2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)

Attention-based Recurrent Neural Network for Location Recommendation

Bin Xia[1], Yun Li[2], Qianmu Li[1], Tao Li[2]
[1]*School of Computer Science and Engineer, Nanjing University of Science and Technology, Nanjing, China*
[2]*School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, Chain*
liqianmu@126.com

***Abstract*—Due to the rapid development of Location-Based Social Networks (LBSNs), the Point of Interest (POI) recommendation has been attracted a lot of research attention. Based on the LBSNs, users are able to share their relevant visiting experience via check-in records. The sequential check-in data not only explicitly show users' moving trajectories, but also implicitly describe personal preferences and corresponding life patterns based on different contexts (e.g., time and geographical locations). The traditional POI recommender systems only consider common contexts (e.g., visit frequency, distance, and social relationship), but ignore the significance of life patterns for individuals during different time periods. In addition, current recommender systems hardly provide interpretable and explainable recommendations based on these limited contexts.**

In this paper, we propose an Attention-based Recurrent Neural Network (ARNN) to provide an explainable recommendation based on the sequential check-in data of the corresponding user. Our proposed approach makes use of the sequential check-in data to capture users' life pattern and utilizes a deep neural network to provide transparent recommendations. The major contribution of this paper are: (1) the proposed model is capable of providing explainable recommendations based on life patterns which implicitly describes the preference of the corresponding user; (2) the proposed approach is able to design a visiting plan (i.e., a series of recommendations) based on users' past visiting patterns instead of simply showing top-N recommendations; (3) we evaluate our proposed approach against a real world dataset and compare it to other start-of-the-art approaches.

***Keywords*-LBSNs; POI; life pattern; attention-based; RNN; series of recommendations;**

I. Introduction

With the rapid popularization of the smart mobile phone and relevant wearable devices, people are able to easily share their check-in information in LBSNs. The information exploration of visiting records in LBSNs brings a big challenge for the recommender system to provide meaningful recommendations. Hence, the POI recommendation has become an attractive research topic and many researchers are taking a lot of efforts on improving the performance of the POI recommender system considering check-in records. With the visiting records, we can easily capture the popular area (i.e., geographical locations) and the category of frequent check-ins (e.g., restaurant, shopping mall, and bar) of the corresponding user. Based on the analysis of historical check-in data, it is effective to provide individuals with personalized recommendations.

Despite the check-in data is abundant and useful for the POI recommender system, it is unable to explicitly describe the preference of the corresponding user. For instance, a user visited SUBWAY (i.e., an American fast food restaurant) for many times but went to Smith & Wollensky (i.e., an America's luxurious steakhouse) only once. It is not sufficient to infer that she prefers SUBWAY to Smith & Wollensky based merely on the statistics of check-in records. Therefore, many context-aware researches [1], [2], [3], [4], [5], [6] and other related methods [7], [8], [9] have been recently proposed to overcome the limitation of check-in data and effectively improve the performance of the original recommender system which is only based on historical visiting data.

The various contexts (e.g., weather, review, and social relationship) bring a lot of extra useful information to infer users' preferences. However, it is not trivial to collect such complex and heterogeneous contexts. For instance, suppose we are given the check-in data of a user from Foursquare and would like to provide her with appropriate recommendation considering her social relationship. It is necessary to collect the information of her friends from Twitter since the entire social relationship are probably not available in Foursquare. It is also non-trivial to integrate these context information from different data sources. In addition, sometimes these information is not available. Therefore, the missing context information limits the development of context-based research studies.

To overcome the aforementioned limitation of the missing context, in this paper, we suppose that each user has her/his own visiting pattern (e.g., watching a film after dinner or doing morning exercise in a gym) and propose an attention-based recurrent neural network to capture users' life patterns based on their check-in data. The traditional top-N recommender system provides a ranking list of locations that users are probably interested in based on the entire historical visits. Different from that, our proposed model provides a visiting plan (i.e., sequential recommendations) for the corresponding user. Different from the top-N recommendations, the visiting plan is generated by iteratively running the proposed RNN based on the comparison between users' latest check-ins and life patterns. In this paper, our proposed model only utilizes the basic information of check-in data such as the visiting time and the geographical location and aims to provide the recommendations considering the category of locations and the active area.

