A Deep Learning Model Based on Sparse Matrix for Point-of-Interest Recommendation

Jun Zeng
School of Big Data & Software Engineering
Chongqing University
Chongqing, China
zengjun@cqu.edu.cn

Haoran Tang
School of Big Data & Software Engineering
Chongqing University
Chongqing, China
tanghaoran@cqu.edu.cn

Yinghua Li
School of Big Data & Software Engineering
Chongqing University
Chongqing, China
yinghuali@cqu.edu.cn

Xin He
School of Big Data & Software Engineering
Chongqing University
Chongqing, China
hexin@cqu.edu.cn

***Abstract*—Point-of-interest (POI) recommendation that consists of location-based social networks (LBSNs) and provides personal services for users has become an important part in the field of recommendation system. Due to the sparseness of user check-in matrix, POI recommendation faces great challenges. However, most researches just consider of spatial and temporal impact on recommendation and do not solve the problem of sparsity. This paper proposes a POI recommendation model called RBMNMF which is based on sparse matrix of user check-ins. Firstly, by stacking restricted Boltzmann machines (RBM), the potential relationship between users and POIs is learned and multiple user-POI matrices are extracted. Second, fill the original sparse matrix by using non-negative matrix factorization (NMF). Finally, fuse those prediction matrices to generate final POI recommendation for users, which is benefit for solving the problem of sparsity effectively. Experiments on real-world data set prove that the model we propose has a better accuracy than traditional algorithms.**

Keywords—Point-of-Interest Recommendation, Social Network, Restricted Boltzmann Machine, Non-Negative Matrix Factorization, Hybrid Mode

I. Introduction

Mobile internet technology has developed at a high speed which makes location-based social networks (LBSNs) become popular such as Foursquare, Gowalla, Yelp and Facebook. Compared with traditional social networks, LBSNs allow users share locations with their friends by check-in on POIs (such as cinemas, amusement parks and restaurants). The number of POIs has increased faster at present, so it's necessary to recommend satisfying POIs to users for saving choice time and improving their experience of life in city. However, mining locations among a large number of POIs that user may have interest in is a great challenge.

Different from traditional recommendation system, the challenge of POI recommendation is bigger because the user-POI check-in matrix is a high-dimension sparse matrix, which makes analysis of user-POI matrix more difficult. From some popular data sets such as Foursquare and Gowalla, we observe that sometimes a user just visited a few locations, leading to unreasonable recommendations under the whole location space. Therefore, data sparsity has become a critical problem for POIs recommendation, which is needed to be emphasized and solved. The classical collaborative filtering (CF) algorithm calculates the similarity between users or recommend locations based on check-ins. But faced with the high-dimension sparse matrix of user check-in data, the similarity calculation of users could not be achieved and CF does not solve the problem of data sparsity.

For overcoming the sparsity, this paper proposes a deep learning model based on sparse user check-in matrix for POI recommendation, which constructs bidirectional analysis of users and POIs and aims to recommend POIs to users that meet users' preferences. The main contributions of this paper are as follows.

- Propose a deep learning model called RBMNMF for POI recommendation that is based on sparse matrix of user check-ins.
- Construct strong correlation values between users and POIs to observe and analyze users' preferences for POIs by calculating bidirectional scores between users and POIs.
- Combine stacking restricted Boltzmann machines with non-negative matrix factorization to solve the problem of data sparsity fundamentally.
- Experiments on real-world data set proves that RBMNMF we propose outperforms other traditional algorithms in terms of data sparsity.

DOI reference number: 10.18293/SEKE2019-156

This paper consists of five sections. Section 2 introduces some related algorithms of POI recommendation in LBSNs. Section 3 explicates the RBMNMF model we propose for sparse matrix of user check-ins. Section 4 shows experiments we have and the performance of our model. Section 5 summarizes this paper and discusses the future work.

