Recommending Points of Interest in LBSNs Using Deep Learning Techniques

Giannis Christoforidis
Aristotle University
54124 Thessaloniki, Greece
icchrist@csd.auth.gr

Pavlos Kefalas
Aristotle University
54124 Thessaloniki, Greece
kefalasp@csd.auth.gr

Apostolos N. Papadopoulos
Aristotle University
54124 Thessaloniki, Greece
papadopo@csd.auth.gr

Yannis Manolopoulos
Open University of Cyprus
2220 Nicosia, Cyprus
yannis.manolopoulos@ouc.ac.cy

***Abstract*—The representation of real-life problems by using k-partite graphs introduced a new era in Machine Learning. Moreover, the merge of virtual and physical layers through Location Based Social Networks (LBSNs) offers a different meaning into the constructed graphs. To this point, multiple models introduced in literature that aim to support users with personalized recommendations. These approaches represent the mathematical models that aim to understand users' behaviour by finding patterns on users' check-ins, reviews, ratings, friendships, etc. With this paper we describe and compare 20 of those state-of-the-art deep learning models to bring into the surface some of their strengths and shortcomings. First, we categorize them according to: data factors or features they use, data representation, methodologies used and recommendation types they support. Then, we highlight the existing limitations that tackles their performance. Finally, we introduce research trends and future directions.**

***Index Terms*—Deep learning, Location Based Social Networks, survey, limitations, new directions.**

I. Introduction

This era is commonly characterized as the information age, where the ever increasing mobile devices generate a plethora of online information regarding people's activities. The growth of Location Based Social Networks (LBSNs) attracted the interest of the research community and thus many methods were proposed that analyze the users' daily influx of data to provide personalized recommendations. The data generated through LBSNs can be interpreted in many ways, revealing different aspects of the users' behavior. Recommender systems take into account those influences aiming to suggest future actions according to their preferences. This has proven to be a challenge in the research community due to the seemingly chaotic behavior of the users.

In the effort to provide personalized recommendations different techniques were proposed, mainly aiming to accurately predict the users' future actions based on their previous behavior. It is important to analyze the different approaches to get a better view of the current state of POI recommendation algorithms. The models vary on many aspects, such as the type of data, how they handle this data to provide new insights and produce information, the way they represent the data, the training and learning procedure and the means to provide recommendations. Experiments on all the aforementioned aspects have proven to be effective in analyzing various aspects of the users' behavior directly connected to the influence factors which inevitably lead to their future actions.

Social networks were greatly developed in the recent years. The increased capabilities and traffic of the LBSNs were naturally followed by an increase on generated structured data regarding several aspects of the users. This raw influx of data streams can be exploited in various ways by computer algorithms. One of the most common goals of analyzing user data is to provide recommendations on numerous factors such as items, locations, friends, etc. In this survey we explore the algorithms focused on the visits of the users on locations, namely "check-ins", to provide recommendations on other places that are close to the users' preferences.

Deep learning is a field of Machine Learning and refers to neural networks with more than two layers that learns multiple levels of representations from data. The learning and training is based on the optimization of an objective function aiming to minimize the error in the

978-1-7281-1862-8/19/$31.00 ©2019 IEEE

TABLE I
CROSS ALGORITHM COMPARISON

		Data factors /features					Data representation				Methodologies and models					Recommendation types		
	Algorithm	Time	Trajectories	Textual	POIs	Ratings	Matrix-based	Graph-based	Tensor-based	Hybrid data	MLP	AE	CNN	RNN	DRL	Route	Location	Review
1	ReEL [1]			✓	✓	✓		✓					✓				✓	
2	ReGS [2]			✓	✓		✓						✓				✓	✓
3	PACE [3]			✓	✓		✓			✓	✓						✓	✓
4	USTT [4]	✓		✓	✓	✓		✓		✓					✓		✓	✓
5	GE [5]	✓		✓	✓			✓		✓					✓		✓	
6	CoTF [6]	✓		✓	✓				✓						✓		✓	
7	Chen et. al. [7]	✓	✓				✓								✓		✓	
8	ST-DME [8]	✓	✓		✓		✓								✓		✓	
9	NH-JTI [9]	✓			✓		✓								✓		✓	
10	JLGE [10]	✓			✓			✓		✓					✓		✓	
11	RWR-HST [11]	✓			✓			✓		✓					✓		✓	
12	Rank-GeoFM [12]	✓			✓				✓						✓		✓	
13	ASMF [13]				✓		✓								✓		✓	
14	BPRLR [14]				✓		✓			✓					✓		✓	
15	SAE-NAD [15]				✓		✓					✓					✓	
16	RecNet [16]				✓			✓			✓						✓	
17	He et. al. [17]		✓		✓				✓						✓	✓		
18	CAPRF [18]			✓			✓								✓		✓	
19	PK-Boosting [19]			✓						✓					✓			✓
20	ST-RNN [20]	✓			✓		✓							✓			✓	

reconstructed output.

