An Interactive Multi-Task Learning Framework for Next POI Recommendation with Uncertain Check-ins

Lu Zhang[1] , **Zhu Sun**[2*] , **Jie Zhang**[1] , **Yu Lei**[3] , **Chen Li**[3] ,
Ziqing Wu[1] , **Horst Kloeden**[4] and **Felix Klanner**[4]

[1]Nanyang Technological University, Singapore
[2]Macquarie University, Australia
[3]Yanshan University, China
[4]BMW Group, Germany
z.sun@mq.edu.au

Abstract

Studies on next point-of-interest (POI) recommendation mainly seek to learn users' transition patterns with certain historical check-ins. However, in reality, users' movements are typically uncertain (i.e., fuzzy and incomplete) where most existing methods suffer from the transition pattern vanishing issue. To ease this issue, we propose a novel interactive multi-task learning (iMTL) framework to better exploit the interplay between activity and location preference. Specifically, iMTL introduces: (1) *temporal-aware activity encoder* equipped with fuzzy characterization over uncertain check-ins to unveil the latent activity transition patterns; (2) *spatial-aware location preference encoder* to capture the latent location transition patterns; and (3) *task-specific decoder* to make use of the learned latent transition patterns and enhance both activity and location prediction tasks in an interactive manner. Extensive experiments on three real-world datasets show the superiority of iMTL.

1 Introduction

Next point-of-interest (POI) recommendation has recently become a hot research topic, which benefits both personal assistant service and business (POI holders). Existing studies, however, are mainly based on the commonly utilized check-in datasets, e.g., Foursquare and Yelp. These datasets are often carefully designed and calibrated so as to enable researchers to access users' precise activities and locations, thus making accurate next POI recommendation [Zhao *et al.*, 2016; Yao *et al.*, 2017; Liao *et al.*, 2018; Zhao *et al.*, 2019a]. In reality, the check-in information delivered by users is typically uncertain, i.e., fuzzy and incomplete [Sengstock *et al.*, 2013; Li *et al.*, 2018], due to the inaccuracy of indoor GPS devices and personal privacy [Zhang *et al.*, 2019]. Nevertheless, next POI recommendation with uncertain check-ins has been rarely investigated by existing studies.
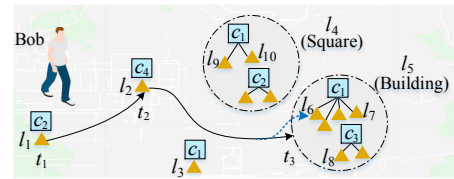
---

*Corresponding Author.

Figure 1: An example of Bob's certain and uncertain check-ins, where yellow triangles are individual POIs (e.g., l_1, l_2); dotted circles are collective POIs (e.g., l_4, l_5); blue squares are categories (e.g., c_1, c_2) that POIs belong to; t_i is the particular check-in time.

As illustrated in Fig. 1, Bob checked at ***individual POIs*** l_1 (e.g., office) at time t_1 and l_2 (e.g., gym) at time t_2 successively, namely ***certain check-in***. He then visited the individual POI l_6 inside the Building l_5, which is defined as ***collective POI*** containing multiple individual POIs (e.g., l_6, l_7, l_8). Most existing studies are built upon Bob's certain check-ins: $l_1 \rightarrow l_2 \rightarrow l_6$. However, Bob may leave a rough footprint, e.g., l_5, instead of the precise POI l_6. The accessible record thus will become: $l_1 \rightarrow l_2 \rightarrow l_5$, where the ***uncertain check-in*** at collective POI l_5 is involved. Based on such records, existing POI recommenders may recommend l_5 at t_3 to Bob instead of l_6 after he visited l_2 at t_2, because POIs l_6, l_7, l_8 are unobserved in his historical records (i.e., cold start POIs). By contrast, we aim to recommend next precise individual POIs (e.g., l_6) to Bob with uncertain check-ins.

In the literature, many efforts are spent on activity or location oriented sequential dependencies for next POI recommendation. Specifically, some studies [Liu *et al.*, 2013; He *et al.*, 2017; Zhang *et al.*, 2019] argue that the interplay between activity and location is significant as the next location visit is affected by the activity. They seek to explore sequential activity patterns, so as to estimate activity first and then predict location given activity. Others attempt to mine sequential location patterns via the recurrent neural network (RNN) [Liu *et al.*, 2016; Huang *et al.*, 2019; Zhao *et al.*, 2019a], due to its capability of capturing sequential information. However, they all ignore the presence of uncertain check-ins, which may weaken the sequential dependencies, hence lead to inaccurate prediction and fail to recommend individual POIs inside the collective POI.

Towards realizing more accurate next POI recommenda-

	# User	# POI	# Check-in	# Category
CLT	1580	1791	20940	239
CAL	301	985	13954	184
PHO	1623	2441	22620	251

Table 1: The statistics of three newly-constructed POI datasets.

tion with uncertain check-ins, we propose an Interactive Multi-Task Learning (iMTL) framework by taking advantage of both types of sequential dependencies (i.e., activity and location). In particular, a two-channel encoder, i.e., temporal-aware activity encoder and spatial-aware location preference encoder, is devised to capture the latent transitions of activities and locations via long-short term memory networks (LSTM) [Hochreiter and Schmidhuber, 1997]. A fuzzy characterization strategy is proposed to better represent activity over uncertain check-ins. Then the task-specific decoder aggregates the latent representations of the two-channel encoder in an interactive manner to perform both activity and location prediction tasks. Moreover, enlightened by [Girshick, 2015], we devise an auxiliary task of POI type (i.e., individual or collective) prediction to enhance the performance of activity prediction. Experimental results on three real-world datasets show iMTL significantly outperforms state-of-the-arts on next POI recommendation with uncertain check-ins.