II. Releted Work

At present, location-based POI recommendation in social networks has attracted wide attention from researchers who comes from different fields. Quan et al. [1] considered that the time factor plays an important role in POI recommendation because users visit different locations in different time periods and proposed a collaborative recommendation algorithm combining time factor. References [2-5] regarded users' mobile trajectories as key information when recommending and mined those trajectories to recommend POIs. Ramesh [6] proposed a fusion algorithm that integrated time, space and social relationships into a unified framework for POI recommendation. There have been many achievements for POI recommendation so far, which can be divided into the following three aspects.

- Collaborative filtering (CF): Although CF is the most famous recommendation algorithm, accurate recognition for similar user is a great challenge to CF because of the sparse matrix of user check-ins. Meanwhile, CF ignores lots of user information that is helpful for recommending. Shu et al. [7] proposed a collaborative filtering recommendation algorithm based on topic model to extract user preference of topics (such as culture, history, landscape and so on) information and recommended comprehensive POIs for users. Yang et al. [8] designed a general semi-supervised learning framework based on context information and reduced data sparsity by smoothing adjacent users.

- POI recommendation based on mobile trajectory: User's daily trajectory of movement is an important behavior pattern. Ya [9] extracted semantic information from user GPS trajectory to recommend POIs for users based on time, space and popularity. Zheng et al. [10] proposed a possible path algorithm for uncertain trajectory based on historical trajectory to reduce the uncertainty of user's trajectory. Thus, it is obvious that trajectory plays an important role for POI recommendation.

- POI recommendation based on geographical impact: Distance of a location has a great impact on user's preference for POI. For example, people will not tend to choose location that is far away from people's current locations. Therefore, recommendation system could filter out distant locations [11-13]. Ying et al. [14] considered user's social intention, preference, location popularity and other factors to calculate prior probability of POIs for users.

All algorithms mentioned above recommend POIs for users only by considering additional information of users' sparse check-in matrix and do not solve the problem of data sparsity effectively. However, some researchers focus on the technologies that aims to solve the problem of data sparsity such as matrix factorization [15, 16]. Yildirim [17] use PageRank algorithm to improve cosine similarity method and alleviated data sparsity. Moreover, Ruslan [18] used restricted Boltzmann machine to handle large-scale data that achieved good performance.

In this paper, we propose a deep learning model called RBMNMF based on sparse matrix of user check-ins for POI recommendation. RBMNMF combines the neural network with non-negative matrix factorization and calculates strong correlation values between users and POIs from bidirectional way, which makes the final prediction matrix smoother. Moreover, RBMNMF are able to display users' preferences for POIs intuitively and solve the problem of data sparsity effectively. Moreover, RBMNMF has a higher accuracy when recommending and outperform some single sparse matrix filling algorithms.

III. Hybrid Model Based on Sparse Matrix of User Check-ins

A. *Definations*

This section defines the sparse user check-in matrix, elaborates on issues of our research and presents the framework of the model we propose. User historical check-ins based on LBSNs includes user ID, location ID, longitude and latitude, check-in time and so on. For simplicity, table 1 shows the meanings of all the symbols in this paper.

TABLE I. The Meanings of All Symbols

Symbol	**Meaning**
u	user ID
l	location ID
$s_{u,l}$	User score of location
$s_{l,u}$	Score of correlation between user and location
x_u	User vector
x_l	Location vector
U	Set of all users
L	Set of all locations
$M_{u,l}$	Sparse user-location matrix based on user check-ins
$M_{l,u}$	Sparse location-user matrix based on user check-ins

Definition 1: (POI). If user u has a check-in at a location l, then location l is regarded as a point-of-interest (POI). For example, l_j and l_k are two different POIs that user u has visited .

Definition 2: (Sparse User Check-in Matrix). From the LBSNs data set, construct the check-in matrix $M_{u,l}$ and s_{u_i,l_j} denotes the number of check-ins of user u_i at the location l_j, where $u_i \in U$, $l_j \in L$.

