

Article

Dynamic Recommendation of POI Sequence Responding to Historical Trajectory

Jianfeng Huang , Yuefeng Liu *, Yue Chen and Chen Jia

Institute of Remote Sensing and Geographic Information System, Peking University, Beijing 100871, China; jianfenghuang@pku.edu.cn (J.H.); chen Yue_sess@pku.edu.cn (Y.C.); jiachen07@pku.edu.cn (C.J.)

* Correspondence: yuefengliu@pku.edu.cn

Received: 9 August 2019; Accepted: 27 September 2019; Published: 30 September 2019



Abstract: Point-of-Interest (POI) recommendation is attracting the increasing attention of researchers because of the rapid development of Location-based Social Networks (LBSNs) in recent years. Differing from other recommenders, who only recommend the next POI, this research focuses on the successive POI sequence recommendation. A novel POI sequence recommendation framework, named Dynamic Recommendation of POI Sequence (DRPS), is proposed, which models the POI sequence recommendation as a Sequence-to-Sequence (Seq2Seq) learning task, that is, the input sequence is a historical trajectory, and the output sequence is exactly the POI sequence to be recommended. To solve this Seq2Seq problem, an effective architecture is designed based on the Deep Neural Network (DNN). Owing to the end-to-end workflow, DRPS can easily make dynamic POI sequence recommendations by allowing the input to change over time. In addition, two new metrics named Aligned Precision (AP) and Order-aware Sequence Precision (OSP) are proposed to evaluate the recommendation accuracy of a POI sequence, which considers not only the POI identity but also the visiting order. The experimental results show that the proposed method is effective for POI sequence recommendation tasks, and it significantly outperforms the baseline approaches like Additive Markov Chain, LORE and LSTM-Seq2Seq.

Keywords: POI sequence recommendation; location-based social networks; deep neural network; sequence-to-sequence

1. Introduction

It is a well-known fact in psychology that humans always tend to behave in a consistent way [1,2], which makes it possible to learn and predict the patterns of human behaviors. On the other hand, the Location-Based Social Networks (LBSNs) [3] are playing an increasingly important role in daily life, through which users can share their locations and location-related contents at any time. LBSNs provide masses of valuable data for researching the patterns of human behaviors. These data have tremendous potential for exploitation for various applications, such as questioning and answering, advertising, activity discovery and recommendations [4], among which Point-of-Interest (POI) recommendation is attracting more and more attention from researchers in recent years.

The consistency of behaviors means that the human behaviors always follow a particular pattern and preference in a certain period. Therefore, in the POI recommendation task, users' behavior patterns and preferences can be captured from the historical trajectories first, and then they can be extended to make the next recommendation. However, most of the existing POI recommenders can only recommend the next POI or a top-k list of candidate POIs [5–8], while sometimes successive POI sequence recommendations are more practical. For example, when one wants to plan an itinerary, what he/she expects is not a single POI recommendation, but a POI sequence recommendation. A POI sequence contains a set of POIs and the order in which they are visited. Itinerary-planning

is a very tedious and time-consuming process because users always need to take into account the time constraint, distance constraint, cost constraint and so on. Therefore, it would be great if there is a POI sequence recommender to automatically recommend POI sequences (i.e., itineraries) for users, which will free users from the tedious and time-consuming itinerary-planning process. Compared with the single POI recommendation, the POI sequence recommendation is more challenging because of the following major reasons: (1) POI sequence recommendation aims to recommend a contextually coherent POI sequence which exactly meets user's interest and preference, instead of just a single POI; (2) users' preferences may change over time, which increases the difficulty of dynamic recommendation; and the (3) POI sequence recommendation is more sensitive to various factors (e.g., spatial, temporal, categorical, etc.) [9].

There are only a few studies focusing on POI sequence recommendation. To model the POI sequence recommendation task, some researchers proposed popularity-based approaches [10,11], which aim to find a POI sequence that maximizes the POI popularity. In these cases, all users will get the same recommendation. In addition, other personalization-based approaches [12–14] have also been developed to recommend a customized and unique tour itinerary for each tourist based on his/her interests and preferences. Nevertheless, most of the existing systems have some common drawbacks: (1) they need to predefine the starting and ending POIs for each recommendation, so they are not completely automatic; and (2) they can not capture the evolution of user preferences over time, so it is difficult to make dynamic recommendations.

This paper proposes a novel POI sequence recommendation framework, named Dynamic Recommendation of POI Sequence (DRPS), which models the POI sequence recommendation as a Sequence-to-Sequence (Seq2Seq) learning task, namely, the input sequence is a historical trajectory, and the output sequence is exactly the POI sequence to be recommended. Many studies have been carried out to address the Seq2Seq learning problem, such as [15,16]. Enlightened by the fact that the Deep Neural Network (DNN) has made a great success in various fields [17], the architecture of DRPS is designed based on the DNN. More specifically, DRPS is mainly composed of an encoder and a decoder. The encoder is designed to learn the contextual information implied in the input sequence, based on the contextual information; then, the decoder will generate the next POI one by one to form a POI sequence recommendation. In addition, in order to achieve better performance, this model comprehensively takes into account the POI embedding feature, the geographical and categorical influences of historical trajectory, and the positional encoding. The proposed method is evaluated on the Weeplace dataset, and the experimental results show the effectiveness of DRPS in a POI sequence recommendation task.

To summarize, the major contributions of this paper are:

- this paper proposes a novel POI sequence recommendation framework named DRPS, which can make dynamic POI sequence recommendation according to the historical trajectory;
- in order to make full use of various information about POIs, the POI embedding feature, the geographical and categorical influences of historical trajectory and the positional encoding are jointly taken into account in DRPS;
- this paper also proposes two new metrics, i.e., the Aligned Precision (AP) and the Order-aware Sequence Precision (OSP), which consider both the POI identity and visiting order, in order to evaluate the recommendation accuracy of the POI sequence;
- detailed experiments are conducted to evaluate the proposed method, and the experimental results demonstrate the effectiveness of DRPS in a POI sequence recommendation task.

The rest of this paper is organized as follows: Section 2 briefly reviews the related work. Section 3 elaborates on the proposed method. In addition, the experimental results are presented in Section 4. Finally, Section 5 concludes this paper.

