Information Processing and Management 58 (2021) 102715

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Hierarchical temporal–spatial preference modeling for user consumption location prediction in Geo-Social Networks

Shuai Xu [a,b,*], Dechang Pi [a], Jiuxin Cao [c], Xiaoming Fu [d]

[a] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China
[b] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
[c] School of Cyber Science and Engineering, Southeast University, Nanjing, China
[d] Institute of Computer Science, University of Göttingen, Göttingen, Germany

ARTICLE INFO

Keywords:
Location prediction
Preference modeling
Hierarchical attention
Feature fusion
Geo-social networks

ABSTRACT

Predicting where people will consume in the future is of great significance for promoting local business. Although the prevalence of Geo-Social Networks (GSNs) has provided sufficient and desirable geo-tagged data for user mobility modeling, most studies attempt to directly fit user's preference toward locations through exploring the complex interaction between $\langle user, location\rangle$ pairs, which is usually hard to incorporate temporal–spatial context and side information. Moreover, the availability of multi-modal data associated with both user and location in GSNs has not yet been comprehensively leveraged. In view of the above-mentioned situations, in this article, we propose a *two-stage* framework composed of a *Temporal Base Model (TBM)* and a *Location Prediction Model (LPM)* to accomplish the task of user consumption location prediction at a given time in the future. In the first stage, based on user sentimental textual reviews, we leverage the hierarchical attention mechanism to capture time-sensitive user latent preference. In the second stage, we fuse the multifaceted context to derive the user's consumption probability toward different locations at the given time. We conduct extensive experiments over three real-world GSN datasets to verify the performance of the proposed approach. The experimental results encouragingly demonstrate the effectiveness of the *two-stage* framework, which outperforms multiple baselines in terms of different evaluation metrics such as accuracy, average percentile rank (APR) and coverage ratio.

1. Introduction

Geo-Social Networks (GSNs) have become an indispensable part of people's daily life by providing location-based service and online social networking service simultaneously. At present, representative GSNs such as Foursquare[1] and Yelp[2] are attracting thousands of new users every day. According to Statista,[3] the number of unique mobile visitors to Yelp has reached 30 million at the 4th quarter of 2020, which enables Yelp to be one of the most popular geo-social networking service providers. A huge amount of user footprints such as check-ins and reviews have been produced in GSNs, offering fine data support for researchers to understand and model users' mobility, which ultimately serves user future consumption location prediction. To help illustrate the

* Corresponding author.
E-mail address: xushuai7@nuaa.edu.cn (S. Xu).

[1] https://foursquare.com/
[2] https://www.yelp.com/
[3] https://www.statista.com/statistics/385440/unique-mobile-visitors-yelp/

https://doi.org/10.1016/j.ipm.2021.102715
Received 3 April 2021; Received in revised form 22 July 2021; Accepted 8 August 2021
Available online 4 September 2021

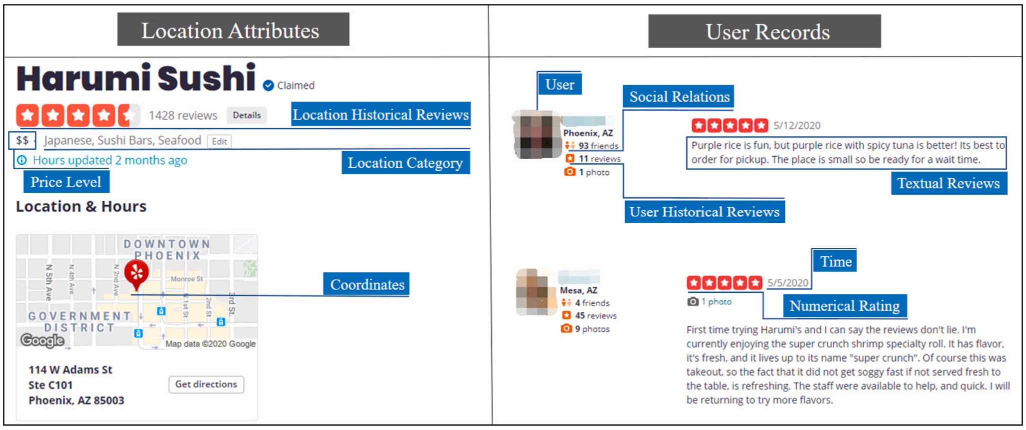


Fig. 1. Overview of an example concerning a restaurant and its visitors on Yelp. The left part illustrates the available attributes associated with this venue, while the right part shows the visitors' attributes and their reviews on this venue.

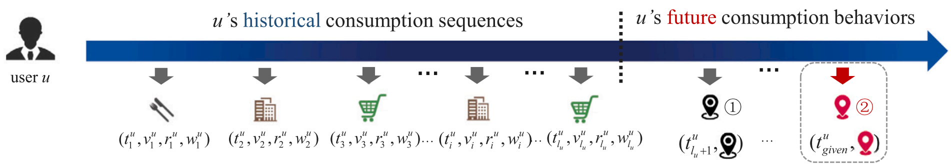


Fig. 2. Two types of user location prediction problems, where ① represents *successive user location prediction* and ② represents *given-time user location prediction*.

characteristics of geo-social networking data, we present a practical example concerning a location[4] and its visitors on Yelp (Fig. 1). For the sake of preserving user privacy, we blur the user profile photos and nicknames.

The problem of user location prediction has long been a hot issue in both academia and industry, as it is able to bring huge value to practical scenarios such as personalized point-of-interest recommendation (Cai, Wen, Wu, & Yang, 2021) and intelligent traffic scheduling (Yang et al., 2020). The essence of user consumption location prediction based on geo-social networking data is to thoroughly mine the temporal–spatial characteristics, sequential relations, as well as user personal preferences hidden in the data, after which machine learning based techniques can be employed to integrate the multi-dimensional information so as to infer where the user will go in the future (Qian, Lu, Han, Du, & Li, 2017; Xu, Fu, Cao, Liu & Wang, 2020). From the perspective of prediction timeliness, current studies concerning user consumption location prediction are basically classified into two types: (1) *successive user location prediction*, which aims to predict a user's next visit location given his/her historical footprints; and (2) *given-time user location prediction*, which aims to predict where a user will be at a given time in the future. For clarity, we display the comparison between two types of location prediction problems in Fig. 2. Suppose we collect the historical consumption sequence of user u and its length is l_u, as each of user u's consumption behavior until $t^u_{l_u}$ is known, whereas his/her consumption behaviors are not available during the period from $t^u_{l_u}$ to t^u_{given}, predicting where user u will consume at time t^u_{given} shall be more challenging than at time $t^u_{l_u+1}$ because there is less information provided. In this article, we aim to solve the given-time user consumption location prediction based on geo-social networking data (enclosed by the gray dashed box shown in Fig. 2). Consider the following scenario: if we can predict where someone will go at a designated time in the future, it not only facilitates neighboring businesses to formulate personalized and timely marketing strategy so as to increase revenue (Gao et al., 2018; Zhang et al., 2021), but also helps provide effective means for authorities to monitor the user of interest (Xu, Cao, Legg, Liu & Li, 2020).

Most of the existing literatures such as Feng et al. (2020), Gao et al. (2019) and Yang, Fankhauser, Rosso, and Cudre-Mauroux (2020) focus on predicting the user's successive consumption location, which basically explore the short-term sequential mobility pattern in user most recent trajectories. For these studies, the common approach is to depict the complex $\langle user, location\rangle$ interaction, so that the user's visit probability toward a specific location can be fitted. Note that in the whole process of $\langle user, location\rangle$ interaction modeling, the user's exact visit time at this location is generally ignored. As a comparison, literature concerning given-time user location prediction is rare. The common approach for such research is to embed heterogeneous vertices (including but not limited to user, location and time) into low-dimensional vector space, based on which the cosine similarity or inner-product between vertices is applied to find the most possible location where a user would go at a given time (Yang, Qu, Yang, & Cudre-Mauroux, 2019b; Zhao, Zhao, Yang, Lyu, & King, 2016). It is well acknowledged that user latent preference modeling is fundamental to user mobility prediction (Zhao, Lou, Qian, & Hou, 2020). However, most of the existing studies tend to characterize user latent preference

[4] In this article, the three terms "location", "point-of-interest (POI)" and "venue" can be used interchangeably unless otherwise stated.

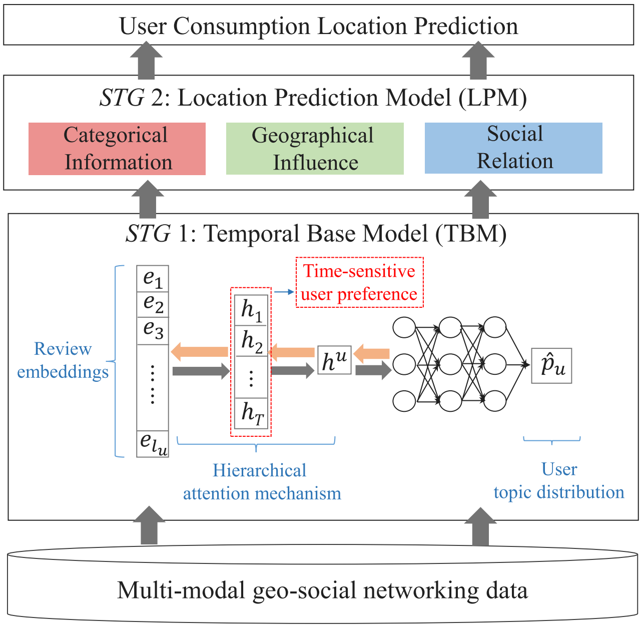


Fig. 3. Overview of the proposed *two-stage* framework.

solely based on location embedding, which is indirect and insufficient. Few studies attempt to model user intrinsic preference, especially the dynamic periodic preference along with time. For example, in Feng et al. (2018) and Gao et al. (2019), location embeddings are sequentially fed into the Recurrent Neural Network (RNN), where the hidden state in the last step is taken as the user latent preference, which can only reflect the short-term user preference toward locations as only sequential features are exploited. Moreover, although user mobility features like temporal cyclic effect, geographical influence, sequential relation, as well as social ties are widely exploited in previous studies, we argue that the multi-modal geo-social networking data, especially the user posted textual reviews, has not been fully utilized. How to incorporate sentimental textual contents and further fuse the multifaceted context are still intractable.

In view of the above-mentioned shortcomings in existing research, in this article, we propose a *two-stage* framework (see Fig. 3) for user consumption location prediction at a given time in the future. As can be seen from Fig. 3:

– In the first stage, a *Temporal Base Model (TBM)* is conceived to model user intrinsic consumption preference within different time windows (see Section 4.1). At this stage, user textual reviews are leveraged to train time-sensitive user latent preference by hierarchical attention mechanism. In order to characterize user intrinsic preference, we propose to connect sentimental review representation (Wu, Dai, Yin, Huang, & Chen, 2018) and topic modeling (Zhou, Noulas, Mascolo, & Zhao, 2018) through multi-layer neural network architecture, which in turn promotes learning dynamic user preference within different time windows.
– In the second stage, a *Location Prediction Model (LPM)* is conceived to portray the time-sensitive $\langle user, location\rangle$ interaction (see Section 4.2), so that the user's consumption probability toward the location at a given time can be well fitted. At this stage, we incorporate categorical information of locations as well as geographical and social influence into *LPM* through non-linear feature fusion. A scoring function is ultimately derived to calculate the overall consumption probability.