This paper categorizes the state-of-the-art deep learning approaches used in LBSNs for recommendations and highlight existing limitations and new directions.

II. THE STATE-OF-THE-ART

In this section, we present an overview of the state-of-the-art POI recommendation models of Table I. In particular, 1) we analyze the various type of data factors/features used, 2) we examine the data representation, 3) we explore the methodologies and models used, and 4) finally, we study the different recommendations types provided to the users.

A. Data factor/features

In this section we present the main data factors/features used as input to the models to personalize their recommendations.

- **Time**: Nowadays, an upcoming feature that diversifies the penalization of the recommendations between periods, is the time dimension. In particular, this feature is used to describe users' preference evolution. It is obvious that users taste evolves, thus models recommendations should change as well. We notice that in eleven models [4]–[12], [19], [21].
- **Trajectories**: Another feature used as input is users' trajectories. With that term we refer to the sequence of POIs that a user attend during a defined time period. The influence of each POI in a trajectory is either high or low depending on how long a user stayed in a location. This information can reveal correlations between important POIs [7], [8], [17] or less significant ones [5], [19].
- **Textual**: Users tend to give reviews to attended locations expressing their opinion. To this point, many models [2]–[6], [12], [18], [19], [21] applied NLP to analyze the trends and correlations of reviews words assigned to each location. This way, the personalization considers the positive or the negative correlation of each word per location.
- **POIS**: Users check-in POIs defining their geographical location on LBSNs. This way models use spatial information to map the virtual and the physical layers [1]–[6], [8]–[13], [15]–[17], [21]. Such models personalize the recommendation given based on geographical proximity.
- **Ratings**: Users quantify their opinion about a POI in a scale of 1-5 stars. Models that use ratings aim to find unvisited locations that are similar to her

TABLE II
NOTATION TABLE

Symbol	Description
U	set of users $U = \{u_1, ..., u_m\}$,
L	set of locations $L = \{l_1, ..., l_n\}$,
$R, \hat{R}$	rating matrix, predicted matrix
θ	network parameters
r_{ul}	preference of user u to location l
$\hat{r}_{ul}$	predicted score
s_u, s_l	side information of users and locations
W, V	weight matrices
σ	sigmoid function
b, μ, λ	bias vectors, regularization parameter
u_i, h_i	the i^{th} visible and hidden units

past preferences [1], [2], [4], [18], [19], [21].

B. Data Representation

There are four main types of frameworks used:

- **Matrix-based**: representation is quite popular at recommender systems [2], [3], [7]–[9], [13]–[15], [18], [20]. They allow for a variety of methods to be used, such as matrix factorization but are greatly affected by the data sparsity.
- **Graph-based**: use k-partite networks to represent the relations between the participant entities whats why they are becoming a trend [1], [4], [5], [10], [11], [16]. In that way they considered as the general case of neural networks.
- **Tensor-based**: approaches use multi-dimensional arrays for data representation [6], [12], [17]. Due computational cost reduction such approaches gain ground everyday.
- **Hybrid**: Hybrid models combine two or more of the aforementioned data structures to form a unified framework [3]–[5], [10], [11], [14], [19].

C. Methodologies and Models

In this section we present the methodologies used for training. In particular, we consider the artificial neural network as a k-partite graph, thus we focus only on the deep learning techniques used in LBSNs. We use a unified notation presented in Table II. Below we present the simplest form of each category and explain how it was adopted to provide recommendations.

1) **Multi-layer Perceptron (MLP):** is the simplest type of a neural network that captures the non-linear relationship between to entities such as users and locations. This method is applied in *Supervised Learning* problems. In particular, the model is trained with an input-output pair to capture the correlations between these pairs. The goal is to minimize the error through propagation that adjusts the parameters or the weights. The propagation is a feed-forward procedure over an Artificial Neural Network (ANN) with at least one hidden layer. Except of the input nodes, the rest are neurons that use a non-linear activation function to propagate to the next layer of neurons. Thus, training is a hierarchical feature representation of these non-linear transformations.