The rest of this paper is organized as follows: Section II describes the architecture of our proposed recommender system and the attention-based recurrent neural network in details; We evaluate the proposed approach in a real world dataset and analyze the experimental results in Section III; Finally, Section IV summarizes our work and introduces the future work.

II. Methodology

Currently, most of POI recommendation strategies are making use of the various context information which is not easily available from an independent LBSN. To effectively address the limitation of context-based recommendation approaches and simultaneously provide explainable recommendations, in this paper, we propose an attention-based recurrent neural network to improve the performance of the recommender system using the basic information of check-in data (i.e., the time and the geographical location). Different from previous recommender systems, our approach aims to provide a visiting plan, which shows several check-in recommendations in temporal order, instead of top-N recommendations. Compared with the traditional recommendation methods, the visiting plan not only provides some recommendations based on users' preference but also shows these recommendations in a storyline as an explainable schedule considering the contexts (e.g., the geographical location and the life pattern). In this section, we introduce the architecture of our recommender system and describe the details of the proposed neural network.

A. System Architecture

Our proposed approach is influenced by the common question and answering model [10], [11]. In this paper, the proposed approach uses an attention-based recurrent neural network, which is a sort of end-to-end memory network similar in [12], to capture the correlation between users' latest check-in and the historical visiting records. Hence, the input of our proposed model consists of two parts: (1) the historical visiting records and (2) the latest check-in, and the output of our proposed model is a list of top-N recommendation. If the model is allowed to output several recommendations once, our proposed approach is equivalent to other traditional recommender systems which provide top-N recommendations. If the model is allowed to output only one location (i.e., top-1 recommendation) each iteration, our recommender system will iteratively run the model and record the corresponding output in order as the visiting plan. For each round of the iteration, the list of historical visiting records will append to the current input of latest check-in and the output will be considered as the input of latest check-in for the next iteration. Figure 1 presents the overview of proposed system architecture.

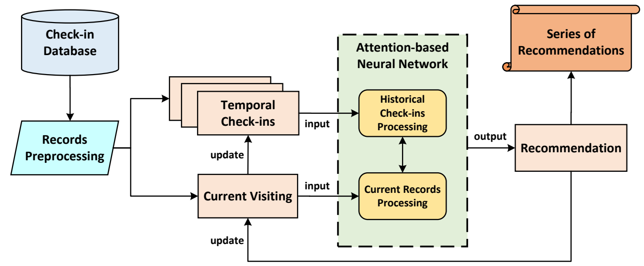
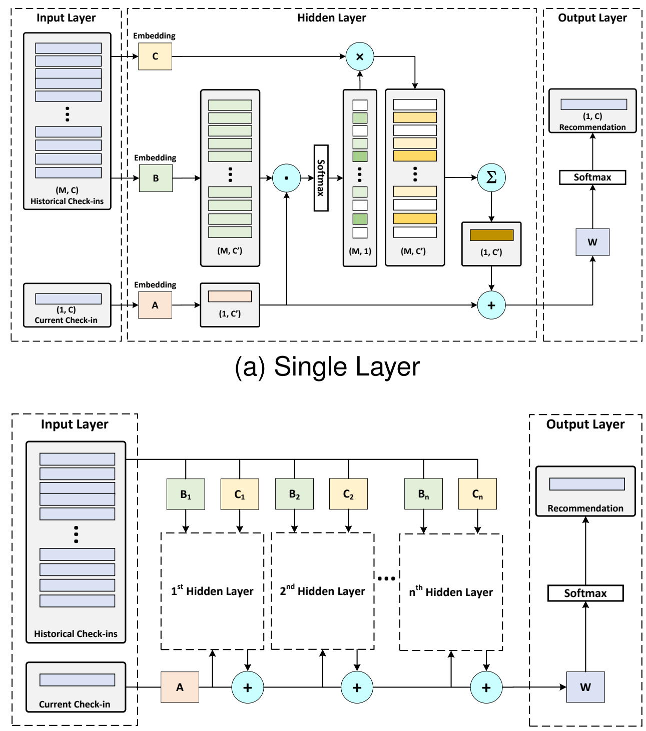


Figure 1: The overview of system architecture



(a) Single Layer

(b) Multiple Layers

Figure 2: The single and multiple version of our deep neural network

B. Attention-based Recurrent Neural Network

The primary goal of the proposed attention-based recurrent neural network is to capture users' preferences based on corresponding life patterns which are described by their check-in records. Our approach is capable of focusing on the parts of user's historical check-ins which are similar with her latest visiting records, and then provides appropriate recommendations based on those similar parts which are considered as the life pattern. In this section, we introduce the attention-based recurrent neural network in two parts: (1) single layer and (2) multiple layers.