Definition 3: (Sparse User Check-in One-Zero Matrix). If the values of elements in check-in matrix $M_{u,l}$ are greater than 0, then set those values to 1, otherwise 0. Then we obtain the sparse one-zero matrix ${}^{0}_{1}M_{u,l}$.

Definition 4: (Location-User Correlation One-Zero Matrix). Transpose the sparse user check-in one-zero matrix ${}^{0}_{1}M_{u,l}$ to location-user correlation one-zero matrix ${}^{0}_{1}M_{l,u}$. Where ${}^{0}_{1}M_{l,u} = ({}^{0}_{1}M_{u,l})^T$ and $s_{l,u} \in {}^{0}_{1}M_{l,u}$ which is the value in the matrix ${}^{0}_{1}M_{l,u}$. s_{l_j,u_i} denotes the score of the correlation for a location l_j on user u_i.

B. Model framework

In order to solve the sparsity problem of user check-in data, this paper proposes a hybrid model called RBMNMF that combines deep learning models with non-negative matrix factorization. The hybrid model considers both user-location and location-user bidirectional information. The model framework is shown in Fig.1.

Sparse user check-in matrix is inputted into stacking RBMs, NMF model and single RBM respectively, and then fuse result of each part to produce a recommendation list after each part calculates its own prediction matrix. RBMNMF model is composed of three parts mentioned above and each part is independent. Moreover, RBMNMF can effectively solve the problem of data sparsity.

C. Restricted Boltzmann Machine Based on Sparse Matrix

Restricted Boltzmann machine (RBM) is an undirected graph probability model with one layer of visible variables and one layer of latent variables, which is shown in Fig.2.