In LBSNs, we usually have a bi-directional interaction between the users' preferences and the locations' features, thus, a dual neural network is required. We may score the user's u preference on location l using their respective side information s_u and s_l as follows:

$$\hat{r}_{ul} = f(U^T \cdot s_u, L^T \cdot s_l | U, L, \theta)$$

where the MLP is represented with the function $f(\cdot)$ and θ is the network's parameters. In its simplest form to train the model is by minimizing the following binary cross-entropy loss function:

$$argmin \sum_{(u,l) \in \mathcal{O}^+ \cup \mathcal{O}^-} r_{ul} \log \hat{r}_{ul} + (1 - r_{ul}) \log (1 - \hat{r}_{ul})$$

where $\mathcal{O}^*$ represent the positive and the negative instances used to train the model. Negative sampling is used to reduce the number of training unobserved instances. To this end, Ding et al. [16] developed networks that focus on co-visited locations according to geographical and categorical proximity. The hidden layers use Rectified Linear Unit as an activation function and also a dropout technique to alleviate the overfitting issue. The model learns users' attendance preference through a binary classification approach of minimizing a cross-entropy loss function.

On the other hand, Yang et al. [3] focused on user's preferences and POIs context and developed a MLP aiming to bridge collaborative filtering and semi-supervised learning. The data are first modeled into latent factors and context of the users and POIs is preserved through a softmax layer. Then, the merging layer is combing the two embedding vectors through an element-wise factorization. The joint training is performed through stochastic gradient decent until convergence is reached.

2) **Auto-Encoder (AE):** is an unsupervised technique for learning data representation of the data through a two steps procedure, the encoding and the decoding. The encoder is used to train the model through an activation function responsible for mapping the input to the latent space. On the other hand the decoder uses another activation function to reconstruct the latent space to the approximated space. The simplest form of an auto-encoder consists of a non-recurrent neural network that feeds forward following the same philosophy with MLP. The main difference is that the reconstructed space have the same number of node with the input. In RS auto encoders can be applied to learn lower dimensional feature representations at the bottleneck layer. Given the partial input vector r_l the reconstruction is defined as:

$$\begin{aligned}\textbf{encoder}: \ & z_l = f(W^{(1)} \cdot r_l + b^{(1)}) \\ \textbf{decoder}: \ & \hat{r}_l = g(W^{(2)} \cdot z_i + b^{(2)})\end{aligned}$$

where $f(\cdot)$ and $g(\cdot)$ are the activation functions for the encoder and the decoder respectively, W is the weight matrices, and b the bias vector. Notice that the z_l is the encoders' representation of input r_l, that is the decoder function uses to approximate the reconstruction of the input denoted as $\hat{r}_l$. The approximation of reconstructed input aims to minimize the following objective function:

$$argmin \sum_{l=1}^{N} \|r_l - \hat{r}_l\|_2^2$$

where $\| \cdot \|_2^2$ is the Frobenius norm over ratings. In this area, Ma et al. [15] developed an AE which consists of a self-attentive encoder and a neighbor-aware decoder. The first component aims to appoint various characteristics of locations based on user preferences, whereas the decoder takes into account the distance between locations expressed by a Gaussian radial basis kernel. The model is trained using Gradient Descent with Back-Propagation.

3) **Convolutional Neural Network (CNN):** is a variant of MLP designed to require less pre-processing. Each layer is connected by their respective receptive fields linking adjacent layer's neurons via a local connectivity pattern. CNNs perform best on large-scale networks as they require reduced memory usage for the training procedure. Moreover, they use convolution over previous layer pulling to generate a feature map. The feature map F^k is calculated as:

$$F_{ij}^i = \tanh((W^k * r)_{ij} + b^k)$$

There is a max-pooling layer where it down-samples the image by a sliding window which keeps the maximum value. This greatly reduces the computational cost as the layer shrink in size. Graph-based CNNs have excellent performance on recommendations on social network data and other similar non-Euclidean types due to the fact that interactions can be seen as a structured dataset (eg. bipartite graphs) and thus can be applied to such tasks. In this direction, Baral et al. [1] developed a model that captures the contextual preference of the users into bipartite graphs. After pre-processing the labels to categorize them into an aspect term, a CNN-based classifier consisting of an activation function, a convolution, a max-pooling, a dense and a soft-max layer is used to label each review sentence. Thus, a bipartite relation between the reviews and the aspects is formed and recommendations are generated by extracting the most dense sub-graphs of this network.

