

Article

A Next POI Recommendation Based on Graph Convolutional Network by Adaptive Time Patterns

Jiang Wu ¹, Shaojie Jiang ^{1,*}  and Lei Shi ² ¹ Informatization Construction and Management Office, Sichuan University, Chengdu 610041, China² State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing 100024, China

* Correspondence: jiangsj@scu.edu.cn

Abstract: Users' activities in location-based social networks (LBSNs) can be naturally transformed into graph structural data, and more advanced graph representation learning techniques can be adopted for analyzing user preferences, which benefits a variety of real-world applications. This paper focuses on the next point-of-interest (POI) recommendation task in LBSNs. We argue that existing graph-based POI recommendation methods only consider user preferences from several individual contextual factors, ignoring the influence of interactions between different contextual information. This practice leads to the suboptimal learning of user preferences. To address this problem, we propose a novel method called hierarchical attention-based graph convolutional network (HAGCN) for the next POI recommendation, a technique which leverages graph convolutional networks to extract the representations of POIs from predefined graphs via different time patterns and develops a hierarchical attention mechanism to adaptively learn user preferences from the interactions between different contextual data. Moreover, HAGCN uses a dynamic preference estimation to precisely learn user preferences. We conduct extensive experiments on real-world datasets to evaluate the performance of HAGCN against representative baseline models in the field of next POI recommendation. The experimental results demonstrate the superiority of our proposed method on the next POI recommendation task.

Keywords: graph representation learning; contextual information; graph convolutional network; hierarchical attention; POI recommendation



Citation: Wu, J.; Jiang, S.; Shi, L. A Next POI Recommendation Based on Graph Convolutional Network by Adaptive Time Patterns. *Electronics* **2023**, *12*, 1241. <https://doi.org/10.3390/electronics12051241>

Academic Editors: Rongyao Hu, Tong Liu and Jiong Wu

Received: 13 February 2023

Revised: 28 February 2023

Accepted: 2 March 2023

Published: 4 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recent years have witnessed the exponential growth of social networks [1]. Location-based social networks (LBSNs), amongst the typical social networks, play an import role in people's daily life. A growing number of people are share their life experiences on the social platforms based on LBSNs. For instance, a user could share their favorite restaurant on the social platform for making friends who have the similar preferences. Understanding user preferences from their activities can not only can improve the service of social platform for users, but also brings economic value to the platform. Hence, one of the most important applications of LBSNs is the recommender system [2]. The advantage of recommender systems in LBSNs is to their ability to help users explore some new places that they have not visited previously according to their check-in habits. As a fundamental task of recommender systems in LBSNs [3], next POI recommendation [4–6] has attracted many researchers.

The goal of next POI recommendation is to recommend several POIs based on the previous check-in activities of the target users. Due to the complex contextual information used in POI recommendations, this recommendation task is more challenging than traditional recommendation tasks, such as music recommendation and movie recommendation. One of the key challenges in next POI recommendation is to preserve the personalized user

preferences [7] for different users. To achieve this goal, many studies have proposed the extraction of user preferences from check-in history.

In this field of next POI recommendation, existing methods could be divided into two main categories, namely Markov chain-based method [8–10] and deep learning based-method [11–13]. The former utilizes the Markov chain-based strategies, such as matrix factorization, to calculate the transition matrix and further predict the next check-in places of users. Conversely, the latter leverages deep learning-based techniques, such as recurrent neural network and its variants [14–19], to learn user preferences. Despite their effectiveness, these methods treat the check-in activities of users as sequences, which is inefficient in capturing the interaction information of different POIs.

To overcome this gap, recent studies have tended to transform the check-in records of users into graph structural data and further leverage graph representation learning-based techniques, such as graph convolutional network [20], to learn user preference. The graph structure data are suitable for analyzing the habits of user check-in behaviors since the check-in records could be naturally transformed into a bipartite graph, which describes the interactions of users and POIs. Based on this operation, many researchers have developed various graphs to explore the user preference. Nevertheless, most existing graph-based methods for next POI recommendation only exploit one type of specific contextual information, such as fixed timestamps for temporal factors, which is inefficient in learning dynamic and complex user preferences. Intuitively, user preference can vary under interactions between different types of contextual information.