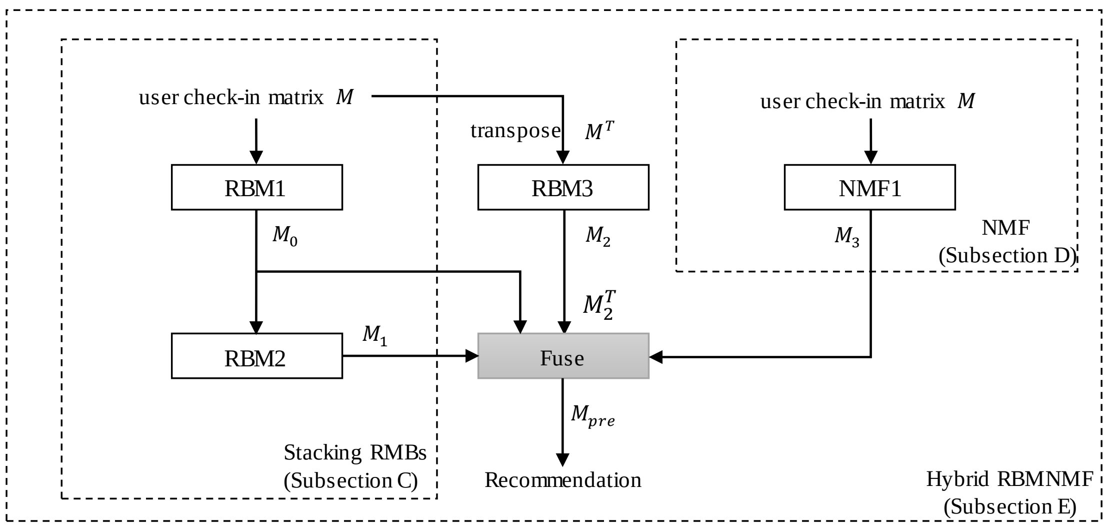

Fig.1 The main architecture of our proposed mode

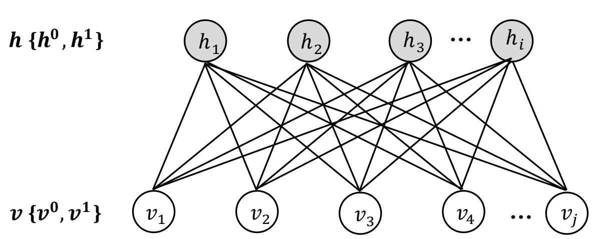

Fig.2 Structure of RBM

RBM model has been widely used in binary data distribution. RBM is defined as binary random vector $\boldsymbol{v} \in \{0,1\}^d$ that is an energy-based model. The energy function is as (1), where $\boldsymbol{a}$ and $\boldsymbol{b}$ are offset vectors of hidden layer and visible layer respectively. Other symbols will be explained latter.

$$E(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{v}^T\boldsymbol{W}\boldsymbol{h} - \boldsymbol{b}^T\boldsymbol{v} - \boldsymbol{a}^T\boldsymbol{h} = -\sum\sum w_{ij}v_i h_j - \sum b_i v_i - \sum a_j h_j \tag{1}$$

The specific training process of our RBM model is as follows, where $\boldsymbol{v}^0$ and $\boldsymbol{v}^1$ are visible-layer vectors ($\boldsymbol{v}^0$ is also a binary check-in vector at the beginning), $\boldsymbol{h}^0$ and $\boldsymbol{h}^1$ are hidden-layer vectors, $\boldsymbol{W}$ denotes the parametric matrix and $\boldsymbol{\beta}$ is learning rate.

Calculate $\boldsymbol{v}^0$:

Input a user vector $\boldsymbol{x}$ and the visible-layer $\boldsymbol{v}^0 = \boldsymbol{x}$

Calculate $\boldsymbol{h}^0$:

Calculating the Opening Probability of Hidden-Layer Neurons:

$$P\left(\boldsymbol{h}_j^0 = 1 \mid \boldsymbol{v}^0\right) = \sigma\left(\sum_i w_{ij}\boldsymbol{v}_i^0 + a_j\right) \tag{2}$$

The hidden-layer vector $\boldsymbol{h}^0$ is obtained by random values filtered from 0 to 1

Calculate $\boldsymbol{v}^1$:

$\boldsymbol{h}^0$ reconstructs visible layer:

$$P\left(\boldsymbol{v}_i^1 = 1 \mid \boldsymbol{h}^0\right) = \sigma\left(\sum_i w_{ij}\boldsymbol{h}_j^0 + b_i\right) \tag{3}$$

The visible-layer vector $\boldsymbol{v}^1$ is obtained by random values filtered from 0 to 1

Calculate $\boldsymbol{h}^1$:

$\boldsymbol{v}^1$ reconstructs hidden layer:

$$P\left(\boldsymbol{h}_j^0 = 1 \mid \boldsymbol{v}^1\right) = \sigma\left(\sum_i w_{ij}\boldsymbol{v}_i^1 + a_j\right) \tag{4}$$

The hidden vector $\boldsymbol{h}^1$ is obtained by random values filtered from 0 to 1

Update Parametric Matrix:

$$\boldsymbol{W} + \boldsymbol{\beta}\left\{\left(\boldsymbol{h}^0\right)^T \times \boldsymbol{v}^0 - \left(\boldsymbol{h}^1\right)^T \times \boldsymbol{v}^1\right\} \to \boldsymbol{W} \tag{5}$$

$$\boldsymbol{b} + \boldsymbol{\beta}\left(\boldsymbol{v}^0 - \boldsymbol{v}^1\right) \to \boldsymbol{b} \tag{6}$$

$$\boldsymbol{a} + \boldsymbol{\beta}\left(\boldsymbol{h}^0 - \boldsymbol{h}^1\right) \to \boldsymbol{a} \tag{7}$$

In order to prevent over-fitting and make our model smoother, data transformation and stacking RBMs are used to train data set, which is shown in Fig.3.

