J. Cent. South Univ. (2021) 28: 494−506
DOI: https://doi.org/10.1007/s11771-021-4617-x

Springer

DP-BPR: Destination prediction based on Bayesian personalized ranking

JIANG Feng(江峰)[1, 2], LU Zhen-ni(卢珍妮)[2, 3], GAO Min(高旻)[2, 3], LUO Da-ming(罗大明)[2, 4]

1. School of Finance and Management, Chongqing Business Vocational College, Chongqing 401331, China;
2. Key Laboratory of Dependable Service Computing in Cyber Physical Society (Ministry of Education), Chongqing 400044, China;
3. School of Big Data and Software Engineering, Chongqing University, Chongqing 400044, China;
4. Chongqing Jiaxun Construction Engineering Co., Ltd., Chongqing 401147, China

Abstract: Destination prediction has attracted widespread attention because it can help vehicle-aid systems recommend related services in advance to improve user driving experience. However, the relevant research is mainly based on driving trajectory of vehicles to predict the destinations, which is challenging to achieve the early destination prediction. To this end, we propose a model of early destination prediction, DP-BPR, to predict the destinations by users' travel time and locations. There are three challenges to accomplish the model: 1) the extremely sparse historical data make it challenge to predict destinations directly from raw historical data; 2) the destinations are related to not only departure points but also departure time so that both of them should be taken into consideration in prediction; 3) how to learn destination preferences from historical data. To deal with these challenges, we map sparse high-dimensional data to a dense low-dimensional space through embedding learning using deep neural networks. We learn the embeddings not only for users but also for locations and time under the supervision of historical data, and then use Bayesian personalized ranking (BPR) to learn to rank destinations. Experimental results on the Zebra dataset show the effectiveness of DP-BPR.

Key words: destination prediction; embedding learning; top-N prediction; Bayesian personalized ranking

Cite this article as: JIANG Feng, LU Zhen-ni, GAO Min, LUO Da-ming. DP-BPR: Destination prediction based on Bayesian personalized ranking [J]. Journal of Central South University, 2021, 28(2): 494−506. DOI: https://doi.org/10.1007/s11771-021-4617-x.

1 Introduction

In recent years, the prediction of vehicle destinations has attracted widespread attention because it can help vehicle-aid systems to realize accuracy advertising and recommend the travel route related services for a user in advance to improve the user's driving experience. The researchers proposed approaches, such as the Bayesian network [1], the neural network-based method [2, 3], and the Markov model [4], to predict the destinations. They usually take the real-time travel trajectory data with spatial concept maps and real-time travel context, cellular data, and location attributes into consideration. However, related research mainly applies the current driving trajectory of vehicles to predict the destination [5, 6].

Foundation item: Project (2018YFF0214706) supported by the National Key Research and Development Program of China; Project (cstc2020jcyj-msxmX0690) supported by the Natural Science Foundation of Chongqing, China; Project (2020CDJ-LHZZ-039) supported by the Fundamental Research Funds for the Central Universities of Chongqing, China; Project(cstc2019jscx-fxydX0012) supported by the Key Research Program of Chongqing Technology Innovation and Application Development, China

Received date: 2020-06-18; **Accepted date:** 2020-09-22

Corresponding author: JIANG Feng, Professor; Tel: +86-13206080900; E-mail: jolivergg@gmail.com; ORCID: https://orcid.org/0000-0002-3178-4640

That leads to two problems: 1) The prediction may be failed in some cases, e.g., the users may reject sharing of the current driving trajectory; 2) It is difficult to achieve the early prediction of the destination according to a driving trajectory.

To this end, this paper uses historical vehicle data to mine the travel mode of users and build a destination prediction model merely according to current travel triple (user, departure location, departure time). There are already some studies [4, 7, 8] focusing on early destination prediction, but they merely focus on mapping relationships between locations and ignore the personalized travel characteristics of users. Besides, they usually predict destinations based on the statistical probabilities of historical user destinations that makes it challenge to predict the destinations users rarely or never visited. Therefore, in this paper, we propose a model of early destination prediction. Firstly, we preprocess the historical data to get user set, time set, and location set, and their interactions. Then we learn the embeddings for users, locations, and time by a neural network model. Specifically, after initializing the corresponding embedding matrix for each set, we take the user vectors, departure time vectors, and departure location vectors from the embedding matrix corresponding to the trip triples as input, and fuse these vectors by a neural network model. Afterward, a Bayesian model-based destination recommendation method is used to maximize the probability of actual destinations under the premise of known trip triples. For this goal, we continuously optimize the embedding matrix. The embeddings finally learned can fully excavate the relationship between the trip triplet and the destination for future destination prediction.

The contributions of the paper are as follows:

1) We propose a novel task for early prediction of the vehicle destination. Different from current vehicle destination prediction based on the trajectory, the new task is to predict the destinations in the early stage, which takes only the departure time and location to predict the destination.

2) We propose a new prediction model DP-BPR based on node ranking. DP-BPR is a destination prediction model based on embedding learning, which takes the specific trip triplet (user, departure location, departure time) as the input vectors of the model, calculates the probability of the candidate destinations, and obtains the recommended results through probability ranking. The training process of the model is the learning process of node embedding, which optimizes the feature embedding of users and locations by optimizing the objective function of prediction tasks.

3) We evaluate our model on a public dataset. The effectiveness of the proposed algorithm is demonstrated by comparing it with other baseline algorithms.

The remainder of this paper is organized as follows: Section 2 introduces related work. Section 3 introduces our destination prediction model. In Section 4, we introduce experiments on a public dataset and analyze experimental results. In the final section, we conclude this paper.

2 Related work

2.1 Destination prediction based on trajectory

This kind of prediction is to predict the destination according to the partial trajectory of the current journey. BESSE et al [9] clustered the users' travel trajectory data, modeled the traffic patterns in each trajectory cluster through a mixed Gaussian distribution model, and then calculated the category of the current trajectory and summarized all the destinations in that category information to predict the travel destination. GAO et al [10] analyzed the behavior of residents' destination selection and obtained the rules of destination selection and the corresponding influencing factors. By discretizing these data and establishing a Bayesian network model for predicting the user's travel destination, an accurate prediction is achieved. LIU [11] used the K-means algorithm to convert the departure time to a time category representation by analyzing the distribution characteristics of the user's travel space and the law of space and time. Finally, a random forest model was constructed to predict the user's travel destination. ZHOU [12] proposed a weight-based map matching algorithm to alleviate the deviation between the vehicle position and the map road network, and then proposed a sub-trajectory destination prediction algorithm.