In light of the above discussions, in this paper, we propose a novel graph structural data-based method for next POI recommendations named the dynamic temporal-based graph convolutional network (HAGCN), which leverages predefined graphs to learn dynamic user preferences via advanced graph representation learning-based techniques. Specifically, HAGCN consists of three main parts: the graph convolutional module, the attention-based fusion module and the dynamic preference estimation module. The first module is to utilize graph convolutional layer to extract the representations of POIs from the constructed graphs. Then, the attention-based fusion module develops a dual attention layer to adaptively fuse the representations under different contextual information. In this way, the influence of interactions between different context could be carefully preserved. Finally, HAGCN proposes a dynamic preference estimation to capture the dynamic user preferences according to the previous check-in activities.

The main contributions of this paper are summarized as follows:

- We propose a novel model, known as HAGCN, for the next POI recommendation task, which can effectively learn user preferences from the check-in activities of users.
- We develop a dual attention layer to adaptively learn the representations of POIs from different contexts and their interactions.
- We apply a dynamic preference estimation strategy to extract the user preferences from the learned POIs' presentations and previous check-in activities.
- We conduct extensive experiments on two real-world datasets. The experimental results show that our proposed HAGCN outperforms powerful baselines, demonstrating the superiority of HAGCN.

2. Related Work

In this section, we provide a brief review of recent works of next POI recommendation. In general, previous studies in this field could be divided into two categories: Markov chain-based methods and deep learning-based methods. The former leverages the Markov chain mechanism to model the check-in behaviors of users, while the latter extracts the information from sequential factors via deep learning-based methods to learn user preferences. We introduce recent works from these two perspectives.

2.1. Markov Chain-Based Methods

The core assumption of Markov chain-based methods is that the next check-in behavior of a target user is highly related with the previous check-in activities. Thus, the Markov chain mechanism is introduced to model the successive check-in behaviors of users via learning the specific transition probability between two check-in activities. One of the typically used Markov chain-based methods is the matrix factorization-based method [8–10], whose core idea is to construct the transition matrix based on users' check-in records and further leverage the matrix factorization technique to learn user preferences from the transition matrix. He et al. [8] develop a third-rank tensor to model the latent check-in behavior patterns from the successive check-in activities, where a personalized Markov chain is adopted to learn diverse user preferences from the check-in history. Besides directly modeling the relations between users and POIs, some researches [9,10] utilize the category information of POIs to construct a transition matrix. He et al. [9] construct the category-aware transition matrix to preserve the check-in behaviors of users in the POIs' category level. Additionally, a list-wise Bayesian personalized ranking is utilized to learn the user preferences from the transition matrix. Li et al. [10] develop a two-step method that first predicts the POI's category of the next check-in activity and then predicts the specific POIs based on the category information. In this way, the check-in habits of users in the category level can be carefully preserved.

Although Markov chain mechanism is useful for modeling the user preference from the successive check-in behaviors, these models suffer from the high data sparsity of the constructed transition matrix since most of users only have several check-in records. Additionally, the highly sparse transition matrix is hard to deeply extract the user preferences. So that, recent methods begin to leverage more powerful methods (e.g., deep learning-based methods) to model the check-in habits of users.

2.2. Deep Learning-Based Methods

Methods falling into this category leverage various neural network-based methods to model the check-in records of users, such as recurrent neural network (RNN) [11–13] and long short-term memory (LSTM [14]) methods [15–18]. Liu et al. [11] develop an RNN-based method called ST-RNN that can extract local temporal and spatial contexts in each neural network layer to learn user preferences from continuous time interval. Yao et al. [12] utilize RNN model to extract the semantics-aware spatiotemporal transition habits of users to improve the performance of POI recommendation service. Chen et al. [13] first leverage a spatial-temporal topic model to extract the user preferences for POIs and then use a supervised RNN model to predict the next check-in POI for users. Guo et al. [15] consider the influence of data sparsity and propose a LSTM-based model to jointly learn the sequential regularity and the transition regularities of similar POIs. In this way, the impact of data sparsity on model performance could be reduced. Yang et al. [16] utilize LSTM to learn contextual features of POIs from the check-in records and further propose an adaptive convolution network, which is developed to learn the short-term user preferences based on the successive check-in behaviors. Wu et al. [17] develop a LSTM-based model named PLSPL, which can learn the specific long- and short-term preferences for different users via the proposed user-based linear combination unit. In this way, the unique preference of different users can be carefully preserved. Huang et al. [18] leverage the LSTM model to learn spatiotemporal preference of users based on the previous check-in activities and using the attention mechanism to capture the relevant check-in activities in the input spatiotemporal contextual information.

