0.4665	3.7134	0.6925
	SCARNN	0.1743	0.4268	0.4911	6.5869	0.5836	0.2393	0.3847	0.4697	3.7239	0.6859
	MCARNN	**0.2011**	**0.4601**	**0.5198**	**6.2961**	**0.6092**	**0.2572**	**0.4217**	**0.5032**	**3.5740**	**0.7058**
TKY	MF	0.1021	0.2041	0.2536	6.6641	0.6018 1	0.2322	0.3672	0.4345	3.6191	0.6871
	CAH	0.1728	0.3877	0.4551	5.9137	0.6662	0.3455	0.5006	0.5690	3.0173	0.7511
	ST-RNN	0.2033	0.4904	0.5703	5.5774	0.7033	-	-	-	-	-
	SERM	0.1796	0.4271	0.4862	5.8377	0.6726	0.3863	0.5840	0.6730	2.6706	0.7892
	MTGRU	0.2000	0.4948	0.5619	5.6804	0.6823	0.4047	0.6062	0.6786	2.5937	0.8129
	SCARNN	0.2431	0.5289	0.6037	5.3974	0.7035	0.4463	0.5709	0.6347	2.5031	0.8248
	MCARNN	**0.2829**	**0.5807**	**0.6391**	**5.0057**	**0.7425**	**0.4706**	**0.6599**	**0.7332**	**2.3320**	**0.8739**

Table 2: Performance comparison with baselines

- **MF**. Most Frequent method assigns the most frequent activity and location of user u at time t as prediction.
- **CAH**. Context-Aware Hybrid model [Liao *et al.*, 2017] predicts activities firstly and then predicts locations given activities.
- **ST-RNN**. Spatial Temporal RNN [Liu *et al.*, 2016] is a RNN model for location prediction based on check-in data without regard to user's activity preference.
- **SERM**. Semantics-Enriched Recurrent Models [Yao *et al.*, 2017] is a LSTM model taking activity semantics and spatial-temporal context as inputs for location prediction.
- **MTGRU**. Multi-Task GRU replaces the proposed CARU with GRU as the shared hidden layer for modeling spatial-activity topics.
- **SCARNN**. Single-Task Context Aware RNNs predict activities or locations separately with the same inputs and hidden layers as proposed model.

Parameter settings

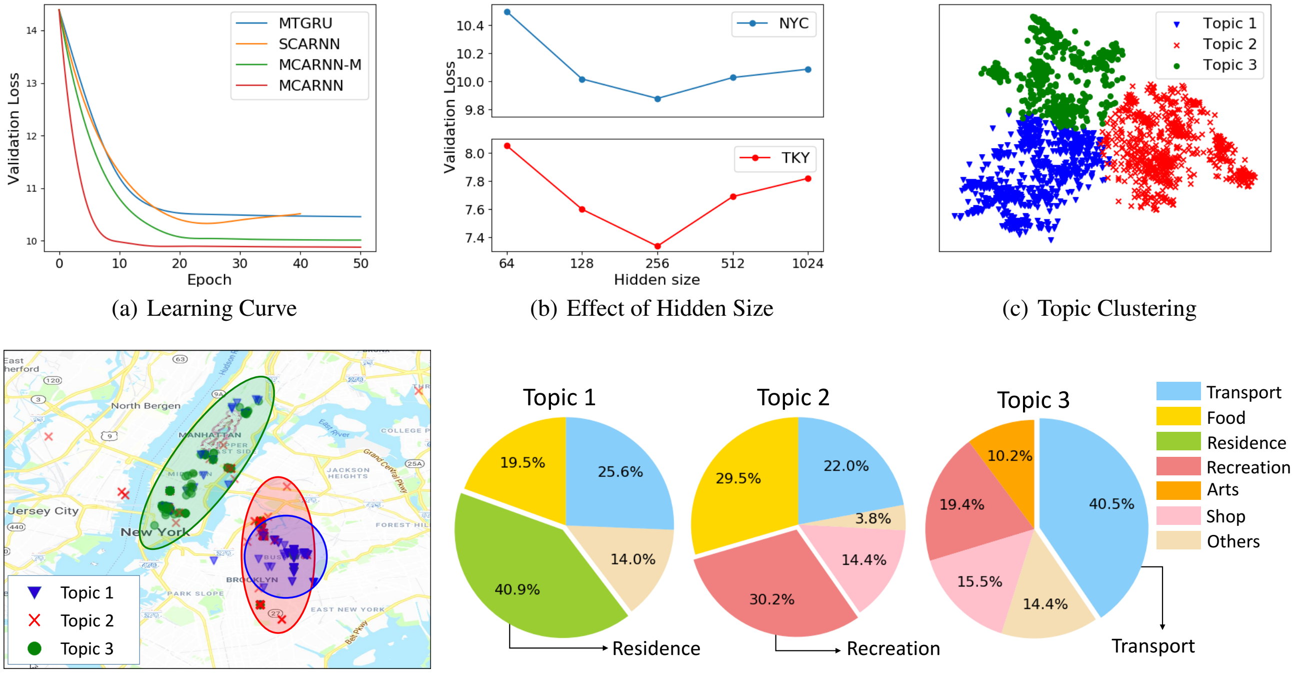
The weights in RNN are initialized with Xavier initialization [Glorot and Bengio, 2010]. Embeddings and initial hidden state are initialized with a random initializer over uniform distribution $(0.1, -0.1)$. The learning rate α starts with 2.00 and decays to one quarter every 4 epochs until it is less than 0.02. We set batch sizes B_a, B_l as 16 and B_G as 64. The timespan threshold is set as 48 hours and the distance threshold is set as 2km. We set the embedding size of $\boldsymbol{u}, \boldsymbol{a}$ and $\boldsymbol{l}$ as the optimal hidden size 256 for convenient calculation. As the value space of time t and timespan s are much smaller, their embedding size is set to a practically effective value, 32. The setting of hidden size of RNN and weighting factors of loss function $\lambda_1, \lambda_2, \lambda_3$ are studied in Sec. 6.4.