Recently years, KE [3] preprocessed the map into grids and then used a recurrent neural network to learn the characteristics of grids and time, and the positional relationships between prediction grids to improve the performance of the prediction. ZHOU [13] adopted multi-scale convolution kernels for the multi-scale characteristics and density differences of trajectory data and introduced an attention mechanism during the convolution process to increase the spatial attention of neural networks, multi-layer perceptron, and convolutional neural network for joint learning to predict the destination. ZHANG et al [14] proposed a novel data-driven ensemble learning approach for destination prediction, which combines the respective superiorities of support vector regression and deep learning at different segments of the whole trajectory.

2.2 Early destination prediction

Pre-trip destination prediction is to predict the destinations before departure, so that related systems can further provide more personalized services based on the prediction results. Research scholars have explored and proposed multiple pre-trip destination prediction models. ZHANG et al [7] proposed a probabilistic destination prediction model, which calculates the time distribution corresponding to each destination based on the historical user orders. The model then uses the Bayesian algorithm to calculate the probability distribution of the user's historical destination according to the current time. HE [4] combined travel data and personal attribute information to analyze the laws of residents' travel destinations, and adopted a Markov chain prediction models for different travel laws on working days and non-working days. ZONG et al [8] combined the Markov chain and multinomial logit model to learn and utilize the habit of multi-day destination choice in developing the pre-trip model. JIANG et al [15] proposed a deep learning model based on spatio-temporal data. First, a candidate destination pool was generated based on the locations that the user frequently went to. Then a neural network model was used to learn user behaviors and the spatial relationships between locations. Finally, the above features are integrated to generate prediction results. LIU et al [16] proposed a "two-stage" destination prediction model. First, the frequency information of historical travel data of users was counted. Second, the probability distribution of candidate destinations was predicted by combining information such as time, geographic location, and frequency of user behavior.

The aforementioned pre-trip destination prediction methods usually focus on the relationship modeling between locations, while ignoring users' personalized travel characteristics, or predict based on users' historical destination statistical probability. Therefore, their prediction performances are limited.

3 Vehicle destination prediction method based on embedding learning

Because the public dataset used in the method of early destination prediction lacks of explicit features of locations, this paper uses embedding learning techniques to discover the relationships among users, locations, and time to obtain the implicit features of users, locations, and time, and then applies the embedding to predict the destinations. In this section, we will introduce the process of data preprocessing and analysis, and present our destination prediction method.

3.1 Data preprocessing and analysis

We first pre-process the data set. Data pre-processing includes pre-processing for time and locations. The preprocessing of time is to convert the travel departure time (dpt_time) expressed in the form of year, month, day, hour, minute, and second into a specific time interval (dpt_hour) and holiday sign (dpt_day: 1/0). For example, dpt_time: 2018-01-20 10:13:43 can be converted into two columns (10, 1) that means dpt_hour is 10 and dpt_day is a holiday.

We use Geohash code to pre-process the geographic locations represented by latitude and longitude, and then we obtain discrete grid ids and divide the map into a grid. The length of the Geohash code can be defined, and the longer the coded length, the smaller the grid division, and the smaller the distance error. Nevertheless, it is not true that the longer the encoding length, the better the pre-process. That is because a long coding length will lead to too many features, and the data set will be too sparse, which is not conducive to model training, and the distance bias can mask the original data noise to a certain extent. In this paper, we set the coding length to 6 because the distance error is only 0.61 km, and the number of the locations in the map will not be too large, as shown in Table 1.

Table 1 Comparison of Geohash coding length, distance error, and number of locations

Geohash code length	Distance error/km	Number of locations
5	2.4	41245
6	0.61	146982
7	0.076	514422

In addition to a certain regularity in travel time, user travel behavior also has a certain regularity in geographical location. For example, the homogeneity of travel locations is usually high in business districts, hospitals, schools, and scenic spots. Therefore, it is also essential to analyze the user’s travel location in the dataset.

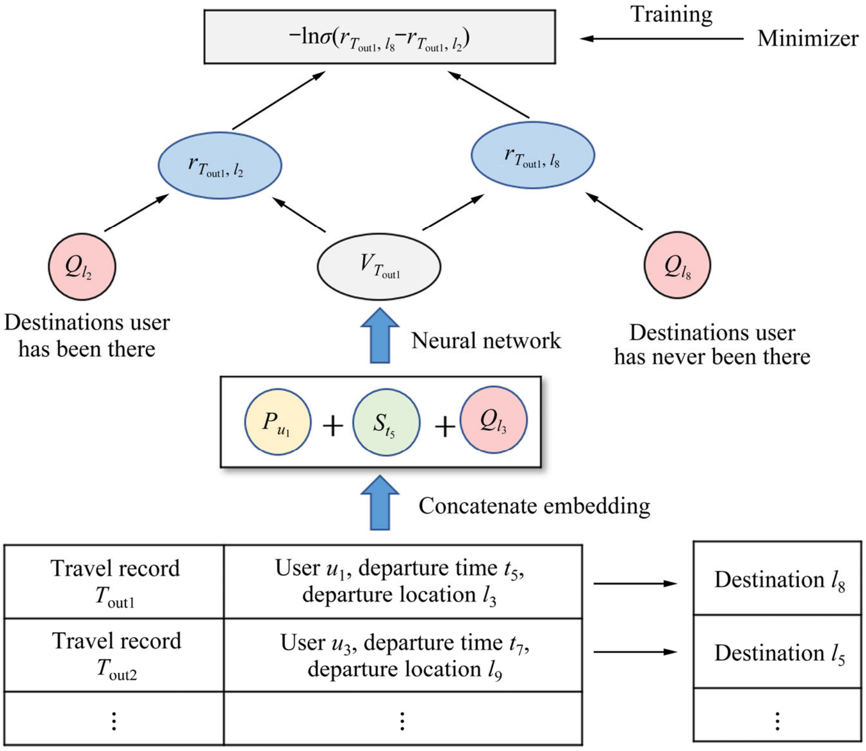
The distribution of the number of locations visited follows the “long tail effect” [17, 18], which means that most people visit a small number of locations. Because the frequently visited locations have more interactive information, when constructing the model, we should give these locations more training weights to make full use of this useful information. Therefore, in this paper, we will fully consider the characteristics of the location in the destination prediction model.