In addition, in relation to the use of applied RNN and its variants to model the user preferences, recent works [19,21–25] focus on introducing graph neural network (GNN) to learn the informative representations of users through a variety graph-based data constructed from check-in history. Li et al. [19] propose a novel spatiotemporal gated graph neural network model based on the attention mechanism to learn the local- and global preference of users. Chen et al. [21] utilize the knowledge graph technique to reduce the

impact of data sparsity and further integrate the spatial–temporal contextual information into graph embeddings to represent the transition patterns for diverse users. Wang et al. [22] propose a novel graph-based spatial dependency modeling (GSD) module to learn the geographical influences depending on distance cost among POIs in successive check-in records. Wang et al. [16] exploit the intrinsic characteristics of POIs and further develop a disentangled representation model based on graph attention networks to capture dynamic user preferences. Kim et al. [24] jointly consider the influence of geographical influence and arrival time of the check-in records and propose a dynamic positional graph neural network (DynaPosGNN) to determine the correlation between arrival time and interaction graphs constructed by check-in histories. Lim et al. [25] construct the POI interaction graphs to explore the influence of spatial–temporal preference by learning neighborhood information via a random walk-based self-attention strategy.

Although GNN-based methods have shown promising performance for next POI recommendation, most existing GNN-based methods fail to extract more informative user preference from the interactions between different contextual information. Thus, we propose HAGCN, which can leverage the attention mechanism to explore the final representations of users from the diverse interaction graphs of different contextual information.

3. Preliminaries

In this section, we provide several key definitions used in this paper.

Definition 1. (POI): A POI is a place with specific functions, such as a restaurant for meals. In this paper, we use $P = \{p_1, \dots, p_m\}$ to represent the set of POIs, where p_m denotes the specific POI with the identity number m . Each POI is associated with several contextual information, such as geographical information which determines the location of the POI. We use l_{p_m} to represent the location of POI p_m , which is described by longitude and latitude in this paper.

Definition 2. (Check-in activity): A check-in activity represents that a user has visited a POI. In this paper, we use a triplet $c_{u,i} = (u, p, t)$ to represent the i -th check-in activity, where u denotes the user, p denotes the POI, and t denotes the timestamp of this check-in activity. We use $C_u = \{c_{u,1}, \dots, c_{u,i}\}$ to represent recent i check-in records of user u .

Definition 3. (Time pattern): The time pattern means the treatment for the check-in timestamp. Check-in records under different time patterns reflect the diverse habits of users. For instance, if we use hours to represent the time pattern, we will observe that a user tends to visit a bar at 12 at night. If we use days (weekdays and weekends) to represent time pattern, we may find that a user tends to visit a restaurant at weekends. As a result, a single time pattern is inefficient in exploring complex user preferences of check-in habits. In this paper, we utilize two type of time patterns, called hours pattern T_h and days pattern T_d , to capture the habits of users' check-in activities. T_h divides the check-in timestamps into 24 h and T_d treats all timestamp as weekdays and weekends. Note that the setting of the time pattern is flexible, which means we can introduce more suitable time patterns to analyze users' check-in activities.

Definition 4. (Temporal interaction graph): A temporal interaction graph $G_T = (V_T, E_T)$ denotes the interactions between POIs under a specific time pattern, where G_T represents the interaction graph under the time pattern T , V_T denotes the node set of G_T which is consisted of POIs, and E_T denotes the edge set of G_T . The edge of G_T represents that two POIs have been visited at the same timestamp. In this paper, we derive two temporal interaction graphs, G_{T_h} and G_{T_d} , from the check-in records of users under the time pattern T_h and T_d , respectively.

Definition 5. (Geographical interaction graph): A geographical interaction graph $G_g = (V_g, E_g)$ describes the similarity between POIs on the geographical influence, where V_g denotes the node set of G_g and E_g denotes the edge set of G_g . The edge of G_g represents that two POIs belong to the same district.

Definition 6. (Top-k next POI recommendation): Given the recent check-in records C_u of user u , the goal of the Top-k next POI recommendation is to generate a POI recommendation list $\{p_1, \dots, p_k\}$ that the user may have interest to visit at the future time t .

The notations used in this paper are summarized in Table 1.

Table 1. Notations and their descriptions.

Notations	Descriptions
P	the set of POIs
U	the set of users
C	the check-in records
T	the time pattern
G_T	temporal interaction graph
G_g	geographical interaction graph
$H^{G_{T_h}}, H^{G_{T_d}}, H^{G_g}$	the representations of nodes from G_{T_h}, G_{T_d}, G_g
H^{hg}, H^{dg}	the interaction representations from G_{T_h}, G_{T_d}, G_g
H^P	the final representations of POIs
H^U	the representations of user preferences

4. Dynamic Temporal-Based Graph Convolutional Network

In this section, we detail our proposed dynamic temporal-based graph convolutional network (HAGCN) for next POI recommendation. This contains three main modules: graph convolutional module, attention-based fusion module and dynamic preference estimation module. The overall framework is shown in Figure 1.

