Electronic Research Archive

https://www.aimspress.com/journal/ERA

ERA, 32(4): 2728–2744.
DOI: 10.3934/era.2024123
Received: 02 February 2024
Revised: 10 March 2024
Accepted: 15 March 2024
Published: 02 April 2024

Research article

FAGRec: Alleviating data sparsity in POI recommendations via the feature-aware graph learning

Xia Liu[1] and Liwan Wu[2,*]

[1] School of Physics and Electronic Information, Yantai University, Yantai 264005, China
[2] Information Technology Center, Guangzhou Health Science College, Guangzhou 510450, China

* **Correspondence:** Email: wuliwan1@gzws.edu.cn.

Abstract: Point-of-interest (POI) recommendation has attracted great attention in the field of recommender systems over the past decade. Various techniques, such as those based on matrix factorization and deep neural networks, have demonstrated outstanding performance. However, these methods are susceptible to the impact of data sparsity. Data sparsity is a significant characteristic of POI recommendation, where some POIs have limited interaction records and, in extreme cases, become cold-start POIs with no interaction history. To alleviate the influence of data sparsity on model performance, this paper introduced FAGRec, a POI-recommendation model based on the feature-aware graph. The key idea was to construct an interaction graph between POIs and their initial features. This allows the transformation of POI features into a weighted aggregation of initial features. Different POIs can share the learned representations of initial features, thereby mitigating the issue of data sparsity. Furthermore, we proposed attention-based graph neural networks and a user preference estimation method based on delayed time factors for learning representations of POIs and users, contributing to the generation of recommendations. Experimental results on two real-world datasets demonstrate the effectiveness of FAGRec in the task of POI recommendation.

Keywords: POI recommendation; data sparsity; cold-start POI; feature-aware graph; graph neural network

1. Introduction

Point-of-interest (POI) recommendation [1] is one of the most important applications in location-based social networks [2]. The goal of POI recommendation [3,4] is to recommend POIs to users based on their historical visits and interests. Understanding users' interest preferences and implementing precise recommendation [5] services can greatly enhance the user experience on social platforms, bringing substantial economic value to the platform. Therefore, in the past decade, many researchers have devoted themselves to constructing powerful models for POI recommendation.

Unlike traditional recommendation tasks [6], in POI recommendation, user behavior is heavily influenced by the contextual environment. For example, users often go to restaurants during mealtime, so recommending a restaurant during these times is highly likely to meet their needs. Additionally, a POI recommendation needs to consider the practical cost of a user's actual visit, namely the distance traveled. Recommending POIs that are too far from the current location is meaningless because users cannot realistically reach those recommended locations. Therefore, existing POI-recommendation methods focus on modeling user preference features from the intricate contextual environment.

Popular techniques include topic models, graph embedding models, and graph neural networks. The key idea of topic model-based methods is to map users' historical visitation behaviors into a topic space, enabling a better understanding and prediction of user interests. These techniques capture the semantic information hidden behind user behaviors, thus enhancing the accuracy of POI recommendations. On the other hand, graph embedding or graph neural network–based methods aim to learn representations of users and POIs from the constructed graph. Since the check-in records of users could be naturally transformed into graph-like data, graph-based methods have achieved remarkable performance for the POI-recommendation tasks.

Despite the impressive performance of existing recommendation models, they require a substantial amount of data for training model parameters. However, this may be challenging to fulfill in practical application scenarios, as user behavior records are often extremely sparse in real-world situations. This poses a significant challenge known as the data sparsity problem in recommendation systems. Additionally, due to the particularities of the POI-recommendation scenario, user check-in records may become even sparser due to privacy concerns and other overhead issues. This implies that POI recommendation faces a more severe issue of data sparsity. In extreme cases, this can lead to the cold-start problem, which involves recommending a new POI where there is limited or no historical data available.

Traditional approaches to this challenge primarily involve content-based methods or generative models such as generative adversarial networks (GANs). For example, Chen et al. [7] introduced ColdGAN, a method that utilizes GANs' data generation prowess to handle uncertain data effectively. However, the reliance on generative model-based methods necessitates the incorporation of extra architectural components, leading to an inevitable increase in model complexity. This situation prompts the question: Is it possible to tackle the cold-start problem for POIs without resorting to additional models?

To tackle this question, in this paper, we propose a novel approach, the feature-aware graph attention network for POI recommendation (FAGRec). The core idea of this method is to construct a feature-aware graph, which connects POIs to features. In other words, we view the representations of POIs as weighted combinations of various features. This allows the learning of user and POI representations to be transformed into the learning of feature representations. Based on the constructed graphs, we introduce an attention-based graph neural network. Since the constructed graphs are of the bipartite type, the aggregation process of the proposed graph neural network is unidirectional and aggregates features only from the odd-hop neighborhoods. This characteristic ensures that the model can adequately learn the feature representations of users and POIs from the constructed graphs. Moreover, we develop a time-delay-based approach to capture dynamic user preferences. The main advantage of FAGRec is that it does not require additional auxiliary models to increase the training effect.

The contributions are summarized as follows:

- We develop a new method, FAGRec, to address the data sparsity problem in the field of POI-recommendation tasks without additional auxiliary models.
- FAGRec introduces a novel feature-aware graph to describe the relations of POIs and initial features, which is beneficial for learning representations for other POIs that share similar features.
- We conduct extensive experiments on two real-world datasets. The experimental results show the effectiveness of our proposed method on the POI-recommendation task.

2. Related work

In this paper, we review recent studies from two perspectives: methods for POI recommendation and methods for cold-start issue.