2. Related Work

The POI recommendation tasks generally fall into two categories: single POI recommendation and POI sequence recommendation. For the single POI recommendation, a lot of literature has developed various recommenders by leveraging different aspects of POIs and users, such as geographical influence, temporal influence, social influence, and user preferences. For example, in order to improve the performance of POI recommendation, Wang et al. [18] integrated the geographical influence of POIs into a standard recommendation model to capture a user's preference. For temporal influence, Hosseini et al. [19] propose a probabilistic generative model, named Multi-aspect Time-related Influence (MATI), to promote the effectiveness of the POI recommendation task. The MATI model firstly detects a user's temporal multivariate orientation using her check-in log in LBSNs, then performs recommendation using temporal correlations between the user and proposed POIs. Griesner et al. [20] used matrix factorization algorithms to model the geographical and temporal influences of POI check-ins, and then proposed the GeoMF-TD model for POI recommendation. Yuan et al. [21] also leveraged the temporal influence to improve the efficiency and effectiveness of the recommendation system. Zhang and Chow [22] proposed the Rank-GeoFM algorithm to learn the geographical, social and categorical correlations from the historical check-in data of users on POIs, and then utilize them to predict the relevance score of a user to an unvisited POI in order to make recommendations for users. Ye et al. [23] combined the social influence with a user-based collaborative filtering model and utilized a Bayesian model to model the geographical influence. In addition, Aliannejadi and Crestani [24] utilized a probabilistic model to construct the mapping between user tags and location taste keywords, which made it possible to exploit various directions to address the data sparsity problem for POI recommendation. Feng et al. [25] proposed an algorithm named Personalized Ranking Metric Embedding (PRME) to jointly learn the sequential transition and user preference implicit in POIs. To recommend a "Smart POI" for a user according to the user preferences, based on the categories and geographical information, Alvarado-Uribe et al. [26] incorporated an aggregation operator into the user-based collaborative filtering algorithm and then proposed the Hybrid Recommendation Algorithm (HyRA). Moreover, because of the powerful learning ability of DNN, some DNN-based approaches were also proposed to enhance the performance of POI recommendation. Ding and Chen [7] designed the RecNet to incorporate various features implicit in LBSNs, such as co-visiting pattern, geographical influence and categorical correlation, and learn their high-order interactions for personalized POI recommendation. Yang et al. [27] proposed a deep neural architecture named Preference and Context Embedding (PACE) to model user preference over POIs, which utilizes the smoothness of semi-supervised learning to alleviate the sparsity of collaborative filtering. Chang et al. [28] also proposed an embedding-based method, called Content-Aware Hierarchical Point-of-Interest Embedding Model (CAPE), to utilize the text content that provides information about the characteristics of a POI and the relationships between POIs for POI recommendation.

For POI sequence recommendation, De Choudhury et al. [10] first proposed an approach to automatically construct a travel itinerary based on the Orienteering Problem, which aims to find a POI sequence that could maximize the POI popularity. Similarly, by modifying the Orienteering Problem, Gionis et al. [29] utilized a POI category to recommend an itinerary that is constrained by a POI category visiting order (e.g., library→restaurant→park). Bolzoni et al. [11] also designed the CLuster Itinerary Planning (CLIP) algorithm to recommend an itinerary based on clustering, where the POIs are clustered first; then, the clusters are used for itinerary generation. Obviously, these approaches heavily focus on the popularity of POIs, while the personalization would be left in the basket. To solve this problem, some personalization-based methods were developed. Lim et al. [14] proposed the PERSTOUR algorithm to recommend personalized POI sequences for the user, which considers not only the POI popularity, but also the user preference and the trip constraints. Baral et al. [30] proposed the Hierarchical Contextual POI Sequence (HiCaPS) model to formulate the user preference as a hierarchical structure and then developed a hierarchy aggregation technique for the POI sequence

recommendation. Debnath et al. [31] designed a preference-aware POI sequence recommendation framework named Preference-Aware, Time-Aware Travel Route Recommendation (PTTR-Reco), which incorporates a time dimension to model the time-specific user preference, so PTTR-Reco aims to recommend the POI sequence that matches the time-specific preference of individual user. As mentioned earlier, the POI sequence recommendation can be modeled as a Seq2Seq learning task. It has been proven that a recurrent neural network (RNN) is quite effective in the Seq2Seq learning task. For instance, Sutskever et al. [16] used a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of fixed dimensionality, and then designed another deep LSTM to decode the target sequence from the vector. This Seq2Seq fashion significantly improved the performance in the English to French translation task. In addition, Cui et al. [32] proposed a model named Multi-View Recurrent Neural Network (MV-RNN) for sequential recommendation. MV-RNN takes multi-view features as inputs and then applies a recurrent structure to dynamically capture the sequential information implicit in inputs. Finally, a separate RNN and a united RNN are respectively designed on the hidden state of MV-RNN to achieve more effective recommendation. Nevertheless, the strong modeling power of DNN has not been well exploited in POI sequence recommendation. To fill this gap, this paper proposed the DRPS to make a dynamic POI sequence recommendation based on the historical trajectory.

3. Methodology

3.1. Problem Statement

A POI sequence consists of a set of POIs and the order of visiting them. Now, the POI sequence recommendation task can be defined as follows: given a user's historical trajectory $H = (h_1, h_2, h_3, \dots, h_n)$, the goal is to recommend a contextually coherent POI sequence $R = (r_1, r_2, r_3, \dots, r_k)$ for the user, where n is the length of historical POI sequence, and k is the length of the recommended POI sequence. Here, a contextually coherent POI sequence is exactly a POI sequence closely following the historical trajectory. The contextual coherence means that it should meet the same interest and preference implicit in the historical trajectory of the user. For instance, if a user has visited such a POI list: (A, B, C, D, E, F, G, H) , it can be considered that (E, F, G, H) is exactly a contextually coherent POI sequence of (A, B, C, D) . POI sequence recommendation is actually a Seq2Seq learning task. Corresponding to the Seq2Seq model, the historical POI sequence and the recommend POI sequence can be denoted as input sequence and target sequence, respectively.

3.2. Overview of the Proposed Framework

Usually, a historical POI sequence implicitly represents the recent preference and behavior pattern of a user. The objective of a POI sequence recommender is to capture this preference and behavior pattern, and extend them to recommend the next POI sequence. This problem is modeled as a Seq2Seq learning task in this paper. Enlightened by the Transformer model [33], a neural machine translation model that is powerful for modeling a Seq2Seq task, this paper proposes a framework named Dynamic Recommendation of POI Sequence (DRPS). The DRPS is mainly composed of an encoder and a decoder (Figure 1). This encoder–decoder structure is the key to model the Seq2Seq learning task. The encoder is used to learn the contextual information implied in the input sequence H , and the decoder is used to generate the POI sequence recommendation R . To capture abundant information from the input POI sequence, the DRPS integrates the POI embedding, category embedding, geographical influence and positional encoding as the input of the encoder. Following the decoder, the main branch will recommend the next POIs one by one to form a POI sequence recommendation, and the other two auxiliary branches are respectively used to predict the categories and locations of the corresponding POIs. As additional constraints, these two auxiliary branches are designed to help train the model. Note that, in the recommended POI sequence, the POIs are generated one by one, and, when generating the next POI, the decoder also takes the previously generated POIs as additional input.