One part of this model that differs from other destination prediction methods is the modeling of user personalized travel patterns. In real life, there is a high probability that two users starting from the same location but have different destinations. According to the historical travel behavior data of different users, mining user travel preferences, and making personalized predictions can improve prediction efficiency and accuracy. Based on the user’s historical behavior data, this paper constructs the user feature embedding, together with the location feature embedding and time feature embedding to predict the destination.

3.2 Prediction model DP-BPR

To discover the implicit relationships between nodes in the heterogeneous vehicle travel network and realize the early prediction of destinations, we propose a destination prediction model based on Bayesian personalized ranking, namely DP-BPR. The framework of the model is shown in Figure 1, which mainly includes two parts: embedding representation of trip triplet and destination prediction based on the Bayesian model.

1) Embedding representation of trip triplet. Firstly, the corresponding location embedding matrix, time embedding matrix, and user embedding matrix are initialized for the location set, time set, and user set, respectively. During training,


$-\ln\sigma(r_{T_{\text{out1}}, l_8} - r_{T_{\text{out1}}, l_2})$
Training
Minimizer
r_{T_{out1}, l_2}
r_{T_{out1}, l_8}
Q_{l_2}
$V_{T_{\text{out1}}}$
Q_{l_8}
Destinations user has been there
Neural network
Destinations user has never been there
P_{u_1} + S_{t_5} + Q_{l_3}
Concatenate embedding

Travel record T_{out1}	User u_1, departure time t_5, departure location l_3	Destination l_8
Travel record T_{out2}	User u_3, departure time t_7, departure location l_9	Destination l_5
⋮	⋮	⋮

Figure 1 Framework of DP-BPR model

for a specific trip triplet, the corresponding vectors are taken from the embedding matrix, and then the concatenated vectors are compressed by a neural network model. The result vector is the features of the trip triplet.

2) Destination prediction based on the Bayesian model. After obtaining the representation of the trip triplet, a Bayesian model is used to maximize the probability of the correct destination under the premise that the trip triplet is known. Moreover, with this goal, it continuously trains and optimizes user embedding, time embedding, and location embedding, so that the learned embedding can fully reflect the relationships between the trip triplet and the destination.

3.2.1 Trip triplet embedding

The user's destination is determined by the user, the departure time, and the departure location, and they have different influences on the destination. How to extract and combine the information contained in the trip triplet and find out the key feature to determine the destination is the main task of this section.

After the data are preprocessed, we can get the location set, time set, and user set. Then we initialize the corresponding location embedding matrix $Q\in R^{L\times d}$, time embedding matrix $S\in T^{L\times d}$, and user embedding matrix $P\in R^{U\times d}$, where d represents the dimension of embedding matrix, U denotes the total number of users, L is the total number of locations, and T means the total number of times. These three embedding matrices model the characteristics of users, locations, and time nodes, where each row of the embedding matrix represents the feature vector of a specific node.

When a specific trip triplet is as an input to the model, the location, time, and user vector corresponding to the trip triplet are taken from the three embedding matrices, and the three vectors are spliced to form a high-dimensional vector $Q_{\text{concat}}\in R^{3d}$, and then the vector is compressed to a low-dimensional vector $V_{T_{\text{out}}}\in R^{d}$, which is the embedding representation of the trip triplet.

There are two reasons to compress the high-dimensional vector to a low-dimensional vector for the trip triplet. One is that the probability needs to be calculated by the dot product operation between the trip triplet embedding and the destination embedding so that the embedding representation of the trip triplet should have the same dimensions with the destination embedding. Second, the user's travel decision has different considerations. For example, at 7 a.m. on a workday, two employees of the same company live at two different communities, but they have the same destination. In this trip triplet, the travel time has the greatest impact on the destination compared to the user and the location of departure.

When a user takes a train station or an airport as a departure point, no matter what time it is, the user usually goes to the location where he will stay. Among the trip triplets currently, the departure point has the greatest impact on the destination. Therefore, the vectors of the three nodes in the trip triplet cannot be simply concatenated to predict the travel destination.

Due to the advantages of the neural network model in constructing complex non-linear features, this paper uses the neural network model to mine the complex and variable internal connection between the trip triplet node and the destination. The multi-layer perceptron model is used in our model. The effective weighted summary of the three-dimensional feature information in the trip triplet aims to excavate the key information that determines the destination.

Finally, we use the stochastic gradient descent algorithm to optimize the prediction objective function and maximize the probability of matching the correct destination with the representation of the trip triplet. For this goal, we continuously train and optimize the corresponding user embedding, time embedding, and location embedding. The embedding matrix finally obtained can well represent the useful information in the nodes. By weighting and summarizing the practical information, the key features can be mined to predict the destination.

3.2.2 Destination prediction based on Bayesian model

We first use a matrix factorization algorithm. The algorithm is usually used in recommendation systems that decomposes the rating matrix into a user latent matrix and an item latent matrix, as the low-dimensional representation of users and items, and calculates the inner product of the user vector and item vector as the prediction of rating. During training, the goal is to minimize the differences between the rating prediction and the original score; the stochastic gradient descent method is used to optimize the objective function to achieve the purpose of updating the user vector and item vector.

After obtaining the embedding of the trip triplet, the inner product of the trip triplet embedding and the candidate destination embedding is calculated as the prediction probability, that is, the probability of this destination is the candidate destination. By minimizing the differences between the predicted probability and the true probability, the embedding of trip triplets and destination embedding are continuously optimized.

However, the point-wise method of the traditional matrix factorization model does not directly optimize the destination ranking, and the Bayesian ranking model [20, 21] is a sorting algorithm based on the maximum posterior probability proposed for the above problem. Therefore, we further adopt the Bayesian personality rank model (BPR) to optimize the embedding parameters of the model.

It assumes that the probability of a destination that has a relationship with a historical trip triplet should be greater than the probability of a destination that has never had a relationship with a historical trip triplet. If it is observed that a trip triplet (user u, departure time t, departure point l_1) has ever been to destination l_2, but has never been to destination l_3, then the probability of the triplet associated with l_2 is greater than that of the triplet associated with l_3.

Based on the above assumptions, we construct two sets of locations for each trip triple T_{out}, a positive feedback destination set PL, and a negative feedback destination set NL. Let L be the set of all locations in the data set. If a location loc was visited by the trip triplet, that is, the loc was used as the true destination of the trip triplet, then $loc \in PL$; otherwise $loc \in NL$, and $L = PL \cup NL$.