1) Single Layer: Figure 2a shows the single layer version of the attention-based neural network. In this section, we introduce the mechanism of our proposed approach which has only one hidden layer.

Input Layer: In order to compare user's current check-in with her visiting pattern in the past, the input layer consists of two parts: (1) the historical visiting records and (2) the latest check-in. Suppose we are given a set $\mathbf{V} = \{v_1, v_2, ..., v_s\}$ ordered by time where $\{v_i\}$ means the user's i^{th} check-in record. The entire set is divided into lots of memory buckets $\mathbf{B} = \{b_1, b_2, ..., b_n\}$ utilizing overlapping subsample. For $\forall b_i \in \mathbf{B}$, there are $M + 1$ consecutive check-ins where the former M records are considered as the historical visits $\mathbf{M} = \{m_1, m_2, ..., m_q | q < s\}$ and the latest one $\mathbf{R} = \{r_1\}$ is regarded as the current check-in.

For each check-in record, the representation of the location consists of three parts: (1) *category*, (2) *id*, and (3) *geographical location*. In this model, One-Hot Encoding is utilized to represent the *category* and the *id* of a location. However, the dimension of location id is huge and will tremendously increase the computational consumption when training a neural network model. Hence, we add the *category*, which not only provides the functional information of a location but also enormously decrease the dimension of the *id* representation, and consider it as a supplementary feature. Then, we can group the locations based on the corresponding categories and the dimension of each group will be less than that of entire locations. For the *geographical location*, the traditional method applies the matrix factorization against the similarity matrix, which describes the correlation of geographical distance between each pair of locations, to describe the geographical information. However, adding or removing a new location will take expensive computational consumption update the representations of related neighborhoods [13]. To overcome the limitation, we use the discrete spatial grid distribution to represent the *geographical location*. Suppose we are given a set of entire locations $\mathbf{L} = \{l_1, l_2, ..., l_o\}$ and a set of spatial grids $\mathbf{G} = \{g_1, g_2, ..., g_p\}$ which is segmented in an equivalent area. We assume that if the location l_i is close to the center of grid g_j, then the location l_i is more likely to influence the choice of users who often check-in in grid g_j and the grids nearby. Based on the assumption that the spatial influence follows the normal distribution, the probability of this influence can be described as:

$$P(g_j|l_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{d_{ij}^2}{2\sigma^2}} S, \tag{1}$$

where d_{ij} presents the distance from the location l_i to the center of grid g_j and S means the area of each grid. The location l_i can be described using a vector $\{P(g_1|l_i), P(g_2|l_i), \ldots, P(g_p|l_i)\}$. After feature extraction, for the input layer, each memory bucket has: (1) an $M \times C$ matrix describes the historical visiting pattern and (2) a $1 \times C$ matrix presents the current check-in record, where C means the dimension of extracted feature.

Hidden Layer: The first thing in the hidden layer is to convert the input matrix into the latent feature representation and decrease the feature dimension by embedding each record in a continuous space. For the current check-in record matrix $\mathbf{R}$, an embedding matrix $\mathbf{A}$ is used to convert the original input into a matrix $\mathbf{R_A}$ with shape $(1, C')$. Because the sequential visiting records are quite important contexts for capturing users' visiting patterns, after embedding, we add a term of constraint and modify the embedded matrix $\mathbf{R_A}$ as $r_1' = \mathbf{A}r_1 + \mathbf{T_A}(1)$, where $\mathbf{T_A}$ is a $(1, C')$ matrix encoding the temporal information and $\mathbf{T_A}(1)$ means the first row in $\mathbf{T_A}$. Similar to the embedding of the current check-in record matrix $\mathbf{R}$, the historical visiting pattern $\mathbf{M}$ is converted into an (M, C') matrix $\mathbf{M_B}$ utilizing an embedding matrix $\mathbf{B}$. A matrix $\mathbf{T_B}$ is also used to encode the temporal information of the embedded matrix $\mathbf{M_B}$, where $m_i' = \sum_j \mathbf{B}m_{ij} + \mathbf{T_B}(i)$. In addition, $\mathbf{M}$ is embedded into the matrix $\mathbf{M_C}$ for the later calculation based on $m_i'' = \sum_j \mathbf{C}m_{ij} + \mathbf{T_C}(i)$ using the embedding matrix $\mathbf{C}$.