Firstly, input ${}_1^0M_{u,l}$ that is the original user check-in matrix to stacking restricted Boltzmann machines model for sparse data training and filling, and then predict the location score M_0 and M_1 for users. The combination of two RBMs will enhance the matrix computation and make prediction more reasonable, which is demonstrated from our experiments.

Secondly, input ${}_1^0M_{l,u}$ to a single RBM and then predict the user score M_2 for locations. ${}_1^0M_{l,u}$ is the transposition of ${}_1^0M_{u,l}$. The reason why we take this measure is that most existing works only consider of users' preference for POIs and ignore the POIs attraction to users. Thus, we decide to take transposition into account. The final predictive matrix M'_{pre} of restricted Boltzmann machine model based on sparse matrix is as follows.

$$M'_{pre} = M_0 + M_1 + (M_2)^T \tag{8}$$

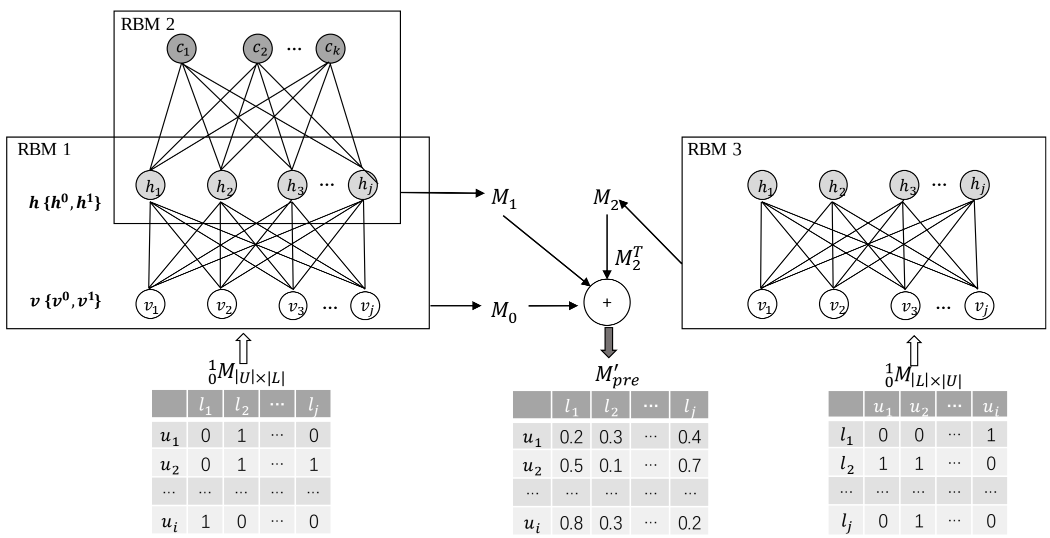

Fig.3 The workflow of stacking RBMs

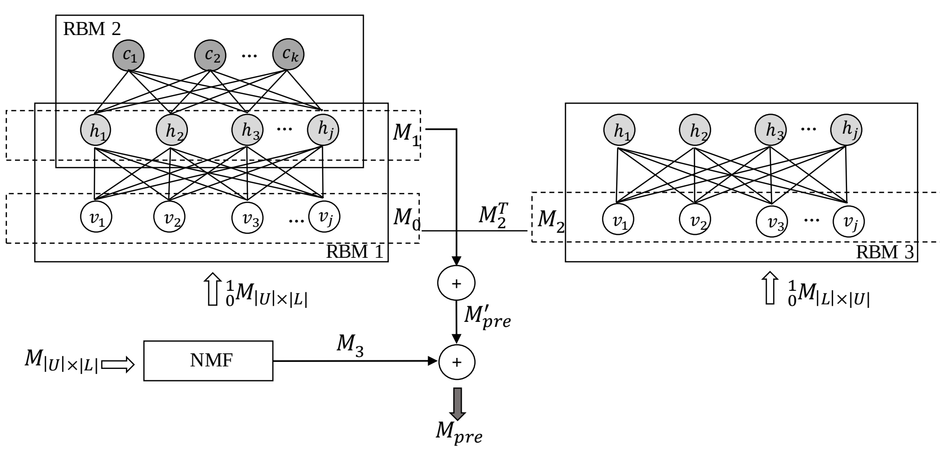

Fig.4 Combination of stacking RBMs and NMF

D. Non-Negative Matrix Factorization Model Based on Sparse Matrix

Values of elements in sparse user check-in matrix are all non-negative. However, the traditional matrix factorization model will get negative values from original matrix, which have no practical significance for user check-in matrix.

Non-negative matrix factorization (NMF) can factorize the user check-in matrix into two non-negative matrices and fill the sparse matrix with positive values. Therefore, it solves the sparse problem of matrix effectively and it’s easy to observe the user's preference for locations. NMF is shown in Fig.4. $M_{u,n}$ and $M_{n,l}$ is the result of factorization of $M_{u,l}$. Thereafter, M_3 is the multiplication of them, which is filled with more negative values.

	k_1	k_2	⋯	k_n
u_1	0	0.8	⋯	0
u_2	2.0	0	⋯	0
⋯	⋯	⋯	⋯	⋯
u_i	0	0	⋯	2.5

$M_{|U|\times|N|}$

×

	l_1	l_2	⋯	l_j
k_1	0	0.5	⋯	1.9
k_2	0	2.3	⋯	0
⋯	⋯	⋯	⋯	⋯
k_n	1.2	1	⋯	0