2 Related Work

Next POI Recommendation. A line of research models sequential activity patterns and generates the most likely locations given activities in a two-fold manner, such as CateMF [Liu *et al.*, 2013], LBPR [He *et al.*, 2017] and CAH [Liao *et al.*, 2017]. Another line adopts variants of RNNs to better capture sequential location patterns, such as ST-RNN [Liu *et al.*, 2016], DeepMove [Feng *et al.*, 2018], STGN [Zhao *et al.*, 2019a] and ATST-LSTM [Huang *et al.*, 2019]. SERM [Yao *et al.*, 2017] further fuses activity with location patterns in a multi-modal fashion. However, these efforts fail to mine the transition patterns due to the presence of uncertain check-ins. The recently proposed HCT [Zhang *et al.*, 2019], built upon Skip-gram [Mikolov *et al.*, 2013], explores this issue by considering both activity and location transition patterns. However, it fails to: (1) well characterize the underlying activity over uncertain check-ins; and (2) exploit the spatial and temporal contexts in a fine-grained way.

Multi-Task Learning for Recommendation. Multi-task learning (MTL) has been widely applied in recommender systems, thanks to its capability in improving model performance and generalization [Caruana, 1997]. Most studies adopt MTL to assist in item recommendation and explanation, such as MT [Lu *et al.*, 2018], CAML [Chen *et al.*, 2019] and MORS [Zhao *et al.*, 2019b]. The recently proposed MCARNN [Liao *et al.*, 2018] considers users' activity and location preferences simultaneously for both activity and location recommendation. However, our iMTL differs from MCARNN in three aspects: (1) iMTL well represents users' uncertain activities via fuzzy characterization strategy while MCARNN ignores the underlying activities over uncertain check-ins; (2) iMTL delicately explores the interplay be-
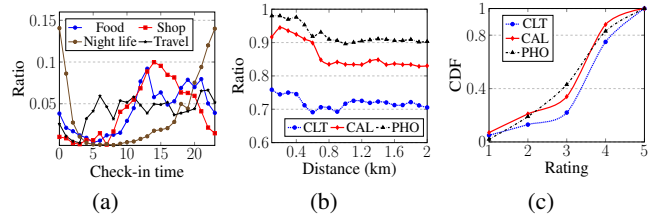

Figure 2: Three observations obtained from the data analysis.

tween activity and location for next POI recommendation via an interactive manner; whilst MCARNN directly adopts the multi-modal embeddings for activity and location as the input of recurrent model; (3) iMTL performs model training by deriving pair-wise ranking loss of activity and location to improve the prediction performance of uncertain activity, while MCARNN solely focuses on observed check-in records.

3 Data Description and Analysis

There is no available dataset with collective POIs, which is suitable for our investigation. Hence, we collect three datasets, i.e., Charlotte (CLT), Calgary (CAL) and Phoenix (PHO), from Foursquare [Yang *et al.*, 2016] and group individual POIs into different collective POIs followed by [Zhang *et al.*, 2019]. For instance, individual POIs, e.g., l_6, l_7, l_8, are grouped into a collective POI l_5 as shown in Fig. 1. The original check-ins $l_1 \rightarrow l_2 \rightarrow l_6$ are converted to $l_1 \rightarrow l_2 \rightarrow l_5$. Each check-in is formed as (u, l, t, c, y, g), meaning that user u visited POI l at time t, where l is associated with category c and POI type y ($y = 0$ denotes individual POI; $y = 1$ refers to collective POI) as well as geocoded by g (latitude and longitude of l). Following [Feng *et al.*, 2015], we remove users and POIs with fewer than 10 check-ins. Besides, each individual POI is assigned with rating derived from Yelp (www.yelp.com/dataset/challenge). The statistics of the newly-constructed datasets are shown in Table 1, and three interesting observations are noted by the data analysis.

Obs.1: temporal-aware activities. User's activities, which can be represented by categories of POIs [Liao *et al.*, 2018], vary with the temporal context [Ye *et al.*, 2013]. Fig. 2(a) shows the ratio of check-ins for the top-4 most popular location categories w.r.t the check-in time on CLT. Due to space limit, we omit the results for CAL and PHO, where similar trends can be observed. The temporal activity pattern is obvious, e.g., the check-ins of shops occur most often between 10am and 8pm. In contrast, the check-ins of night life start to rise quickly at 7pm and peak at 11pm. These imply that users' activities exhibit the strong temporal pattern.

Obs.2: choice-driven check-ins at collective POIs. Due to the extreme challenge of modeling uncertain check-ins, we mine underlying factors that affect users' check-ins at collective POIs. Specifically, we explore the consecutive check-ins containing collective POIs, e.g., $l_2 \rightarrow l_5$ in Fig. 1. We search for the POIs (e.g., l_k) near l_5, i.e., $dist(l_k, l_5) \leq \triangle d$. We find that Bob visited l_5 instead of l_3 and l_4 (both are the accessible POIs which meet Bob's activity c_1). This is in align with the intuition that users tend to visit collective POIs with

more choices (choice-driven), e.g., l_5 contains more individual POIs under c_1. Fig. 2(b) shows the ratio of POIs satisfies the choice-driven over all accessible POIs varying $\triangle d$ within $2km$. We find that more than 75% uncertain check-ins within $0.4km$ are affected by the scale of POIs (choices) .