In this way, multiple destination ranking pairs can be constructed for each trip triplet. Because the trip behaviors are independent of each other, the posterior probability can be calculated, as shown in Eq. (1).

$$\prod_{T_{\text{out}},PL,NL} p(\Theta \mid PL >_{T_{\text{out}}} NL) \propto \prod_{T_{\text{out}},PL,NL} p(PL >_{T_{\text{out}}} NL \mid \Theta) p(\Theta) \tag{1}$$

where T_{out} represents the trip triplet, and $PL >_{T_{\text{out}}} NL$ indicates that for the trip triplet, the probability of its destination in the set PL is higher than that in the set NL, that is, the historical destination of the trip triplet has high probability of being recommended.

The calculation equation of conditional probability is:

$$p(PL >_{T_{\text{out}}} NL \mid \Theta) = \sigma(r_{T_{\text{out}},PL} - r_{T_{\text{out}},NL}) \tag{2}$$

where $r_{T_{\text{out}},PL}$ is the predicted probability, which is obtained by the dot product operation from the embedding of the trip triplet and the embedding of the candidate destination:

$$r_{T_{\text{out}},PL} = V_{T_{\text{out}}} \cdot Q_{loc} \tag{3}$$

Finally, taking the log-likelihood function for the posterior probability, the optimization objective can be obtained as:

$$\begin{aligned} L &= \arg \mathrm{ma}\, x_{\Theta} \ln \prod_{T_{\text{out}},l_p \in PL,l_n \in NL} p(l_p >_{T_{\text{out}}} l_n \mid \Theta) p(\Theta) \\ &= \sum_{T_{\text{out}},l_p \in PL,l_n \in NL} \ln \sigma(r_{T_{\text{out}},l_p} - r_{T_{\text{out}},l_n}) + \ln p(\Theta) \\ &= \sum_{T_{\text{out}},l_p \in PL,l_n \in NL} \ln \sigma(r_{T_{\text{out}},l_p} - r_{T_{\text{out}},l_n}) - \lambda_{\Theta} \| \Theta \|^2 \end{aligned} \tag{4}$$

where Θ is the parameters that the model needs to be trained, including user latent factor matrix, location latent factor matrix, time latent factor matrix, and weight parameters w and bias vector b in the neural network model. We use a gradient descent method [21] iteratively to get the final model parameters.

The above optimization method based on the Bayesian model can give the frequently visited locations more weight in the training process because the frequently visited locations can frequently appear in the set of positive feedback destinations, and their feature embeddings can be constantly optimized and updated. While the infrequently visited locations are more likely to appear in the negative feedback destination set. Due to the random sampling method of the locations in the negative feedback destination set during the model training process, the feature embeddings of these locations are updated infrequently.

The flowchart of the DP-BPR model is shown in Figure 2.

When predicting destinations, the model first calculates the dot product of the embedding of the travel triplet and the embedding of the candidate destination. It then obtains the probability of each locations and arranges them in descending order to form top N recommended list.

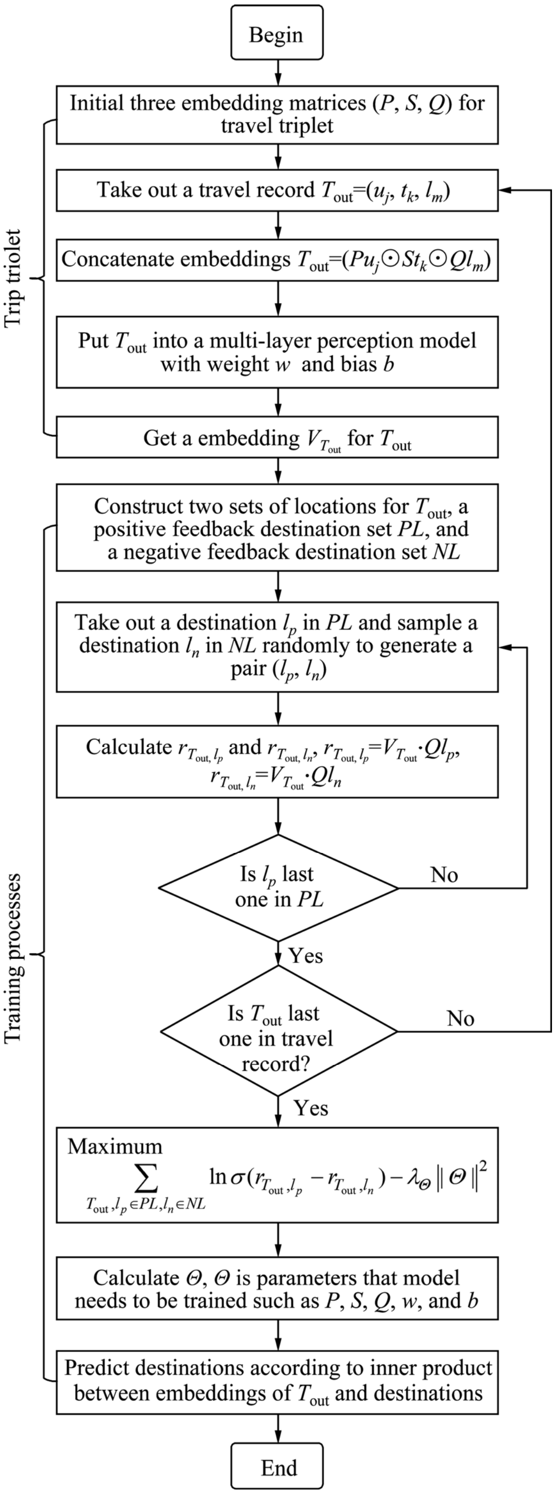

Begin
Initial three embedding matrices (P, S, Q) for travel triplet
Take out a travel record $T_{\text{out}}=(u_j, t_k, l_m)$
Concatenate embeddings $T_{\text{out}}=(Pu_j \odot St_k \odot Ql_m)$
Put T_{out} into a multi-layer perception model with weight w and bias b
Get a embedding $V_{T_{\text{out}}}$ for T_{out}
Trip triolet
Construct two sets of locations for T_{out}, a positive feedback destination set PL, and a negative feedback destination set NL
Take out a destination l_p in PL and sample a destination l_n in NL randomly to generate a pair (l_p, l_n)
Calculate r_{T_{out},l_p} and r_{T_{out},l_n}, $r_{T_{\text{out}},l_p}=V_{T_{\text{out}}}\cdot Ql_p$, $r_{T_{\text{out}},l_n}=V_{T_{\text{out}}}\cdot Ql_n$
Is l_p last one in PL
No
Yes
Is T_{out} last one in travel record?
No
Yes
Maximum $\sum_{T_{\text{out}},l_p\in PL,l_n\in NL} \ln\sigma(r_{T_{\text{out}},l_p}-r_{T_{\text{out}},l_n})-\lambda_{\Theta}\|\Theta\|^2$
Calculate Θ, Θ is parameters that model needs to be trained such as P, S, Q, w, and b
Predict destinations according to inner product between embeddings of T_{out} and destinations
Training processes
End