We verify the effectiveness of the proposed *two-stage* framework using three Yelp datasets, which are generated from three representative metropolises in North America. Extensive experiments from different aspects have been conducted to evaluate the performance of the proposed approach. The empirical results encouragingly demonstrate the superior prediction accuracy and ranking capacity of our approach, as it is not only at least 19.4%, 11.7% and 16.1% better than the state-of-the-art approach on $Acc@5$ metric using different datasets respectively, but also delivering higher APR metric values than the comparisons. Last but not least, as it generally has the best performance in terms of coverage ratio metric, the approach also shows fine usability for user consumption location prediction in real-world scenarios.

To sum up, the core contributions of this work are three-fold:

• First, based on user sentimental textual reviews, we propose to portray user intrinsic periodic preference by the hierarchical *Temporal Base Model*, to our knowledge, this is the first attempt to directly learn time-sensitive user latent preference by bridging the gap between topic modeling and sentimental review representation using deep neural networks.

- Second, the multi-modal geo-social networking data is leveraged. We incorporate textual content, categorical information, geographical influence, as well as social relation into the *Location Prediction Model*, so as to predict the user's consumption location at a given time in the future.
- Third, a comprehensive evaluation based on three real-world GSN datasets is conducted. Quantitative comparisons on multiple metrics show that our approach outperforms several state-of-the-art methods in terms of user consumption location prediction performance.

The rest of the manuscript is organized as follows. Section 2 reviews the related works. Section 3 introduces the datasets and gives the formal definition of the problem. Section 4 explains the *Temporal Base Model* and the *Location Prediction Model*, respectively. Section 5 elaborates the experiments. Finally, Section 6 concludes the article and introduces the direction of future work.

2. Related works

2.1. Indicative features for user mobility modeling

Temporal Cyclic Effect: Gao, Tang, Hu, and Liu (2013) introduce the concept of time window to simulate the temporal periodicity of user mobility for the first time. They propose a temporal-context based framework to explore the impact of periodicity on the predictive performance for user future visits. In recent years, temporal cyclic effect hidden in user trajectories has been further explored. Hao, Cheang, and Chiang (2019) and Yu et al. (2020) analyze user check-in pattern during weekdays and weekends, respectively, and design time-aware embedding models to depict user preference under two temporal contexts. Specifically, Yu et al. (2020) apply a density-sensitive method to partition user check-in records, so that the division could be finer-grained in the peak period of user visits, while it could be relatively coarse-grained in slack visit period. Cao, Xu, Zhu, Lv, and Liu (2018) quantify the predictive accuracy of various models under different temporal contexts. The key findings can be summarized as follows, for the time periods when people tend to have regular activities like going to office and taking public transportation, the predictive performance can be the best, while for people's leisure time, the predictive performance will decrease. Zhou, Mascolo, and Zhao (2019) extend the traditional LDA model with temporal factor. They propose to infer user topic distribution under a given time window, which can be regarded as the temporal-specific representation of the user. However, in Zhou et al. (2019), only 37 phrases standing for location categories are used, which cannot reflect users' sentiment and opinion toward locations.

Another strategy to incorporate temporal factor is to model the impact of time interval on users' future mobility. Intuitively, human beings tend to keep steady visit preference within a short time period, while a large time gap may change their visit preference. Based on this assumption, literatures Yang et al. (2020) and Xu et al. (2021) redesign the architecture of RNNs by introducing a time gate to update users' visit preference after a certain time interval. Xu, Cao et al. (2020) adopt a time-decay manner to compute a user's preference after a given time interval, by this way, locations that are visited by the user more recently will be assigned higher weights.

Geographical Influence: It is well recognized that every user in GSNs has his/her own mobility center, and the closer a venue is to that center, the higher probability he/she will visit there (Gonzalez, Hidalgo, & Barabasi, 2008; Lian et al., 2018). Ma, Zhang, Wang, and Liu (2018) adopt the Gaussian radial basis function to incorporate geographical influence into auto-encoders, through which users' historical visits would exert more influence on nearby unseen locations. Similarly, Zhou et al. (2019) derive a geographical score between $\langle user, location \rangle$ pair, so that a nearby location to the user's check-in center would have higher probability to be visited. Yu et al. (2020) propose to filter locations using geographical coordinates, which helps to reduce the size of candidate location set. There are also studies fusing geographical influence into RNN architecture, for example, in Kong and Wu (2018) and Zhao et al. (2019), the "spatial gates" are added to LSTM in order to control the influence of previously visited locations.

Semantic Information: Semantic information in GSNs includes the user's published textual contents, the venue's categorical description, as well as the numerical ratings that can reflect the user's sentiment toward venues (Zhu, Shen, Jin, Xie, & Zheng, 2015). (1) *Textual contents*. Many studies show that textual information can offer complementary knowledge to profile users and venues respectively, and it is helpful to incorporate textual contents to enhance user preference modeling. Chen, Zhang, and Qin (2019) propose to profile user properties using textual reviews by merging sentence embeddings. In Tal and Liu (2019), textual contents of a user and a location are embedded into vectors, then they are fed into convolution layers to capture informative phrases that best represent the user and the location, respectively. Inspired by Word2Vec model (Mikolov, Chen, Corrado, & Dean, 2013), Chang, Park, Park, Kim, and Kang (2018) design a hierarchical embedding model that captures the characteristics of a location from text contents associated with POIs. Based on the learned location embeddings, they obtain user preference using traditional models like LSTM for user mobility prediction. (2) *Categorical description*. The categories of locations implicitly reflect users' activities when they visit these places. Recent literatures such as Wu, Li, Zhao, and Qian (2019a) and Yu et al. (2020) feed the category sequences of user trajectories into RNNs to learn category-level user preference, which is then combined with POI-level user preference for user next visit location prediction. (3) *Sentimental polarity*. Users' sentiment is often embedded in their reviews, and the sentimental reviews can reflect a user's opinion toward a venue. By taking the numerical rating of each review as the ground-truth, Wu et al. (2018) design a sentiment classification model specifically for reviews on Yelp. The learned review representation not only contains user and location properties, but also includes sentimental attribute, which is suitable for user preference modeling.

Social Relation: Social ties are inherent attributes in GSNs, which have been widely adopted in recommender systems. Considering that social friends are more likely to share common interests than strangers with regard to POIs, Cai et al. (2021) quantify the impact of social features on the predictability of a user's future visits. Enlightened by the recent advantage of Graph Convolutional Network (GCN), Wu et al. (2019) propose a social influence diffusion model to capture the recursive dynamic interactions between users. In Fan et al. (2019), GCN also proves to be effective to obtain user latent representation by integrating social friends' embeddings.

2.2. Neural models for user location prediction

In recent years, neural models have been the mainstream strategy for user mobility modeling and prediction. Existing neural models can be grouped into three categories: (1) RNN-based models, (2) Graph embedding based models, and (3) Encoder–decoder based models.

RNN-based models: A lot of works have been done on the variants of RNNs, which model sequential influence and temporal dynamics in user trajectories. Liu, Wu, Wang, and Tan (2016) extend the architecture of conventional RNN cell by incorporating temporal and spatial factors (ST-RNN) for the first time. Following ST-RNN, more variants such as DeepMove (Feng et al., 2018), STGN (Zhao et al., 2019), Flashback (Yang et al., 2020), LATL (Xu et al., 2021) are further proposed by introducing temporal–spatial gate to update users' next visit preference. However, as we note in Section 1, these works only focus on the sequential correlation for user next location prediction, ignoring the exact visit time of $\langle user, location \rangle$ interaction, thus are not able to predict a user's consumption location at a given time in the future.

Graph embedding based models: Graph embedding aims to project heterogeneous vertices (like user, location and time) into dense continuous space, so that the time-specific similarity between user and location can be calculated. Xie et al. (2016) propose a graph-based embedding framework that incorporates the joint effect of temporal cyclic effect, sequential relation as well as geographical influence to learn vector representations of venues, time slots and textual words. Yang et al. (2019b) and Zhao et al. (2016) use both social network and user check-in data to learn user embedding and location embedding, respectively. In Yang, Liu, Sun, and Bertino (2019a), in addition to the entity vertices, a transition vector is learned to bridge the gap between user and location, which plays the translation role to connect user and location.

Encoder–decoder based models: Auto-encoders can effectively capture the non-linear and non-trivial relationships between users and locations, and enable more complex data representation in the latent space. Ma et al. (2018) conceive a self-attentive encoder and a neighbor-aware decoder to adaptively select indicative locations that can reflect the user's preference. When the whole model is trained, the bottle-neck layer of the stacked encoders can be used as the user's latent representation. In Li, Shen, and Zhu (2018), a LSTM-based encoder–decoder framework is proposed to learn the spatial–temporal representation of user check-ins, which integrates multiple contextual factors in a unified manner.

In conjunction with neural models, attention mechanism also plays a key role when modeling user preference. Attention mechanism not only strengthens the ability of neural networks in capturing the long-range dependencies, but also improves the interpretability for prediction results. For example, Chen et al. (2019) propose to learn sentence-level user embedding by assigning weights to different textual reviews of a location. Similarly, Ma, Kang, Wu, Wang, and Liu (2019) aggregate word embeddings to characterize a location using a vanilla attention mechanism (self-attention), which assigns weights for different words. Guan et al. (2019) apply the aspect-level attention module and the user-level attention module to select salient features in order to find out users' most concerned items, respectively.

Discussion. As far as we know, there is no study concerning user consumption location prediction at a given time based on multi-modal geo-social networking data. In this article, we propose a *two-stage* approach comprising of a *Temporal Base Model* and a *Location Prediction Model*, where the former captures user preference in different time windows, and the latter fuses the temporal–spatial context as well as the side information to characterize a user's consumption probability toward a location at a given time.

3. Preliminaries

3.1. Problem definition

In this article, following common symbolic notation, calligraphic letters denote sets, upper case bold letters denote matrices, lower case bold letters represent column vectors, and non-bold letters represent scalars. Table 1 presents the notations used in this article.

Without loss of generality, we formally define the user consumption location prediction problem as follows. Suppose we have a user set $\mathcal{U} = \{u_1, u_2, \dots, u_M\}$ and a location set $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$, where M and N are the number of users and locations, respectively. For each user u, we chronologically organize his/her historical consumption behaviors as a sequence of quadruples $Q^u = \{(t_i{}^u, v_i{}^u, r_i{}^u, w_i{}^u)\}_{i=1}^{l_u}$, where $t_1{}^u \le t_2{}^u \le \dots \le t_{l_u}{}^u$, l_u is the number of u's consumption records, and each element $(t_i{}^u, v_i{}^u, r_i{}^u, w_i{}^u)$ means that user u visited POI $v_i{}^u$ at time $t_i{}^u$ with numerical rating $r_i{}^u$ and textual review $w_i{}^u$. Given the historical trajectory and the social link information of user u, as well as a future time t accurate to hour-level, we aim to predict a top-K location list for user u, so that the correct location he/she will visit at time t can be ranked as high as possible.

3.2. Dataset

Our study in this article is conducted using real-world Yelp data, as it provides sufficient time-sensitive $\langle user, location \rangle$ interactions as well as semantic information (see Fig. 1). To be specific, three datasets generated in three representative metropolises, i.e., *Toronto*, *Phoenix* and *Las Vegas* in North America, are built based on the original raw data published by Yelp.[5] Following the general

[5] https://www.yelp.com/dataset/challenge, access date: December 2019. The scale of original data is very large, which involves many cities in North America. We extract the above-mentioned three cities to verify this work since they are typical in geography and culture.