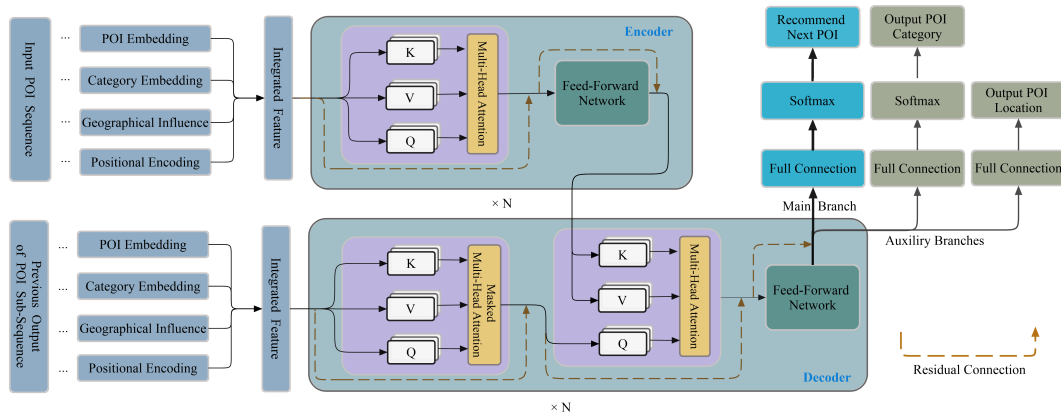


Figure 1. The architecture of DRPS.

3.3. Details of Module Design

The DRPS framework can be divided into three modules, i.e., an input module, an encoder–decoder module, and an output module. The details about each module are described in the following sections.

3.3.1. Input Module

In order to learn rich contextual information implicit in the POI sequence, four features of POI sequences are considered in the input module, including POI embedding, category embedding, geographical influence, and positional encoding, as shown in Figure 1.

POI embedding Intuitively, a POI sequence can be represented with a set of ordered POI IDs. However, the ID is insufficient to characterize a POI. Feature embedding is an important representation learning technique that can embed the original feature into a more effective vector representation [34]. Similarly, a learnable POI embedding is used to map each POI ID to a d -dimension latent vector f_{pe} in this research, which actually describes the intrinsic feature of POI. Formally, the POI embedding of a POI sequence $S \in \mathbb{R}^l$ is denoted by a matrix $F_{pe} \in \mathbb{R}^{l \times d}$, where the l is the length of POI sequence, and d is the dimension of latent vector. The POI embedding is first randomly initialized and then it can be learnt by training the neural network.

Category embedding Category is an important attribute of POI, which is widely used in many POI recommendation systems, such as Ding and Chen [7], Bolzoni et al. [11], Lin et al. [35]. The categorical influence of a POI sequence is also taken into account in DRPS. A POI sequence actually corresponds to a POI category sequence. Similar to POI embedding, the category embedding is also used to map each POI category to a d -dimension latent vector f_{ce} . Thus, the category embedding of a POI sequence S can be denoted by a matrix $F_{ce} \in \mathbb{R}^{l \times d}$.

Geographical influence It has been proven that geographical influence has a significant impact on POI recommendation [18]. The geographical influence exists not only between POIs and POIs, but also between POIs and users. According to Tobler’s first law of geography, the closer two POIs are, the more similar they are, which makes the geographical coordinates an important way to measure the similarity between POIs. On the other hand, the geographical influence can also reflect the preference of users to locations. For example, users may prefer to visit places close to home. Therefore, the geographical influence is an important aspect of POI characteristics. To model the geographical influence of POI, a Multi-Layer Perceptron (MLP) [36] is adopted to convert the coordinates of each POI to a d -dimension vector f_{gi} . Namely, the f_{gi} is given by Equation (1):

$$f_{gi}(c) = \text{ReLU}(\text{ReLU}(cW_1 + b_1)W_2 + b_2), \quad (1)$$

where $c = (x, y)$ is the coordinates of a POI, b_1 and b_2 are the biases, W_1 and W_2 are the weight matrices, and ReLU [37] is a nonlinear activation transformation. The biases and the weight matrices are the

parameters that need to be trained. Similarly, a matrix $F_{gi} \in \mathbb{R}^{l \times d}$ is used to describe the geographical influence of a POI sequence S .

Positional encoding To make use of the order of the POI sequence, which provides vital contextual information, the “positional encoding” is introduced into DRPS. As in Vaswani et al. [33], each position is encoded to a d -dimension vector; in this way, a matrix $F_{pos} \in \mathbb{R}^{l \times d}$ can be constructed to represent the positional encoding of a POI sequence. Specifically, each element of F_{pos} is given by Equation (2):

$$F_{pos}(p, i) = \begin{cases} \sin\left(\frac{p}{1000^{\frac{i}{d}}}\right), & \text{if } i \text{ is an even number,} \\ \cos\left(\frac{p}{1000^{\frac{i-1}{d}}}\right), & \text{otherwise,} \end{cases} \quad (2)$$

where $p \in \{1, 2, 3, \dots, l\}$ is the position of a POI in the sequence, and $i \in \{1, 2, 3, \dots, d\}$ denotes the i -th dimension of the d -dimension vector.

After the above features of POI sequence are calculated, they will be integrated together by following naive summation:

$$F_{int} = F_{pe} + F_{ce} + F_{gi} + F_{pos}. \quad (3)$$

Then, F_{int} is fed into the downstream module.

3.3.2. Encoder–Decoder Module

The encoder–decoder structure has strong power for a Seq2Seq learning task [15,16]. This paper also follows this structure. Specifically, an encoder is designed to capture the contextual information implied in the input sequence H ; then, a decoder is used for generating the POI sequence recommendation R based on the output of encoder, as shown in Figure 1.

Encoder The encoder consists of a stack of N identical blocks. Each block contains a multi-head attention (MHA) layer and a feed-forward layer. The attention mechanism has proven to be an effective approach for a sequence modeling task in a deep neural network [38]. The attention function can be perceived as mapping a set of key-value pairs (K-V) and a query (Q) to an output, where the keys, values and queries are all derived from different transformation of the integrated feature. The output is actually a weighted sum of the values, where the weight assigned to each value depends on the similarity of the query to the corresponding key. Here, a MHA structure similar to Vaswani et al. [33] is added into DRPS. Formally, MHA is given by Equation (4):

$$MHA(K, V, Q) = \text{Concat}(Att_1, Att_2, \dots, Att_m)W_o, \quad (4)$$

where the *Concat* function is used to concatenate $Att_1 \sim Att_m$, and

$$Att_i = \text{Attention}(KW_i^K, VW_i^V, QW_i^Q), \quad i \in \{1, 2, 3, \dots, m\}. \quad (5)$$

The *Attention* function is defined as Equation (6)

$$\text{Attention}(K, V, Q) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V, \quad (6)$$

where d_K is the dimension of K , and the *softmax* function is used for normalizing the output probabilities. The above parameters' matrices W_i^K , W_i^V , W_i^Q , W_o are all trainable. Following the MHA layer, a two-layer Feed-Forward Network (FFN) is used to generate the output of encoder. The FFN is given by Equation (7),

$$FFN(x) = \text{ReLU}(xW_1^{FFN} + b_1^{FFN})W_2^{FFN} + b_2^{FFN}, \quad (7)$$

where W_1^{FFN} and W_2^{FFN} are the weight matrices, b_1^{FFN} and b_2^{FFN} are the biases. In addition, the residual connection [39] is also employed to improve the performance.