Obs.3: rating-driven check-ins at cold start POIs. All individual POIs inside the collective POIs are unobserved in users' check-in records in our scenario, i.e., cold start POIs. It is hard to recommend such POIs even with their geo-location as claimed in [Zhang *et al.*, 2019]. We assign a rating for each individual POI by merging Foursquare and Yelp datasets based on its geographical and category information, as such side information could help ease the cold start issue [Sun *et al.*, 2019]. We compute the ratio of check-ins over the ratings, and plot the cumulative distribution function (CDF) in Fig. 2(c). We find that users prefer individual POIs with higher ratings inside the collective POI. For instance, more than 70% check-ins inside the collective POIs correspond to individual POIs with ratings larger than 3 in CLT.

4 The Proposed Methodology

This section first formulates the investigated research problem, and then presents the proposed Interactive Multi-task Learning (iMTL) framework, followed by the model learning strategy and complexity analysis.

For each user, we split his historical check-in records, i.e., (u, l, t, c, y, g), into check-in sequences by day, where each record is sorted by timestamps as in [Zhao *et al.*, 2017]:

- The i-th temporal-aware activity (category) sequence of user u is denoted by a set of activity tuples, i.e., $\mathcal{A}^{u,i} = \{A_{t_1}^u, A_{t_2}^u, \ldots\}$, where $A_{t_k}^u = (c_{t_k}^u, y_{t_k}^u, t_k^u)$, $y_{t_k}^u \in \{0, 1\}$.
- The i-th spatial-aware location sequence of user u is denoted by a set of location tuples, i.e., $\mathcal{L}^{u,i} = \{L_{t_1}^u, L_{t_2}^u, \ldots\}$, where $L_{t_k}^u = (l_{t_k}^u, g_{t_k}^u)$.

Given $\mathcal{A}^{u,i}$ and $\mathcal{L}^{u,i}$, our goal is to predict user u's next activity $c_{t_{n+1}}$ and location $l_{t_{n+1}}$ at time t_{n+1}. If $c_{t_{n+1}}$ happens at a collective POI $l_{t_{n+1}}$, we need to further recommend precise individual POIs inside $l_{t_{n+1}}$ given $c_{t_{n+1}}$.

4.1 The iMTL Framework

The proposed iMTL is outlined in Fig. 3, composed of a two-channel encoder (i.e., temporal-aware activity and spatial-aware location preference encoders) and a task-specific decoder. The two-channel encoder, equipped with embedding, aggregation and recurrent layers, aims to capture the sequential correlations of activities and location preferences. Then the representations encoded by the recurrent layer are utilized in the task-specific decoder to interactively perform three (i.e., the next activity, POI type and POI) prediction tasks.

Temporal-aware Activity Encoder

Given an activity tuple $A_{t_k}^u = (c_{t_k}^u, y_{t_k}^u, t_k^u)$ of user u, $y_{t_k}^u = 1$ represents $c_{t_k}^u$ is an uncertain activity, which happens at collective POI $l_{t_k}^u$ (e.g., Building l_5 in Fig. 1). To better represent such an uncertain activity, we propose a fuzzy characterization strategy inspired by Obs.2:

$$\mathbf{c}_{t_k}^u = \sum\nolimits_{j=1}^{M} \beta_j \mathbf{c}_j, \qquad (1)$$

where M is the total number of categories inside $l_{t_k}^u$; $\mathbf{c}_j \in \mathbb{R}^{D_c}$ is the embedding of j-th category. Since a user prefers a collective POI with more choices given his activity (Obs.2), we thus use β_j to control the possibility of each activity inside $l_{t_k}^u$, which is defined by the scale of individual POIs belong to c_j inside $l_{t_k}^u$ (see Fig. 3). Therefore, $\mathbf{c}_{t_k}^u$ can be used to denote the fuzzy representation of user u's uncertain activity, as well as the functionality (e.g., food-oriented) of the collective POI $l_{t_k}^u$. Meanwhile it also well accommodates the scenario where $y_{t_k}^u = 0$, that is, $c_{t_k}^u$ is a certain activity. In that case, $M = 1$, and $\mathbf{c}_{t_k}^u$ is thus the representation of user u's certain activity.

Furthermore, inspired by Obs.1, modeling sequential activities by incorporating temporal contexts is essential to generate accurate user's next activity. Hence, the temporal-aware activity aggregation $\mathbf{x}_{t_k}^u$ is ultimately represented by,

$$\mathbf{x}_{t_k}^u = \mathbf{W}_c \mathbf{c}_{t_k}^u + \mathbf{W}_t \mathbf{t}_k^u + \mathbf{W}_y \mathbf{y}_{t_k}^u + \mathbf{b}, \qquad (2)$$

where $\mathbf{W}$ is the weight matrix; $\mathbf{b}$ is the bias term; $\mathbf{t}_k^u \in \mathbb{R}^{D_t}$ is the embedding of temporal context with one day mapped into 24 hours; and $\mathbf{y}_{t_k}^u \in \mathbb{R}^{D_y}$ is the embedding of POI type. Subsequently, $\mathbf{x}_{t_k}^u$ is fed into the recurrent layer to infer the hidden state of user u's activity at t_k:

$$\mathbf{h}_{t_k}^u = \text{LSTM}(\mathbf{x}_{t_k}^u, \mathbf{h}_{t_{k-1}}^u), \qquad (3)$$

where $\text{LSTM}(\cdot)$ captures the sequential correlations of activities, and $\mathbf{h}_{t_{k-1}}^u$ encodes the previous activity until t_{k-1}.