Xing et al. [2] proposed a joint convolution matrix factorization model which models the users' social behavior, geographical influence and review information. The three frameworks are unified using an integration of a CNN into a probability matrix factorization. They use the gradient descent and coordinate descent methods to optimize the objective function and the weights and biases of each layers are learned through the back-propagation algorithm.

4) **Recurrent Neural Network (RNN):** is similar to previous model but it preserves information learned, and apply it to future inputs. Thus, it able to remember and recognize patterns encountered across time. This is quite efficient when applied to sequential data. Also, it uses 1) a dynamic user state u_{ut} and location state v_{lt} learned from the LSTM, along with 2) the respective static attributes u_u and v_l learned from MF techniques. The predicted rating of a location can be calculated as:

$$\hat{r}_{ul|t} = f(u_{ut}, v_{lt}, u_u, v_l)$$

Once again the goal is to minimize the square error between the predicted and actual values.

Modelling spatio-temporal information can be challenging in RNNs. To model the temporal context, Liu et al. [20] proposed to replace the transition matrix with timespecific transition matrices to include continuous time intervals. Similarly, they incorporate distance-specific transition matrices to represent the geographical distances between locations. The algorithm is trained through Bayesian Personalized Ranking and back-propagation through time and stochastic gradient decent is applied until convergence is reached.

5) **Deep Reinforcement Learning (DRL):** is a popular technique used in RS to rank scores of future candidate location for a target user. In particular, a 'walker' learns users behaviour based on a k-partite graph. The approximated values are computed using gradient decent equations that aim to represent users spatial behaviour. Many approaches tend to incorporate temporal dimension along with the spacial one.

Aliannejadi et al. [19] proposed to maps users' tags with the locations' keywords. By using a gradient probabilistic approach they predict future users' tags on unvisited locations. Additionally, the preferred categories of each user is investigated according to his frequent positive and negative ratings on various locations. After the model's parameters are estimated using an Expectation-Maximization algorithm, they follow two approaches to predict future users' tags on locations: using the Maximum Likelihood criterion and Sequence Labeling.

Likewise, Gao et al. [18] proposed a sentiment enhanced weighting scheme where the importance of each sentiment information is being adjusted by the observed check-ins which are scaled appropriately. The model uses proximal gradient decent to optimize the difference of the word latent topics between users and POIs, and a projected strategy for the other parameters. Chen et al. [7] developed an algorithm that takes into account both POIs relevance to previous locations and their diversity. As those two factors are opposite to each other, a Chebyshev polynomial approximation method is used to balance the weight of each factor. Finally, he uses Stochastic Gradient Descent to optimize the parameters.

Focusing on the geographical information, Cheng et al. [14] used a Multi-center Gaussian Model to capture the geographical impact of the locations on the users' preferences, where large number of POIs tend to form centers of attention on the map increasing the probability of a user visiting a location close to its center. The user's preference is represented through Matrix Factorization and the two frameworks are combined and the final recommendations are produced through a Bayesian Personalized Ranking which uses stochastic gradient descent. In the same direction, Li et al. [12] proposed a ranking-based geographical factorization method where they combine geographical influence with users' preference. To capture the spatial influence they use geographical factorization and stochastic gradient decent is used to learn the incompatibility of the user over a location. In addition, they incorporate temporal information through extending the objective function to illustrate the model's capability of accepting new type of context.

Xie et al. [5] formed 4 networks that represent temporal, contextual and spatial features. The model is trained simultaneously on those models producing embeddings which indicate the influence of the aforementioned factors on the users' preference. Following the same direction, Christoforidis et al. [10] extracts 6 information networks and trains their model simultaneously on all of them producing embeddings through finding possible future second proximity connections between nodes of the aforementioned networks. Then, they provide personalized recommendations based on these embeddings.

Targeting temporal influence, Hosseini et al. [9] focused on characterizing the preference of users over locations for different time periods. Thus, they took into account the hourly, daily, weekly and monthly influence of time over the choices of the users. To find the best values over the probabilities of the user visiting a location on the aforementioned four temporal influences, and also the spatial impact, the model uses the Expectation-Maximization method over the historical data.