Decoder The decoder also consists of a stack of N identical blocks. While each block, compared with the encoder, has an additional Masked Multi-Head Attention (Masked-MHA) layer, which takes the previous output of a POI sub-sequence as input when generating the next POI. The mask ensures that the POI recommendation for position i depends only on the known outputs at positions less than i . In addition, the MHA layer in the decoder also takes the output of the encoder as input and is then followed by a two-layer FFN. The residual connection is also exploited in the decoder.

3.3.3. Output Module

The output module contains three branches, i.e., a main branch and two auxiliary branches. The main branch is used to recommend the next POI one by one to form a POI sequence recommendation. In addition, the other two auxiliary branches are respectively used to predict the categories and locations of the corresponding POIs, which aims to add additional constraints to the learning task to help train the model and improve the performance. Combining these three branches, the total loss can be written as follows:

$$\mathcal{L} = \sum_{i=1}^k CE(y_i, \hat{y}_i) + \sum_{i=1}^k CE(cat_i, \hat{cat}_i) + \sum_{i=1}^k MSE(loc_i, \hat{loc}_i), \quad (8)$$

where CE is the Cross Entropy function, y_i is the one-hot encoding of true POI ID on position i , and \hat{y}_i is the recommendation probabilities of all POIs on position i . Similarly, cat_i is the one-hot encoding of a true POI category and \hat{cat}_i is the predicted probabilities of all categories on position i . MSE is the Mean Square Error function, loc_i and \hat{loc}_i are respectively the true and predicted location of POI on position i . The training objective is to minimize the loss function \mathcal{L} .

3.4. Dynamic Recommendation

Users' historical trajectories always change over time, which means that their interests and preferences may also change over time. Therefore, a dynamic recommendation system is more practical than a static one. In this research, when the model finishes training, DRPS can easily make dynamic recommendations by allowing the input POI sequence to change over time. When the input POI sequence changes, the DRPS can dynamically capture the recent interest and preference of the user, and then recommend the most suitable POI sequence. In contrast to other approaches, which require expensive computation resources to re-calculate the features of the changing input, the DRPS can easily recommend a POI sequence dynamically in an end-to-end way.

4. Experiments

In this section, a series of experiments are conducted to evaluate the performance of the proposed framework DRPS.

4.1. Experimental Settings

4.1.1. Dataset

In this paper, the experiments are conducted on the Weeplace dataset [40], which is collected from Weeplace, a website that aims to visualize users' check-in activities in LBSN. This dataset contains 7,658,368 check-ins generated by 15,799 users over 971,309 POIs. These POIs cover the whole world. In addition, the category and location (i.e., the longitude and latitude) of each POI are also provided in this dataset. Considering the practicability, the dataset is divided into different parts by the city, and then the proposed method is evaluated on some major cities, including New York, San Francisco, Brooklyn, and London. Table 1 presents the statistic details of the data of each city.

Table 1. Statistics of the data of each city.

City	#Check-In	#User	#POI
New York	720,350	4811	28,333
San Francisco	330,975	3220	13,366
Brooklyn	159,946	2724	7334
London	147,610	1935	10,405

In the dataset, the users with less than 40 check-ins and the POIs with less than 10 visits are ignored. Each user may have a long historical trajectory, therefore, many samples can be constructed for each user by moving a lag along the history. In each sample, the first n POIs compose the input POI sequence, denoted by $I = (i_1, i_2, i_3, \dots, i_n)$, and the latter k POIs compose the target POI sequence, denoted by $T = (t_1, t_2, t_3, \dots, t_k)$, where n and k are the lengths of input and output, respectively. In this paper, the lag is also set as k .

4.1.2. Evaluation Metrics

In the single POI recommendation tasks, precision@ k and recall@ k are usually chosen as evaluation metrics, both of which are evaluated on the top k POIs in a ranked recommendation list. Nevertheless, what needs to be recommended in the POI sequence recommendation tasks is not a ranked POI list, but an ordered POI sequence. Thus, precision@ k and recall@ k are no longer appropriate in this work. The most used metrics for evaluating POI sequence recommendation are precision, recall, and F1 score [6,30,35,41]. However, a common drawback of these metrics is that they all ignore the order of POI in the sequence, while the order is actually an important aspect of POI sequence recommendation. To solve the problem, two new metrics named Aligned Precision (AP) and Order-aware Sequence Precision (OSP) are proposed to evaluate the recommendation accuracy of POI sequence, considering both the POI identity and visiting order.

Specifically, AP is calculated by Equation (9):

$$AP = \frac{\sum_{i=1}^k I(g_i = r_i)}{k}, \quad (9)$$

where g_i and r_i are the ground truth and recommendation on position i , respectively, and

$$I(g_i = r_i) = \begin{cases} 1, & \text{if } g_i = r_i, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

Equation (9) shows that the AP is equal to 1 only when each position of sequence is correctly recommended. In other words, if a POI sequence recommendation contains correct POI identities but the wrong order, its AP will be still 0. Obviously, the metric AP is so strict. Therefore, a gentler but still effective metric OSP is also proposed in this research, which is given by Equation (11):

$$OSP = \frac{G \cap R}{k} \cdot \frac{M}{C}, \quad (11)$$

where $\frac{G \cap R}{k}$ measures the degree of overlap between the recommended POI sequence R and ground truth G (without considering the order), and $\frac{M}{C}$ measures the order precision of the overlapped sub-sequence, where C is the number of all POI pairs in the overlapped sub-sequence, and M is the number of the pairs that contain correct order.

For example, if a recommended POI sequence is (B, A, D, C, F) and the corresponding ground truth is (A, B, C, D, E) , it can be easily calculated that $AP = 0$. As for OSP, the overlapped sub-sequence (B, A, D, C) can be obtained first, which contains four common POIs, so $\frac{G \cap R}{k} = 0.8$. In addition, all the ordered POI pairs in the overlapped sub-sequence include $\{BA, BD, BC, AD, AC, DC\}$, i.e., $C = 6$.

Nevertheless, only $\{BD, BC, AD, AC\}$ have correct orders as ground truth, so $M = 4$. Naturally, it can be calculated that the $OSP = 0.53$. This example shows that AP is much stricter than OSP, and OSP considers not only the degree of overlap between sequences, but also the order in sequence.

Besides AP and OSP, the most used metrics, i.e., the precision and recall, are also used to evaluate the models in this research. The precision and recall can be calculated from the confusion matrix, and they are both averaged over all the POIs.

4.1.3. Baseline Methods

The proposed model DRPS is compared with the following baseline methods:

Random Selection (RAND) In the RAND model, each POI in the recommended sequence is randomly selected from all POIs.

Additive Markov Chain (AMC) [42] The AMC model recommends POI sequence by exploiting the sequential influence. Specifically, given a historical trajectory S_u of user u , when recommending the POI on position p , AMC first calculates the probability of the user visiting each POI based on all the POIs before position p ; then, the POI with maximum probability will be recommended on position p .