2.1. Methods for POI recommendation

Recently, various techniques have been developed to improve the performance of the POI-recommendation task. Zhang et al. [8] proposed context-enhanced probabilistic diffusion, which integrates contextual information into a probabilistic diffusion process to generate effective POI recommendations in sparse data environments. Rahmani et al. [9] emphasized the importance of appropriately selecting and combining various contextual factors that can significantly impact POI recommendation performance, quantifying the impact of different contextual information. Ji et al. [10] introduced a probabilistic generative model, STARec, which focuses on activities. STARec defines users' social preferences separately from individual interests, combining both to accurately depict user preferences and model the inconsistency between social preferences and decisions. Sun et al. [11] proposed the STSP with category- and location-aware encoders for users, enhancing the next POI prediction by fusing rich context features.

Besides these efforts, Chen et al. [12] proposed a POI-recommendation method to incorporate time and distance irregularities and dynamic weight decay values in the learning process. The awareness weights provide an interpretable way to understand the emphasis on each context in the prediction. Sey et al. [13] incorporated social, geographical, and temporal information into the matrix factorization (MF) model to overcome data sparsity. Liu et al. [14] employed a spatial-temporal heterogeneous information network (HIN) to jointly model various context features, including spatiotemporal effects, POI types, and social relations. HIN defines a spatial-temporal effects entity (St) and utilizes a four-way neural interaction model to effectively mine information from meta-path-based context and spatiotemporal effects, resulting in improved recommendation performance. Liu et al. [15] introduced a novel ensemble learning framework called PG-PRE, which constructs multiple similar user groups using a roulette selection–based sampling method, improves diversity, and combines recommendations using a Gaussian mixture–based approach. Wang et al. [16] proposed VirHpoi, a heterogeneous hypergraph embedding method for POI recommendation that leverages the hypergraph to capture complex interactions and learns the hypergraph by preserving homophily and interaction attribute affinity. Qin et al. [17] introduced a disentangled representation learning method that employs a geo-constrained negative sampling strategy to find reliable negative samples and a geo-enhanced soft-weighted loss function to quantify the trade-off between geographical and user interest factors.

Different from previous POI methods, our proposed method introduces a new concept named feature-aware graph, which captures the interactions between POIs and features. This merit allows the proposed method to alleviate the impact of the cold-start issue.

2.2. Methods for cold-start issue

Here, we also introduce recent efforts for addressing the cold-start issue in the field of recommender systems. Puthiya et al. [18] proposed a novel soft-cluster embeddings approach that represents items as soft-cluster embeddings in the space defined by associated side information. Li et al. [19] proposed the conversational Thompson sampling model, which aims to address this limitation by unifying attributes and items in the same arm space, automatically achieving exploration-exploitation trade-offs using Thompson sampling. Qian et al. [20] developed an attribute-graph neural network framework that leverages the attribute graph instead of the interaction graph, allowing it to learn embeddings for strict cold-start users or items. Zhang et al. [21] proposed a FINI framework that incorporates a global-local context attention mechanism for dynamic feature importance and a mixed interaction mechanism to enhance weighted neighbor interactions. The key idea of FINI is to strengthen the expressive capability of user/item embeddings and improve the quality of the mapping function for cold-start scenarios. Wu et al. [22] proposed a new method called UIMA, which addresses this problem using zero-shot learning principles. UIMA includes two auto-encoders for learning user and item embeddings and a matching network to explore the relationship between these embeddings. Cai et al. [23] developed the IHGNN model, which converts new users, items, and associated multimodal information into a modality-aware heterogeneous graph, preserving rich and heterogeneous relationship information. Rehman et al. [24] proposed CAML, which leverages meta-learning and incorporates contextual features to enhance tasks' adaptation capabilities. The key idea of CAML is to introduce a data augmentation unit that uses a hybrid similarity to augment data samples from similar neighbors.

Though many efforts have been undertaken for the cold-start problem, they have rarely been designed for POI recommendation. In this paper, we develop a novel solution for overcoming the cold-start issues in the field of POI-recommendation tasks.

3. Preliminaries

In this section, we introduce the notations and basic neural networks used in this paper.

3.1. Notations

Here, we first introduce several definitions of this paper.

Definition 1. (POI) A POI represents a place with a certain function, such as a restaurant for food and a cinema for movies. Here, we use y to represent a POI and use Y to represent the POI set. Each POI has the location information, which is represented by l_y in this paper.

Definition 2. (Check-in record) A check-in record represents one user's check-in behavior. We use $c_u = (y, l, t)$ to represent a check-in record of user u, where y denotes the checked POI, l denotes the location, and t denotes the check-in timestamp. Also, we use C_u that denotes all check-in records of user u.

Definition 3. (Cold-start POI) A cold-start POI means the corresponding item has very less or no interactions. In this paper, we regard a number of interactions less than five as the cold-start POI.

Definition 4. (POI latent graph) A POI latent graph $G_y = (V_y, E_y)$ represents the latent graph of POIs. G_v is a bipartite graph where $|V_v| = |Y| + |D^v|$. $|Y|$ denotes the POI set and $|D^v|$ denotes the dimensions of the POI features. The goal of G_v is to model the interactions between POIs and features. G_v treats each dimension of the POI features as the independent node. This way, the POI features can be regarded as the combination of a series of feature nodes.

Definition 5. (Top-k POI recommendation) Given the check-in history of a user C_u, the goal of the Top-k POI recommendation is to generate a POI-recommendation list $Y^R = \{Y_1, ..., Y_k\}$ that the target user may have interest to check in.

Notations are summarized in Table 1.

Table 1. Summarization of notations.