$M_{|N|\times|L|}$

=

	l_1	l_2	⋯	l_j
u_1	0	1.8	⋯	0
u_2	0	1	⋯	3.8
⋯	⋯	⋯	⋯	⋯
u_i	3	$\boxed{2.5}$	⋯	0

M_3

Fig.5 The result of factorization of NMF

In $M_{u,l}$, the value of each element represents the times that the user has visited the location. As shown above, using NMF model, predictive matrix M_3 fills the elements of $M_{u,l}$. Each element’s value of M_3 denotes the preference of a user for a location.

E. Hybrid RBMNMF Model

In order to make our model smoother, we propose a hybrid model called RBMNMF that fuses the models mentioned in the above sections, as shown in figure 4.

The whole process of RBMNMF is as follows.

1) Input ${}_1^0 M_{u,l}$ into two stacking RBMs model for sparse data training and filling. Then predict location score matrices M_0 and M_1.

2) Input ${}_1^0 M_{l,u}$ into a single RBM and predict the location-user correlation score matrix M_2. Then transpose the M_2.

3) Factorize $M_{u,l}$ into two non-negative matrices to obtain another location score matrix M_3 for users.

4) Predict the final recommendation matrix as follows.

$$M_{pre} = \left(M_0 + M_1 + (M_2)^T\right) + M_3 \tag{9}$$

5) Recommend Top-N POIs to users according to M_{pre}.

IV. Experiments

A. Datasets

Foursquare is a social networking site that records a large number of geographic information of users' current locations. This paper uses the data set provided by Quan et al. [1] which contains 342850 check-ins from August 2010 to July 2011 in Singapore. Yuan [1] deletes users whose check-ins are less than 5 and locations checked by less than 5 users. After processing, the foursquare data set contains 2321 users, 5596 POIs and 194108 check-ins. For each user, choose 12.5% of the user’s POIs for tuning parameters and another 25% of the user’s POIs for testing data randomly. The left POIs are used for training. The formats of each check-in is user ID, location ID, longitude and latitude, check-in time and time ID respectively.