Table 1
Notations used in this article.

Symbol	Description
u, $\mathcal{U}$	A user and the set of users
v, $\mathcal{V}$	A location and the set of locations
M, N	The number of users and the number of locations
$\mathcal{Q}^u$, l_u	User u's historical consumption sequence and its length
$\mathcal{L}^u$	User u's consumption location set
$(t_i{}^u, v_i{}^u, r_i{}^u, w_i{}^u)$	User u visited POI $v_i{}^u$ at time $t_i{}^u$ with numerical rating $r_i{}^u$ and textual review $w_i{}^u$
$\mathbf{W}^u$, $\mathbf{w}_i^u$	The review embedding matrix and the embedding for a certain review of user u
$\mathbf{W}_{trans}$	A trainable matrix for review embedding normalization
$\mathbf{e}_i$	The normalized embedding for a certain review
$\mathbf{h}_t^u$	User u's preference embedding within a specific time slot t
$\mathbf{h}_u^{week}$, $\mathbf{h}_u^{day}$	User u's time-sensitive preference embedding under week-mode and day-mode
$\mathbf{p}_u$, $\hat{\mathbf{p}}_u$	User u's intrinsic latent representation and the output of *TBM*
$\{\mathbf{w}\}$, $\{\mathbf{b}\}$	The set of weight vectors and biases in the attention layer of *TBM*
$\{\mathbf{W}\}$	The set of transformation matrices in *LPM*
$\mathbf{u}^{Cat\_t_d}$, $\mathbf{u}^{Cat\_t_s}$	User u's consumption preference embedding toward location categories for week-mode and day-mode
$\mathbf{h}_v$, $\mathbf{v}^{Cat}$	Location v's averaged review embedding and category embedding
$\varphi(u, v, t)$	A real value indicating user u's normalized temporal visit probability toward location v at time slot t
$g(u, v)$	A real value indicating user u's normalized spatial preference toward location v
$\xi(u, v)$	A real value indicating the impact of user u's social friends on visiting location v
$\hat{s}(u, v, t)$	A real value indicating the final probability that user u visits location v at time slot t

Table 2
Statistics of the selected datasets.

Dataset	# Users	# Locations	# Reviews	Density	# Social links
Toronto	4,625	4,847	265,153	1.13%	47,392
Phoenix	11,225	7,879	546,344	0.58%	100,442
Las Vegas	13,542	7,392	641,652	0.60%	191,864

preprocessing way (Ma et al., 2018), for each dataset, we recursively remove inactive users and non-popular locations, so that each user has at least 20 consumption records and each location has been visited at least 20 times. After preprocessing, we show the statistics of the selected datasets in Table 2.

We plot the tag clouds of location categories over three datasets in Fig. 4. Note that as the vast majority of locations share common tag words like "Restaurants", "Food", "Nightlife" and "Bar", we just remove these words in the tag clouds to strengthen the representativeness of locations. For example, with regard to this category "Restaurants, Food, Asian Fusion, Chinese, Hotpot", we keep it as "Asian Fusion, Chinese, Hotpot" when visualizing the tag clouds. As can be seen, locations in *Toronto* are highly related to Asian elements since there are lots of Asian residents in this city; locations in *Las Vegas* are clearly associated with recreational attributes, which is inseparable from the urban function; by contrast, locations in *Phoenix* have a strong flavor of American lifestyles. Considering the scale and diversity of geo-tagged data over the three datasets, we believe they are qualified to validate our approach.

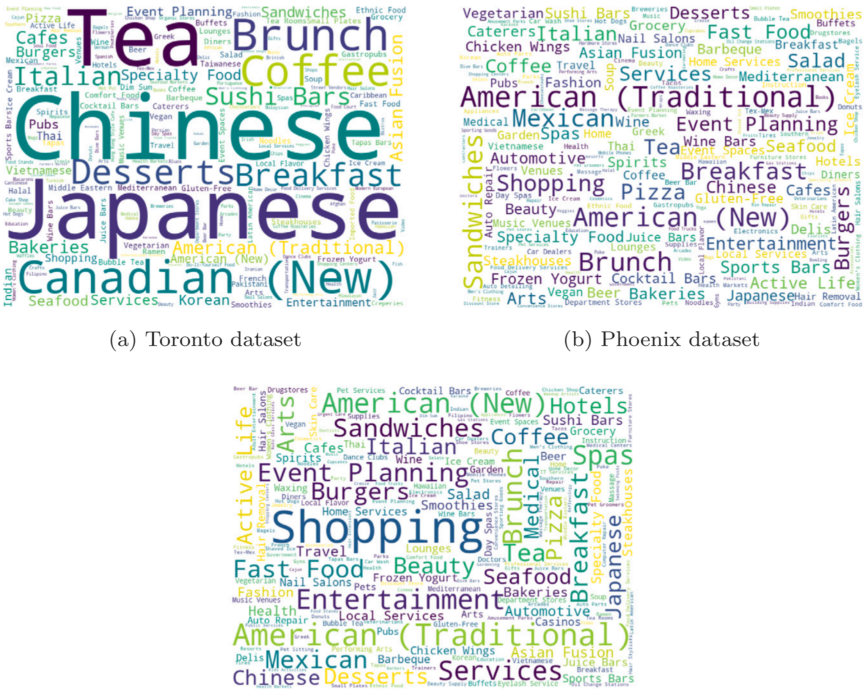
4. The methodologies

4.1. Temporal base model

4.1.1. Motivation

Different from existing studies, we propose to model user intrinsic time-sensitive consumption preference toward locations based on sentimental textual reviews (user textual reviews with numerical ratings). The motivation can be explained from two aspects. First, we hold that the functionality of a location is the critical factor that drives a user's consumption behavior, as it reveals his/her inner requirement or the intended purpose under specific temporal context (Li, Larson, & Hanjalic, 2017). Second, each user has unique consumption preference toward different types of locations. User preference can be learned from places where he/she leaves digital footprints, in this situation, user posted sentimental textual reviews are the natural medium to connect WHO (user), WHEN (time), WHERE (location) and WHAT (consumption topic).

Before introducing the *Temporal Base Model*, it is necessary to divide continuous time stamps into discrete time windows. Inspired by Gao et al. (2013) which shows that user mobility pattern may be cycled around one day and one week, we combine the 7-day week-mode and 24-hour day-mode to form a fixed number of time windows. Specifically, to alleviate data sparsity and express the user periodical mobility pattern more clearly, we group the 24-hour day-mode into five time slots, i.e., night (0–6 am), morning (6–11 am), noon (11 am–14 pm), afternoon (14–19 pm) and evening (19–24 pm). In this way, we divide one day into 5 slots and one week into 7 days, so that we have totally 35 (5×7) time windows to represent the temporal context.



(a) Toronto dataset (b) Phoenix dataset

(c) Las Vegas dataset

Fig. 4. Tag clouds of location categories over three datasets, where larger font indicates higher tag frequency and vice versa. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.1.2. Model components

In Fig. 5, we display the graphical overview of the conceived *Temporal Base Model (TBM)*,[6] where the left part models user preference under week-mode, and the right part models user preference under day-mode. By combining hierarchical neural review representation and temporal topic modeling through multi-layer neural networks, *TBM* aims to learn time-sensitive user preference under week-mode and day-mode time windows (shown in the pink shadow area).

As can be seen from Fig. 5, both week-mode and day-mode user preference modeling have the same input, i.e., the vector representation of user textual reviews. Based on the hierarchical attention mechanism (review-level attention and window-level attention), the two parts output time-sensitive user preference under week-mode ($\mathbf{h}_u^{week}$) and day-mode ($\mathbf{h}_u^{day}$), respectively. By concatenating $\mathbf{h}_u^{week}$ and $\mathbf{h}_u^{day}$ as $\mathbf{h}_u$, *TBM* maps it to the user intrinsic latent representation $\mathbf{p}_u$, through the multi-layer neural network architecture. Note that $\mathbf{p}_u$ is obtained based on user textual reviews using TLDA model (Zhou et al., 2018), which not only considers the number of times each word appears in a document (“bag-of-words” assumption), but also involves temporal factors, so as to depict user topic distribution under temporal context (Chen, 2017). In this article, following Zhou et al. (2019), we treat $\mathbf{p}_u$ as the ground-truth user intrinsic latent representation, so as to guide the learning of time-sensitive user preference. For this purpose, the concatenation vector $\mathbf{h}_u$ is treated as the input of the multi-layer neural network architecture, and $\hat{\mathbf{p}}_u$ is treated as the output with $\mathbf{p}_u$ being the ground-truth, where each dimension of the vector represents a class. In this way, we transform the user preference modeling problem into a multi-class classification task.

Input Module. To embed user sentimental textual reviews into low dimensional vector space, we apply the HUAPA model (Wu et al., 2018) to obtain the embedding of each review. The learned review embedding not only contains user and location properties, but also encodes the user’s sentiment toward the location, which is suitable for our task. Once the HUAPA model is trained using our datasets described in Section 3.2, we can obtain a review embedding matrix $\mathbf{W}^u$ for each user u, where each column $\mathbf{w}_i^u \in \mathbb{R}^d$ in $\mathbf{W}^u$ represents a review embedding and d is the size of review embeddings.

Normalization Module. We keep $\mathbf{W}^u$ fixed in the whole training process of *TBM*. Above the input module, we employ a trainable matrix $\mathbf{W}_{trans} \in \mathbb{R}^{d\times d}$ to customize the pre-trained review embedding $\mathbf{w}_i^u$. In this way, the customized review embeddings can be task oriented. The customized review embedding $\mathbf{e}_i$ for a certain review is normalized according to Eq. (1):

$$\mathbf{e}_i = \frac{\mathbf{W}_{trans} \cdot \mathbf{w}_i^u}{\|\mathbf{W}_{trans} \cdot \mathbf{w}_i^u\|} \tag{1}$$

[6] An alternative Temporal Base Model based on all 35 time windows is introduced in Section 4.3.