To capture the correlation between the current check-in record and the historical visiting pattern, in the embedding space, we compare the embedded matrix $\mathbf{R_A}$ with each row of the embedded matrix $\mathbf{M_B}$ using the inner product. In order to increase the influence of the records with strong correlations and suppress those weak correlations, a softmax function (i.e., $Softmax(x_i) = e^{x_i} / \sum_{k=1}^{K} e^{x_k}$, for $i = 1, 2, ..., K$) is utilized following the inner product:

$$\mathbf{S} = Softmax(\mathbf{R_A}^T \mathbf{M_B}). \tag{2}$$

Hence, an $(M, 1)$ weight matrix $\mathbf{S}$ is available to present the influence of each check-in record in historical visits. Then, we sum up the product of each row between the weight matrix $\mathbf{S}$ and the matrix $\mathbf{M_C}$ using:

$$\mathbf{O} = \sum_i \mathbf{S}(i) \mathbf{M_C}(i), \tag{3}$$

where $\mathbf{S}(i)$ and $\mathbf{M_C}(i)$ mean the i^{th} row in the corresponding matrix. In order to output the result of current layer and keep the information of previous layer (i.e., input layer), the last thing in the hidden layer is to sum the matrix $\mathbf{O}$ with the matrix $\mathbf{R_A}$ and deliver the result to the next layer (i.e., output layer).

Output Layer: Now, the recommendation from the previous hidden layer is available, however, it is still in the embedding space. In order to provide the final predicted recommendation, we pass the sum of $\mathbf{O}$ and $\mathbf{R_A}$ though a weight matrix $\mathbf{W}$ followed by a softmax function:

$$\hat{l} = Softmax((\mathbf{O} + \mathbf{R_A})\mathbf{W}), \tag{4}$$

where the shape of $\mathbf{W}$ is (C', C). The proposed model applies the stochastic gradient decent to jointly learn the parameters $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{T_A}$, $\mathbf{T_B}$, $\mathbf{T_C}$, and $\mathbf{W}$ by minimizing the cross entropy loss between the predicted recommendation $\hat{l}$ and the ground truth l.

2) Multiple Layers: Based on the description of the single layer version, the proposed model can be easily extended to a multiple layers version. Figure 2b illustrates the multiple-layer version of our approach. The hidden layers are connected by the following methods:

1) Suppose the inputs of the i^{th} hidden layer are $\mathbf{R_A}^i$, $\mathbf{M_B}^i$, $\mathbf{M_C}^i$ and the output is $\mathbf{O}^{i+1}$. Then the input of the $(i+1)^{th}$ hidden layer are:

$$\begin{aligned}\mathbf{R_A}^{i+1} &= \mathbf{R_A}^i + \mathbf{O}^i,\\ \mathbf{M_B}^{i+1} &= embedding(M, B^{i+1}) + T_B^{i+1},\\ \mathbf{M_C}^{i+1} &= embedding(M, C^{i+1}) + T_C^{i+1},\end{aligned} \tag{5}$$

where $embedding()$ is the embedding function.

2) For the n^{th} hidden layer, the output $\mathbf{O}^{n+1}$ is passes through the weight matrix $\mathbf{W}$ followed by a softmax function in the output layer. Finally, the predicted recommendation $\hat{l} = Softmax((\mathbf{O}^n + \mathbf{R_A}^n)\mathbf{W})$ is available.

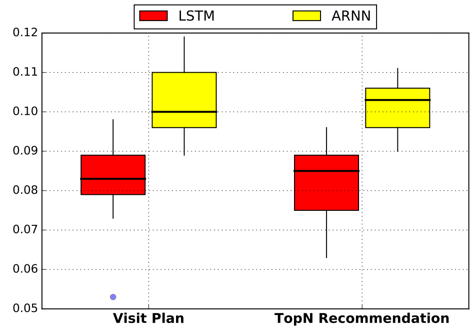
In this paper, we apply the traditional strategy of RNN model where the same embedding matrices are utilized in each hidden layer (i.e., $\mathbf{B}^1 = \mathbf{B}^2 = ... = \mathbf{B}^n$, $\mathbf{C}^1 = \mathbf{C}^2 = ... = \mathbf{C}^n$, $\mathbf{T_B}^1 = \mathbf{T_B}^2 = ... = \mathbf{T_B}^n$, and $\mathbf{T_C}^1 = \mathbf{T_C}^2 = ... = \mathbf{T_C}^n$). In addition, the output of each hidden layer is divided into internal and external parts, where the internal output contains the information in the corresponding hidden layer and the external part accumulating the information of current and previous layers is delivered to the next hidden layer or the output layer.