B. Evaluation Metric

We use $Precision@N$ and $Recall@N$ to evaluate our model, which are as follows.

$$precision@N = \frac{|R(u) \cap T(u)|}{R(u)} \tag{10}$$

$$Recall@N = \frac{|R(u) \cap T(u)|}{T(u)} \tag{11}$$

$R(u)$ denotes the list of POIs recommended to user u and $T(u)$ denotes the list of POIs that are actually checked in by user u in the test data. $|R(u)|$ is the total number of POIs in $R(u)$ and $|T(u)|$ is the total number of POIs in $T(u)$. The final precision and recall are the average of all users.

C. Experimental Result

In order to verify the recommendation effectiveness of our RBMNMF model, we compare RBMNMF with single sparse matrix filling algorithms in the same training data and test data. Those algorithms are as table 2 shows. The core of this paper is to integrate non-negative matrix factorization and stacking RBMs. So we focus on comparing our proposed algorithm with single NMF or single RBM but CF is a popular algorithm that we should also consider.

TABLE II. Comparison Algorithms

Algorithm	Description
CF	Collaborative filtering (CF) is the most popular and classical recommendation algorithm
RBM	Single restricted Boltzmann machine based on sparse matrix of user check-ins
NMF	Single non-negative matrix factorization based on sparse matrix of user check-ins
RBMNMF	The hybrid model we propose

The experimental results are shown in Fig.6 and Fig.7. We recommend N (N = 5, 10, 20, 30) POIs for each user. The performance of CF is the worst among all algorithms and possibly due to the data set, the precision of CF does not change significantly. No matter what the value of N is, the precision and recall of RBMNMF are generally better than other single algorithms. Hence, even though there is a lack of user information, RBMNMF we propose are able to fill the sparse matrix well and have an excellent performance when recommending. The results of experiments demonstrate RBMNMF could be used into improving traditional algorithms.

However, there is a limitation in RBMNMF. When stacking RBMs process check-in data, we adopt the behavior of check-in to represent the user's preference for a location by 0 or 1. That treats all locations equally that a user has visited even though they have different numbers of check-ins.

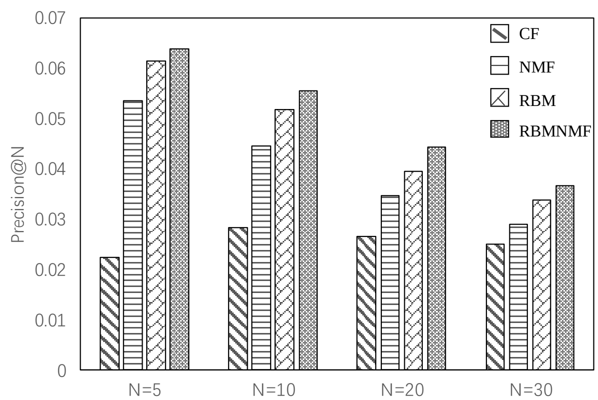

Figure 6. Precision with different N

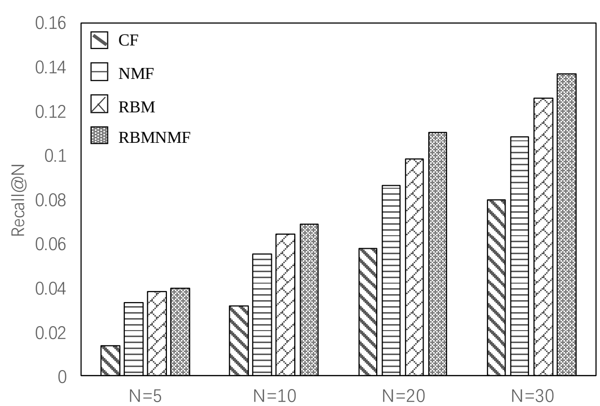

Figure 7. Recall with different N

V. CONCLUSION

For the sake of overcoming the sparsity problem of user check-in matrix, we propose a deep learning model called RBMNMF that combines stacking restricted Boltzmann machines with non-negative matrix factorization to make the prediction model smoother. RBMNMF fills the user check-in matrix and alleviates the problem of data sparsity effectively. Therefore, we could observe each user's preference value for each POI intuitively and then recommend satisfying POIs to users. In the future, more information will be taken into account such as time, geographical effect and location popularity to improve our model.