Analyzing the temporal and geo-sequential user behavior, Ding et al. [8] captures users' preferences and assign them into groups with related personalities. This model uses stochastic gradient decent to estimate the parameters and personalize recommendations based on the estimated rank of each unvisited location. Similarly, He et al. [17] exploited users trajectories to capture the locations a user has visited during a session using a tensor representation of the users' personal, spatial and transition preferences. Learning process consist an Expectation Maximization algorithm that optimizes the attendance probability. Another Tensor

Maroulis et al. [6] developed a model based on Tensor Factorization that takes into account geographical, temporal and location-context data. They first provide the loss function and add regularization terms to minimize the objective function so the users, POIs, context and core tensor matrix is formed. The tensor is then decomposed by mapping the three latent factors into a joint latent space model. The algorithm is trained through Stochastic Gradient Descent.

Similarly, Kefalas et al. [4] proposed a model that incorporates spatial, temporal and textual information to provide both POI and review recommendations. They used a close-proximity matrix to calculate weather a user is interested in reading a review or visiting a location. Furthermore, they introduced a regularization parameter to combine the two methods aiming to provide a unified framework which is accessed by a random walker that produces the final united recommendations. In a future work, Kefalas et al. [11], examined the impact of co-visited locations - namely 'session'. They developed 7 network graphs and applied a random walk with restart to calculate the probability of a user visiting a location.

Li et al. [13] distinct social friends, co-visiting location friends and users that live in close proximity. In addition, they calculate the geographical influence by using Maximum Likelihood Estimation. Similarly, to predict the probability of a user visiting a friend's location they use Random Walks. Stochastic Gradient Decent is used to optimize the model parameters.

D. Recommendation types

Finally, we categorize models based on the recommendation type they provide. The supported types are: (i) route recommendation, (ii) location recommendation, and (iii) review recommendation. Notice that some models support any of these type to a target user separately or combined. The majority of models support location recommendations [1]–[16], [18], [20], while some of them also support review recommendations [2]–[4]. We note that only two of them use spatial data to provide route [17] and review recommendations [19].

III. Conclusions and Future Work

LBSNs have been greatly developed in the recent years, where users daily produce information regarding their activities. This vast plethora of information led to the development of models that analyze and learn users' behavior to providing recommendations. With this paper we categorize the state-of-the-art recommendation algorithms in LBSNs, aiming to provide insight to the current approaches. At the same time we aim to point out the limitations and new directions. Since techniques have evolved from collaborative and content-based filtering to deep learning methods recommender systems aim to provide solutions to the challenges of this research area.

The ongoing research focuses on enhancing the models with additional features. The data can be augmented with side information to improve the understanding of the user's behavior. Furthermore, the introduction of an auto-tuned hyper-parameter can balance the impact of each influence network taken into account in the recommendation process. Finally, a combination of current models can form a hybrid construct which combines the advantages of each but limits their drawbacks.

References

[1] R. Baral, X. Zhu, S. S. Iyengar, and T. Li, "Reel: Review aware explanation of location recommendation," in *Proceedings of the 26th Conf. on User Modeling, Adaptation & Personalization (UMAP)*, Singapore, 2018, pp. 23–32.

[2] S. Xing, F. Liu, X. Zhao, and T. Li, "Points-of-interest recommendation based on convolution matrix factorization," *Applied Intelligence*, vol. 48, no. 8, pp. 2458–2469, 2018.

[3] C. Yang, L. Bai, C. Zhang, Q. Yuan, and J. Han, "Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation," in *Proceedings of the 23rd ACM International Conf. on Knowledge Discovery & Data Mining (KDD)*, Halifax, Canada, 2017, pp. 1245–1254.

[4] P. Kefalas and Y. Manolopoulos, "A time-aware spatio-textual recommender system," *Expert Systems with Applications*, vol. 78, pp. 396–406, 2017.

[5] M. Xie, H. Yin, H. Wang, F. Xu, W. Chen, and S. Wang, "Learning graph-based POI embedding for location-based recommendation," in *Proceedings of the 25th ACM International Conf. on Information & Knowledge Management (CIKM)*, Indianapolis, IN, 2016, pp. 15–24.