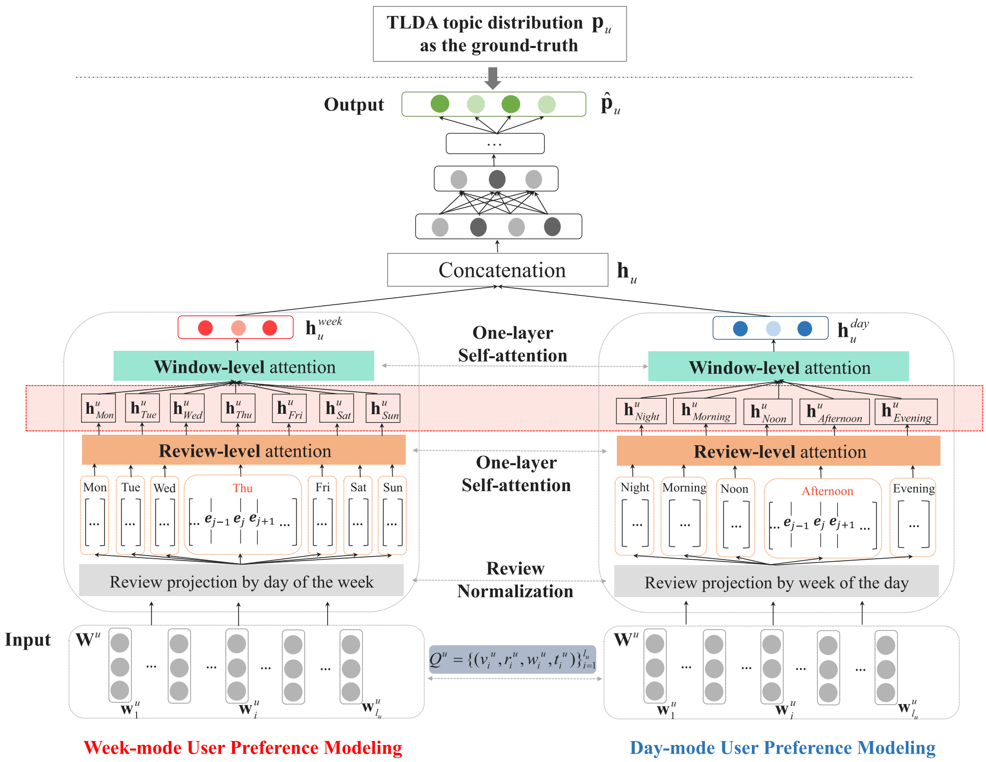


Fig. 5. Temporal Base Model for hierarchical consumption preference modeling. Each grid in the pink shadow area (for example, $\mathbf{h}^u_{Mon}$) represents the time-sensitive user consumption preference under corresponding time window. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where $\|\cdot\|$ represents the Euclidean norm. By normalizing review embeddings with their corresponding Euclidean norms, the calculation of inner-product between user and location is equivalent to calculating their cosine similarity.

Hierarchical Attention Module. According to the time stamp associated with each review, we can group reviews into separate week-mode and day-mode time windows. In this way, we get the review embedding matrix within each time window. The review embedding matrix of the tth time window is represented as $\mathbf{D}^u_t$:

$$\mathbf{D}^u_t = \begin{bmatrix} & | & | & | & \\ \dots & \mathbf{e}_{j-1} & \mathbf{e}_j & \mathbf{e}_{j+1} & \dots \\ & | & | & | & \end{bmatrix} \tag{2}$$

where $\mathbf{D}^u_t \in \mathbb{R}^{d\times L_t}$ and $\mathbf{e}_j \in \mathbb{R}^d$. L_t is number of reviews within time window t.

For $\mathbf{D}^u_t$, the review-level attention mechanism is applied to assign different importance on each review, after which the review embeddings in $\mathbf{D}^u_t$ are aggregated in a weighted manner to characterize the corresponding time window. In other words, we linearly aggregate review embeddings in $\mathbf{D}^u_t$ to represent the time-sensitive user preference for time window t. To achieve this goal, the one-layer self-attention is adopted to compute the attention weight of each review according to Eq. (3):

$$\mathbf{a}_t = softmax(tanh(\mathbf{w}_r{}^T \cdot \mathbf{D}^u_t + \mathbf{b}_r)) \tag{3}$$

where $\mathbf{a}_t$ is a vector containing the weights of each review, $\mathbf{w}_r \in \mathbb{R}^d$ and $\mathbf{b}_r \in \mathbb{R}^{1\times L_t}$ are the weight vector and bias in the review-level attention layer to be learned, $softmax(\cdot)$ ensures that the sum of computed weights equals to 1.

Based on the review weight vector $\mathbf{a}_t$, we aggregate reviews within $\mathbf{D}^u_t$ according to Eq. (4):

$$\mathbf{h}^u_t = \sum_{1\le j\le L_t, \mathbf{e}_j\in \mathbf{D}^u_t} a_{tj}\cdot \mathbf{e}_j \tag{4}$$

where a_{tj} is the weight of the jth review in $\mathbf{D}^u_t$.

Note that we actually have two different review-level attention layers for week-mode preference modeling and day-mode preference modeling. For simplicity, we do not discriminate them by giving another couple of equations. Hereafter, we obtain 7 representations for week-mode user preference and 5 representations for day-mode user preference, as shown in the pink shadow area of Fig. 5.

Similarly, we hold that different time windows contribute unequally when characterizing a user. Therefore, we further apply the window-level attention mechanism to assign different weights on each time windows, after which the time-sensitive user representations can be aggregated to characterize the global user representation under week-mode and day-mode, respectively. We take week-mode as the example, the attention weight of each time window is computed according to Eq. (5), and we aggregate different time windows according to Eq. (6):

$$\mathbf{b}^{week} = softmax(tanh(\mathbf{w}_w{}^T \cdot \mathbf{H}_u + \mathbf{b}_w)) \tag{5}$$

$$\mathbf{h}_u^{week} = \sum_{j\in\{0,1,\dots,6\}} b_j^{week} \cdot \mathbf{h}_j^u \tag{6}$$

where $\mathbf{H}_u \in \mathbb{R}^{d\times 7}$ is the week-mode user preference matrix, $\mathbf{w}_w \in \mathbb{R}^d$ and $\mathbf{b}_w \in \mathbb{R}^{1\times 7}$ are the weight vector and bias in the window-level attention layer to be learned, and $\mathbf{h}_j^u$ is the jth time-sensitive user preference in $\mathbf{H}_u$. The obtained vector $\mathbf{h}_u^{week}$ represents the global week-mode user preference, following the similar way, the global day-mode user preference $\mathbf{h}_u^{day}$ can also be obtained.

Output Module. The concatenation representation $\mathbf{h}_u = [\mathbf{h}_u^{week} \oplus \mathbf{h}_u^{day}]$ is fed into a multi-layer neural network architecture, which finally outputs a probability distribution $\hat{\mathbf{p}}_u$:

$$\hat{\mathbf{p}}_u = softmax(MLP(\mathbf{h}_u)) \tag{7}$$

where $MLP(\cdot)$ is the multi-layer perceptron function to project the input vector $\mathbf{h}_u$ into the probability distribution $\hat{\mathbf{p}}_u$. For the activation function in each layer of $MLP(\cdot)$, we use the $ReLu$ function as it yields favorable results. A tentative validation using *Toronto* dataset indicates that two fully connected layers are sufficient for this task, as deeper network adds more complexity without bringing in evident improvement in return.

Loss Function. With user intrinsic latent representation $\mathbf{p}_u$ as ground-truth, we regard user preference modeling as a multi-class classification task, so that the cross-entropy loss is used to optimize *TBM*:

$$Loss\_tbm = -\sum_{u\in\mathcal{U}} \sum_i \mathbf{p}_u(i) \cdot \log(\hat{\mathbf{p}}_u(i)),\ \forall i = 1, 2, \dots, Z \tag{8}$$

where i is one of the Z topics (classes) trained on textual reviews using TLDA (Zhou et al., 2018), and $\mathbf{p}_u(i)$ is the probability of topic i in the ground-truth user latent representation $\mathbf{p}_u$.

Model Learning. From bottom up, parameters to be learned in *TBM* include the matrix $\mathbf{W}_{trans}$ for review embedding normalization, the set of weight vectors $\{\mathbf{w}\}$ and biases $\{\mathbf{b}\}$ in the hierarchical attention module, as well as the weight vectors and biases in the multi-layer neural network architecture. Note that *TBM* is indeed a feed-forward neural network, all the parameters in the above loss function are differentiable.

Because we treat user preference learning as a multi-class classification task where user intrinsic temporal topic distribution $\mathbf{p}_u$ is the ground-truth (obtained by TLDA (Zhou et al., 2018)), the training process of *TBM* is actually a supervised learning problem, where gradient descent can be applied to optimize the loss function. We employ Adam (Kingma & Ba, 2015) with mini-batch to train the parameters automatically. Following Wu et al. (2018), the weight matrices (vectors) and biases are initialized according to the uniform distribution $U(-0.01, 0.01)$. We set the initial learning rate of Adam as 0.001, and the batch size is set to 256.

4.2. Location prediction model

Based on the learned time-sensitive user preference, i.e., $\mathbf{h}_t^u$ shown in the pink shadow area of Fig. 5, we build the *Location Prediction Model (LPM)* which characterizes the interaction between user and location at a given time. The graphical overview of *LPM* is shown in Fig. 6. As we can see, *LPM* is composed of three components, namely, temporal interaction $\varphi(u, v, t)$, geographical interaction $g(u, v)$, as well as social interaction $\xi(u, v)$. The details of each component are explained as follows.

4.2.1. Model components

Input Module. Our aim is to fit user u's consumption probability toward location v at time t, therefore, we take each of user u's consumption behavior $\langle u, v, t\rangle$ as the input of *LPM*. For each input sample $\langle u, v, t\rangle$, we project t into day-of-week time window t_d and slot-of-day time window t_s, respectively, so as to obtain the time-sensitive user consumption preference, i.e., $\mathbf{h}_{t_d}^u$ and $\mathbf{h}_{t_s}^u$, according to the learned time-sensitive user preference by *TBM*. As we hold that the functionality of a location drives users' consumption behavior, we obtain the user's consumption topic preference by averaging the location category embeddings in his/her footprints.[7] In this way, we obtain user consumption preference toward location categories for week-mode and day-mode, i.e., $\mathbf{u}^{Cat\_t_d}$ and $\mathbf{u}^{Cat\_t_s}$, respectively.

Meanwhile, for current location v, we take the average review embeddings associated with v, so as to capture the latent representation of v, i.e., $\mathbf{h}_v$, from the review level. In addition, we use $\mathbf{v}^{Cat}$ to represent the category embedding of location v.

Feature Fusion Module. We introduce non-linear fusion layers to fuse the multifaceted contexts.

First, we fuse different kinds of time-sensitive information to capture user u's specific preference at time t. Following Li, Pi, Lin, Ahmed Khan, Cui (2020) and Li, Wu et al. (2020), the fusion layers take $[\mathbf{h}_{t_d}^u, \mathbf{h}_{t_s}^u]$ and $[\mathbf{u}^{Cat\_t_d}, \mathbf{u}^{Cat\_t_s}]$ as input, and output $\mathbf{h}_u^{Tem}$ and

[7] For the tokenized categories associated with locations, we use the pre-trained 100d Glove word embedding as the word vector, and average the word vectors to get the final location category embedding. The Glove word embeddings can be accessed here: https://nlp.stanford.edu/projects/glove/.

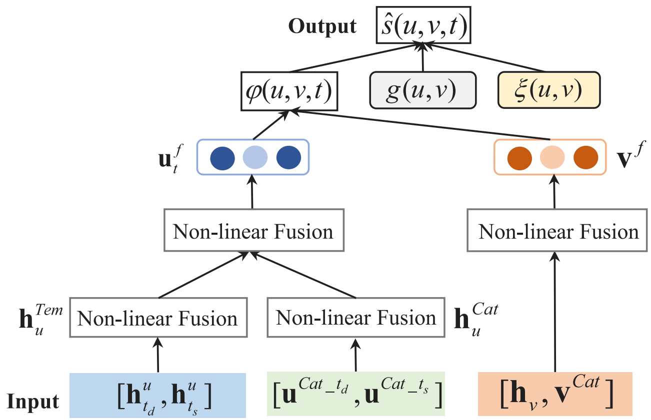


Fig. 6. Location Prediction Model for user consumption location prediction at a given time.