Location REcommendation (LORE) [41] The LORE model first mines sequential patterns implicit in POI sequences and represents the sequential patterns as a dynamic Location–Location Transition Graph (L2TG). Based on the L2TG and geographical influence, LORE can then predict the user's probability of visiting each POI.

LSTM-Seq2Seq [16] The LSTM-Seq2Seq model adopts a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimension, and then uses another deep LSTM to decode the target sequence from the vector.

4.1.4. Parameter Setting

In the experiments, all the samples are divided into three subsets, i.e., a training set, a validation set and a test set, and their proportions are 70%, 20% and 10%, respectively. The training set is used to train the model parameters of DRPS, the validation set is used to select the parameters when training, and the test set is used to evaluate the performance of the model. In addition, the dimension d of input feature is set to 64, the number of blocks N in encoder (decoder) is set to 2, and the number of the "header" of the MHA layer m is set to 8. In addition, the experiments are conducted with two different settings of input/output POI sequence lengths (n, k) , including (30, 5) and (25, 10). In order to evaluate the performance of the proposed algorithm more effectively and reduce the risk of overfitting, every experiment in this paper is repeated 10 times, and the mean and variance of multiple results are used to measure the performance of the model. In addition, all the experiments are implemented with TensorFlow [43].

4.2. Experimental Results

A series of experiments are carried out according to the above settings. The performances of different POI sequence recommendation approaches in terms of AP and OSP are presented in Tables 2 and 3. The difference between Tables 2 and 3 is the setting of input/output POI sequence lengths, that is, the former is (30, 5) and the latter is (25, 10).

It is apparent in Tables 2 and 3 that DRPS significantly outperforms the other baselines when measured with AP and OSP on all four cities. The RAND method gives the recommendation just by naively random selection without utilizing any additional information, so its accuracy is very low but stable. The AMC and LORE achieve similar performance. More specifically, when the input length is longer (30) and output length is shorter (5), LORE is better than AMC in most cases. Conversely, when the input length gets shorter (25) and output length gets longer (10), AMC almost always outperforms LORE. LSTM-Seq2Seq reaches the closest performance to DRPS, while, no matter how long the input/output sequences are, DRPS always achieves the best performance. On the other hand,

it is easy to find that the values of OSP are always higher than the ones of AP, which verifies that, as mentioned in the previous analysis, AP is stricter than OSP.

Table 2. Performance comparison on four cities in terms of AP and OSP under the setting where the input and output lengths are, respectively, 30 and 5. The numbers in bold face indicate the ones with the best performances.

%	New York		San Francisco		Brooklyn		London	
	AP	OSP	AP	OSP	AP	OSP	AP	OSP
RAND	0.011 ± 0.0006	0.052 ± 0.0015	0.023 ± 0.0022	0.108 ± 0.0031	0.049 ± 0.0039	0.221 ± 0.0057	0.035 ± 0.0019	0.157 ± 0.0049
AMC	7.373 ± 0.4386	10.040 ± 0.5494	8.160 ± 0.3533	10.840 ± 0.3217	9.267 ± 0.7745	13.173 ± 1.3283	11.680 ± 0.2993	16.471 ± 0.2554
LORE	8.107 ± 0.3785	10.680 ± 0.3767	9.013 ± 0.2660	10.427 ± 0.3583	10.653 ± 0.8213	12.507 ± 1.1437	13.467 ± 0.5310	15.547 ± 0.4031
LSTM-Seq2Seq	9.014 ± 0.0092	13.060 ± 0.0297	11.352 ± 0.6135	15.766 ± 1.0097	10.772 ± 0.5439	22.293 ± 0.6308	14.666 ± 0.7064	27.723 ± 0.7408
DRPS	9.982 ± 0.3985	19.511 ± 0.2400	13.653 ± 0.1611	18.747 ± 0.3048	11.933 ± 0.4434	23.240 ± 0.9827	15.413 ± 0.2294	29.513 ± 0.8359

Table 3. Performance comparison on four cities in terms of AP and OSP under the setting where the input and output lengths are, respectively, 25 and 10. The numbers in bold face indicate the ones with the best performances.

%	New York		San Francisco		Brooklyn		London	
	AP	OSP	AP	OSP	AP	OSP	AP	OSP
RAND	0.011 ± 0.0005	0.093 ± 0.0011	0.022 ± 0.0012	0.190 ± 0.0016	0.053 ± 0.0022	0.387 ± 0.0015	0.039 ± 0.0021	0.278 ± 0.0061
AMC	7.067 ± 0.3117	11.698 ± 0.4837	6.710 ± 0.1445	11.530 ± 0.4276	8.560 ± 0.0712	14.892 ± 0.1127	11.473 ± 0.6824	20.490 ± 0.7428
LORE	6.833 ± 0.4873	10.347 ± 0.5788	6.920 ± 0.1728	8.860 ± 0.1840	8.727 ± 0.2265	11.320 ± 0.4109	12.947 ± 0.3488	16.090 ± 0.3813
LSTM-Seq2Seq	8.208 ± 0.3165	19.068 ± 0.3939	8.136 ± 0.3347	19.083 ± 0.5741	9.452 ± 0.4460	18.227 ± 0.5413	12.951 ± 0.5010	27.850 ± 0.7589
DRPS	9.547 ± 0.0929	21.040 ± 0.2204	8.927 ± 0.2877	19.502 ± 0.0408	11.475 ± 0.8351	21.223 ± 0.0719	14.485 ± 0.3118	29.533 ± 0.1363

In addition to AP and OSP, the precision and recall are also chosen to evaluate the above models. Tables 4 and 5 present the experimental results. Table 4 is for the case where the input and output lengths are, respectively, 30 and 5, and Table 5 is for the case where the input and output lengths are, respectively, 25 and 10.

Based on the results, it can be observed that DRPS can still outperform almost all the baselines when measured with precision and recall. In addition, with the input length getting shorter and the output length getting longer, the POI sequence recommendation also gets harder. Surprisingly, DRPS can still achieve competitive (even better) performance, which proves that DRPS is powerful for the POI sequence recommendation task.

Table 4. Performance comparison on four cities in terms of precision and recall under the setting where the input and output lengths are, respectively, 30 and 5. The numbers in bold face indicate the ones with the best performances.