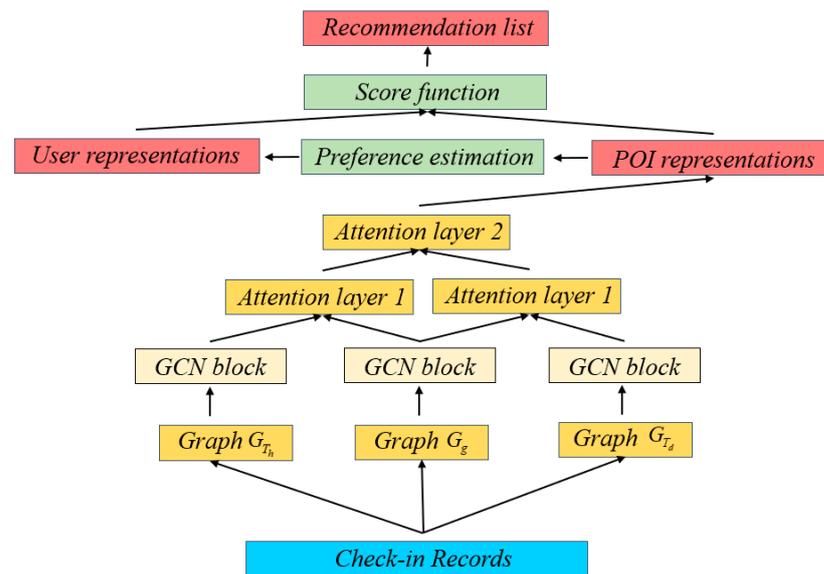


Figure 1. The overall of HAGCN.

4.1. Graph Convolutional Module

As mentioned before, the graph structural data are naturally suitable for analyzing the check-in behaviors of users. Thus, we first transform the check-in history into graph structural data to describe the interactions between POIs under different context. Then, we leverage the graph convolutional network to learn the representations of POIs.

For constructing the graph structural data, we obey the aforementioned Definition 4 and Definition 5. Specifically, we construct three graphs G_{T_h} , G_{T_d} and G_g to represent the interactions between POIs under varying temporal and geographical contexts. Note that constructing the graph is a flexible operation. Definition 4 and Definition 5 are general settings. The interactions between POIs can be redefined according to specific

requirements. In this paper, we study the influence of temporal and geographical context on user preferences. Additionally, we leave investigating more contextual information, such as semantic context, to future studies. After constructing the graph structural data, we focus on learning the informative representations of POIs from these constructed graphs. In this paper, we utilize the graph convolutional network to achieve this goal.

The graph convolutional network (GCN) [20] consists of several graph convolutional layers. Each graph convolutional layer is defined as follows:

$$\mathbf{H}^{(l+1)} = \sigma(\hat{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}) \tag{1}$$

where $\mathbf{H}^{(l+1)}$ and $\mathbf{H}^{(l)}$ denote the representations in the $(l + 1)$ -th and l -th graph convolutional layer, respectively. $\hat{\mathbf{A}}$ denotes the normalized adjacency matrix of the input graph. $\mathbf{W}^{(l)}$ denotes the parameter matrix of the l -th graph convolutional layer. Additionally, $\sigma(\cdot)$ denotes the nonlinear activation function, such as RELU.

For each constructed graph, we develop a GCN block with two graph convolutional layers to learn the representations of POIs. As a result, we obtain the representations of nodes $\mathbf{H}^{G_{T_h}} \in \mathbb{R}^{m \times d}$, $\mathbf{H}^{G_{T_d}} \in \mathbb{R}^{m \times d}$ and $\mathbf{H}^{G_g} \in \mathbb{R}^{m \times d}$ from the interaction graphs G_{T_h} , G_{T_d} and G_g respectively.

4.2. Attention-Based Fusion Module

Previous studies [14,15] reveal that user preference varies with different contextual information. This observation indicates that considering the influence of multiple contexts leads to discovery of the informative user preference. Although existing methods aim to extract user preferences under different context factors, such as temporal factors and geographical factors, they ignore the influence of the interaction between different context factors on user preferences, leading to the obtention of suboptimal user preferences.