Notations	Description
y, Y	a POI, a POI set
l_y	the location of POI y
u, U	a user, a user set
t	the check-in timestamp
C_u	all check-in records of user
G_y	the POI latent graph
$\mathbf{X}^Y_i$	the initial features of POI
$\mathbf{H}^Y$, $\mathbf{H}^U$	the representations of POIs and users

3.2. Graph attention network

Graph attention network (GAT) is a popular architecture of graph neural networks. GAT leverages the attention mechanism to learn the aggregation weights and aggregate the information from the neighbor nodes. The calculation of the aggregation weights in GAT is as follows:

$$a_{ij} = \frac{\exp((\mathbf{H}_i \,\|\, \mathbf{H}_j)\mathbf{W}^a)}{\sum_{v_k \in N_i} \exp((\mathbf{H}_i \,\|\, \mathbf{H}_j)\mathbf{W}^a)}, \tag{1}$$

where $\mathbf{H}_i, \mathbf{H}_j$ are the representations of the corresponding nodes. $\|$ denotes the vector concatenation operation. $\mathbf{W}^a$ is the learnable parameter matrix. a_{ij} denotes the aggregation weights. Then, GAT adopts the following aggregation operation to collect information from neighbors:

$$\mathbf{H}^{out}{}_{i} = \sum_{v_j \in N_i} a_{ij} \cdot \mathbf{H}_j \,, \tag{2}$$

where $\mathbf{H}^{out}{}_{i}$ denotes the output of the node i.

4. Methodology

In this section, we introduce the detailed designs of the proposed method FAGRec. Specifically, we first introduce how to construct the POI latent graphs according to the known information. Then, we introduce the GAT-based neural network, which could effectively learn the representations of users and POIs. Finally, we introduce how to use the proposed method to generate the POI-recommendation list.

4.1. Latent graph construction

The key problem of cold-start POI recommendation is to calculate the representations of POIs even with no interactions with users. So, the crucial step is to establish the connections between known POIs and cold-start POIs. In this paper, we address this problem by designing a POI latent graph.

Suppose a POI y_i with the associated feature vector $\mathbf{X}_i \in \mathbb{R}^{1\times d}$ that has d-dimensional features; we attempt to generate a graph to reflect the relations of the target POI and the corresponding features. In practice, we obtain the initial features of the nodes through random initialization. Hence, we construct G_v for this purpose. For a POI y_i, we treat a POI and each dimension of the feature vector as nodes on G_v. This way, we have d + 1 nodes on the graph. Further, we generate d edges between the POI node and each feature node. Then, for the POI node v_{y_i} and the feature node v_{d_j}, the weight of the edge between the above nodes is $\mathbf{X}_{ij}$, which denotes the relevance of the target POI to the specific feature. For the feature nodes, we generate the corresponding feature matrix to represent their hidden features $\mathbf{X}^F \in \mathbb{R}^{d\times d'}$. In this way, the hidden feature can be calculated as follows:

$$\mathbf{X}_i^Y = \sum_{v_j \in N_i} w_{ij} \cdot \mathbf{X}_j^F \tag{3}$$

where w_{ij} denotes the edge weight between POI node and the corresponding feature node. $\mathbf{X}_i^Y$ denotes the hidden feature of POI y_i calculated by the associated feature vectors.

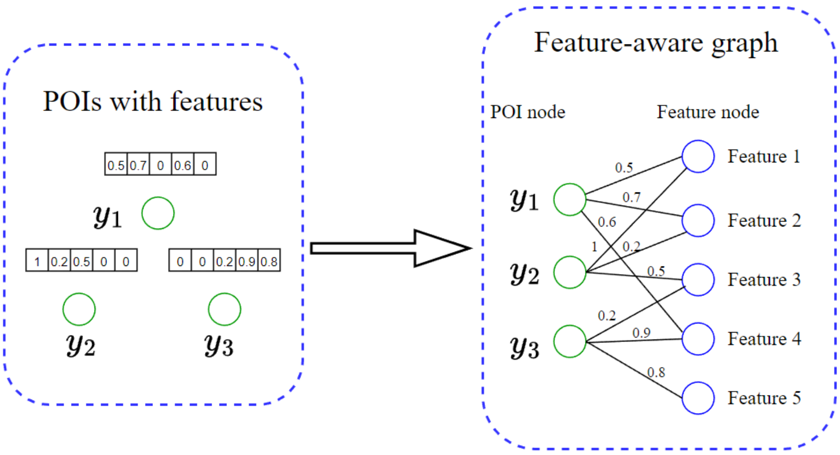

Figure 1. The toy example of constructing the latent graph.

The advantage of transforming into the graph-style data is that it allows us to calculate the features of POIs based on the associative relationships between POIs and the initial features. This requires training only the representation vectors of the feature nodes. For the representation vector of a POI node, it can be calculated through weighted aggregation. This way, for a cold-start POI, it is only necessary to identify its initial features, and its representation vector can be quickly computed from the constructed graph, thereby addressing the POI cold-start problem.

4.2. Attention-based graph neural network

After constructing the graph, we need to design a neural network module to learn the representations of POIs from the constructed graph. In this paper, we build a graph neural network framework based on GAT for the constructed graph. Its key idea is to utilize information from odd-order neighborhoods to learn the representation of POIs.

Specifically, we first utilize Eq (3) to obtain the initial feature vectors of POIs. Then, we design the following attention-based neural network to learn the representations of POIs:

$$\mathbf{H}_i^Y = \sum_{v_j \in N_i^{2k+1}} {}^1\!/_k \cdot \alpha_{ij} \cdot \mathbf{H}_j^F \tag{4}$$

where $\mathbf{H}_i^Y$ and $\mathbf{H}_j^F$ are the hidden representations of POI node *i* and the feature node *j*. In practice, we use two linear layers to obtain the above representations based on the initial features of the POI node and the feature node. k denotes the neighborhood range. ${}^1\!/_k$ denotes topology-aware aggregation weights, which means that nodes in a large range of the neighborhood will contribute less to the final representations than those in the local neighborhood. Hence, the proposed method can effectively preserve the topology information during the aggregation process. α_{ij} is learnable aggregation weight, which is calculated as follows:

$$\alpha_{ij} = \tanh((\mathbf{H}_i^Y \| \mathbf{H}_j^F)\mathbf{W}^a) . \tag{5}$$

Similar to the calculation of GAT shown in Eq (1), we also adopt the projection layer to learn the aggregation weights between nodes. However, different from the attention mechanism in GAT, we utilize the $\tanh(\cdot)$ function to generate the attention weights between nodes. The advantage of this strategy is that $\tanh(\cdot)$ can generate signed attention weights. Since neighbors in the large-range neighborhood can contain irrelevant feature nodes for POI nodes, the generated signed attention weights can help the model to learn distinguishable representations from these irrelevant nodes.