Spatial-aware Location Preference Encoder

As a user's check-in is generally affected by the distance between the current location and the next visiting one [Feng *et al.*, 2015], the spatial-aware location preference encoder aims to capture sequential location correlations by considering the spatial contexts. Hence, the aggregation $\widetilde{\mathbf{x}}_{t_k}^u$ for each location tuple $L_{t_k}^u = (l_{t_k}^u, g_{t_k}^u)$ is formulated as:

$$\widetilde{\mathbf{x}}_{t_k}^u = \mathbf{W}_l \mathbf{l}_{t_k}^u + \mathbf{W}_d \mathbf{d}_{t_k}^u + \widetilde{\mathbf{b}}, \qquad (4)$$

where $\mathbf{l}_{t_k}^u \in \mathbb{R}^{D_l}$ is the embedding of POI $l_{t_k}^u$; and $\mathbf{d}_{t_k}^u \in \mathbb{R}^{D_d}$ is the embedding of the distance $d_{t_k}^u$ between $l_{t_{k-1}}^u$ and $l_{t_k}^u$ based on $g_{t_{k-1}}^u$ and $g_{t_k}^u$. Note that we round the distance into integer (e.g., $2.12 \rightarrow 2$), to reduce the number of parameters [Liu *et al.*, 2016]. $\widetilde{\mathbf{x}}_{t_k}^u$ is then fed into the recurrent layer to infer the hidden state of location at t_k, given by,

$$\widetilde{\mathbf{h}}_{t_k}^u = \text{LSTM}(\widetilde{\mathbf{x}}_{t_k}^u, \widetilde{\mathbf{h}}_{t_{k-1}}^u). \qquad (5)$$

Task-specific Decoder

The task-specific decoder aims to interactively perform three (i.e., the next activity, POI type and location) prediction tasks based on the latent representations, i.e., $\mathbf{h}_{t_n}^u$ and $\widetilde{\mathbf{h}}_{t_n}^u$, learned from the two-channel encoder.

Activity Prediction with Auxiliary Task. We predict user u's next activity in the dot-product way, and the probability of next possible activity $c_{t_{n+1}}$ at time t_{n+1} is calculated by:

$$r_{t_{n+1}, c_{t_{n+1}}}^u = (\mathbf{W}_h^c \mathbf{h}_{t_n}^u)^\top (\mathbf{W}_c \mathbf{c}_{t_{n+1}} + \mathbf{W}_y \mathbf{y}_{t_{n+1}} + \mathbf{W}_t \mathbf{t}_{n+1}^u + \mathbf{b}), \qquad (6)$$

where the first term $\mathbf{W}_h^c \mathbf{h}_{t_n}^u$ denotes the user representation, and the second term $\mathbf{W}_c \mathbf{c}_{t_{n+1}} + \mathbf{W}_y \mathbf{y}_{t_{n+1}} + \mathbf{W}_t \mathbf{t}_{n+1}^u + \mathbf{b}$ denotes the temporal-aware activity representation.
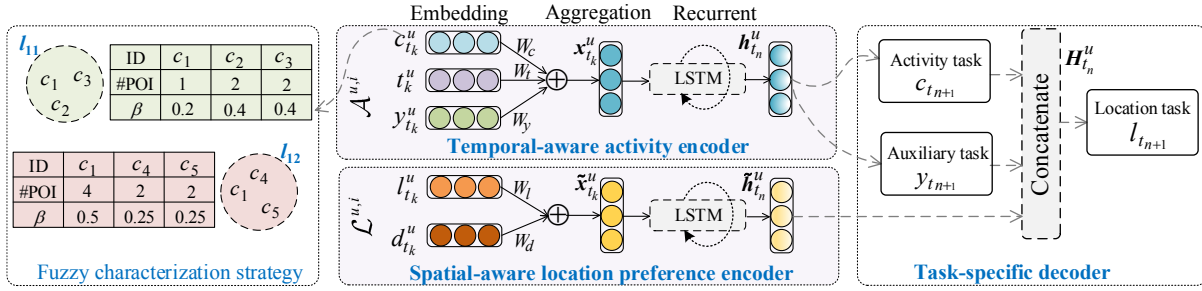
Figure 3: The architecture of the proposed iMTL, which mainly consists of three modules: (1) temporal-aware activity encoder including a fuzzy characterization strategy, (2) spatial-aware location preference encoder, and (3) task-specific decoder. The left subfigure illustrates the fuzzy characterization strategy for the uncertain activity representation based on Obs.2. For instance, collective POI l_{12} contains three categories as shown in the first row of the table; the second row contains the corresponding number of individual POIs under each category within l_{12}; β in the third row is the ratio of #POI to total #POI inside l_{12}.