To overcome this limitation, we develop an attention-based fusion module that adaptively learns the representations from different interactions between multi-contextual information. Specifically, we consider the interactions between different time patterns and geographical factors via the attention mechanism in this paper. For this purpose, we propose dual attention layer to adaptively adjust the contribution of different contextual information. The first attention layer is used to learn the interaction representations from different contextual information. Additionally, the second attention layer is developed to learn the final representations from the extracted interaction representations.

We first learn the interaction representations based on $\mathbf{H}^{G_{T_h}}$ and \mathbf{H}^{G_g} . We leverage the learnable weight layer to obtain the attention weights:

$$\mathbf{a}^{hg} = (\mathbf{H}^{G_{T_h}} \parallel \mathbf{H}^{G_g})\mathbf{W}^{hg} \tag{2}$$

where \parallel denotes the concatenation operation, $\mathbf{W}^{hg} \in \mathbb{R}^{2d \times 2}$ denotes the weight parameter matrix of the linear layer and $\mathbf{a}^{hg} \in \mathbb{R}^{m \times 2}$ denotes the attention matrix. Then, we can leverage the weighted aggregation strategy to obtain the interaction representations as follows:

$$\mathbf{H}^{hg} = \mathbf{a}_0^{hg} \cdot \mathbf{H}^{G_{T_h}} + \mathbf{a}_1^{hg} \cdot \mathbf{H}^{G_g} \tag{3}$$

where $\mathbf{H}^{hg} \in \mathbb{R}^{m \times d}$ denotes representations extracted from the interaction graphs, G_{T_h} and G_g . In this way, we can adaptively learn the representations under different contextual information. Additionally, we can utilize the same strategy to obtain the representations from G_{T_d} and G_g as follows:

$$\mathbf{a}^{dg} = (\mathbf{H}^{G_{T_d}} \parallel \mathbf{H}^{G_g})\mathbf{W}^{dg} \tag{4}$$

$$\mathbf{H}^{dg} = \mathbf{a}_0^{dg} \cdot \mathbf{H}^{G_{T_d}} + \mathbf{a}_1^{dg} \cdot \mathbf{H}^{G_g} \tag{5}$$

where $\mathbf{H}^{dg} \in \mathbb{R}^{m \times d}$ denotes representations extracted from the interaction graphs G_{T_d} and G_g .

After extracting the interaction representations, we need to determine the final representations of POIs. As mentioned before, different contextual information contributes differently to the final representations. To capture the contributions of different contexts, we also develop the attention layer to adaptively fuse the interaction representations to obtain the final representations:

$$\mathbf{H}^P = \psi((\mathbf{H}^{con}(\mathbf{H}^{con})^T) \cdot (\mathbf{H}^{con}\mathbf{W}^{con})) \quad (6)$$

where $\mathbf{H}^{con} = (\mathbf{H}^{hg} || \mathbf{H}^{dg})$, $\psi(\cdot)$ denotes the softmax function and \mathbf{W}^{con} denotes the projection layer. \mathbf{H}^P denotes the final presentations of POIs. In this way, we can adaptively fuse the representations from different contextual information.

4.3. Dynamic Preference Estimation Module

Estimating the user preferences is also an important part of analyzing the check-in habits of users. Intuitively, user preferences vary at different timestamps. Additionally, the influences of different check-in records are different. To capture dynamic user preferences, we propose a time-aware estimation strategy that adjusts the influence of history records according to the check-in timestamp. Specifically, for each check-in record in check-in history, we leverage the following method to estimate user preferences:

$$\mathbf{H}_u^U = \sum_{c_{u,i} \in C_u} \exp^{-(t_i - t_0)} \cdot \mathbf{H}_{p_i}^P \quad (7)$$

where t_i denotes the check-in timestamp and t_0 denotes the current timestamp. \mathbf{H}_u^U denotes the representations of user preferences. In this way, we can capture the dynamic user preferences among different timestamps.

4.4. Model Training

As for model training stage, we adopt the widely used learning method, known as Bayesian personalized ranking (BPR) [26–28], to learn the model parameters. We first define the ranking score function to calculate the preference scores of unvisited POIs for each user. For each pair (u, p) , we have the following:

$$s_{u,p} = \mathbf{H}_u^U \cdot (\mathbf{H}_p^P)^T \quad (8)$$

Based on Equation (8), the loss function of HAGCN is as follows:

$$Loss = -\frac{1}{|U|} \sum_{u \in U} \sum_{(p_p, p_n) \in C_u} \ln \rho(s_{u,p_p} - s_{u,p_n}) \quad (9)$$

where U denotes the set of users, p_p denotes the positive POI that appears in check-in history C_u of user u , and p_n denotes the negative POIs that have not appeared in C_u . Additionally, $\rho(\cdot)$ denotes the sigmoid function. By minimizing the loss function, we can utilize the stochastic gradient descent (SGD) method to learn the model parameters. The overall learning process is reported in Algorithm 1.