6.2 Results

The performance comparison by Acc@K, NLL and MAP@K is shown in Table 2. We can observe that the proposed MCARNN achieves the best result in both prediction tasks.

MF is a naive method without any insight of activity and location prediction, so that it is not surprising of getting the worst results. CAH is a two-stage method based on the intuition that the users' activity purpose determines the location preferences. Comparing with CAH, the proposed spatial-activity topic is more appropriate to leverage the interplay of users' activity and location preferences. SERM and ST-RNN are two RNN models for location prediction of check-ins. SERM can not handle the vanishing of sequential dependency and ST-RNN ignored the temporal context and activity information. Thus our MCARNN significantly outperforms these state-of-the-art methods. Concretely, in location prediction, MCARNN shows an increase of 20.92% on the NYC dataset and 39.15% on the TKY dataset improvement of top-1 accuracy. In activity prediction, the proposed model outperforms CAH by 10% on the NYC dataset and 36% on the TKY dataset of top-1 accuracy.

The proposed model is also superior to its two variants (i.e. MTGRU and SCARNN). For one thing, as we have discussed, GRU is unsuitable for highly sparse check-in data, while the proposed CARU can handle the decayed sequential dependency and integrate the temporal context. As expected, MCARNN performs much better than MTGRU. For another, SCARNN integrates temporal context and sequential pattern by CARU. However, the hidden layers of single-task network only learn the latent regularity in single aspects, instead of spatial-activity topics. Actually, single-task networks may suffer the overfitting problem, which is illustrated by the learning curves on the NYC dataset shown in Fig. 3(a). Learning curves on the TKY dataset show similar trends thus we omit them due to the space limit. The multi-task learning fashion can not only model users' latent topics in both activity and spatial aspects, but also improve generalization because each task can be seen as a regularization for other tasks. The MCARNN-M in Fig. 3(a) is another simplified version of MCARNN without the auxiliary embedding learning task. It can be observed that the auxiliary representation learning task has a sizable promotion of convergence rate for location and activity prediction.



(a) Learning Curve

(b) Effect of Hidden Size

(c) Topic Clustering

(d) Spatial Distribution of Topics

(e) Activity Distribution of Topics

Figure 3: Experiment Results

λ_1	λ_2	λ_3	NLL_l	NLL_a	NLL
1	2	0	6.6068	3.6064	10.2132
	1		6.4526	3.5799	10.0325
	0.5		6.4061	**3.5778**	**9.9839**
	0.25		**6.3941**	3.6862	10.0803
	0.5	0.025	6.3989	**3.5762**	9.9751
		0.05	**6.2879**	3.5782	**9.8561**
		0.1	6.3096	3.6022	9.9118
		0.2	6.3445	3.6040	9.9485

Table 3: Effect of $\lambda_1, \lambda_2, \lambda_3$

6.3 Parameter Study

The size of hidden state determines the capacity of RNN. The performance comparison with the varying hidden size is displayed in Fig. 3(b). When the hidden layer size grows from 64 to 256, the model capacity increases with validation loss becoming smaller gradually. However, when the hidden layer size grows from 256 to 1024, overfitting occurs and validation loss becomes larger gradually due to excessive model capacity. Thus we set the hidden layer size as 256.