Although the fuzzy characterization strategy can help mine the user's underlying activity, it may introduce some noise in the process of activity prediction. Inspired by [Girshick, 2015], we thus perform the major task (i.e., activity prediction) with the help of the auxiliary task (i.e., POI type prediction). This is mainly because of two reasons: (1) POI type prediction is related to the activity prediction, as it triggers the presence of either certain or uncertain activities; (2) jointly learning both tasks facilitates to enhance the model generalization. The POI type prediction task is formulated as below, where σ is the sigmoid function.

$$r^u_{t_{n+1}, y_{t_{n+1}}} = \sigma(\mathbf{W}^y_h \mathbf{h}^u_{t_n}), \qquad (7)$$

Location Prediction with Interactive Fashion. Considering the interplay between the user's activity and location visit, that is, the next location check-in is affected by the activity [Zhang *et al.*, 2019], we propose an interactive multi-task learning strategy to generate the next location prediction under the help of activity prediction with auxiliary task. Inspired by [Chen *et al.*, 2019], we concatenate the latent representations learned in the two-channel encoder together with the predicted results of activity and POI type:

$$\mathbf{H}^u_{t_n} = [\tilde{\mathbf{h}}^u_{t_n}; \mathbf{h}^u_{t_n}; r^u_{t_{n+1}, c_{t_{n+1}}} \mathbf{c}_{t_{n+1}}; r^u_{t_{n+1}, y_{t_{n+1}}} \mathbf{y}_{t_{n+1}}], \quad (8)$$

Hence, the probability of u's next POI $l_{t_{n+1}}$ at time t_{n+1} is:

$$r^u_{t_{n+1}, l_{t_{n+1}}} = (\mathbf{W}_H \mathbf{H}^u_{t_n})^\top (\mathbf{W}_l \mathbf{l}_{t_{n+1}} + \mathbf{W}_d \mathbf{d}^u_{t_{n+1}} + \tilde{\mathbf{b}}), \quad (9)$$

where the first term $\mathbf{W}_H \mathbf{H}^u_{t_n}$ denotes the user representation, and the second term $\mathbf{W}_l \mathbf{l}_{t_{n+1}} + \mathbf{W}_d \mathbf{d}^u_{t_{n+1}} + \tilde{\mathbf{b}}$ denotes the spatial-aware location representation. In sum, the activity prediction task assists in the location prediction task, which in turn influences the activity representation learning during the model training with back-propagation, that is, they are interactively enhanced by each other. Furthermore, we consider two cases due to the presence of collective POIs: 1) if $y^u_{t_{n+1}} = 0$ meaning $l^u_{t_{n+1}}$ is an individual POI, we directly recommend individual POIs related to activity $c^u_{t_{n+1}}$; and 2) if $y^u_{t_{n+1}} = 1$, we further select individual POIs characterized by $c^u_{t_{n+1}}$ inside $l^u_{t_{n+1}}$ with higher ratings according to Obs.3.

4.2 Learning Strategy and Complexity Analysis

We adopt the Bayesian Personalized Ranking [Rendle *et al.*, 2009] to define the loss functions of activity and location pre-

---

Algorithm 1: Learning Strategy of iMTL.

Input: $\Theta, \mathcal{A}, \mathcal{L}$, user set $\mathcal{U}$, max_iter, λ, η
Output: iMTL model $\{\Theta\}$

1 Initialize the set of model parameters Θ;
      // Training instances generation
2 **foreach** $u \in \mathcal{U}$ **do**
3   **foreach** $\mathcal{A}^{u,i} \in \mathcal{A}^u$ and $\mathcal{L}^{u,i} \in \mathcal{L}^u$ **do**
4   Sample negative activities and locations

   // Parameter Update
5 **for** $iter = 1; iter \leq max\_iter; iter++$ **do**
6   **foreach** $u \in \mathcal{U}$ **do**
7   Randomly select a batch of instances
8   **foreach** $\theta \in \Theta$ **do**
9          $\theta \leftarrow \theta - \eta * \frac{\partial \mathcal{J}}{\partial \theta}$

---

diction tasks. Followed by [Huang *et al.*, 2019], we generate the training instances Ω in a recursive way for each temporal-aware activity sequence and spatial-aware location sequence. For each instance, we then uniformly sample the negative activities and locations. Particularly, w.r.t. collective POIs, the negative samples come from the nearby POIs as defined in Obs.2. The loss functions for the two prediction tasks are:

$$\mathcal{J}_c = \sum_{(c \succ c') \in \Omega} \ln \left(1 + e^{-(r^u_{t,c} - r^u_{t,c'})}\right),$$
$$\mathcal{J}_l = \sum_{(l \succ l') \in \Omega} \ln \left(1 + e^{-(r^u_{t,l} - r^u_{t,l'})}\right), \qquad (10)$$

where c' is the negative activity for c, and where l' is the negative location for l. Meanwhile, the loss function of POI type prediction (binary classification) is defined by:

$$\mathcal{J}_y = \sum_{k=1}^{|\Omega|} y_{t_k} \log(r^u_{t_k, y_{t_k}}) + (1 - y_{t_k}) \log(1 - r^u_{t_k, y_{t_k}}). \quad (11)$$

Ultimately, we seek to minimize the sum loss:

$$\mathcal{J} = \lambda_c \mathcal{J}_c + \lambda_l \mathcal{J}_l + \lambda_y \mathcal{J}_y + \frac{\lambda}{2} ||\Theta||^2, \qquad (12)$$

where λ_c, λ_l, and λ_y ($\lambda_c + \lambda_l + \lambda_y = 1$) are weights to balance the importance of different losses; λ is the regularization coefficient; and $\Theta = (\mathbf{W}, \mathbf{c}, \mathbf{y}, \mathbf{t}, \mathbf{l}, \mathbf{d}, \mathbf{b}, \tilde{\mathbf{b}})$, is the set of model parameters to be learned. We use AdaGrad [Duchi *et*