[6] S. Maroulis, I. Boutsis, and V. Kalogeraki, "Context-aware point of interest recommendation using tensor factorization," in *Proceedings of the IEEE International Conf. on Big Data*, Washington, DC, 2016, pp. 963–968.

[7] B. Chen, S. Yu, J. Tang, M. He, and Y. Zeng, "Using function approximation for personalized point-of-interest recommendation," *Expert Systems with Applications*, vol. 79, pp. 225–235, 2017.

[8] R. Ding, Z. Chen, and X. Li, "Spatial-temporal distance metric embedding for time-specific POI recommendation," *IEEE Access*, vol. 6, pp. 67 035–67 045, 2018.

[9] S. Hosseini, H. Yin, M. Zhang, X. Zhou, and S. W. Sadiq, "Jointly modeling heterogeneous temporal properties in location recommendation," in *Proceedings of 22nd International Conf. Database Systems for Advanced Applications (DASFAA)*, Suzhou, China, 2017, pp. 490–506.

[10] G. Christoforidis, P. Kefalas, A. N. Papadopoulos, and Y. Manolopoulos, "Recommendation of points-of-interest using graph embeddings," in *Proceedings of the 5th IEEE International Conf. on Data Science & Advanced Analytics (DSAA)*, Turin, Italy, 2018, pp. 31–40.

[11] P. Kefalas, P. Symeonidis, and Y. Manolopoulos, "Recommendations based on a heterogeneous spatio-temporal social network," *World Wide Web*, vol. 21, no. 2, pp. 345–371, 2018.

[12] X. Li, G. Cong, X. Li, T. N. Pham, and S. Krishnaswamy, "Rank-GeoFM: A ranking based geographical factorization method for point of interest recommendation," in *Proceedings of the 38th International ACM Conf. on Research & Development in Information Retrieval (SIGIR)*, Santiago, Chile, 2015, pp. 433–442.

[13] H. Li, Y. Ge, R. Hong, and H. Zhu, "Point-of-interest recommendations: Learning potential check-ins from friends," in *Proceedings of the 22nd ACM International Conf. on Knowledge Discovery & Data Mining (KDD)*, San Francisco, CA, 2016, pp. 975–984.

[14] C. Cheng, H. Yang, I. King, and M. R. Lyu, "A unified point-of-interest recommendation framework in location-based social networks," *ACM Transactions on Intelligent Systems & Technology*, vol. 8, no. 1, pp. 10:1–10:21, 2016.

[15] C. Ma, Y. Zhang, Q. Wang, and X. Liu, "Point-of-interest recommendation: Exploiting self-attentive autoencoders with neighbor-aware influence," in *Proceedings of the 27th ACM International Conf. on Information & Knowledge Management (CIKM)*, Torino, Italy, 2018, pp. 697–706.

[16] R. Ding and Z. Chen, "RecNet: A deep neural network for personalized POI recommendation in location-based social networks," *International Journal of Geographical Information Science*, vol. 32, no. 8, pp. 1631–1648, 2018.

[17] J. He, X. Li, L. Liao, and M. Wang, "Inferring continuous latent preference on transition intervals for next point-of-interest recommendation," in *Proceedings of the European Conf. on Machine Learning & Knowledge Discovery in Databases (ECML-PKDD)*, Dublin, Ireland, 2018, pp. 741–756.

[18] H. Gao, J. Tang, X. Hu, and H. Liu, "Content-aware point of interest recommendation on location-based social networks," in *Proceedings of the 29th AAAI Conf. on Artificial Intelligence*, Austin, TX, 2015, pp. 1721–1727.

[19] M. Aliannejadi and F. Crestani, "Personalized context-aware point of interest recommendation," *ACM Transactions on Information Systems*, vol. 36, no. 4, pp. 45:1–45:28, 2018.

[20] A. Liu, W. Wang, Z. Li, G. Liu, Q. Li, X. Zhou, and X. Zhang, "A privacy-preserving framework for trust-oriented point-of-interest recommendation," *IEEE Access*, vol. 6, pp. 393–404, 2018.

[21] H. Yin, X. Zhou, Y. Shao, H. Wang, and S. W. Sadiq, "Joint modeling of user check-in behaviors for point-of-interest recommendation," in *Proceedings of the 24th ACM International Conf. on Information & Knowledge Management (CIKM)*, Melbourne, Australia, 2015, pp. 1631–1640.