III. Evaluation

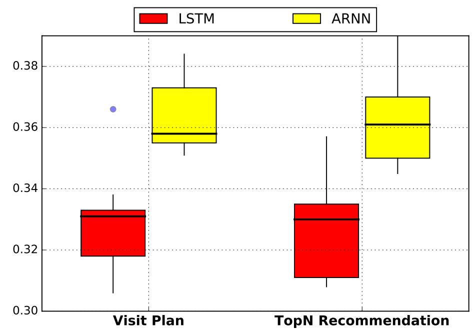
In this paper, we evaluate the proposed attention-based neural network using a real world LBSN dataset. We collect 419,509 check-in records published by 49,823 users among 18,899 locations from August 2012 to July 2013 in Manhattan via the API of Foursquare [14], [15]. In the dataset, there are 378 categories (e.g., Office, Hotel, and Food Truck) where the category 'Office' has the largest number of location (i.e., 2,145). In this section, we compare our proposed approach with the Long Short-Term Memory network (LSTM) to evaluate the performance of the visiting plan recommendation, and also show the performance of ARNN and other baseline methods in the TopN recommendation task.

A. Visiting Plan VS TopN Recommendation

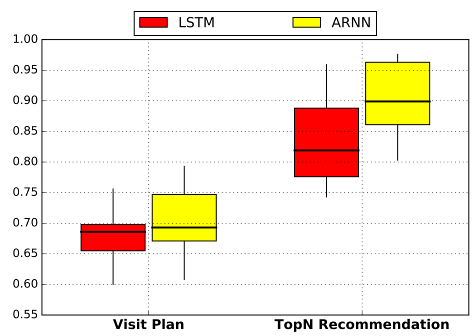
The recommendation of visiting plan is the major contribution of this paper, which is a new type of recommendation. Different from the traditional TopN recommendation, the visiting plan is capable of providing a temporal series of POIs, where each POI is dependent on the previous one in the recommendation list. To evaluate the effectiveness of the visiting plan in the recommendation task, we compare ARNN with LSTM in both the visiting plan and TopN



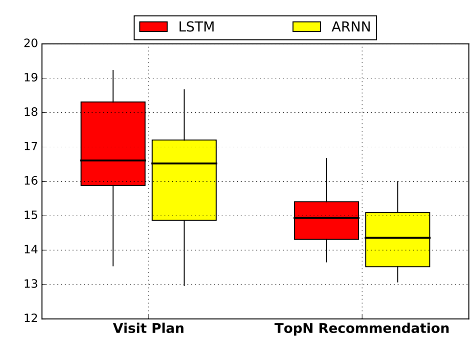
(a) Precision@20



(b) Recall@20



(c) Coverage_@20



(d) Coverage_Gini@20

Figure 3: The comparison between the visiting plan and TopN recommendation

recommendation. In order to show the performance of the proposed approach during different time periods, we segment the check-in records based on the temporal information (i.e., month) [15]. For each segment, we consider the current segment as the test dataset and apply the previous segments to train the models. Figure 3 illustrates the performance of ARNN and LSTM in the TopN recommendation and the visiting plan respectively.

As shown in Figure 3, ARNN and LSTM have the similar performance of recall and precision in the task of the visiting plan and the TopN recommendation, where the recommendation of visiting plan outperforms the TopN recommendation in some time periods. The experimental result shows that the recommendation of visiting plan is effective in the recommendation, however, the TopN recommendation outperforms the visiting plan in the coverage of recommendation. In fact, some recommendations (i.e., POI) will be repeatedly provided in the list, since each POI in the visiting plan is the most appropriate selection (i.e., Top1) of a TopN recommendation and each TopN recommendation may have the intersection of POIs. Therefore, the recommendation mechanism of visiting plan will affect the performance in the recommendation coverage. However, the goal of visiting plan recommendation is to convert the potential demands of users to the actual check-in records. For instance, in our Foursquare check-in dataset, a user often visited bars at night and sometimes enjoyed the dinner in the restaurant near the bar. Based on the recommendation of visiting plan, ARNN will provide a POI in the category of ‘Food’ before the bar in the recommendation list. According to this kind of recommendation, the user is more likely to accept the visiting plan and the frequency of visiting a restaurant before visiting a bar will be increased. In summary, the visiting plan provides users a new kind of recommendation instead of the traditional TopN recommendation.