Alternate training can easily become biased towards a specific task, thus we need to carefully select the weighting factors λ_1, λ_2 and λ_3. In this study, we fix the λ_1 as 1, and adjust λ_2, λ_3 one after the other. Table 3 shows the performance of the location prediction NLL, activity prediction NLL and total NLL on the NYC validation set. For one thing, as the λ_2 decrease, the location prediction performance gets better. However, when λ_2 is smaller than 0.5, the model become biased to the location prediction task so that the activity prediction performance degenerates dramatically. For another, we fix the λ_2 as 0.5 and increase λ_3 from 0 to 0.2. With the embedding learning task, the location prediction performance is further enhanced. When λ_3 comes up to 0.1, the prediction performance gets even worse due to the same bias issue. Thus the best setting of λ_1, λ_2 and λ_3 is 1, 0.5 and 0.05.

6.4 Case Study of Spatial-Activity Topic

To intuitively interpret the spatial-activity topic, we select one random user with fruitful check-ins in NYC dataset as an example. We feed all this user's check-ins into a fine-trained MCARNN, and cluster outputs of the CARU layer with K-means, representing spatial-activity topics of this user.

Figure 3(c) shows the clustering result with dimension reduction by Principal Components Analysis. It is obvious that the appropriate cluster number is three. Figure 3(d) shows the spatial distribution of these topics. The percentage of activity categories of each topic is shown in Fig 3(e). Here, we classify activities by 9 root categories in Foursquare: transport, food, residence, college, nightlife, arts, shop, recreation and professional. It is difficult to exactly understand this user's behavior from only one aspect. From the spatial aspect, topic 1 and topic 2 check-ins locate in overlapped areas, while show a great difference in activity distribution. From the activity aspect, topic 2 and topic 3 are both about outdoor categories, but locate in completely different regions. In contrast, considering both two aspects, these topics can easily be interpreted as "stay in the residence", "recreation in the neighborhood" and "travel to Manhattan". This is why we state the spatial-activity topic is a reasonable way to capture the complex relationship between user's activities and locations.

7 Conclusion

In this paper, we have proposed a novel activity and location prediction method called MCARNN. The multi-task learning fashion is implemented for utilizing the spatial-activity topics. To integrate the temporal context and sequential pattern, we introduce a Context Aware Recurrent Unit in the shared hidden layer. The experimental results on two public datasets show that MCARNN outperforms the state-of-art methods in terms of both activity and location prediction.

References

[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*, 2013.

[Cho *et al.*, 2011] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011.

[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.

[Glorot and Bengio, 2010] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[Graves *et al.*, 2013] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 2013.

[Kingma and Ba, 2014] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Liao *et al.*, 2017] Dongliang Liao, Yuan Zhong, and Jing Li. Location prediction through activity purpose: Integrating temporal and sequential models. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 711–723. Springer, 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*, 2016.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

[Song *et al.*, 2004] Libo Song, David Kotz, Ravi Jain, and Xiaoning He. Evaluating location predictors with extensive wi-fi mobility data. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*. IEEE, 2004.

[Tang *et al.*, 2015] Jian Tang, Meng Qu, and Qiaozhu Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1165–1174. ACM, 2015.

[Wang *et al.*, 2015] Yingzi Wang, Nicholas Jing Yuan, Defu Lian, Linli Xu, Xing Xie, Enhong Chen, and Yong Rui. Regularity and conformity: location prediction using heterogeneous mobility data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1275–1284. ACM, 2015.

[Wu *et al.*, 2017] Hao Wu, Ziyang Chen, Weiwei Sun, Baihua Zheng, and Wei Wang. Modeling trajectories with recurrent neural networks. 2017.

[Yang *et al.*, 2015] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):129–142, 2015.

[Yang *et al.*, 2017a] 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 *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1245–1254. ACM, 2017.

[Yang *et al.*, 2017b] Cheng Yang, Maosong Sun, Wayne Xin Zhao, Zhiyuan Liu, and Edward Y Chang. A neural network approach to jointly modeling social networks and mobile trajectories. *ACM Transactions on Information Systems (TOIS)*, 35(4):36, 2017.

[Yao *et al.*, 2013] Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. Recurrent neural networks for language understanding. In *Interspeech*, pages 2524–2528, 2013.

[Yao *et al.*, 2017] Di Yao, Chao Zhang, Jianhui Huang, and Jingping Bi. Serm: A recurrent model for next location prediction in semantic trajectories. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2411–2414. ACM, 2017.

[Ye *et al.*, 2013] Jihang Ye, Zhe Zhu, and Hong Cheng. What is your next move: User activity prediction in location-based social networks. In *Proceedings of the SIAM International Conference on Data Mining. SIAM*. SIAM, 2013.

[Yuan *et al.*, 2013] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. Who, where, when and what: discover spatio-temporal topics for twitter users. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 605–613. ACM, 2013.