4.3. User representation generator

After obtaining representations of POIs via the developed attention-based graph neural network, we need to estimate the representation of users. In real-world scenarios, user preferences change over time. Aggregating the representations of POIs that a user has visited directly through summation is insufficient to capture the dynamic changes in user preferences. Therefore, we employ a time-delay-based approach to estimate the user preferences.

The key idea of the time-delay-based approach is to define the time-delay coefficients that are used to estimate the dynamic user preference. In this paper, we develop the following strategy to calculate the user preference via the checked POIs based on the time-delay coefficients:

$$\mathbf{H}_q^U = \sum_{(y_i, t_i) \in C^q} \exp(-(t_0 - t_i)) \cdot \mathbf{H}_i^Y , \tag{6}$$

where t_i denotes the check-in timestamp of POI y_i by the user q. t_0 is the current timestamp. $\exp(-(t_0 - t_i))$ is the time-delay coefficient, which means that the impact of POI on user preferences diminishes as the check-in records for the POIs become more distant, while the influence increases for more recent check-in records.

By calculating time-delay coefficients based on user check-in times, we ultimately compute the user's representation using a weighted sum method depicted in Eq (6).

4.4. POI-recommendation generator

After obtaining the representations of the user and the POI, the final step of RAGRec is to generate POI recommendations for users based on the learned representations. The key step of generating the POI-recommendation list is to estimate the recommendation scores of POIs for the target user. Traditional strategies directly calculate the similarity scores of POIs and the target user via the learned hidden representations. However, these strategies ignore the influence of the geographical factor. One of the biggest differences between POI recommendation and traditional item recommendation is that the user must afford the cost of visiting the recommended POI. Hence, recommending faraway POIs to the target user is unadvisable and unreasonable. This situation motivates us to consider the geographical factor in estimating the recommendation scores of POIs. In this paper, we introduce the geographical information to generate the recommendation score. For the target user q, the recommendation score of POI y_i is calculated as follows:

$$s_{y_i}{}^{u} = \frac{d_{\max}}{d_{y_i}} \cdot (\mathbf{H}_q{}^{U} \cdot \mathbf{H}_i{}^{Y\mathrm{T}}), \tag{7}$$

where $s_{y_i}{}^{u}$ denotes the recommendation score of POI y_i for the user q. d_{y_i} denotes the distance between the POI and the current location of the user. $d_{\max}$ is the threshold of the geographical distance. Considering the actual circumstances, we have set $d_{\max}$ to 60 km, which happens to be a one-hour drive. $\frac{d_{\max}}{d_{y_i}}$ is the geographical factor, which controls the influence of the geographical distance on the recommendation suggestions. If the distance between the target user and the POI is larger than $d_{\max}$, the geographical factor will weaken the recommendation score of the POI.

4.5. Model parameter learning

As for model parameter learning, we adopt the widely used BPR loss function in the field of recommender systems. For applying BPR loss to learn the model parameters, we construct the training samples, which contain three elements (q, y_i, y_j). y_i, y_j denotes the POIs that appear and do not appear in the check-in history of user q. Following this strategy, we can construct the set of training samples D^t. Then, the loss function of FAGRec is as follows:

$$Loss_{\mathrm{FAGRec}} = \sum_{(q,y_i,y_j)\in D^t} -\ln(\sigma(\mathbf{H}_q{}^{U} \cdot \mathbf{H}_i{}^{Y\mathrm{T}} - \mathbf{H}_q{}^{U} \cdot \mathbf{H}_j{}^{Y\mathrm{T}})) + \lambda \|\theta\|^2, \tag{8}$$

where $\sigma(\cdot)$ denotes the nonlinear activation function. λ is the regularization coefficient and θ denotes the model parameters. By minimizing Eq (8), the SGD technique can be used for learning the model parameters. As for the computational complexity, the main training cost of FAGRec is learning representations of POIs from the feature-aware graph. Suppose the number of edges in the generated graph is $|E|$, and the total computational cost is $O(|E|d)$, where d is the dimension of the representations. The overall framework of FAGRec is shown in Figure 2.

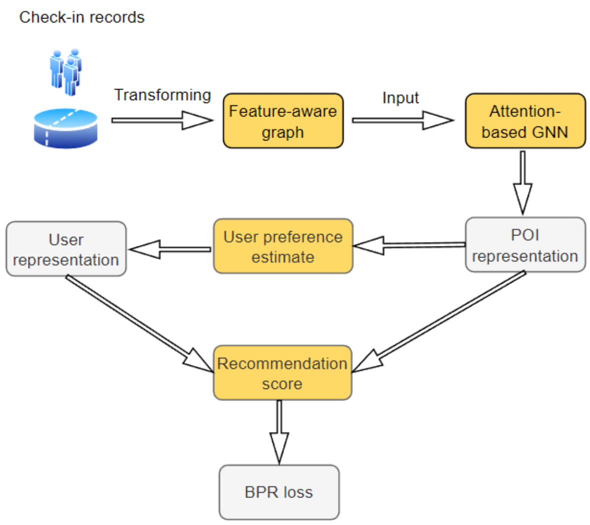

Figure 2. Overall framework of FAGRec.