B. ARNN VS Baselines

In order to evaluate the effectiveness of our proposed approach, in this section, we compare ARNN with other baseline methods which have good performance of providing accurate recommendations in POI recommender system, including UserCF (User-based Collaborative Filtering), BPR (Bayesian Personalized Ranking [16]), NMF (Non-negative Matrix Factorization [17]), ESSVM (Embedded Space ranking SVM [14]), LBIMC (Linearized Bregman Iteration for Matrix Completion [15]), and LSTM (Long Short-Term Memory network [18]). Figure 4 presents the comparison between our proposed approach and other baseline methods.

As shown in Figure 4, ARNN outperforms other baseline approaches except for LBIMC under Top-20 recommendation in recall. LBIMC is a noise-tolerance and matrix factorization-based POI recommendation approach [15]. The experimental result shows that the noise in the check-in records negatively impacts the performance of ARNN

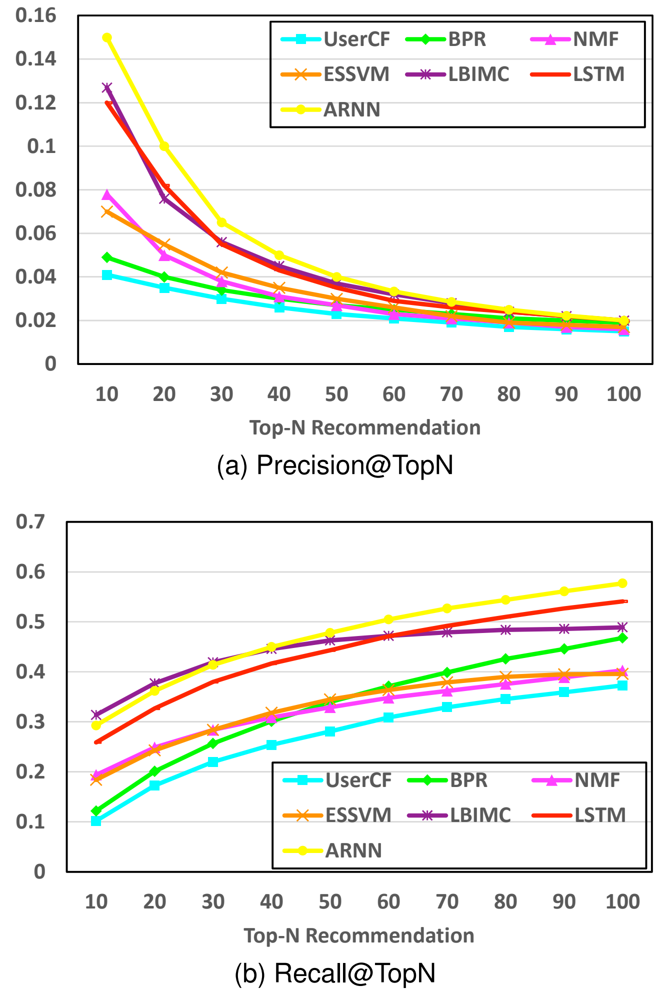


Figure 4: The comparison between ARNN and other baseline methods in TopN recommendation

since the performance of the other matrix factorization-based approach (i.e., NMF) is worse than LBIMC. In addition, our proposed approach outperform ESSVM which is a context-based POI recommendation method. From the experimental result, we can consider that the check-in pattern is capable of representing users’ preferences more appropriate than the contextual information. For instance, a user would like to enjoy the dinner before watching a film, thus she is more likely to choose a restaurant near the cinema where the restaurant may not fit her preference. To this end, the context-based approach can hardly provide an obscure recommendation (i.e., distance priority or preference priority), however, the check-in pattern-based method is capable of capturing this kind of preference if she used to have similar check-in records.