$\mathbf{h}_u^{Cat}$, which represent user u's temporal location preference and consumption topic preference, respectively. The non-linear feature fusions are described in Eqs. (9) and (10), respectively.

$$\mathbf{h}_u^{Tem} = f(\mathbf{W}^{Tem} \cdot [\mathbf{h}_{t_d}^u, \mathbf{h}_{t_s}^u]) \tag{9}$$

$$\mathbf{h}_u^{Cat} = f(\mathbf{W}^{Cat} \cdot [\mathbf{u}^{Cat\_t_d}, \mathbf{u}^{Cat\_t_s}]) \tag{10}$$

where $\mathbf{W}^{Tem}, \mathbf{W}^{Cat} \in \mathbb{R}^{d \times d}$ are the transformation matrix to be learned, $f(\cdot)$ is the non-linear function. Note that we just omit the bias term in the fusion layer for the convenience of notation.

Next, after $\mathbf{h}_u^{Tem}$ and $\mathbf{h}_u^{Cat}$ are derived, we use another one-layer fully connected neural network to fuse them (Eq. (11)). In this way, we get user u's global latent representation at time t, i.e., $\mathbf{u}_t^f$.

$$\mathbf{u}_t^f = f(\mathbf{W_1}^f \times [\mathbf{h}_u^{Tem}, \mathbf{h}_u^{Cat}]) \tag{11}$$

where $\mathbf{W_1}^f$ is the transformation matrix to be learned. Please note that, to merge heterogeneous information, we have multiple fusion options, one of which is to concatenate them by setting $\mathbf{W_1}^f$ as the identity matrix and setting $f(\cdot)$ as the concatenation function.

In the end, we can also derive location v's global latent representation $\mathbf{v}^f$, which is time-independent. We present the fusion process using Eq. (12):

$$\mathbf{v}^f = f(\mathbf{W_2}^f \cdot [\mathbf{h}_v, \mathbf{v}^{Cat}]) \tag{12}$$

where $\mathbf{W_2}^f$ is the transformation matrix to be learned. In this way, we can align user latent representation $\mathbf{u}_t^f$ and location latent representation $\mathbf{v}^f$ as both of them are derived using textual review and categorical information.

Interaction Module. For temporal interaction between user u and location v at time t, we apply cosine similarity to indicate the normalized temporal visit probability of user u toward location v:

$$\varphi(u, v, t) = \mathbf{u}_t^f \odot \mathbf{v}^f \tag{13}$$

where $\odot$ stands for the inner product.

For geographical interaction, we aim to enable user u's historical consumption trajectory to exert more influence on nearby unvisited locations. According to Ma et al. (2018), we adopt kernel density estimation (KDE) to incorporate geographical influence. Specifically, given user u's consumption location set $\mathcal{L}^u$ and a location v, we compute a normalized distance factor for v according to Eq. (14):

$$g(u, v) = p(v|\mathcal{L}^u) = \frac{1}{\delta|\mathcal{L}^u|} \sum_{v_i \in \mathcal{L}^u} \phi\left(\frac{dist(v_i, v)}{\delta}\right) \tag{14}$$

where $dist(x, y)$ represents the geographical distance between location x and location y, $\phi(\cdot)$ is the standard Gaussian distribution function, δ is the bandwidth parameter and we set $\delta = 1.06\tilde{\delta}|\mathcal{L}^u|^{\frac{1}{5}}$ according to Zhang and Chow (2015).

For social interaction, we consider the user's social friends as another indicator to strengthen the predictive power of *LPM*. Specifically, for each friend a of user u, we compute the frequency of location v in a's footprints $\mathcal{L}^a$, multiplied by the normalized counter-based similarity between u and a, so as to quantify the social influence of a to u with regard to location v. The normalized social influence is quantified according to Eq. (15):

$$\xi(u, v) = \frac{\sum_{a \in \mathcal{F}(u)} sim(u, a) \cdot freq(a, v)}{\sum_{a \in \mathcal{F}(u)} sim(u, a)} \tag{15}$$

where $\mathcal{F}(u)$ is the social neighbors of user u, $sim(u,a)$ indicates the similarity between u and a, and $freq(a,v)$ stands for the frequency of v in a's footprints $\mathcal{L}^a$.

Output Module. Based on the afore-mentioned three types of interactions between user u and location v, we derive the consumption probability of user u toward location v at time t by a scoring function:

$$\hat{s}(u,v,t) = \varphi(u,v,t) + \eta \cdot g(u,v) + \lambda \cdot \xi(u,v) \tag{16}$$

where η and λ are weighting parameters indicating the influential levels of geographical interaction and social interaction, respectively.

4.2.2. Model training

In this article, we train *LPM* with a learning-to-rank method, which is widely used in location prediction research (Gao et al., 2018; Guan et al., 2019). Specifically, we employ Bayesian Personalized Ranking (BPR) (Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, 2009) to optimize *LPM*, as BPR can make use of the unobserved time-sensitive $\langle u,v\rangle$ interactions by learning a pair-wise ranking loss in the training process of *LPM*. Given an observed triple $\langle u,v^+,t\rangle$ (generally called a positive example), and a corresponding unobserved triple $\langle u,v^-,t\rangle$ (generally called a negative example), where the former means user u visits location v^+ at time t and the latter means user u does not visit location v^- at time t, BPR should give a higher consumption probability $\hat{s}(u,v^+,t)$ for $\langle u,v^+,t\rangle$ than $\hat{s}(u,v^-,t)$ for $\langle u,v^-,t\rangle$. Following Gao et al. (2018), the loss function with a regularization term for optimizing *LPM* is described below:

$$Loss\_lpm = -\sum_{u\in\mathcal{U}}\sum_{\langle v^+,v^-\rangle\in\mathfrak{R}_u} \ln\sigma(\hat{s}(u,v^+,t) - \hat{s}(u,v^-,t)) + \varepsilon\|\Theta\|^2 \tag{17}$$

where $\sigma(\cdot)$ is the sigmoid function, Θ is the parameters of the non-linear fusion layers, ε is the regularization coefficient.

As we employ the BPR loss function, it is necessary to retrieve negative samples for each positive sample. Specifically, for each positive sample $\langle u,v^+,t\rangle$, we randomly sample 10 negative samples $\langle u,v^-,t\rangle$. We use $\mathfrak{R}_u$ to represent user u's training set, which consists of positive samples and negative samples. The training set for all users is denoted as $\mathfrak{R}$. Note that for each epoch during model training, we would re-sample negative samples for each positive sample, so that each negative sample only gives very weak negative signal in the training process. Parameters to be learned in *LPM* include the set of transformation matrices in feature fusion layers. As *LPM* is also a feed-forward neural network, where all the parameters in the loss function are differentiable, we apply Adam (Kingma & Ba, 2015) with mini-batch to train the parameters automatically. The parameter set is initialized according to the uniform distribution $U(-0.01, 0.01)$, and the batch size is set to 256. Algorithm 1 below illustrates the whole training process of *LPM*.

Algorithm 1: Training algorithm of *LPM*

Input : users' training set $\mathfrak{R}$
Output: the parameter set Θ of *LPM*.

initialize the parameter set Θ;
while *exceed(maximum number of iterations)==FALSE* **do**
    Randomly select a batch of training instances $\mathfrak{R}^b$;
    for *each user u in $\mathfrak{R}^b$* **do**
        for *each pair $\langle u,v^+,t\rangle$ and $\langle u,v^-,t\rangle$ of user u in $\mathfrak{R}^b$* **do**
            compute the consumption probability $\hat{s}(u,v^+,t)$ and $\hat{s}(u,v^-,t)$, according to Eq. (16);
        end
    end
    find Θ minimizing the objective function (Eq. (17)) with $\mathfrak{R}^b$;
end
return *the parameter set Θ*

4.2.3. User consumption location prediction

Once *LPM* is trained, we can predict a user's consumption location at a given time by producing a top-K location list. Specifically, for user u and the designated time t_{given}, we compute the consumption probability $\hat{s}(u,v,t_{given})$ for each candidate location v, and select the locations with top-K probabilities as the prediction result. Algorithm 2 below illustrates the whole prediction process.

4.3. An alternative to temporal base model

In addition to the proposed *Temporal Base Model* in Section 4.1, we introduce an alternative model that incorporates all the 35 time windows, the graphical framework of the model is shown in Fig. 7. In this case, after the model is trained, we can directly get time-sensitive user preference $\mathbf{h}_u^{Tem}$ and $\mathbf{h}_u^{Cat}$ without the first two non-linear fusion layers in *LPM*. The way to obtain user u's global latent representation $\mathbf{u}_t^f$ as well as location v's global latent representation v^f is the same to Eqs. (11) and (12), respectively.

To distinguish the two models, we name the *Temporal Base Model* proposed in Section 4.1 as *TBM1*, and the model proposed here as *TBM2*. For *TBM1*, the sizes of the last two fully-connected layers are empirically set as 128 and 64. For *TBM2*, as it does not involve the concatenation operation, we set them as 64 and 32, respectively. The predictive performance of these two models will be compared in the following section.

Algorithm 2: User consumption location prediction algorithm

Input : u, t_{given}, $\mathcal{Q}^u$, $\mathcal{V}$, $\mathcal{F}(u)$, Θ, K.
Output: the predicted top-K list L.
transform time t_{given} into t_d and t_s;
obtain time-sensitive user preference $\mathbf{h}^u_{t_d}$, $\mathbf{h}^u_{t_s}$, $\mathbf{u}^{Cat\_t_d}$ and $\mathbf{u}^{Cat\_t_s}$;
for *each candidate location* $v \in \mathcal{V}$ **do**
  compute $\varphi(u, v, t_{given})$ according to Eq. (13);
  compute $g(u, v)$ according to Eq. (14);
  compute $\xi(u, v)$ according to Eq. (15);
  compute the consumption probability $\hat{s}(u, v, t_{given})$ according to Eq. (16);
end
sort the candidate locations in a descending order;
produce the prediction list L=top-$K(\mathcal{V})$;
return L

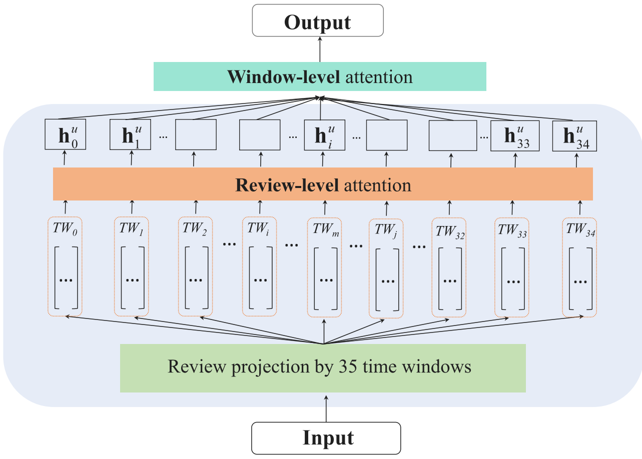


Fig. 7. An alternative Temporal Base Model based on all 35 time windows.

5. Experiments

We carry out the experiments on a Dell workstation with dual processors (2 × Intel Xeon E5 @ 2.10 GHz), four graphic processing units (NVIDIA TITAN Xp, 12GB), and 188GB RAM. The operating system of the workstation is 64-bit Ubuntu 16.04. All the codes are written in Python 3.7 with the deep learning framework PyTorch[8] 1.1.0.