5. Experiment

In this section, we first present the experimental setup of this paper including datasets, baselines, and parameter configurations. Then, we report the experimental results, containing model performance comparisons, sparsity analysis, parameter sensitivity analysis, and ablation studies.

5.1. Dataset

Here, we adopt two widely used datasets extracted from real-world applications: Gowalla and Foursquare. Each dataset contains the check-in records of all users. We reorder all check-in records of users according to the check-in timestamp. Each check-in record in the dataset contains four elements: the user, the POI, the location information, and the check-in timestamp. We choose the early 80% check-in records as the training set, the next 10% as the validation set, and the last 10% as the test set. Detailed information of the above datasets is reported in Table 2.

Table 2. Statistics of datasets.

Dataset	**#users**	**#POIs**	**#records**	**density**
Gowalla	3,318	33,665	635,600	0.3%
Foursquare	10,180	16,561	867,107	0.15%

5.2. Baselines

In this paper, we adopt the following seven representative baselines for comparison:

POI2Vec [25]: POI2Vec is a latent representation model based on knowledge graph, which hierarchically divides POIs into various regions, employing a binary tree structure to capture check-in sequences along with location and temporal information.

TMCA [26]: TMCA is a framework employing an LSTM-based encoder-decoder architecture that integrates multi-level contextual factors, incorporating temporal information.

STAN [27]: STAN is a method based on attention mechanisms, which explicitly incorporates spatial-temporal correlations within a trajectory.

SAE-NAD [28]: SAE-NAD is the autoencoder-based method that comprises a self-attentive encoder and a neighbor-aware decoder.

STACP [29]: STACP utilizes the matrix factorization to model user preferences on geographical and temporal factors.

LGLMF [30]: LGLMF leverages matrix factorization techniques that utilize geographical information to harness a user's personal geographic profile.

CPIR [31]: CPIR also is a matrix factorization–based method that adopts the matrix factorization strategy to calculate the similarity of users and POIs.

SLS-REC [32]: SLS-REC utilizes the hypergraph neural network to learn user preferences based on users' check-in behaviors in short-term dynamic interests.

SGM-GCN [33]: STACP constructs various subgraphs to capture the influence of different contextual factors and further leverages graph neural networks to learn user preferences.

5.3. Parameter settings

Here, we detail the parameter settings of experiments. For each baseline, we adopt the official implementation in their homepage. For our proposed FAGRec, we try the dimension of the hidden representation in $\{64, 128, 256\}$, and the learning rate in $\{0.01, 0.005, 0.001\}$. Moreover, we leverage the grid search to determine the optimal values of hyperparameters. The implementation is based on Python 3.6. All experiments are conducted on a personal computer with one Intel 13900K CPU, 64 G RAM.

5.4. Performance comparison

In this paper, we adopt two popular metrics, $Hits@k$ and MRR, to evaluate the performances of all methods. $Hits@k$ is used to measure whether the recommended POI appears in the target user's check-in records. MRR is used to represent the average reciprocal rank of positive examples, serving as a metric that indicates the model's overall ranking ability. The calculation of each evaluation metric is as follows:

$$Hits@k = \frac{\#Hit@k}{|D^{test}|} \tag{9}$$

$$MRR = \frac{1}{|D^{test}|} \sum_{i=1}^{|D^{test}|} \frac{1}{rank(Y_i)} \tag{10}$$

where $Hit@k$ denotes the true checked POI appearing in the recommendation list. $rank(Y_i)$ denotes the rank of the checked POI in the recommendation list. D^{test} denotes the test set.

We run each model five times with different random seeds and report the average values of each metric on two datasets. The results are shown in Tables 3 and 4.

Table 3. Performances (%) of all methods on the Gowalla dataset.

Methods	***Hits@5***	***Hits@10***	***Hits@20***	***MRR***
SAE-NAD	10.52	11.34	12.12	8.25
STACP	11.68	15.61	16.68	9.98
LGLMF	12.14	15.75	16.79	10.21
CPIR	13.54	17.96	21.15	11.28
POI2Vec	15.63	18.96	23.65	12.75
TMCA	17.85	21.11	24.14	13.52
STAN	18.68	22.86	28.34	14.65
SLS-REC	18.24	21.58	25.68	13.74
SGM-GCN	18.52	22.49	26.95	14.21
FAGRec	20.01	24.85	31.21	16.13

From the experimental results in the table, we have the following observations: First, methods based on matrix factorization perform the worst. This is because matrix factorization is highly sensitive to data sparsity, and in real-world scenarios, data sparsity is high, making it challenging to adequately support the operations of matrix factorization. Second, our proposed method achieves the best performance under various evaluation metrics on both datasets, especially compared to the advanced graph-based POI-recommendation approaches SLS-REC and SGM-GCN. This experiment validates the effectiveness of our proposed approach in real-world application scenarios. Third, it is worth noting that all methods exhibit weaker results on Foursquare compared with Gowalla. This is attributed to the higher sparsity of the Foursquare dataset, as evident from statistical analyses of the datasets. This further confirms that data sparsity significantly influences the model's performance.

Table 4. Performances (%) of all methods on the Foursquare dataset.