						Activity Prediction Task					
Data	Metrics	MostPop	CateMF	ME	LBPR	ST-RNN	ATST-LSTM	MCARNN	HCT	iMTL	Improve
CLT	Rec@10	0.1321	0.1516	-	0.1829	-	-	0.1935	0.1984*	**0.2213**	11.5%
	MAP@10	0.0435	0.0482	-	0.0558	-	-	0.0621	0.0614*	**0.0709**	15.5%
CAL	Rec@10	0.1063	0.1238	-	0.1425	-	-	0.1579	0.1602*	**0.1724**	7.6%
	MAP@10	0.0417	0.0454	-	0.0618	-	-	0.0751	0.0764*	**0.0887**	16.1%
PHO	Rec@10	0.1102	0.1214	-	0.1647	-	-	0.1974	0.2010*	**0.2311**	15.0%
	MAP@10	0.0430	0.0506	-	0.0624	-	-	0.0776	0.0781*	**0.0947**	21.3%
						Location (POI) Prediction Task					
Data	Metrics	MostPop	CateMF	ME	LBPR	ST-RNN	ATST-LSTM	MCARNN	HCT	iMTL	Improve
CLT	Rec@10	0.0305	0.0323	0.0401	0.0446	0.0421	0.0465	0.0458	0.0477*	**0.0534**	11.9%
	MAP@10	0.0104	0.0124	0.0137	0.0174	0.0162	0.0201	0.0210	0.0204*	**0.0238**	13.3%
CAL	Rec@10	0.0317	0.0341	0.0416	0.0483	0.0479	0.0522	0.0545	0.0617*	**0.0691**	12.0%
	MAP@10	0.0119	0.0155	0.0205	0.0226	0.0230	0.0328	0.0364	0.0372*	**0.0443**	16.4%
PHO	Rec@10	0.0323	0.0352	0.0434	0.0525	0.0506	0.0591	0.0608	0.0662*	**0.0769**	16.2%
	MAP@10	0.0125	0.0180	0.0207	0.0218	0.0243	0.0260	0.0275	0.0301*	**0.0352**	16.9%

Table 2: Performance comparison of activity and location tasks on the three datasets. The best performance is boldfaced; the runner up is labeled with '*'; 'Improve' refers to the improvements (Paired t-test with p-value < 0.01) that iMTL achieves relative to the '*' results.

al., 2011] to optimize the network parameters. Alg. 1 shows the training process composed of two parts: training instances generation (lines 2-4) and parameter update (lines 5-9).

Complexity Analysis. In the training process of iMTL, the computational complexity of the two-channel encoder and decoder is $\mathcal{O}(D^2)$, where D is the embedding size. The complexity of recurrent units is $\mathcal{O}(1)$. Consequently, given the training instances Ω with the average sequence length $\bar{S}$, the overall complexity for each training iteration would be $\mathcal{O}(|\Omega| \cdot \bar{S} \cdot D^2)$. In sum, the complexity of iMTL is linear to $|\Omega|$ and quadratic to the embedding size D.

5 Experiments

We investigate the effectiveness of iMTL with the goal of answering three research questions[1]. **RQ1**: does our proposed iMTL outperform the baselines w.r.t. both activity and POI recommendation tasks? **RQ2**: how do different components of iMTL (e.g., fuzzy characterization strategy) affect its performance? **RQ3**: how do hyper-parameters affect iMTL?

5.1 Experimental Setup

Datasets and Metrics. We conduct experiments on the three datasets as shown in Table 1. Following [Huang et al., 2019], we treat the first 80% sequences of each user as training set, the latter 10% as the validation set and the last 10% as test set. Note that for validation and test sets, we also remain the real visited individual POIs under collective POIs as ground-truth for POI prediction. Two standard metrics are adopted to evaluate the performance of all methods, namely, Recall (Rec@K) and Mean Average Precision (MAP@K).

Baselines. We compare iMTL with eight state-of-the-arts: (1) **MostPop**: it recommends next activity and POI via popularity; (2) **CateMF** [Liu et al., 2013]: it generates next activity and POI via factorizing user-category/location matrices; (3) **LBPR** [He et al., 2017]: it is a Listwise Bayesian Personalized Ranking method, which recommends next activity and

POI in a two-fold way; (4) **ME** [Feng et al., 2015]: it is a state-of-the-art metric embedding model for next POI recommendation; (5) **ST-RNN** [Liu et al., 2016]: it is a RNN-based method that incorporates spatial-temporal contexts for next location prediction; (6) **ATST-LSTM** [Huang et al., 2019]: it embeds the POIs and spatial-temporal contexts in a multimodal manner for next POI prediction; (7) **MCARNN** [Liao et al., 2018]: it exploits a multi-task learning framework for next activity and POI prediction; (8) **HCT** [Zhang et al., 2019]: it is the recently proposed method for next activity and POI recommendation with uncertain check-ins.

Note that if the predicted activity happens at a collective POI, we need to further recommend next precise individual POIs inside the collective POI given the activity. As those individual POIs are cold start ones, we rank such POIs related to the predicted activity via their ratings based on Obs 3.