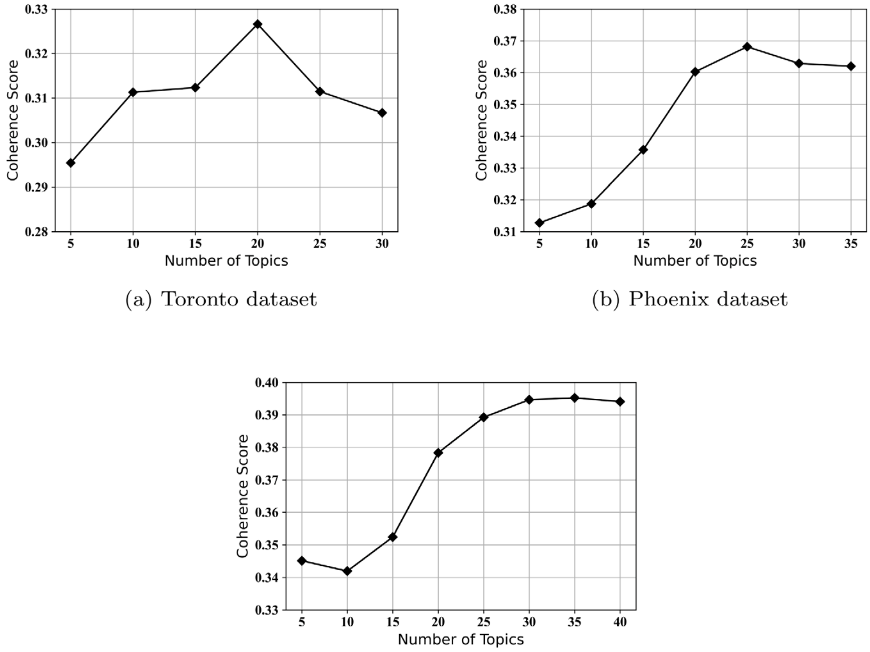
The experiment is based on train–validation–test mode. For each user u in each dataset described in Section 3.2, we use the chronologically former 80% footprints in $\mathcal{Q}^u$ for training, the middle 10% for validating, and the last 10% for testing. We conduct extensive experiments using three real-world Yelp datasets to answer the following four questions (RQ1 ∼ RQ4), which aim at verifying the effectiveness of the proposed *two-stage* framework:

- **RQ1**: How many topics should be assigned on each dataset for the TLDA model?
- **RQ2**: Which *Temporal Base Model* performs better when combined with the *Location Prediction Model*?
- **RQ3**: How does the *two-stage* framework perform compared to other state-of-the-art models?
- **RQ4**: How do the key factors in *LPM* impact the performance of our prediction framework?

5.1. Evaluation metrics

According to Xu, Fu et al. (2020), which reviews various evaluation metrics used for user location prediction, we select three metrics from different aspects to verify the predictive performance of the *two-stage* framework.

[8] https://www.pytorch.org/



(a) Toronto dataset (b) Phoenix dataset

(c) Las Vegas dataset

Fig. 8. Coherence value scores for different number of topics among three datasets.

The first metric is $Accuracy@K$ (abbreviated as $Acc@K$), which measures the ratio of successfully predicted locations in the top-K list (Feng et al., 2018; Xu et al., 2021). The calculation method for $Acc@K$ is as follows:

$$Acc@K = \frac{\#hit@K}{|Test|} \tag{18}$$

where $\#hit@K$ denotes the number of times that we successfully predict in the top-K list. $|Test|$ denotes the number of tests, i.e., the number of test cases.

The second metric is Average Percentile Rank (APR), which considers the rank of correctly predicted location in the top-K list (Cao et al., 2018). The intended purpose of APR metric is that: the higher the rank of ground-truth location in the list, the larger the APR metric value, and vice versa. For this purpose, Percentile Rank (PR) is firstly calculated for each single test as follows:

$$PR = \frac{|\mathcal{L}| - rank(k)}{|\mathcal{L}|} \tag{19}$$

where $|\mathcal{L}|$ is the size of candidate location set, and $rank(k)$ is the rank of the ground-truth location in the sorted list. Specifically, as there are too many unvisited locations for each user, to reduce the computational cost, for each test case we randomly select 100 unvisited locations and combine them with the positive location (i.e., $|\mathcal{L}| = 101$). We repeat this procedure 10 times and report the average result of PR. Finally, we average all the test cases to calculate APR metric.

The third metric is coverage ratio, which measures the proportion of the users who are given at least one correct prediction with regard to $Acc@K$. This metric is often used to measure the usability of a location prediction model for different user groups.

5.2. Baseline models

As we aim to predict where a user will consume at a given time in the future, most of the existing studies (for example, the RNN-based models) concerning user next location prediction cannot be used. As a consequence, we compare the proposed *two-stage* framework with the following competitive approaches that involve exact time as the temporal context for user visit location prediction.

- **Random Embedding**: This is a simple baseline embedding method, which uses the randomly initialized vectors for the input of *LPM*.
- **Score_Rank** (Cao et al., 2018): This model fuses a set of hand-crafted features including user mobility feature, global popularity feature, and temporal feature to design a supervised learning framework. It combines a classification model and a scoring model to calculate a user's visit probability toward a location at the given time.

- **UCGT** (Yin et al., 2016): This is a Bayesian generative model that characterizes the generation of user check-ins within GSNs. Users sharing similar spatial–temporal topics are firstly clustered into communities. Based on the discovered communities, given a user u and a time slot t, we can estimate the probability of u visiting each candidate location v through a probability distribution.
- **STELLAR** (Zhao et al., 2016): This is a tensor factorization based ranking model which learns a scoring function to evaluate the probability of a user visiting a location at a given time. Although it is conceived for predicting a user's next visit location, we can use it in our task by removing the sequential influence between two successive check-ins in the testing phase.
- **CAPE** (Chang et al., 2018): Based on a hierarchical embedding strategy, this model learns location embeddings using both textual contents and temporal–spatial context in user trajectories. We use the learned location embeddings as the input of *LPM*.
- **Venue2Vec** (Xu, Cao et al., 2020): By treating each location in the sampled user trajectories as a word, this model learns location embedding based on Skip-gram model. Then, users' preference can be estimated by averaging all location embeddings in a specific time window. For location prediction, a distance factor using kernel density estimation is added to the similarity between the user and a candidate location.
- **LBSN2Vec** (Yang et al., 2019b): This is the state-of-the-art model for user visit location prediction at a given time. It conceives a novel random walk strategy to jointly sample friendships and check-in hyper-edges from the GSN hyper-graph, and then propose to learn node embeddings from hyper-edges by preserving the cosine proximity between nodes. Based on the learned user embedding, time slot embedding and POI embedding, it computes two similarities, one between the user and the POI, and one between the time slot and the POI. Then the two similarities are added up to represent the final preference score of the user toward a location.

5.3. Experimental settings

For training *TBM*, we empirically set the size of review embedding $d = 128$, and set the size of location category embedding as 100. Depth of the multi-layer neural network architecture that maps $\mathbf{h}_u$ to $\hat{\mathbf{p}}_u$ is set to 2. The optimal size of user intrinsic preference $\mathbf{p}_u$ will be determined by experiments. The maximum number of reviews within each time window is set to 50. Parameters (i.e., the normalization matrix $\mathbf{W}_{trans}$, the set of weight vectors $\{\mathbf{w}\}$ and biases $\{\mathbf{b}\}$ in the hierarchical attention layers) are initialized according to the uniform distribution $U(-0.01, 0.01)$. The batch size is set to 256. The initial learning rate of Adam is 0.001.

For training *LPM*, we employ a maximum of 500 epochs with early stopping strategy. For each training sample $\langle u, v^+, t\rangle$, we randomly choose 10 negative samples $\langle u, v^-, t\rangle$ in order to optimize the BPR loss. Parameters (i.e., $\{\mathbf{W}\}$ in the non-linear fusion layers) are initialized according to the uniform distribution $U(-0.01, 0.01)$. The batch size is set to 256. The initial learning rate of Adam is 0.001. The weighting parameters for geo-influence η and social relation λ in Eq. (16) will be discussed in the experiments to find the best combination. In the end, for the regularization parameter ϵ in Eq. (17), we determine the optimal value by grid search in the range of $\{0.001, 0.01, 0.1\}$.

The parameters for the baseline approaches are initialized as in the corresponding papers, and are then carefully tuned to achieve optimal performance.

5.4. Empirical analysis on user intrinsic representation learning(RQ1)

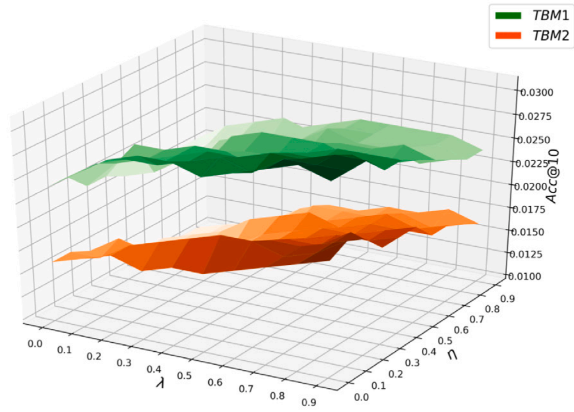
As we emphasize in Section 4.1, the key part of *TBM* is the user intrinsic latent representation $\mathbf{p}_u$, which is taken as the ground-truth for hierarchical user preference modeling. To determine the size of $\mathbf{p}_u$, each user's textual reviews are combined into one document, and the whole corpus is fed into TLDA (Zhou et al., 2018) to obtain the topic distribution of each user.

In order to find out the optimal number of topics for each dataset, we plot the coherence value (CV) curve (Röder, Both, & Hinneburg, 2015) among different number of topics (Fig. 8). Technically, the higher the CV score, the better the clustering effect of topic modeling. Based on the CV curves shown in Fig. 8, we empirically select 20, 25 and 30 as the optimal number of topics for *Toronto* dataset, *Phoenix* dataset and *Las Vegas* dataset, respectively. As a consequence, the size of $\mathbf{p}_u$ for each dataset is determined as 20, 25 and 30, respectively. As we regard the learning of *TBM* as a multi-class classification task, this means the best dimensions (i.e., the number of classes) of user intrinsic embedding are 20, 25 and 30 for three cities in our datasets.

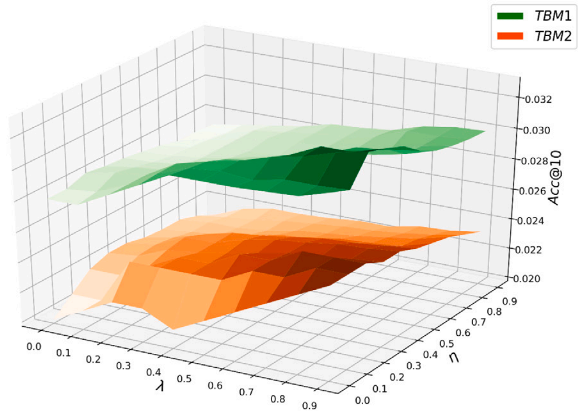
5.5. Performance comparison between TBM1 and TBM2 (RQ2)

Note that the *two-stage* prediction framework is indeed a hierarchical pipeline, where the *Temporal Base Model* learns the time-sensitive user latent representation, after which the *Location Prediction Model* learns the scoring function to calculate a user's consumption probability toward a location at a given time. For clarity, we denote the *two-stage* framework based on *TBM1* as *TBM1_LPM*, and denote the *two-stage* framework based on *TBM2* as *TBM2_LPM*.