Methods	*Hits@5*	*Hits@10*	*Hits@20*	*MRR*
SAE-NAD	11.28	12.35	13.84	9.15
STACP	12.74	13.54	15.68	9.65
LGLMF	13.52	15.24	18.01	10.41
CPIR	13.98	16.52	19.24	10.68
POI2Vec	15.03	17.55	21.35	12.01
TMCA	16.85	18.72	22.36	12.62
STAN	18.21	21.32	24.67	14.65
SLS-REC	17.34	20.67	23.96	13.85
SGM-GCN	18.36	21.54	25.12	14.96
FAGRec	19.32	23.21	26.75	15.68

5.5. Study of data sparsity

To verify the model's performance in the face of adversarial data sparsity, we adjusted the number of cold-start POIs in the dataset. Specifically, for each dataset, we randomly selected a certain proportion of nodes in the test set, deleted their associated records with users, and then observed the performance of each method on these datasets. The experimental results are shown in Figures 3 and 4.

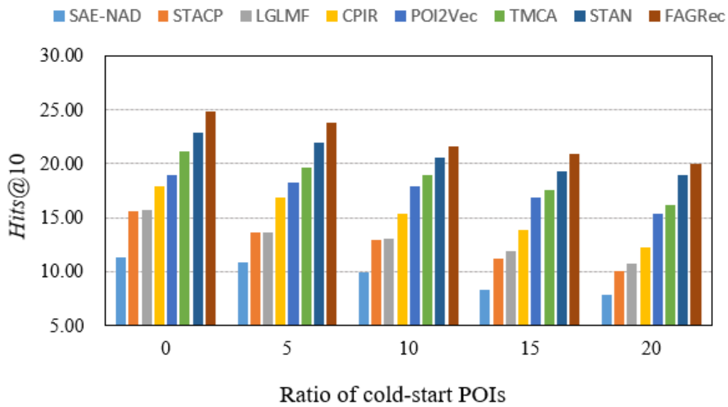

Figure 3. Performance of each model under different ratios of cold-start POIs on Gowalla.

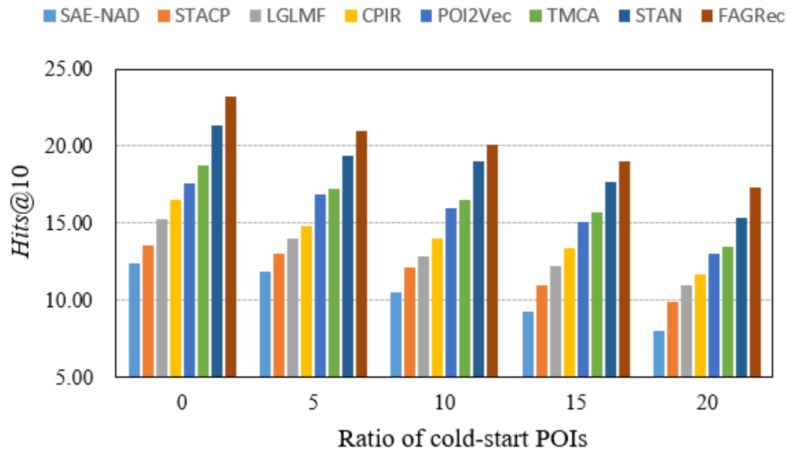

Figure 4. Performance of each model under different ratios of cold-start POIs on Foursquare.

The experimental results from two datasets indicate that our approach exhibits better performance in addressing adversarial data sparsity issues. This is attributed to our utilization of graph-structured features to depict the associative relationships between interest points and features. The data in the form of this graph structure is more adept at capturing the latent representations of POIs, leading to improved performance when confronted with data sparsity challenges.

5.6. Parameter sensitive analysis

In this section, we explore the impact of propagation steps on the model's performance. Propagation steps determine the range of aggregated neighborhood information. A larger propagation step implies aggregating information from a larger neighborhood. While obtaining more crucial information, it also introduces more noise. To investigate the influence of propagation steps on the model's performance, we conducted a performance analysis for different propagation step values, and the results are depicted in Figures 5 and 6. We observe that, on both datasets, a larger propagation step leads to a significant decrease in the model's performance, aligning with our initial hypothesis. Additionally, we note that the optimal propagation step required on Gowalla is smaller than that on Foursquare, which is attributed to the denser nature of the Gowalla dataset.

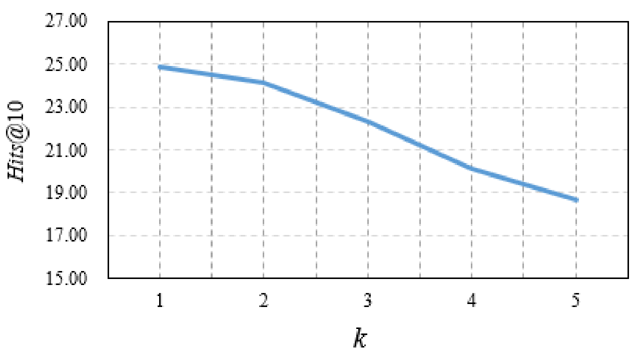

Figure 5. Performance of FAGRec under different propagation steps on Gowalla.

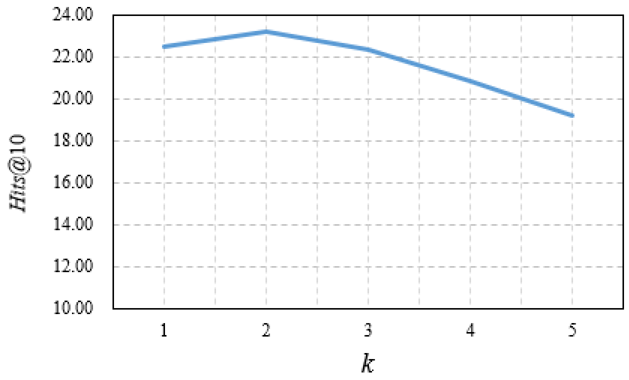

Figure 6. Performance of FAGRec under different propagation steps on Foursquare.

5.7. Ablation study

Here, we develop experiments to analyze the gains of the time-delay-based approach used in learning user performance. Specifically, we develop a variant of FAGRec, named FAGRec-m, which uses the average aggregation strategy to calculate user preferences. Then, we run two models on all datasets, and the results are shown in Tables 5 and 6.