Figure 2 Flowchart of DP-BPR model

3.2.3 Destination prediction algorithm

The proposed vehicle travel destination prediction algorithm based on embedding learning is shown in Algorithm 1. The algorithm takes a specific trip triplet (user, departure time, and departure location) as input, concatenates the corresponding user vector, departure time vector, and departure location vector, and compresses it through the neural network model to obtain a new trip triplet embedding $V_{T_{\text{out}}}$. $V_{T_{\text{out}}}$ represents the summarization of valid information in the trip triplet for destination prediction. Then it uses the optimization algorithm based on the Bayesian model to maximize the probability of historical destinations under the premise of known trip triples and optimizes the parameters in the model, including user latent factor matrix, location latent factor matrix, time latent factor matrix, and the weight parameter w and bias b in the neural network model, which enables the learned embedding to fully mine the potential relationship between the triple node and the destination.

Algorithm 1 Vehicle travel destination prediction algorithm based on embedding learning

Input: Vehicle travel record R; Initialization: user embedding matrix P, location embedding matrix Q, time embedding matrix S, multilayer perceptron weight matrix w, multilayer perceptron bias b

Output: Destination prediction results

1: Pre-process the location and time in vehicle travel record R to form trip triples (user, departure time, and departure location)

2: for trip triple T_{out} in R do

3: Take the user vector P_u, the departure location vector Ql, and departure time vector S_t from the corresponding embedding matrix and concatenate them

4: The multi-layer perceptron compresses the concatenated vector to obtain a new vector as an representation of the trip triplet

5: Divide the set of locations into a visited location set PL, and an unvisited location set NL

6: To form location ranking pairs: ($l_p\in PL$, $l_n\in NL$)

7: end for

8: By maximizing the conditional probability of location ranking pairs, continuously update the corresponding user vector, departure vector, time vector, destination vector, and multi-layer perceptron weights w and bias b.

9: When predicting, calculate the dot product of the embedding of the travel triplet and the embedding of the candidate destination, obtain the probability of each location, and arrange them in descending order.

10: Choose top N locations as recommended destinations.

4 Experiments and analysis

In this section, we first introduce the experiment setup and then compare the performances of the model and baselines. Furthermore, we analyze the effectiveness of the parameters in the model, demonstrate the role of BPR, and compare the performance of the prediction on cold-start users.

4.1 Experiment setup

(1) Dataset

The Zebra Mathematics Dataset [22] is used to evaluate the DP-BPR model. There are 1495814 rows of data for 5817 users in the dataset. Each row represents a specific travel behavior, including the sample id, user id, travel start and end time, and the latitude and longitude information of the departure and destination. The dataset contains user travel data from January to July. In the experiments, we use data from January to June as the training set, and the data in July as the test set.

(2) Metrics

This paper expands the traditional destination prediction to destination top-*N* prediction [23]. The value of *N* is set to 1, 3, and 5 in the experiments. When *N*=1, the destination with the highest probability is selected as the prediction result, i.e., the traditional destination prediction task. Two common evaluation metrics for top-*N* recommendation tasks are used in the experiment: HR (hit ratio) and NDCG (normalized discounted cumulative gain). HR is based on the recall rate, which measures whether the prediction results in line with users' preferences:

$$HR@N=\frac{Hits@N}{GT} \tag{5}$$

where *Hits@N* means how many prediction results appear in the list of items that the user really likes in the top *N* recommendation list, and *GT* is the number of items in the test set.

NDCG is used to measure the consistency between the destination ranking of users in the test set and the ranking of locations in the top-*N* prediction list:

$$NDCG@N=\frac{DCC@N}{IDCG} \tag{6}$$

$$DCG@N=\sum_{i=1}^{N}\frac{rel_i}{\log_2(i+1)} \tag{7}$$

$$IDCG@N=\sum_{i=1}^{|REL|}\frac{2^{rel_i}-1}{\log_2(i+1)} \tag{8}$$

where rel_i is the graded relevance of the result at position *i*, and *REL* represents the list of recommendation items.

(3) Baselines

To demonstrate the effectiveness of the proposed algorithm, we compare the proposed model with several baselines. These baselines include closely related work based on KNN (*K* nearest neighbors), NB (naive Bayes) [7], and commonly used models in pre-trip destination prediction such as LR (logistic regression) and MLP (multilayer perceptron) [25−28]. It should be noticed that we achieved the up-to-date models by ourselves in the experiments because they are not open-sourced.

1) Prediction model based on KNN [7, 24]

This model uses KNN to find *K* historical trips similar to the current trip. The destination for most of the *K* historical trips is the prediction result of the current trip.

2) Prediction model based on NB [7]

This model learns the joint probability distribution from input to output based on the Bayes theorem and the independent hypothesis of feature conditions. It then obtains the destination with the maximum posterior probability as the prediction result.

3) Prediction model based on RF (random forest) [25]

The model is composed of several decision trees, and the final prediction results are obtained based on the majority voting principle.

4) Prediction model based on LR [26, 27]

This prediction model uses linear regression to combine the independent variables (departure time, departure place) and parameters, and then maps the predicted value to the destination probability value through sigmoid function. The place with the maximum probability is the destination prediction result.

5) Prediction model based on MLP [17, 28]

This model combines travel information with weight and bias parameters and then limits the output to [0,1] through an activation function. After the multi-level calculation, the model finally outputs the probability distribution of the destination.

Although the trajectory-based destination prediction models are also related to our work, they are on-trip destination task, which is different from the pre-trip destination prediction task in this paper. The two prediction tasks are different stages for destination prediction. Therefore, the proposed model in this paper cannot be compared with the destination prediction models based on trajectory.