Parameter Settings. The parameters for all the baselines are tuned to achieve the best results or set as suggested by the original papers. For iMTL, the weight matrices, embeddings and hidden states are randomly initialized over uniform distribution. We apply a grid search in $[20, 200]$ stepped by 20 to find the optimal settings for embedding size $D_c = D_y = D_t = D_l = D_d$, and they are set to 120/140/120 for CLT, CAL and PHO, respectively (Fig. 5(a)). The number of recurrent layers is 3; the learning rate $\eta = 0.0001$; the iteration numbers are 25/20/30 for CLT, CAL and PHO, respectively (Fig. 5(b)); $\lambda_c = 0.4$, $\lambda_y = \lambda_l = 0.3$, $\lambda = 0.0025$ (Fig. 5(c)). The distance threshold $\triangle d = 2km$ for searching the negative POIs near the collective POI.

5.2 Performance Comparison (RQ1)

Table 2 presents the performance (Rec@K and MAP@K) of all methods across the three datasets. We test $K = 5, 10, 15$. Due to space limitation, we only show the results with $K = 10$. Similar trends can be observed with $K = 5, 15$. Note that, ME, ST-RNN and ATST-LSTM do not consider category information, thus cannot predict next activity.

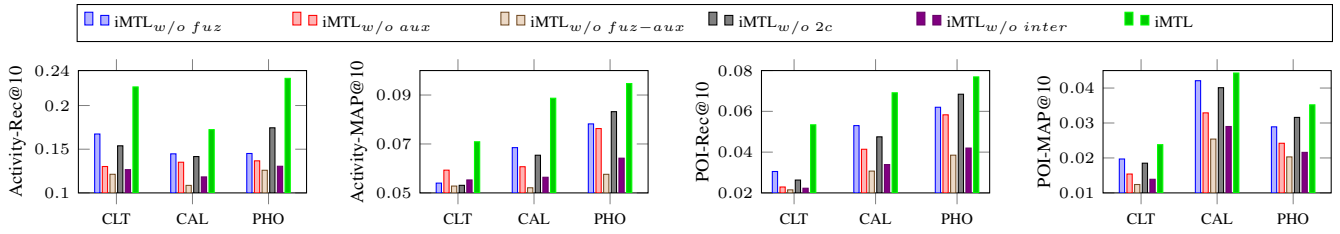In terms of both tasks, the non-RNN based methods

Figure 4: Performance comparison for variants of iMTL on the three datasets w.r.t activity and POI prediction tasks.
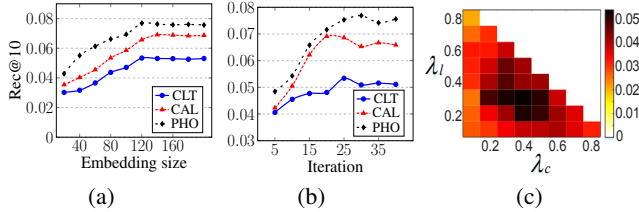

Figure 5: Parameter sensitivity analysis on POI prediction task.

(Pop, CateMF and ME, LBPR) generally perform worse than RNN based baselines (ST-RNN, ATST-LSTM, MCARNN), demonstrating the efficacy of RNN on modeling the sequential dependency. As for RNN based methods, ATST-LSTM performs better than ST-RNN, which indicates the effectiveness of attention mechanism in modeling check-in sequence. They, however, are outperformed by MCARNN, due to its ability of capturing both sequential activities and locations via the MTL method. HCT performs better than RNN based methods, as it carefully designs and models the hierarchical dependencies between activity and location to ease the issue of next POI recommendation with uncertain check-ins. Overall, our proposed iMTL consistently achieves the best performance w.r.t. the two tasks across the three datasets. This is mainly because: (1) iMTL models user's uncertain activity via fuzzy characterization strategy; (2) iMTL adopts the auxiliary task (i.e., POI type prediction) to assist in activity prediction; and (3) iMTL exploits the interplay between activity and POI via an interactive multi-task learning framework.

5.3 Detailed Study of iMTL (RQ2)

To investigate the effectiveness of different components of iMTL, we compare it with five different variants: (1) $iMTL_{w/o\ fuz}$ removes the fuzzy characterization on users' uncertain activities. Following [Zhang *et al.*, 2019], we directly use the category of a collective POI to represent the activity, instead of the fuzzy representation; (2) $iMTL_{w/o\ aux}$ removes the auxiliary task, i.e., POI type prediction; (3) $iMTL_{w/o\ fuz-aux}$ ignores both fuzzy characterization and auxiliary task; (4) $iMTL_{w/o\ 2c}$ merges the two-channel encoder into one by using multi-modal embedding manner; and (5) $iMTL_{w/o\ inter}$ removes the interactive learning method by performing the three prediction tasks in parallel.

We report the results in Fig. 4, where iMTL significantly outperforms its variants regarding to both activity and location prediction tasks. We notice that $iMTL_{w/o\ fuz-aux}$

performs worse than either $iMTL_{w/o\ fuz}$ or $iMTL_{w/o\ aux}$, which suggests that both fuzzy characterization strategy and auxiliary task indeed improve the recommendation performance. Generally, the performance decrease of $iMTL_{w/o\ aux}$ far exceeds that of $iMTL_{w/o\ fuz}$, implying that the auxiliary task (i.e., POI type prediction) plays a more important role than the fuzzy characterization strategy. Both $iMTL_{w/o\ 2c}$ and $iMTL_{w/o\ inter}$ underperform iMTL, indicating the advantages of the two-channel encoder and interactive multi-task learning strategy. In summary, our proposed iMTL benefits from the four delicately designed components.