%	New York		San Francisco		Brooklyn		London	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
RAND	0.021 ± 0.0194	0.033 ± 0.0310	0.013 ± 0.0137	0.043 ± 0.0117	0.015 ± 0.0138	0.025 ± 0.0232	0.005 ± 0.0055	0.011 ± 0.0109
AMC	0.823 ± 0.6556	1.840 ± 1.6125	3.573 ± 2.0724	3.252 ± 0.5875	5.409 ± 1.6053	14.732 ± 3.4203	4.260 ± 0.5859	11.411 ± 0.6116
LORE	3.151 ± 1.1616	7.355 ± 2.8026	6.652 ± 1.6381	1.004 ± 0.7220	2.857 ± 0.6719	11.401 ± 2.2280	5.400 ± 1.0526	11.999 ± 4.4562
LSTM-Seq2Seq	4.703 ± 0.0036	9.035 ± 0.0065	6.021 ± 0.9436	4.673 ± 0.7343	7.191 ± 0.3673	15.853 ± 0.8100	6.239 ± 0.4209	12.459 ± 0.7952
DRPS	5.254 ± 0.6595	11.152 ± 0.8343	7.806 ± 1.0156	6.353 ± 1.0308	8.557 ± 0.8870	15.669 ± 1.0639	7.846 ± 1.2103	14.577 ± 1.3540

Table 5. Performance comparison on four cities in terms of precision and recall under the setting where the input and output lengths are, respectively, 25 and 10. The numbers in bold face indicate the ones with the best performances.

%	New York		San Francisco		Brooklyn		London	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
RAND	0.002 ± 0.0022	0.003 ± 0.0037	0.025 ± 0.0121	0.021 ± 0.0116	0.034 ± 0.0149	0.061 ± 0.0286	0.017 ± 0.0127	0.028 ± 0.0210
AMC	0.752 ± 0.6590	3.326 ± 2.7543	5.352 ± 1.9037	2.667 ± 0.7595	3.977 ± 0.9521	11.555 ± 1.7694	3.083 ± 0.7702	6.964 ± 1.7802
LORE	1.916 ± 0.3494	7.978 ± 1.6981	3.676 ± 2.2748	2.419 ± 0.2754	2.225 ± 0.3266	12.097 ± 2.0696	2.255 ± 0.4506	8.856 ± 2.6079
LSTM-Seq2Seq	2.134 ± 0.0829	8.375 ± 0.2178	8.415 ± 0.3744	3.597 ± 0.3681	4.956 ± 0.3066	13.645 ± 0.7883	3.902 ± 0.7636	10.975 ± 2.1515
DRPS	3.142 ± 0.4779	8.466 ± 1.1623	8.207 ± 0.6381	4.000 ± 0.5911	6.536 ± 0.7484	14.424 ± 0.9999	4.737 ± 0.5935	11.190 ± 1.1647

4.3. Effect of Components

As described in Section 3.3.1, four features are considered in the input module of DRPS, including the POI embedding (PE), category embedding (CE), geographical influence (GI), and positional encoding (Pos). This part aims to investigate the effect of each of these components. Specifically, each component is removed from the input module, respectively, for demonstrating the effectiveness of the corresponding component. The experimental results have been shown in Table 6.

Table 6. Effect of different components on four cities in terms of AP and OSP under the setting where the input and output lengths are, respectively, 30 and 5.

%	New York		San Francisco		Brooklyn		London	
	AP	OSP	AP	OSP	AP	OSP	AP	OSP
DRPS	9.982 ± 0.3985	19.511 ± 0.2400	13.653 ± 0.1611	18.747 ± 0.3048	11.933 ± 0.4434	23.240 ± 0.9827	15.413 ± 0.2294	29.513 ± 0.8359
Without PE	2.012 ± 0.0160	3.052 ± 0.0382	3.020 ± 0.0155	5.088 ± 0.0312	3.054 ± 0.0237	6.222 ± 0.0707	4.030 ± 0.0205	7.176 ± 0.0720
Without CE	6.396 ± 0.3026	9.938 ± 0.4332	7.332 ± 0.5356	10.830 ± 0.4786	8.086 ± 0.3146	14.426 ± 0.4782	11.712 ± 0.4892	17.404 ± 0.4020
Without GI	7.154 ± 0.2549	9.988 ± 0.2268	8.026 ± 0.2367	10.170 ± 0.2587	8.718 ± 0.3214	13.338 ± 0.3804	12.478 ± 0.4352	14.644 ± 0.4751
Without Pos	3.014 ± 0.0092	4.060 ± 0.0297	6.352 ± 0.6135	9.766 ± 1.0097	7.772 ± 0.5439	11.293 ± 0.6308	9.666 ± 0.7064	12.723 ± 0.7408

According to the results presented in Table 6, when the POI embedding and positional encoding are removed from the input module, the AP and OSP always get lower values, and the AP and OSP always get higher values when removing the category embedding and geographical influence. It means that POI embedding and positional encoding are the more important features in POI sequence recommendation. Meanwhile, the category embedding and geographical influence are also effective at improving the model performance.

4.4. Cold-Start Scenario

Recommendation systems always face the cold-start problem, where the recommendation systems have difficulty recommending reliable results because of the initial lack of data. Similarly, DRPS also faces the challenge of a cold-start when the input data are limited. In this paper, those users who have less than 35 and more than 15 check-ins are used to validate the performance of the proposed algorithm in a cold-start scenario. Specifically, each incomplete trajectory is extended to a sequence of a fixed length (35) by padding with leading-zeros. Then, the first 30 POIs are input into the trained model to get a recommendation, and the last five POIs are used to evaluate the recommendation accuracy. The experimental results have been presented in Table 7. It can be found that the performances of all methods (except for RAND) in a cold-start scenario get worse (compared with Table 2). However, DRPS is still significantly superior to the other baselines.

Table 7. Performance comparison in a cold-start scenario. The numbers in bold face indicate the ones with the best performances.

%	New York		San Francisco		Brooklyn		London	
	AP	OSP	AP	OSP	AP	OSP	AP	OSP
RAND	0.010 ± 0.0013	0.044 ± 0.0021	0.018 ± 0.0020	0.098 ± 0.0059	0.038 ± 0.0029	0.222 ± 0.0037	0.032 ± 0.0023	0.172 ± 0.0067
AMC	6.376 ± 0.5018	9.792 ± 0.5886	4.330 ± 0.3404	5.876 ± 0.3287	6.268 ± 0.3375	8.330 ± 0.4973	5.952 ± 0.5250	8.344 ± 0.4870
LORE	6.998 ± 0.3104	9.614 ± 0.4888	4.830 ± 0.2738	5.276 ± 0.3725	7.588 ± 0.4761	9.330 ± 0.6523	8.126 ± 0.3813	10.654 ± 0.5957
LSTM-Seq2Seq	6.568 ± 0.3652	10.666 ± 0.4190	6.776 ± 0.4029	9.009 ± 0.7750	9.302 ± 0.4369	18.241 ± 0.3784	11.038 ± 0.3721	21.633 ± 0.6624
DRPS	7.128 ± 0.5427	13.312 ± 0.6444	8.212 ± 0.2969	12.454 ± 0.6282	9.506 ± 0.3323	19.529 ± 0.4302	11.256 ± 0.5722	25.293 ± 0.6157

4.5. An Illustrative Example

Figure 2 illustrates an example from New York. The solid grey lines represent the historical trajectory of a user. This trajectory starts with P and ends with Q. The solid blue lines are the ground truth transitions after Q of this user. In addition, the dash purple lines are the recommended sequence by DRPS. In this case, the ground truth of output sequence is (A, B, C, D, E), and the recommended sequence is (A, C, B, D, F). According to Equations (9) and (11), the AP and OSP can be easily calculated, that is, AP = 0.4 and OSP = 0.67.