4.2 Performance of DP-BPR model

As shown in Table 2, the performance of the proposed DP-BPR model is better than that of other baselines from the top-1 prediction task to the top-5 prediction task. For the best-performing RF classification model in the baselines, DP-BPR can also achieve a performance improvement of at least 4.3%. The result shows that the proposed DP-BPR model incorporates the potential relationships, e.g., the complex interrelationships between users, departure locations, departure time triplets, and the destinations, into the learning process, which makes the generated embedding representation can generate more accurate destination prediction results.

As the *N* value in the top-*N* recommendation increases, all the models' performance improves significantly. Notably, when the value of *N* is 5, the performance of the model improves greatly improved. This shows that both the DP-BPR model and baselines can put the destination that the user really visits in top of the recommendation list, and DP-BPR outperforms others in destination prediction task, i.e., it puts the correct result in top of the recommendation list.

The performance of the NB-based model is the worst among those of the models. This shows that NB is challenging to generate the prediction results by probability statistics of historical user travel destinations. Meanwhile, the performance of RF and KNN is better than other baselines. The results demonstrate that the models based on simple attribute feature division, i.e., KNN and RF, are more suitable than the models that construct sophisticated non-linear features, i.e., LR and MLP.

4.3 Parameter analysis

There are two hyperparameters in the DP-BPR model proposed in this paper: the dimension *d* of the embedding vector and the network layers number *n* of the multi-layer perceptron.

We first fix the number of network layers *n*, and observe the predication effect of the model by changing the value of the embedding dimension *d*. The value of *d* is limited to the range [10, 50], and the prediction effect is shown in Figure 3(a) and Figure 3(b) that respond to the performance of top-3 and top-5 predictions, respectively. We omit the top-1 prediction result since it has the same trend of top-3 and top-5. According to the figures, we see that when *d* is less than 30, increasing the dimension can effectively improve the model prediction performance, and the model performance is optimized at the point that *d* is equal to 30. After that, the increase of *d* can no longer improve the model performance. That is because a large *d* value makes the model overfit. Hence, we set *d* to 30 in the experiments.

Then we fix the embedding dimension *d* and adjust the number of MLP network layers to observe the performance changes of the model. Here set the network layer number to 1, 2, 3. The experimental results are shown in Figure 4(a) and Figure 4(b). Due to space limitations, only the results of top-3 and top-5 predictions are displayed, and the top-1 recommendation has the same trend. According to the figures, we see that the performance of the model is optimal when *n* is 2 on both top-3 and top-5 recommendations. If *n* is too small, the layers of the neural network model are too few to obtain the complex non-linear relationship between the destination with the trip triplet. On the contrary, if the value of *n* is too large, the model is too complicated, and over-fitting

Table 2 Experimental comparison results

Top-*N*	Metrics	KNN	NB	RF	LR	MLP	DP-BPR
Top-1	HR@1	0.5432	0.1752	0.5632	0.3126	0.4501	**0.5876**
Top-1	NDCG@1	0.5432	0.1752	0.5632	0.3126	0.4501	**0.5876**
Top-3	HR@3	0.6319	0.5100	0.6763	0.4543	0.5654	**0.7295**
Top-3	NDCG@3	0.5975	0.3641	0.6304	0.3935	0.5179	**0.6707**
Top-5	HR@5	0.6540	0.6630	0.7029	0.5233	0.6341	**0.7805**
Top-5	NDCG@5	0.6187	0.4401	0.6415	0.4215	0.5459	**0.6920**

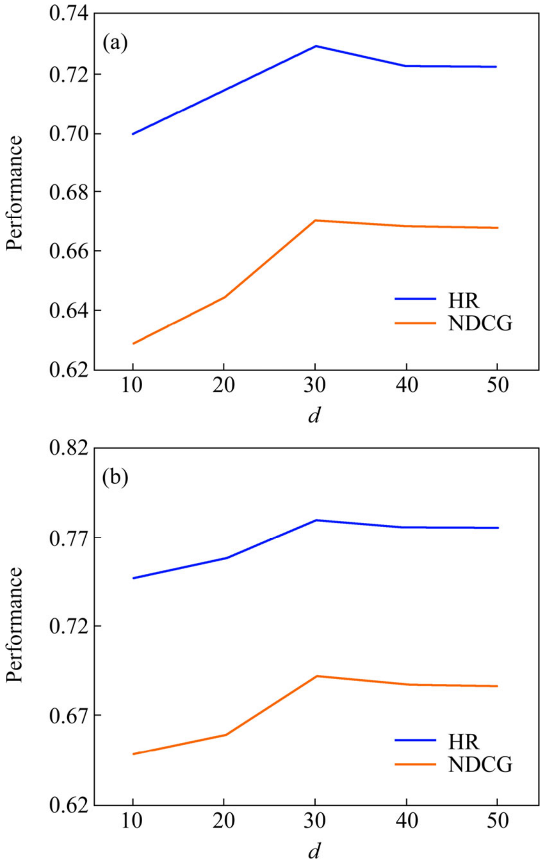


Figure 3 Impact of embedding dimension value *d* on model performance: (a) Top-3; (b) Top-5

occurs. Therefore, we set *n* to be 2 in the experiments.

4.4 Performance of model with or without BPR

As mentioned in Section 3.2.2, the Bayesian model-based posterior probability maximization optimization strategy is more suitable for personalized sorting tasks. To demonstrate the effectiveness of the BPR optimization strategy, we compare the error minimization optimization strategy used by traditional matrix factorization with the optimization strategy of the maximum posterior probability of the Bayesian model.

The results shown in Table 3 indicate that the Bayesian model-based optimization strategy can significantly improve the prediction performance of the model, especially for the top-1 recommendation, which is the traditional destination prediction task. That is because, under the premise of a given user, the ranking algorithm based on the Bayesian model can maximize the posterior probability of a real destination and reduce the probability of locations that it has never visited so that the learned embedding can effectively distinguish the travel behavior patterns of users, and achieve personalized destination prediction.