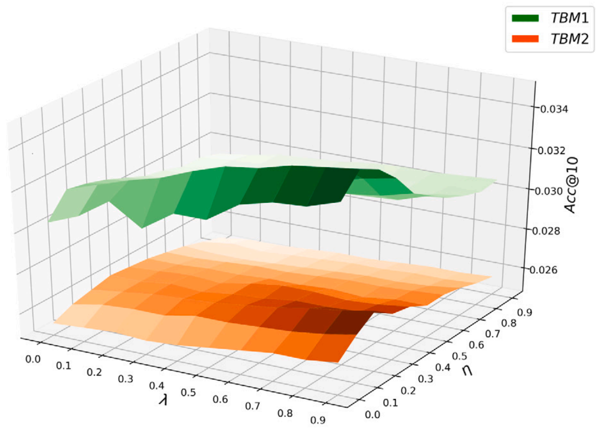
In order to compare the predictive performance of *TBM1* and *TBM2*, we should actually train *TBM1_LPM* and *TBM2_LPM*, respectively, and then quantify the prediction accuracy of these two models. To ensure the comprehensiveness of comparison, we need to obtain the prediction accuracy by traversing all possible hyper-parameters, such as the geo-influence weight η and social weight λ in Eq. (16), and the regularization coefficient ϵ in Eq. (17). For clarity, we use $Acc@10$ as the evaluation metric, and tune η and λ by grid search from 0 to 1 with step 0.1. The experimental results using three datasets are plotted in Fig. 9, where the regularization parameter ϵ is set to 0.001, because this value generally works well on all datasets. As can be seen in Fig. 9, no



(a) Toronto dataset



(b) Phoenix dataset



(c) Las Vegas dataset

Fig. 9. $Acc@10$ values along with different η and λ using three datasets. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

matter which dataset is used, the prediction accuracy of *TBM1_LPM* is significantly greater than that of *TBM2_LPM*. This observation can be explained from two aspects.

Table 3
$Acc@K$ results of various models where the best results are highlighted using **bold font**.

	Toronto			Phoenix			Las Vegas		
	$Acc@5$	$Acc@10$	$Acc@20$	$Acc@5$	$Acc@10$	$Acc@20$	$Acc@5$	$Acc@10$	$Acc@20$
TBM1_LPM	**0.0233**	**0.0291**	**0.0334**	**0.0276**	**0.0347**	**0.0403**	**0.0295**	**0.0394**	**0.0461**
Random	0.0104	0.0111	0.0157	0.0122	0.0165	0.0264	0.0138	0.0226	0.0271
Score_Rank	0.0146	0.0182	0.0249	0.0185	0.0233	0.0292	0.0233	0.0294	0.0359
UCGT	0.0138	0.0185	0.0242	0.0173	0.0216	0.0278	0.0205	0.0281	0.0347
STELLAR	0.0126	0.0164	0.0231	0.0162	0.0199	0.0256	0.0187	0.0264	0.0322
CAPE	0.0152	0.0187	0.0253	0.0188	0.0242	0.0285	0.0236	0.0291	0.0366
Venue2Vec	0.0182	0.0232	0.0266	0.0214	0.0257	0.0331	0.0231	0.0310	0.0364
LBSN2Vec	0.0195	0.0264	0.0282	0.0247	0.0283	0.0358	0.0254	0.0323	0.0387

Table 4
APR results of various models where the best results are highlighted using **bold font**.

	Toronto	Phoenix	Las Vegas
TBM1_LPM	**0.384**	**0.442**	**0.454**
Random	0.257	0.296	0.306
Score_Rank	0.379	0.408	0.433
UCGT	0.373	0.394	0.418
STELLAR	0.307	0.325	0.337
CAPE	0.364	0.402	0.432
Venue2Vec	0.366	0.416	0.435
LBSN2Vec	0.371	0.439	0.451

First, based on the model structures of *TBM1* and *TBM2* (Figs. 5 and 7), as *TBM1* has much fewer time windows than *TBM2*, each time window of *TBM1* has much more training data than that of *TBM2*, which allows *TBM1* to learn user preference more comprehensively. By contrast, training data in each time window of *TBM2* is much sparser, where user reviews maybe missing in a few time windows, making it difficult for the model to fully capture user preference. Second, the *Location Prediction Model* in *TBM1_LPM* has two more non-linear fusion layers than that in *TBM2_LPM*, which enables *TBM1_LPM* to be more powerful in extracting high-level features.

In light of the findings above, we use *TBM1_LPM* as the final *two-stage* framework for user consumption location prediction. Based on Fig. 9, we configure the weighting hyper-parameters η and λ for *TBM1_LPM* to achieve the best performance: $\eta = 0.1$, $\lambda = 0.8$ for *Toronto* dataset; $\eta = 0.2$, $\lambda = 0.9$ for *Phoenix dataset*; $\eta = 0.1$, $\lambda = 0.9$ for *Las Vegas* dataset.

5.6. Performance comparison between TBM1_LPM and baseline models (RQ3)

We compare the predictive performance of *TBM1_LPM* with the baseline models. For better illustration, we summarize the results on $Acc@K$ metric ($K = \{5, 10, 20\}$) in Table 3, and the results on APR metric in Table 4, which represent the prediction accuracy and ranking capacity of different models, respectively. All parameters in the baseline models are carefully tuned to ensure the best performance and fair comparison.

As we can see from Table 3, in terms of $Acc@N$ metric, the proposed *two-stage* framework *TBM1_LPM* consistently outperforms other models with obvious advantage on three datasets. Among the baseline models, LBSN2Vec model has the closest performance to *TBM1_LPM*, followed by Venue2Vec. As LBSN2Vec model preserves multiple types of interactions among user vertices, location vertices and time vertices, it can capture the time-sensitive user preference more accurately than Venue2Vec. However, LBSN2Vec model only takes into account the topology information of GSNs, which limits its capacity to perceive users' sentiment toward locations. Compared with these models, *TBM1_LPM* learns time-sensitive user consumption preference from sentimental textual reviews, spatial context, as well as other side information, which enables it to have more powerful ability in feature fusion. In terms of $Acc@5$, which is widely used to evaluate a location prediction model in relevant research Feng et al. (2018) and Gao et al. (2019), *TBM1_LPM* exceeds the most competitive model LBSN2Vec by 19.4%, 11.7% and 16.1% on *Toronto*, *Phoenix* and *Las Vegas* dataset, respectively, which means *TBM1_LPM* can be competitive in practical applications.

From Table 4, with regard to APR metric which takes the position of the ground-truth location into account, *TBM1_LPM* also performs desirably as it can generally rank the correct locations at the highest positions in most cases. Note that this means we are able to rank a location where a user would consume in the future at a higher position compared with other models, it is indeed a remarkable performance as there are tens of thousands candidate locations in a city.

We also notice that the evaluation metrics based on *Phoenix* and *Las Vegas* datasets are basically higher than that of *Toronto* dataset. One possible explanation is that the average visitors per location in *Phoenix* and *Las Vegas* are larger than that of *Toronto*, indicating that users in *Phoenix* and *Las Vegas* are more likely to visit constant venues, which means users in these two cities have more regular mobility patterns.

Table 5
Comparison in terms of coverage ratio. Best performance is in **boldface**.

	Toronto	Phoenix	Las Vegas
TBM1_LPM	**0.931**	**0.936**	**0.944**
Random	0.916	0.919	0.928
Score_Rank	0.923	0.934	0.936
UCGT	0.920	0.926	0.931
STELLAR	0.922	0.924	0.935
CAPE	0.928	0.926	0.933
Venue2Vec	0.926	0.934	0.937
LBSN2Vec	0.928	0.934	0.939

We further compare the coverage ratio of different models, and the results are summarized in Table 5. Note that this metric measures the proportion of users who are given at least one correct prediction in terms of $Acc@10$, which actually measures the usability of different models for GSN users. Based on Table 5, we notice that in each dataset there are about 7% users who cannot be given any correct prediction. When rechecking the trajectories of selected users, we find that the time interval between two successive consumption behaviors is often very large, thus it is hard to discover intuitive consistency within their trajectories. Based on our observation, such users do exist in Geo-Social Networks, who do not use location-based service frequently and never revisit a location (such as a restaurant). In general, as *TBM1_LPM* consistently has the largest coverage ratio among all models, we believe that it has fine usability in consumption location prediction for GSN users.

5.7. Ablation study on TBM1_LPM (RQ4)

According to Eq. (16), *TBM1_LPM* not only incorporates textual reviews and categorical information, but also fuses geo-influence and social relation. In the following, we compare the predictive performance of the full location prediction framework (Eq. (16)) with its weakened version (Eq. (16) removing geographical and social interactions). For clarity, we name them as *Full_LPM* and *Weak_LPM*, respectively. Again, we use $Acc@K$ metric for comparison where $K = \{5, 10, 20\}$, and the results are plotted in Fig. 10.

It is clear that the full framework *Full_LPM* performs better than its weakened version *Weak_LPM* with large margin. For one thing, both geo-influence and social relation play an important role in reducing the size of candidate locations, which in turn makes the prediction result closer to the user's true consumption preference. For another thing, *Full_LPM* incorporates various kinds of information, including textual content, categorical information, geographical influence as well as social relation, which effectively improves the learning capacity for users' preference. Based on the results, we believe that the fusion of multi-modal GSN data can mutually reinforce each other under the designed framework, and will finally improve the prediction performance.

6. Conclusion

In this article, we study the problem of user consumption location prediction using geo-social networking data. In view of the shortcomings of current literatures, we present a *two-stage* framework that consists of the *Temporal Base Model (TBM)* and the *Location Prediction Model (LPM)* for fine-grained user location prediction. Based on three real-world datasets, we conduct extensive experiments to verify the effectiveness of the proposed approach.

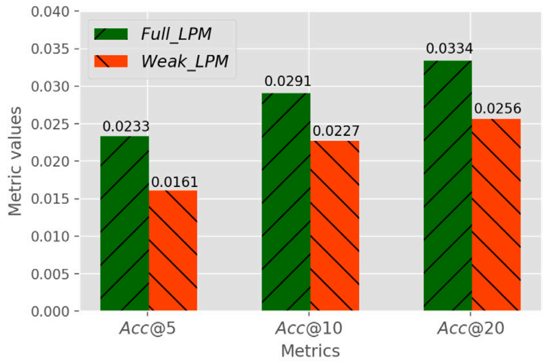
As for future work, we consider to extend current studies from two aspects. First, as we focus on geo-social networks where social influence should vary with the geographic distance among users in the real situation, we plan to characterize a user's next movement by considering not only the similarity between the user and his/her friends, but also the geo-distance and time-interval between the user's latest consumption behavior and his/her friends' latest consumption behavior. Second, we note that existing approaches are basically designed for in-town user location prediction, however, a user may frequently visit places far away from home (hometown), how to properly predict a user's out-of-town consumption location remains to be studied in our future work.