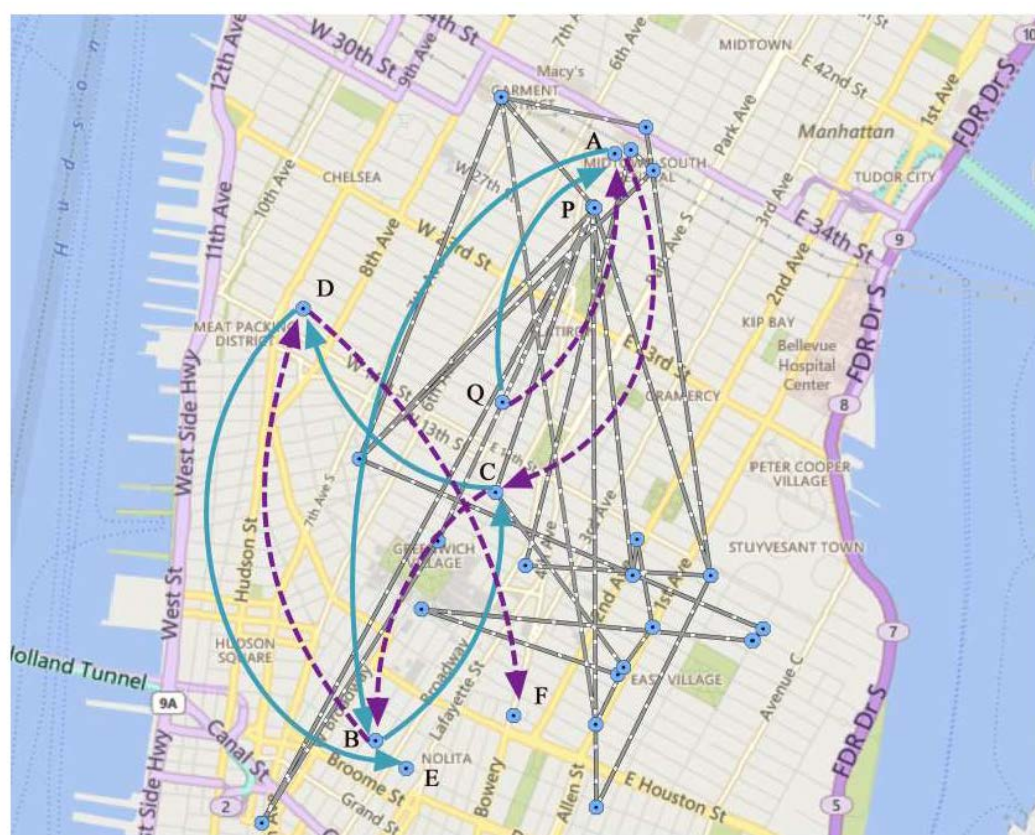


Figure 2. An example of POI sequence recommendation.

5. Conclusions and Future Work

This paper proposes the framework DRPS to recommend POI sequence based on users' historical trajectories. DRPS first models the POI sequence recommendation as a Seq2Seq learning task, and then develops a DNN-based architecture to solve it. DRPS takes the POI embedding feature, the geographical and categorical influences of historical trajectory as input, and outputs the POI sequence recommendation that the user is most interested in. Owing to the end-to-end workflow, DRPS can easily make dynamic POI sequence recommendation by allowing the input to change over time. In addition, other than precision and recall, two new metrics named AP and OSP are proposed to evaluate the recommendation accuracy of POI sequence. Differing from precision and recall, AP and OSP both take into account the visiting order of POI sequence, which provide a more reasonable way to evaluate the recommendation accuracy of POI sequence. In addition, the experimental results of each of the above metrics demonstrate the significant advantages of DRPS in the POI sequence recommendation task.

DRPS has shown its effectiveness; however, there are still some works worthy of further exploration. First, in order to get better recommendations, some other helpful information, such as the social relationship in LBSNs, should be also considered in the future. Second, considering more practical constraints when making recommendations is another important work in the future. These constraints may include time constraint, cost constraint, distance constraint, etc.

Author Contributions: Jianfeng Huang conceived and conducted this research; Yuefeng Liu, Yue Chen and Chen Jia helped in date analysis and language correction; Jianfeng Huang and Yuefeng Liu wrote the paper.

Funding: This research was funded by the National Natural Science Foundation of China, Grant No. U1433102.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Funder, D.C.; Colvin, C.R. Explorations in behavioral consistency: Properties of persons, situations, and behaviors. *J. Personal. Soc. Psychol.* **1991**, *60*, 773–94. [[CrossRef](#)]
2. Khan, S.A.; Arif, S.; Bölöni, L. Emulating the Consistency of Human Behavior with an Autonomous Robot in a Market Scenario. In Proceedings of the 13th AAAI Conference on Plan, Activity, and Intent Recognition, Bellevue, WA, USA, 14–15 June 2013; pp. 17–23.
3. Zheng, Y.; Zhou, X. *Computing with Spatial Trajectories*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2011.
4. Mei, T.; Hsu, W.H.; Luo, J. Knowledge Discovery from Community-Contributed Multimedia. *IEEE Multimed.* **2010**, *17*, 16–17. [[CrossRef](#)]
5. Cheng, C.; Yang, H.; Lyu, M.; King, I. Where you like to go next: Successive point-of-interest recommendation. In Proceedings of the IJCAI International Joint Conference on Artificial Intelligence, Beijing, China, 3–9 August 2013; pp. 2605–2611.
6. Zhao, P.; Zhu, H.; Liu, Y.; Li, Z.; Xu, J.; Sheng, V.S. Where to Go Next: A Spatio-temporal LSTM model for Next POI Recommendation. *arXiv* **2018**, arXiv:1806.06671.
7. Ding, R.; Chen, Z. RecNet: A deep neural network for personalized POI recommendation in location-based social networks. *Int. J. Geogr. Inf. Sci.* **2018**, *32*, 1631–1648. [[CrossRef](#)]
8. Liu, Q.; Wu, S.; Wang, L.; Tan, T. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 194–200.
9. Baral, R.; Li, T. Exploiting the roles of aspects in personalized POI recommender systems. *Data Min. Knowl. Discov.* **2017**, *32*, 320–343. [[CrossRef](#)]
10. De Choudhury, M.; Feldman, M.; Amer-Yahia, S.; Golbandi, N.; Lempel, R.; Yu, C. Automatic Construction of Travel Itineraries Using Social Breadcrumbs. In Proceedings of the 21st ACM Conference on Hypertext and Hypermedia, Toronto, ON, Canada, 13–16 June 2010; ACM: New York, NY, USA, 2010; pp. 35–44.
11. Bolzoni, P.; Helmer, S.; Wellenzohn, K.; Gamper, J.; Andritsos, P. Efficient Itinerary Planning with Category Constraints. In Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Dallas, TX, USA, 4–7 November 2014; ACM: New York, NY, USA, 2014; pp. 203–212.
12. Bin, C.; Gu, T.; Sun, Y.; Chang, L.; Sun, W.; Sun, L. Personalized POIs Travel Route Recommendation System Based on Tourism Big Data. In Proceedings of the 15th Pacific Rim International Conference on Artificial Intelligence, Nanjing, China, 28–31 August 2018; pp. 290–299.
13. Zhao, S.; Chen, X.; King, I.; Lyu, M.R. Personalized Sequential Check-in Prediction: Beyond Geographical and Temporal Contexts. In Proceedings of the 2018 IEEE International Conference on Multimedia and Expo (ICME), San Diego, CA, USA, 23–27 July 2018.
14. Lim, K.H.; Chan, J.; Leckie, C.; Karunasekera, S. Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency. *Knowl. Inf. Syst.* **2017**, *54*, 375–406. [[CrossRef](#)]
15. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; Association for Computational Linguistics: Doha, Qatar, 2014; pp. 1724–1734.
16. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*; Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2014; pp. 3104–3112.
17. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
18. Wang, H.; Shen, H.; Ouyang, W.; Cheng, X. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3877–3883.
19. Hosseini, S.; Yin, H.; Zhou, X.; Sadiq, S.; Kangavari, M.R.; Cheung, N.M. Leveraging multi-aspect time-related influence in location recommendation. *World Wide Web* **2019**, *22*, 1001–1028. [[CrossRef](#)]
20. Griesner, J.B.; Abdessalem, T.; Naacke, H. POI Recommendation: Towards Fused Matrix Factorization with Geographical and Temporal Influences. In Proceedings of the 9th ACM Conference on Recommender Systems, RecSys, Vienna, Austria, 16–20 September 2015; ACM: Vienne, Austria, 2015; pp. 301–304.