5.4 Parameter Sensitivity Analysis (RQ3)

Fig. 5 depicts the results (Rec@10) of parameter sensitivity analysis on POI prediction. Due to the space limitation, we omit the results on activity prediction where similar trends can be observed. In Fig. 5(a), the performance of iMTL climbs up as the embedding size increases, and gradually becomes stable with the size around 120. We further study the convergence property of iMTL as shown in Fig. 5(b), where we observe that iMTL can converge within 30 iterations on the three datasets. Fig. 5(c) shows the performance of varying the combination weights λ_c and λ_l, which control the importance of the activity and location prediction tasks, respectively. Note that the importance of the POI type prediction task λ_y is determined by $1 - \lambda_c - \lambda_l$. From the results, we find that iMTL performs best with $\lambda_c = 0.4$, $\lambda_l = \lambda_y = 0.3$, which implies that all the three tasks are vital for more accurate next POI recommendation with uncertain check-ins.

6 Conclusion

In this paper, we propose an interactive multi-task learning (iMTL) framework for the next POI recommendation with uncertain check-ins. In particular, we devise a two-channel encoder, i.e., temporal-aware activity encoder and spatial-aware location preference encoder, to capture the transitions of activities and locations, whereby a fuzzy characterization strategy is proposed to better represent activity over uncertain check-ins. The task-specific decoder then interactively aggregates the latent representations of the two-channel encoder to perform both activity and location prediction tasks. Experimental results shows the superiority of iMTL over state-of-the-arts w.r.t next activity and location recommendation tasks.

Acknowledgments

References

[Caruana, 1997] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[Chen *et al.*, 2019] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. Co-attentive multi-task learning for explainable recommendation. In *IJCAI*, pages 2137–2143, 2019.

[Duchi *et al.*, 2011] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR*, 12(Jul):2121–2159, 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.

[Feng *et al.*, 2018] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *WWW*, pages 1459–1468, 2018.

[Girshick, 2015] Ross Girshick. Fast R-CNN. In *CVPR*, pages 1440–1448, 2015.

[He *et al.*, 2017] Jing He, Xin Li, and Lejian Liao. Category-aware next point-of-interest recommendation via listwise bayesian personalized ranking. In *IJCAI*, pages 1837–1843, 2017.

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

[Huang *et al.*, 2019] Liwei Huang, Yutao Ma, Shibo Wang, and Yanbo Liu. An attention-based spatiotemporal lstm network for next poi recommendation. *IEEE Transactions on Services Computing*, 2019.

[Li *et al.*, 2018] Ming Li, Rene Westerholt, Hongchao Fan, and Alexander Zipf. Assessing spatiotemporal predictability of LBSN: A case study of three foursquare datasets. *GeoInformatica*, 22(3):541–561, 2018.

[Liao *et al.*, 2017] Dongliang Liao, Yuan Zhong, and Jing Li. Location prediction through activity purpose: Integrating temporal and sequential models. In *KDD*, pages 711–723, 2017.

[Liao *et al.*, 2018] Dongliang Liao, Weiqing Liu, Yuan Zhong, Jing Li, and Guowei Wang. Predicting activity and location with multi-task context aware recurrent neural network. In *IJCAI*, pages 3435–3441, 2018.

[Liu *et al.*, 2013] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*, pages 733–738, 2013.

[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.

[Lu *et al.*, 2018] Yichao Lu, Ruihai Dong, and Barry Smyth. Why I like it: multi-task learning for recommendation and explanation. In *Recsys*, pages 4–12, 2018.

[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 *NIPS*, pages 3111–3119, 2013.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461, 2009.

[Sengstock *et al.*, 2013] Christian Sengstock, Michael Gertz, Florian Flatow, and Hamed Abdelhaq. A probablistic model for spatio-temporal signal extraction from social media. In *SIGSPATIAL*, pages 274–283, 2013.

[Sun *et al.*, 2019] Zhu Sun, Qing Guo, Jie Yang, Hui Fang, Guibing Guo, Jie Zhang, and Robin Burke. Research commentary on recommendations with side information: A survey and research directions. *ECRA*, 37:100879, 2019.

[Yang *et al.*, 2016] Dingqi Yang, Daqing Zhang, and Bingqing Qu. Participatory cultural mapping based on collective behavior data in location-based social networks. *ACM (TIST)*, 7(3):30, 2016.

[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 *CIKM*, pages 2411–2414. ACM, 2017.

[Ye *et al.*, 2013] Jihang Ye, Zhe Zhu, and Hong Cheng. What's your next move: User activity prediction in location-based social networks. In *ICDM*, pages 171–179, 2013.

[Zhang *et al.*, 2019] Lu Zhang, Zhu Sun, Jie Zhang, Horst Kloeden, and Felix Klanner. Modeling hierarchical category transition for next poi recommendation with uncertain check-ins. *Information Sciences*, 2019.

[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 315–321, 2016.

[Zhao *et al.*, 2017] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R Lyu. Geo-teaser: geo-temporal sequential embedding rank for point-of-interest recommendation. In *WWW*, pages 153–162, 2017.

[Zhao *et al.*, 2019a] 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.

[Zhao *et al.*, 2019b] Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. Recommending what video to watch next: a multi-task ranking system. In *RecSys*, pages 43–51. ACM, 2019.