ACKNOWLEDGMENT

This research is supported by the National Natural Science Foundation of China (Grant No. 61502062, Grant No. 61672117 and Grant No. 61602070).

REFERENCES

[1] Yuan, Quan , et al. "Time-aware point-of-interest recommendation." International Acm Sigir Conference on Research & Development in Information Retrieval ACM, pages 363-372, 2013.

[2] Yu, Zheng, et al. "Recommending friends and locations based on individual location history." Acm Transactions on the Web, pages 1-44, 2011.

[3] Yu, Zheng, and X. Xing. "Learning Location Correlation from GPS Trajectories." Eleventh International Conference on Mobile Data Management, pages 27-32,2010.

[4] Zheng, Yu , and X. Xie . "Learning travel recommendations from user-generated GPS traces." ACM Transactions on Intelligent Systems and Technology, pages 1-29, 2011.

[5] Ye, Mao , et al. "Exploiting geographical influence for collaborative point-of-interest recommendation." Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 325-334, 2011

[6] Baral, Ramesh , and T. Li . "MAPS: A Multi Aspect Personalized POI Recommender System." Acm Conference on Recommender Systems ACM, pages 281-284, 2016.

[7] Jiang, Shuhui , et al. "Author topic model-based collaborative filtering for personalized POI recommendations." IEEE Transactions on Multimedia, pages 907-918, 2015

[8] Yang, Carl , et al. "Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation." Acm Sigkdd International Conference on Knowledge Discovery & Data Mining ACM, pages 1245-1254, 2017.

[9] Liu, Yaqiong , and H. S. Seah . "Points of interest recommendation from GPS trajectories." International Journal of Geographical Information Systems, pages 953-979, 2015.

[10] Zheng, Kai , et al. "Reducing Uncertainty of Low-Sampling-Rate Trajectories." 2012 IEEE 28th International Conference on Data Engineering IEEE Computer Society, pages 1144-1155, 2012.

[11] Bao, Jie , Y. Zheng , and M. F. Mokbel . "Location-based and preference-aware recommendation using sparse geo-social networking data." Proceedings of the 20th International Conference on Advances in Geographic Information Systems, pages 199-208, 2012.

[12] Ference, Gregory , M. Ye , and W. C. Lee . "Location recommendation for out-of-town users in location-based social networks." Proceedings of the 22nd ACM international conference on Conference on information & knowledge management ACM, pages 721-726, 2013.

[13] Wang, Hao , M. Terrovitis , and N. Mamoulis ."Location recommendation in location-based social networks using user check-in data." Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pages 364-373, 2013.

[14] Ying, Jia Ching , et al. "Mining User Check-In Behavior with a Random Walk for Urban Point-of-Interest Recommendations." ACM Transactions on Intelligent Systems and Technology, pages 1-26, 2014.

[15] Ma, Hao , et al. "SoRec: Social recommendation using probabilistic matrix factorization." ACM, pages 931-940, 2008.

[16] Lian, Defu , et al. "GeoMF : Joint Geographical Modeling and Matrix Factorization for Point-of-Interest Recommendation." Acm Sigkdd International Conference on Knowledge Discovery & Data Mining ACM, pages 831-840, 2014.

[17] Yildirim, Hilmi , and M. S. Krishnamoorthy . "A random walk method for alleviating the sparsity problem in collaborative filtering." Acm Conference on Recommender Systems ACM, pages 131-138, 2008.

[18] Salakhutdinov, Ruslan , A. Mnih , and G. Hinton . " [ACM Press the 24th international conference - Corvalis, Oregon (2007.06.20-2007.06.24)] Proceedings of the 24th international conference on Machine learning - ICML \"07 - Restricted Boltzmann machines for collaborative filtering." (2007):791-798.