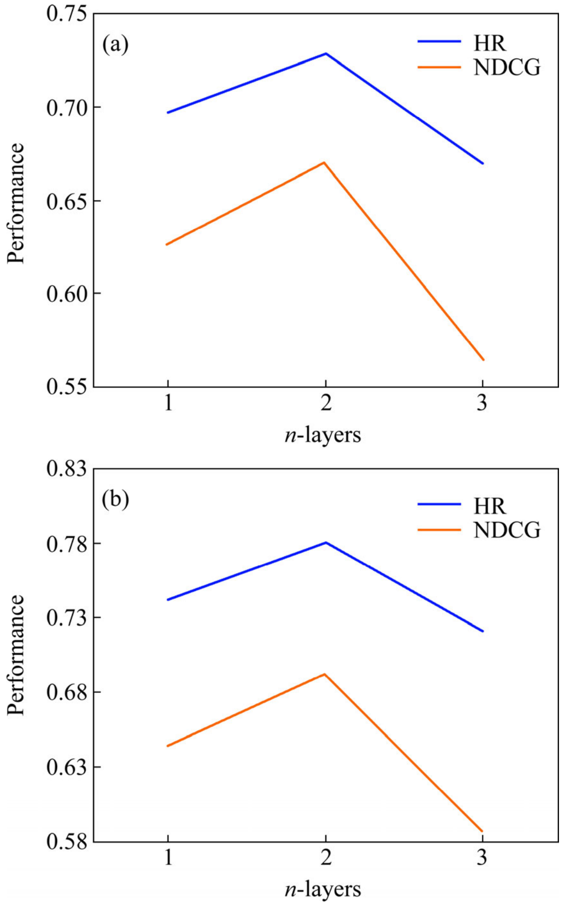


Figure 4 Impact of number of layers in MLP network on model performance: (a) Top-3; (b) Top-5

Table 3 Impact of BPR optimization strategy on model performance

Strategy	With_BPR	Without_BPR
Top-1_HR	0.5876	0.4013
Top-1_NDCG	0.5876	0.4013
Top-3_HR	0.7295	0.6608
Top-3_NDCG	0.6707	0.5476
Top-5_HR	0.7805	0.7716
Top-5_NDCG	0.692	0.5938

4.5 Performance for cold-start users

In the system, the cold-start users have less travel behavior information than regular users, and it is challenging to explore their travel patterns due to the sparse data. In this section, we focus on the performance of the models for cold-start users and demonstrate whether the DP-BPR model can better mine and combine the useful information of the users to generate predictions. The results are shown in Table 4.

Table 4 Experimental comparison results for cold-start users

Recommendation	Metrics	KNN	NB	RF	LR	MLP	DP-BPR
Top-1	HR@1	0.4528	0.3396	0.4592	0.4664	0.4717	**0.5094**
Top-1	NDCG@1	0.4528	0.3396	0.4592	0.4664	0.4717	**0.5094**
Top-3	HR@3	0.5094	0.6038	0.5283	0.5849	0.6038	**0.6226**
Top-3	NDCG@3	0.4836	0.4989	0.5004	0.5180	0.5476	**0.5759**
Top-5	HR@5	0.5109	0.6303	0.5472	0.6029	0.6226	**0.6604**
Top-5	NDCG@5	0.4932	0.5224	0.5086	0.5326	0.5557	**0.5913**

The following conclusions can be obtained from Table 4:

(1) For cold start users, the recommendation effect of the DP-BPR model is still better than all comparison methods. That shows that the DP-BPR model is more suitable for cold-start users than baselines. Our model is still able to use embedding learning techniques to model implicit features of user nodes even the user’s travel history is sparse. Thanks to the DP-BPR model uses a deep neural network model that carries out a weighted combination of the information in the travel information of the user, the departure location, and the departure time, the model can make full use these abundant features to get the embedding of travel triple when lack of the user information.

(2) The performance of LR and MLP is better than other comparison methods. The result is opposite to the result in Section 4.2 because the travel history of cold users is fewer than that of regular users. In this cold user scenario, the models such as LR and MLP can construct sophisticated non-linear features and mine the profound relationships between travel information and destination.

5 Conclusion and future work

In the paper, we propose a destination prediction model that discovers the potential connection between departure triples (user, departure time, departure location) and destinations. In the model, we first use embedding learning-based neural networks to model the user’s specific travel triplet. We then calculate the probability of each candidate destination’s embedding and arrange the candidate destinations in descending order of probability. After that, we select the top N destinations as the result for destination recommendation. The optimization of our model is to minimize the target function of the prediction task, and meanwhile regularize the embedding of three types of nodes: user, location, and time. Finally, the effectiveness of the proposed model is verified by evaluating and comparing the performance of other prediction models and ours.

In the future work, we will consider more implicit relationships, such as the relations and potential connections between users, to further improve the performance of the prediction model. Moreover, more specific features will be explored for the destination prediction.

Contributors

JIANG Feng developed the overarching research goals. JIANG Feng and LU Zhen-ni wrote the initial draft of the manuscript. LU Zhen-ni and GAO Min carried out the experiments. JIANG Feng and LUO Da-ming analyzed the experimental results.

Conflict of 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.

References

[1] DASH M, KOO K K, KRISHNASWAMY S P, JIN Y, SHI-NASH A. Visualize people’s mobility-both individually and collectively-using mobile phone cellular data [C]// IEEE International Conference on Mobile Data Management. IEEE, 2016, 1: 341−344. DOI: 10.1109/MDM.2016.59.

[2] de BRÉBISSON A, SIMON É, AUVOLAT A, VINCENT P, BENGIO Y. Artificial neural networks applied to taxi destination prediction [EB/OL]. 2015: arXiv: 1508.00021 [cs.LG]. https://arxiv.org/abs/1508.00021. DOI: 10.13140/RG.2.2.20264.26888.

[3] KE Yang-bin. Travel destination prediction based on deep learning and regression classification model [D]. Hangzhou: Zhejiang University, 2018. (in Chinese)

[4] HE Ya-nan. Travel destination prediction based on Markov model [D]. Changchun: Jilin University, 2017. (in Chinese)

[5] LV Jian-ming, SUN Qing-hui, LI Qing, MOREIRA-MATIAS L. Multi-scale and multi-scope convolutional neural networks for destination prediction of trajectories [J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(8): 3184−3195. DOI: 10.1109/TITS.2019.2924903.

[6] XU Jia-jie, ZHAO Jing, ZHOU Rui, LIU Cheng-fei, ZHAO Peng-peng, ZHAO Lei. Destination prediction a deep learningbased approach [J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 33(2): 651−666. DOI: 10.1109/ TKDE.2019.2932984.

[7] ZHANG Ling-yu, HU Tao, MIN Yue, WU Guo-bin, ZHANG Jun-ying, FENG Peng-cheng, GONG Ping-hua, YE Jie-ping. A taxi order dispatch model based on combinatorial optimization [C]// Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. HalifaxNS Canada. New York, NY, USA: ACM, 2017: 2151−2159. DOI: 10.1145/3097983.3098138.