The experimental results reveal that the performance of FAGRec-m is inferior to FAGRec on both datasets, validating the effectiveness of our design. Additionally, it is evident that the impact of this design is more pronounced on the Foursquare dataset. This is attributed to it being sparser than Gowalla, with fewer check-in records per user. Consequently, it exhibits higher sensitivity to the time factor. Our proposed method excels at incorporating temporal factors to calculate user preferences, making it more advantageous on the Foursquare dataset.

Table 5. Performances (%) of FAGRec and its variant on Gowalla.

Methods	*Hits@5*	*Hits@10*	*Hits@20*	*MRR*
FAGRec-m	18.69	23.15	29.98	15.23
FAGRec	20.01	24.85	31.21	16.13

Table 6. Performances (%) of FAGRec and its variant on Foursquare.

Methods	*Hits@5*	*Hits@10*	*Hits@20*	*MRR*
FAGRec-m	17.39	20.96	23.15	13.65
FAGRec	19.32	23.21	26.75	15.68

6. Conclusions

In this paper, we propose the FAGRec model to alleviate the impact of data sparsity on POI-recommendation models. Specifically, the FAGRec model constructs a feature-aware graph to model the relationships between POIs and each feature. This enables the effective representation of the association between POI representations and initial features. Through the constructed graph, the representation of POIs can be viewed as a weighted sum of initial features. Thus, for newly emerging

POIs, their latent features can be well represented based on the distribution of initial features. Furthermore, we introduce an attention-based graph neural network model to learn the representations of POIs and initial features from the constructed graph. In terms of learning node representations, we employ a time-delayed computation method to capture dynamic changes in user preferences. We conducted extensive experiments on two real datasets, and the results demonstrate the effectiveness of our approach in POI-recommendation tasks.

As mentioned before, the main advantage of FAGRec is that it deeply captures the interactions between POIs and their semantic features. This allows the model to alleviate the impact of the cold-start issue, since the representations of new POIs can be learned according to the relations of features via the feature-aware graph. The main limitation of FAGRec is that it only leverages the naïve graph neural network architecture, which may be inefficient in learning complex relations between POIs and feature elements. Hence, the promising way to enhance the FAGRec is to introduce advanced GNNs to learning informative representations from the constructed feature-aware graph.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work is supported by the Special Research Project of the Higher Education Research and Development Center of the Ministry of Education (No. ZJXF2022209).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. Y. Hwangbo, K. J. Lee, B. Jeong, K. Y. Park, Recommendation system with minimized transaction data, *Data Sci. Manage.*, **4** (2021), 40−45. https://doi.org/10.1016/j.dsm.2022.01.001
2. L. Shi, G. Song, G. Cheng, X. Liu, A user-based aggregation topic model for understanding user's preference and intention in social network, *Neurocomputing*, **413** (2020), 1−13. https://doi.org/10.1016/j.neucom.2020.06.099
3. W. Ji, X. Meng, Y. Zhang, STARec: Adaptive learning with spatiotemporal and activity influence for POI recommendation, *ACM Trans. Inf. Syst.*, **40** (2021), 1−40. https://doi.org/10.1145/3485631
4. L. Shi, J. Luo, C. Zhu, F. Kou, G. Cheng, X. Liu, A survey on cross-media search based on user intention understanding in social networks, *Inf. Fusion*, **91** (2023), 566−581. https://doi.org/10.1016/j.inffus.2022.11.017
5. W. Ji, X. Meng, Y. Zhang, SPATM: A social period-aware topic model for personalized venue recommendation, *IEEE Trans. Knowl. Data Eng.*, **34** (2020), 3997−4010. https://doi.org/10.1109/TKDE.2020.3029070