21. Yuan, Q.; Cong, G.; Ma, Z.; Sun, A.; Thalmann, N.M. Time-aware Point-of-interest Recommendation. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, 28 July–1 August 2013; ACM: New York, NY, USA, 2013; pp. 363–372.
22. Zhang, J.D.; Chow, C.Y. 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, Santiago, Chile, 9–13 August 2015; ACM: New York, NY, USA, 2015; pp. 443–452.
23. Ye, M.; Yin, P.; Lee, W.C.; Lee, D.L. Exploiting Geographical Influence for Collaborative Point-of-interest Recommendation. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 25–29 July 2011; ACM: New York, NY, USA, 2011; pp. 325–334.
24. Aliannejadi, M.; Crestani, F. Personalized Context-Aware Point of Interest Recommendation. *ACM Trans. Inf. Syst.* **2018**, *36*, 45. [[CrossRef](#)]
25. Feng, S.; Li, X.; Zeng, Y.; Cong, G.; Chee, Y.M.; Yuan, Q. Personalized Ranking Metric Embedding for Next New POI Recommendation. In Proceedings of the 24th International Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 2069–2075.
26. Alvarado-Uribe, J.; Gómez-Oliva, A.; Barrera-Animas, A.Y.; Molina, G.; Gonzalez-Mendoza, M.; Parra-Meroño, M.C.; Jara, A.J. HyRA: A Hybrid Recommendation Algorithm Focused on Smart POI. Ceutí as a Study Scenario. *Sensors* **2018**, *18*, 890. [[CrossRef](#)] [[PubMed](#)]
27. Yang, C.; Bai, L.; Zhang, C.; Yuan, Q.; Han, J. Bridging Collaborative Filtering and Semi-Supervised Learning: A Neural Approach for POI Recommendation. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; ACM: New York, NY, USA, 2017; pp. 1245–1254.
28. Chang, B.; Park, Y.; Park, D.; Kim, S.; Kang, J. Content-aware Hierarchical Point-of-interest Embedding Model for Successive POI Recommendation. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3301–3307.
29. Gionis, A.; Lappas, T.; Pelechrinis, K.; Terzi, E. Customized Tour Recommendations in Urban Areas. In Proceedings of the 7th ACM International Conference on Web Search and Data Mining, New York, NY, USA, 24–28 February 2014; ACM: New York, NY, USA, 2014; pp. 313–322.
30. Baral, R.; Iyengar, S.S.; Li, T.; Zhu, X. HiCaPS: Hierarchical Contextual POI Sequence Recommender. In Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 6–9 November 2018; ACM: New York, NY, USA, 2018; pp. 436–439.
31. Debnath, M.; Tripathi, P.K.; Biswas, A.K.; Elmasri, R. Preference Aware Travel Route Recommendation with Temporal Influence. In Proceedings of the 2nd ACM SIGSPATIAL Workshop on Recommendations for Location-based Services and Social Networks, Seattle, WA, USA, 6 November 2018; ACM: New York, NY, USA, 2018.
32. Cui, Q.; Wu, S.; Liu, Q.; Zhong, W.; Wang, L. MV-RNN: A Multi-View Recurrent Neural Network for Sequential Recommendation. *IEEE Trans. Knowl. Data Eng.* **2018**. [[CrossRef](#)]
33. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; pp. 5998–6008.
34. Covington, P.; Adams, J.; Sargin, E. Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; ACM: New York, NY, USA, 2016; pp. 191–198.
35. Lin, I.C.; Lu, Y.S.; Shih, W.Y.; Huang, J.L. Successive POI Recommendation with Category Transition and Temporal Influence. In Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018.
36. Hornik, K. Approximation Capabilities of Multilayer Feedforward Networks. *Neural Netw.* **1991**, *4*, 251–257. [[CrossRef](#)]
37. Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 11–13 April 2011; Gordon, G., Dunson, D., Dudík, M., Eds.; PMLR: Fort Lauderdale, FL, USA, 2011; Volume 15, pp. 315–323.

38. Kim, Y.; Denton, C.; Hoang, L.; Rush, A.M. Structured Attention Networks. *arXiv* **2017**, arXiv:1702.00887.
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
40. Liu, X.; Liu, Y.; Aberer, K.; Miao, C. Personalized Point-of-interest Recommendation by Mining Users' Preference Transition. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, San Francisco, CA, USA, 27 October–1 November 2013; ACM: New York, NY, USA, 2013; pp. 733–738.
41. Zhang, J.D.; Chow, C.Y.; Li, Y. LORE: Exploiting Sequential Influence for Location Recommendations. In Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Dallas, TX, USA, 4–7 November 2014; ACM: New York, NY, USA, 2014; pp. 103–112.
42. Usatenko, O. *Random Finite-Valued Dynamical Systems: Additive Markov Chain Approach*; Kharkov Series in Physics and Mathematics; Cambridge Scientific Publishers: Cambridge, UK, 2009.
43. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A System for Large-Scale Machine Learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; USENIX Association: Savannah, GA, USA, 2016; pp. 265–283.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).