IV. Conclusion

In this paper, we propose an Attention-based Recurrent Neural Network (ARNN) which aims to capture users’ life patterns based on the historical check-in records. By comparing users' latest check-in information to the corresponding historical visiting records, our proposed approach is capable of providing explainable and appropriate visiting plan (i.e., a series of recommendations) based on users' past visiting patterns. There are some limitations in the proposed model, such as the cold start problem since the captured life pattern should be supported by the check-in data. As the future work, we would like to search the similarity between life patterns and providing group recommendations while maintaining personalized recommendations for the corresponding user.

ACKNOWLEDGMENT

This research has been supported by the Chinese National Natural Science Foundation under grant 91646116 the Fundamental Research Funds for the Central Universities (No. 30916015104), the National key research and development program: key projects of international scientific and technological innovation cooperation between governments (No. 2016YFE0108000), CERNET Next Generation Internet Technology Innovation Project (NGII20160122), the Project of ZTE Cooperation Research (2016ZTE04-11), and Jiangsu province key research and development programs: social development project (BE2017739) and industry outlook and common key technology projects (BE2017100), the Scientific and Technological Support Project (Society) of Jiangsu Province No. BE2016776, the Ministry of Education/China Mobile joint research grant under Project No. 5-10, Jiangsu Provincial Natural Science Foundation of China under Grant BK20171447, and Jiangsu Provincial University Natural Science Research of China under Grant 17KJB520024.

REFERENCES

[1] H. Gao, J. Tang, X. Hu, and H. Liu, "Content-aware point of interest recommendation on location-based social networks." in *AAAI*, 2015, pp. 1721–1727.

[2] L. Yao, Q. Z. Sheng, Y. Qin, X. Wang, A. Shemshadi, and Q. He, "Context-aware point-of-interest recommendation using tensor factorization with social regularization," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2015, pp. 1007–1010.

[3] A. Majid, L. Chen, G. Chen, H. T. Mirza, I. Hussain, and J. Woodward, "A context-aware personalized travel recommendation system based on geotagged social media data mining," *International Journal of Geographical Information Science*, vol. 27, no. 4, pp. 662–684, 2013.

[4] B. Liu and H. Xiong, "Point-of-interest recommendation in location based social networks with topic and location awareness," in *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM, 2013, pp. 396–404.

[5] K. H. Lim, "Recommending tours and places-of-interest based on user interests from geo-tagged photos," in *Proceedings of the 2015 ACM SIGMOD on PhD Symposium*. ACM, 2015, pp. 33–38.

[6] R. Baral, D. Wang, T. Li, and S.-C. Chen, "Geotecs: Exploiting geographical, temporal, categorical and social aspects for personalized poi recommendation," in *Information Reuse and Integration (IRI), 2016 IEEE 17th International Conference on*. IEEE, 2016, pp. 94–101.

[7] W. Zhou, T. Li, L. Shwartz, and G. Y. Grabarnik, "Recommending ticket resolution using feature adaptation," in *Network and Service Management (CNSM), 2015 11th International Conference on*. IEEE, 2015, pp. 15–21.

[8] W. Zhou, L. Tang, T. Li, L. Shwartz, and G. Y. Grabarnik, "Resolution recommendation for event tickets in service management," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 287–295.

[9] R. Baral and T. Li, "Maps: A multi aspect personalized poi recommender system," in *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, 2016, pp. 281–284.

[10] J. Weston, S. Chopra, and A. Bordes, "Memory networks," *arXiv preprint arXiv:1410.3916*, 2014.

[11] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[12] S. Sukhbaatar, J. Weston, R. Fergus *et al.*, "End-to-end memory networks," in *Advances in neural information processing systems*, 2015, pp. 2440–2448.

[13] D. Lian, Z. Zhang, Y. Ge, F. Zhang, N. J. Yuan, and X. Xie, "Regularized content-aware tensor factorization meets temporal-aware location recommendation," in *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 1029–1034.

[14] B. Xia, Z. Ni, T. Li, Q. Li, and Q. Zhou, "Vrer: context-based venue recommendation using embedded space ranking svm in location-based social network," *Expert Systems with Applications*, vol. 83, pp. 18–29, 2017.

[15] B. Xia, T. Li, Q. Li, and H. Zhang, "Noise-tolerance matrix completion for location recommendation," *Data Mining and Knowledge Discovery*, pp. 1–24, 2017.

[16] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," in *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 2009, pp. 452–461.

[17] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[18] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, 2016.