6. L. Shi, J. P. Du, G. Cheng, X. Liu, Z. G. Xiong, J. Luo, Cross-media search method based on complementary attention and generative adversarial network for social networks, *Int. J. Intell. Syst.*, **37** (2021). https://doi.org/10.1002/int.22723
7. C. C. Chen, P. L. Lai, C. Y. Chen, ColdGAN: An effective cold-start recommendation system for new users based on generative adversarial networks, *Appl. Intell.*, **53** (2023), 8302−8317. https://doi.org/10.1007/s10489-022-04005-1
8. Z. Zhang, M. Dong, K. Ota, Y. Zhang, Y. Kudo, Context-enhanced probabilistic diffusion for urban point-of-interest recommendation, *IEEE Trans. Serv. Comput.*, **15** (2021), 3156−3169. https://doi.org/10.1109/TSC.2021.3085675
9. H. A. Rahmani, M. Aliannejadi, M. Baratchi, F. Crestani, A systematic analysis on the impact of contextual information on point-of-interest recommendation, *ACM Trans. Inf. Syst.*, **40** (2022), 1−35. https://doi.org/10.48550/arXiv.2201.08150
10. W. Ji, X. Meng, Y. Zhang, STARec: Adaptive learning with spatiotemporal and activity influence for POI recommendation, *ACM Trans. Inf. Syst.*, **40** (2021), 1−40. https://doi.org/10.1145/3485631
11. Z. Sun, C. Li, Y. Lei, L. Zhang, J. Zhang, S. Liang, Point-of-interest recommendation for users-businesses with uncertain check-ins, *IEEE Trans. Knowl. Data Eng.*, **34** (2021), 5925−5938. https://doi.org/10.1109/TKDE.2021.3060818
12. Y. C. Chen, T. Thaipisutikul, T. K. Shih, A learning-based POI recommendation with spatiotemporal context awareness, *IEEE Trans. Cybern.*, **52** (2020), 2453−2466. https://doi.org/10.1109/TCYB.2020.3000733
13. K. Seyedhoseinzadeh, H. A. Rahmani, M. Afsharchi, M. Aliannejadi, Leveraging social influence based on users activity centers for point-of-interest recommendation, *Inf. Process. Manage.*, **59** (2022), 102858. https://doi.org/10.48550/arXiv.2201.03450
14. Y. Liu, Z. Yang, T. Li, D. Wu, A novel POI recommendation model based on joint spatiotemporal effects and four-way interaction, *Appl. Intell.*, **52** (2022), 5310−5324. https://doi.org/10.1007/s10489-021-02677-9
15. S. Liu, L. Yang, W. Zheng, Y. Xiao, L. Liu, An ensemble learning model for preference-geographical aware point-of interest recommendation, *Appl. Intell.*, **52** (2022), 13763-13780. https://doi.org/10.1007/s10489-022-04035-9
16. C. Wang, M. Yuan, R. Zhang, K Peng, L Liu, Efficient point-of-interest recommendation services with heterogenous hypergraph embedding, *IEEE Trans. Serv. Comput.*, **16** (2022), 1132−1143. https://doi.org/10.1109/TSC.2022.3187038
17. Y. Qin, C. Gao, Y. Wang, S. Wei, D. Jin, J. Yuan, et al., Disentangling geographical effect for point-of-interest recommendation, *IEEE Trans. Knowl. Data Eng.*, (2022). https://doi.org/10.1109/TKDE.2022.3221873
18. S. A. Puthiya Parambath, S. Chawla, Simple and effective neural-free soft-cluster embeddings for item cold-start recommendations, *Data Min. Knowl. Discovery*, **34** (2020), 1560−1588. https://doi.org/10.1007/s10618-020-00708-6
19. S. Li, W. Lei, Q. Wu, X. He, P. Jiang, T. S. Chua, Seamlessly unifying attributes and items: Conversational recommendation for cold-start users, *ACM Trans. Inf. Syst.*, **39** (2021), 1−29. https://doi.org/10.1145/3446427
20. T. Qian, Y. Liang, Q. Li, S. Wei, D. Jin, J. Yuan, et al., Attribute graph neural networks for strict cold start recommendation, *IEEE Trans. Knowl. Data Eng.*, **34** (2020), 3597−3610. https://doi.org/10.1109/TKDE.2020.3038234

21. J. Zhang, C. Ma, C. Zhong, P. Zhao, X. Mu, Combining feature importance and neighbor node interactions for cold start recommendation, *Eng. Appl. Artif. Intell.*, **112** (2022), 104864. https://doi.org/10.1016/j.engappai.2022.104864
22. H. Wu, C. W. Wong, J. Zhang, Y. Yan, D. Yu, J. Long, et al., Cold-start next-item recommendation by user-item matching and auto-encoders, *IEEE Trans. Serv. Comput.*, (2023). https://doi.org/10.1109/TSC.2023.3237638
23. D. Cai, S. Qian, Q. Fang, J. Hu, C. Xu, User cold-start recommendation via inductive heterogeneous graph neural network, *ACM Trans. Inf. Syst.*, **41** (2023), 1−27. https://doi.org/10.1145/3560487
24. I. Rehman, W. Ali, Z. Jan, Z. Ali, H. Xu, J. Shao, CAML: Contextual augmented meta-learning for cold-start recommendation, *Neurocomputing*, **533** (2023), 178−190. https://doi.org/10.1016/j.neucom.2023.02.051
25. S. Feng, G. Cong, B. An, Y. M. Chee, Poi2vec: Geographical latent representation for predicting future visitors, in *2017 Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, (2017), 102−108. https://doi.org/10.1609/aaai.v31i1.10500
26. R. Li, Y. Shen, Y. Zhu, Next point-of-interest recommendation with temporal and multi-level context attention, in *Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM)*, (2018), 1110−1115. https://doi.org/10.1109/ICDM.2018.00144
27. Y. Luo, Q. Liu, Z. Liu, Stan: Spatio-temporal attention network for next location recommendation, *in 2021 Proceedings of the web conference (WWW)*, (2021), 2177−2185. https://doi.org/10.48550/arXiv.2102.04095
28. C. Ma, Y. Zhang, Q. Wang, X. Liu, Point-of-interest recommendation: Exploiting self-attentive autoencoders with neighbor-aware influence, *in Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM)*, (2018), 697−706. https://doi.org/10.48550/arXiv.1809.10770
29. M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, et al., Semeval-2016 task 5: Aspect based sentiment analysis, in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval)*, (2016), 19−30. https://doi.org/10.18653/v1/S16-1002
30. H. A. Rahmani, M. Aliannejadi, S. Ahmadian, M. Baratchi, M. Afsharchi, F. Crestani, LGLMF: local geographical based logistic matrix factorization model for POI recommendation, *in Proceedings of the Information Retrieval Technology: 15th Asia Information Retrieval Societies Conference*, (2020), 66−78. https://doi.org/10.48550/arXiv.1909.06667
31. P. Han, Z. Li, Y. Liu, P. Zhao, J. Li, H. Wang, et al., Contextualized point-of-interest recommendation, *in Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, **344** (2021), 2484−2490. https://doi.org/10.24963/ijcai.2020/344
32. J. Fu, R. Gao, Y. Yu, J. Wu, J. Li, D. Liu, et al., Contrastive graph learning long and short-term interests for POI recommendation, *Expert Syst. Appl.*, **238** (2024), 121931. https://doi.org/10.1016/j.eswa.2023.121931
33. F. Mo, H. Yamana, EPT-GCN: Edge propagation-based time-aware graph convolution network for POI recommendation, *Neurocomputing*, **543** (2023), 126272. https://doi.org/10.1016/j.neucom.2023.126272