CRediT authorship contribution statement

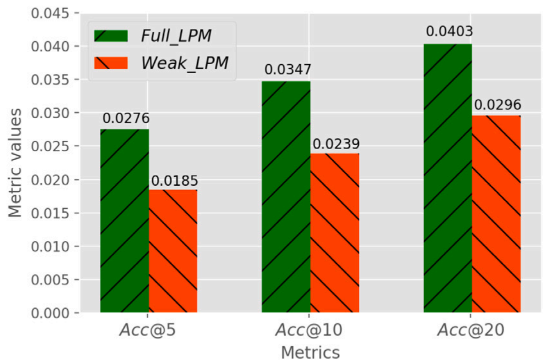
Shuai Xu: Conceptualization, Methodology, Data curation, Validation, Writing - original draft. **Dechang Pi:** The analysis with constructive discussions. **Jiuxin Cao:** Writing - review & editing, Funding support. **Xiaoming Fu:** Writing - review & editing, Funding support.

Declaration of competing interest

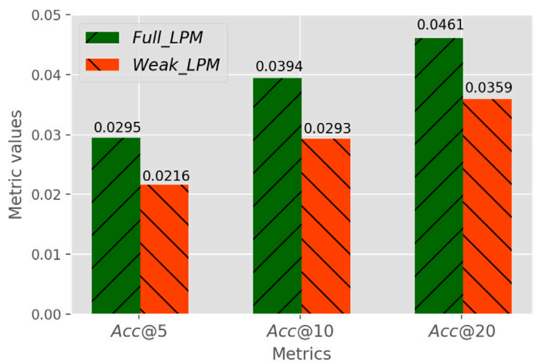
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



(a) Toronto dataset



(b) Phoenix dataset



(c) Las Vegas dataset

Fig. 10. $Acc@K$ comparison between the full framework and weakened framework.

Acknowledgments

This work is supported by the Natural Science Foundation of Jiangsu Province of China, National Natural Science Foundation of China under Grants No. 61772133, National Social Science Foundation of China under Grants No. 19@ZH014, and Jiangsu

Provincial Key Project, China under Grants No. BE2018706. Besides, the financial support provided by China Scholarship Council (CSC) during a visit of Shuai Xu to University of Göttingen is acknowledged. In the end, thanks to open-minded researchers for sharing codes and data resources.

References

Cai, L., Wen, W., Wu, B., & Yang, X. (2021). A coarse-to-fine user preferences prediction method for point-of-interest recommendation. *Neurocomputing*, *422*, 1–11.

Cao, J., Xu, S., Zhu, X., Lv, R., & Liu, B. (2018). Effective fine-grained location prediction based on user check-in pattern in LBSNs. *Journal of Network and Computer Applications*, *108*, 64–75.

Chang, B., Park, Y., Park, D., Kim, S., & Kang, J. (2018). Content-aware hierarchical point-of-interest embedding model for successive POI recommendation. In *Proceedings of the 27th international joint conference on artificial intelligence* (pp. 3301–3307).

Chen, L.-C. (2017). An effective LDA-based time topic model to improve blog search performance. *Information Processing & Management*, *53*(6), 1299–1319.

Chen, X., Zhang, Y., & Qin, Z. (2019). Dynamic explainable recommendation based on neural attentive models. In *Proceedings of the 33rd AAAI conference on artificial intelligence* (pp. 53–60).

Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J., et al. (2019). Graph neural networks for social recommendation. In *Proceedings of the 2019 world wide web conference* (pp. 417–426).

Feng, J., Li, Y., Zhang, C., Sun, F., Meng, F., Guo, A., et al. (2018). DeepMove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference* (pp. 1459–1468).

Feng, J., Yang, Z., Xu, F., Yu, H., Wang, M., & Li, Y. (2020). Learning to simulate human mobility. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 3426–3433).

Gao, R., Li, J., Li, X., Song, C., Chang, J., Liu, D., et al. (2018). STSCR: Exploring spatial-temporal sequential influence and social information for location recommendation. *Neurocomputing*, *319*, 118–133.

Gao, H., Tang, J., Hu, X., & Liu, H. (2013). Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th ACM conference on recommender systems* (pp. 93–100).

Gao, Q., Zhou, F., Trajcevski, G., Zhang, K., Zhong, T., & Zhang, F. (2019). Predicting human mobility via variational attention. In *Proceedings of the 2019 world wide web conference*. (pp. 2750–2756).

Gonzalez, M. C., Hidalgo, C. A., & Barabasi, A.-L. (2008). Understanding individual human mobility patterns. *Nature*, *453*(7196), 779–782.

Guan, X., Cheng, Z., He, X., Zhang, Y., Zhu, Z., Peng, Q., et al. (2019). Attentive aspect modeling for review-aware recommendation. *ACM Transactions on Information Systems*, *37*(3), 1–27.

Hao, P.-Y., Cheang, W.-H., & Chiang, J.-H. (2019). Real-time event embedding for POI recommendation. *Neurocomputing*, *349*, 1–11.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd international conference on learning representation* (pp. 1–15).

Kong, D., & Wu, F. (2018). HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction. In *Proceedings of the 27th international joint conference on artificial intelligence* (pp. 2341–2347).

Li, X., Larson, M., & Hanjalic, A. (2017). Geo-distinctive visual element matching for location estimation of images. *IEEE Transactions on Multimedia*, *20*(5), 1179–1194.

Li, B., Pi, D., Lin, Y., Khan, I. A., & Cui, L. (2020). Multi-source information fusion based heterogeneous network embedding. *Information Sciences*, *534*, 53–71.

Li, R., Shen, Y., & Zhu, Y. (2018). Next point-of-interest recommendation with temporal and multi-level context attention. In *Proceedings of the 2018 IEEE international conference on data mining* (pp. 1110–1115).

Li, J., Wu, L., Hong, R., Zhang, K., Ge, Y., & Li, Y. (2020). A joint neural model for user behavior prediction on social networking platforms. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *11*(6), 1–25.

Lian, D., Zheng, K., Ge, Y., Cao, L., Chen, E., & Xie, X. (2018). GeoMF++: scalable location recommendation via joint geographical modeling and matrix factorization. *ACM Transactions on Information Systems*, *36*(3), 1–29.

Liu, Q., Wu, S., Wang, L., & Tan, T. (2016). Predicting the next location: A recurrent model with spatial and temporal contexts. In *Proceedings of the 30th AAAI conference on artificial intelligence* (pp. 194–200).

Ma, C., Kang, P., Wu, B., Wang, Q., & Liu, X. (2019). Gated attentive-autoencoder for content-aware recommendation. In *Proceedings of the 12th ACM international conference on web search and data mining* (pp. 519–527).

Ma, C., Zhang, Y., Wang, Q., & Liu, X. (2018). Point-of-interest recommendation: Exploiting self-attentive autoencoders with neighbor-aware influence. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 697–706).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, 1–12, arXiv:1301.3781.

Qian, X., Lu, X., Han, J., Du, B., & Li, X. (2017). On combining social media and spatial technology for POI cognition and image localization. *Proceedings of the IEEE*, *105*(10), 1937–1952.

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th conference on uncertainty in artificial intelligence* (pp. 452–461).

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. In *Proceedings of the 8th ACM international conference on web search and data mining* (pp. 399–408).

Tal, O., & Liu, Y. (2019). A joint deep recommendation framework for location-based social networks. *Complexity*, *2019*, 1–11.

Wu, Z., Dai, X.-Y., Yin, C., Huang, S., & Chen, J. (2018). Improving review representations with user attention and product attention for sentiment classification. In *Proceedings of the 32nd AAAI conference on artificial intelligence* (pp. 5989–5996).

Wu, Y., Li, K., Zhao, G., & Qian, X. (2019). Long-and short-term preference learning for next POI recommendation. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 2301–2304).

Wu, L., Sun, P., Fu, Y., Hong, R., Wang, X., & Wang, M. (2019). A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval* (pp. 235–244).

Xie, M., Yin, H., Wang, H., Xu, F., Chen, W., & Wang, S. (2016). Learning graph-based POI embedding for location-based recommendation. In *Proceedings of the 25th ACM international on conference on information and knowledge management* (pp. 15–24).

Xu, S., Cao, J., Legg, P., Liu, B., & Li, S. (2020). Venue2Vec: An efficient embedding model for fine-grained user location prediction in geo-social networks. *IEEE Systems Journal*, *14*(2), 1740–1751.

Xu, S., Fu, X., Cao, J., Liu, B., & Wang, Z. (2020). Survey on user location prediction based on geo-social networking data. *World Wide Web*, *23*(3), 1621–1664.

Xu, J., Zhao, J., Zhou, R., Liu, C., Zhao, P., & Zhao, L. (2021). Predicting destinations by a deep learning based approach. *IEEE Transactions on Knowledge and Data Engineering*, *33*(2), 651–666.

Yang, D., Fankhauser, B., Rosso, P., & Cudre-Mauroux, P. (2020). Location prediction over sparse user mobility traces using RNNs: Flashback in hidden states. In *Proceedings of the 29th international joint conference on artificial intelligence* (pp. 2184–2190).

Yang, H., Liu, T., Sun, Y., & Bertino, E. (2019). Exploring the interaction effects for temporal spatial behavior prediction. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 2013–2022).

Yang, D., Qu, B., Yang, J., & Cudre-Mauroux, P. (2019). Revisiting user mobility and social relationships in lbsns: a hypergraph embedding approach. In *Proceedings of the 2019 world wide web conference* (pp. 2147–2157).

Yang, Z., Sun, H., Huang, J., Sun, Z., Xiong, H., Qiao, S., et al. (2020). An efficient destination prediction approach based on future trajectory prediction and transition matrix optimization. *IEEE Transactions on Knowledge and Data Engineering*, *32*(2), 203–217.

Yin, H., Hu, Z., Zhou, X., Wang, H., Zheng, K., Nguyen, Q. V. H., et al. (2016). Discovering interpretable geo-social communities for user behavior prediction. In *Proceedings of the IEEE 32nd international conference on data engineering* (pp. 942–953).

Yu, F., Cui, L., Guo, W., Lu, X., Li, Q., & Lu, H. (2020). A category-aware deep model for successive POI recommendation on sparse check-in data. In *Proceedings of the 2020 world wide web conference* (pp. 1264–1274).

Zhang, J.-D., & Chow, C.-Y. (2015). GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 443–452).

Zhang, L., Shen, J. J., Zhang, J., Xu, J., Li, Z., Yao, Y., et al. (2021). Multimodal marketing intent analysis for effective targeted advertising. *IEEE Transactions on Multimedia*.

Zhao, G., Lou, P., Qian, X., & Hou, X. (2020). Personalized location recommendation by fusing sentimental and spatial context. *Knowledge-Based Systems*, *196*, Article 105849.

Zhao, S., Zhao, T., Yang, H., Lyu, M. R., & King, I. (2016). STELLAR: Spatial-temporal latent ranking for successive point-of-interest recommendation. In *Proceedings of the 30th AAAI conference on artificial intelligence* (pp. 315–321).

Zhao, P., Zhu, H., Liu, Y., Xu, J., Li, Z., Zhuang, F., et al. (2019). Where to go next: A spatio-temporal gated network for next POI recommendation. In *Proceedings of the 33rd AAAI conference on artificial intelligence* (pp. 5877–5884).

Zhou, X., Mascolo, C., & Zhao, Z. (2019). Topic-enhanced memory networks for personalised point-of-interest recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 3018–3028).

Zhou, X., Noulas, A., Mascolo, C., & Zhao, Z. (2018). Discovering latent patterns of urban cultural interactions in wechat for modern city planning. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1069–1078).

Zhu, L., Shen, J., Jin, H., Xie, L., & Zheng, R. (2015). Landmark classification with hierarchical multi-modal exemplar feature. *IEEE Transactions on Multimedia*, *17*(7), 981–993.