[8] ZONG Fang, TIAN Yong-da, HE Ya-nan, TANG Jin-jun, LV Jian-yu. Trip destination prediction based on multi-day GPS data [J]. Physica A: Statistical Mechanics and its Applications, 2019, 515: 258−269.

[9] BESSE P C, GUILLOUET B, LOUBES J M, ROYER F. Destination prediction by trajectory distribution-based model [J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 19(8): 2470−2481. DOI: 10.1109/TITS.2017.2749413.

[10] GAO Jing-xin, JUAN Zhi-cai, NI An-ning. Modeling and application of traveler destination selection behavior based on Bayesian network [J]. Journal of System& Management, 2015, 24(1): 32−37. DOI: 10.1016/S0169-4332(96)00598-3.

[11] LIU Rui-han. Research on user behavior analysis based on vehicle trajectory data [D]. Beijing: University of Chinese Academy of Sciences, 2019. (in Chinese)

[12] ZHOU Ling-tong. Research on the destination prediction algorithm based on the historical traveling track [D]. Guangzhou: South China University of Technology, 2015. (in Chinese)

[13] ZHOU Xiao-yun. Research on traveling destination prediction technology based on multi-scale convolution neural network [D]. Beijing: Beijing University of Posts and Telecommunications, 2019. (in Chinese)

[14] ZHANG Xiao-cai, ZHAO Zhi-xun, ZHENG Yi, LI Jin-yan. Prediction of taxi destinations using a novel data embedding method and ensemble learning [J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(1): 68−78.

[15] JIANG Jian, LIN Fei, FAN Jin, LV Hang, WU Jia. A destination prediction network based on spatiotemporal data for bike-sharing [J]. Complexity, 2019: 7643905. DOI: 10.1155/2019/7643905.

[16] LIU Yang, JIA Ruo, XIE Xue, LIU Zhi-yuan. A two-stage destination prediction framework of shared bicycles based on geographical position recommendation[J]. IEEE Intelligent Transportation Systems Magazine, 2018, 11(1): 42−47. DOI: 10.1109/MITS.2018.2884517.

[17] LI Wen-tao, GAO Min, LI Hua, XIONG Qing-yu, WEN Jun-hao, LING Bin. A shilling attack detection algorithm based on the feature of popularity classification [J]. Acta Automatica Sinica, 2015, 41(9): 1563−1576. DOI: 10.16383/ j.aas.2015.c150040.

[18] ZHOU Tao, HAN Xiao-pu, YAN Xiao-yong, YANG Zi-mo, ZHAO Zhi-dan, WANG Bing-hong. Statistical mechanics of time and space characteristics of human behavior [J]. Journal of University of Electronic Science and Technology of China, 2013, 42(4): 481−540. DOI: 10.3969/j.issn.1001-0548.2013. 04.001.

[19] WANG Xin, LU Wei, ESTER M, WANG Can, CHEN Chun. Social recommendation with strong and weak ties [C]// Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. New York, NY, USA: ACM, 2016: 5−14. DOI: 10.1145/2983323.2983701.

[20] ZHAO Tong, MCAULEY J, KING I. Leveraging social connections to improve personalized ranking for collaborative filtering [C]// Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management-CIKM'14. Shanghai, China. New York: ACM Press, 2014: 261−270. DOI: 10.1145/2661829.2661998.

[21] WANG Liang, YU Zhi-wen, GUO Bin, KU Tao, YI Fei. Moving destination prediction using sparse dataset [J]. ACM Transactions on Knowledge Discovery from Data, 2017, 11(3): 1−33. DOI: 10.1145/3051128.

[22] DC competition [EB/OL]. https://www.pkbigdata.com/ common/bbs/topicDetails.html?tid=2949.

[23] TANG Jia-xi, WANG Ke. Personalized top-*N* sequential recommendation via convolutional sequence embedding [C]// Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining-WSDM'18. New York, NY: ACM Press, 2018: 565−573. DOI: 10.1145/3159652.3159656.

[24] ZHANG Shi-chao, LI Xue-long, ZONG Ming, ZHU Xiao-feng, WANG Rui-li. Efficient KNN classification with different numbers of nearest neighbors [J]. IEEE Transactions on Neural Networks and Learning Systems, 2017, 29(5): 1774−1785. DOI: 10.1109/TNNLS.2017. 2673241.

[25] BREIMANL. Random forests [J]. Machine Learning, 2001, 45(1): 5−32. DOI: 10.1023/A:1010933404324.

[26] HWANG C S, KAO Y C, YU Ping. Integrating multiple linear regression and multicriteria collaborative filtering for better recommendation [C]// 2010 International Conference on Computational Aspects of Social Networks. IEEE, 2010: 229−232. DOI: 10.1109/CASoN.2010.59.

[27] LIU Ling-bo, QIU Zhi-lin, LI Guan-bin, WANG Qing, OUYANG Wan-li, LIN Liang. Contextualized spatial–temporal network for taxi origin-destination demand prediction [J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(10): 3875−3887. DOI: 10.1109/TITS.2019.2915525.

[28] TOPALLI I, KILINC S. Modelling user habits and providing recommendations based on the hybrid broadcast broadband television using neural networks [J]. IEEE Transactions on Consumer Electronics, 2016, 62(2): 182−190. DOI: 10.1109/ TCE.2016.7514718.

(Edited by YANG Hua)

中文导读

基于贝叶斯个性化排序的目的地预测

摘要：目的地预测可以帮助车辆辅助系统提前推荐相关服务，改善用户的驾驶体验，受到研究者的广泛关注。然而，相关研究主要是基于车辆的行驶轨迹来预测目的地，难以实现早期的目的地预测。为此，本论文提出了一个早期目的地预测模型 DP-BPR，通过用户的出行时间和地点来预测目的地。该模型的实现有三个方面的挑战：1）稀疏的历史数据使得直接从原始数据中预测目的地非常困难；2）目的地不仅与出发点有关，而且与出发时间有关，在预测时应将两者都考虑在内；3）如何从历史数据中准确地学习目的地偏好。为了应对这些挑战，我们利用深度神经网络将稀疏的高维数据映射到稠密的低维空间，并学习用户、位置和时间的嵌入，然后，使用贝叶斯个性化排序学习并对目的地进行排名。在 Zebra 数据集上进行了实验，实验结果表明了 DP-BPR 的有效性。

关键词：目的地预测；嵌入学习；排序预测；贝叶斯个性化排序