Algorithm 1: the learning process of HAGCN

Input: Check-in records: C ; Max training epoch T .

Output: Model parameters: θ .

1. Constructing the interaction graphs based on C according the Definition 4 and Definition 5;
2. Calculating the interaction representations of POIs based on Equations (3) and (5);
3. Calculating the final representations of POIs based on Equation (6);
4. Calculating the user preferences based on Equation (7);
5. for epoch in T do:
6. Calculating the BPR loss function based on Equation (9);
7. Learning model parameters θ with SGD;
8. end for
9. Return model parameters θ .

5. Experiments

In this section, we first introduce the experimental settings of this paper, including datasets, baseline models and evaluation metric. Then, we provide the experimental results of model comparison, parameter analysis and ablation study. Finally, we provide the corresponding analysis for each experiment.

5.1. Dataset

Following the method outlined in a recent study [17], we adopt two real-world datasets collected from the famous location based social network Foursquare, called NYC and TKY. NYC contains the check-in records in New York from Apr. 2012 to Feb. 2013, while TKY consists of check-in records which are from Tokyo in the same period of NYC. Each check-in record contains the information of user, POI, location and timestamp. We also follow the data preprocess operation in [17] that divide the check-in records into several sessions for each user, which meets the task of next POI recommendation. After data preprocessing, we split the dataset into training set, validation set and test set for model training, validation and testing. Specifically, we select the first 70 percent check-in records as a training set for training the model, then 10 percent as a validation set for tuning the model parameters, and the rest as a test set for evaluating the model performance. The statistics of the used datasets are summarized in Table 2.

Table 2. Statistics of datasets.

Dataset	#Users	#POIs	#Sessions
NYC	1121	38,643	11,433
TKY	2134	62,003	29,131

5.2. Baseline Model

In this paper, we choose the following models for model comparison from three perspectives, Markov chain-based method, RNN-based method and graph-based method:

TAD-FPMC [10]: TAD-FPMC is a tensor factorization-based method that first predicts the POI category and then generates the recommendation POIs based on the prediction results outlined in the previous step.

ARNN [15]: ARNN first leverages a random walk-based method to assess the influence of location neighbors and further utilizes the RNN-based model to extract the sequential regularity of all users.

PLSPL [17]: PLSPL is an RNN-based method that develops a user-based linear combination unit to learn the dynamic preference for different users.

STKGRc [21]: STKGRc derives the user preference from the constructed knowledge graphs via a novel spatial-temporal transfer relation.

STP-UDGAT [25]: STP-UDGAT is a graph attention network-based method that leverages the attention mechanism to learn user preference from the spatial-temporal-preference neighborhoods.

TAD-FPMC is a Markov chain-based method. ARNN and PLSPL are RNN-based methods. STKGRec and STP-UDGAT are graph-based methods.

5.3. Evaluation Metric

For evaluating the model performance on the task of next POI recommendation, we adopt two widely used evaluation metrics, Precision@ k [17] and MAP@ k [17], where k denotes the length of the POI recommendation list. In this paper, we vary k in {5, 10, 15, 20}. Precision@ k is used to indicate the rate of the ground truth POI appears in the recommendation list. A higher value of Precision@ k means that model can predict more POIs correctly. Additionally, MAP@ k is used to the quality of the recommendation list. A higher value means that the ground truth POI is higher in the recommendation list. The calculations of these evaluation metrics are performed as follows:

$$\text{Precision@}k = \frac{1}{n} \sum_{i=1}^n \frac{|D_{rec}^i \cap D_{vis}^i|}{|D_{vis}^i|} \quad (10)$$

$$\text{MAP@}k = \frac{1}{n} \sum_{i=1}^n \frac{|D_{rec}^i \cap D_{vis}^i|}{rank} \quad (11)$$

where n denotes the number of training items, D_{rec}^i denotes the recommendation list, D_{vis}^i denotes the visited list and $rank$ denotes the position of the correctly recommended POI in the ranked list [17].

5.4. Performance Comparasion

To evaluate the model performance on the task of next POI recommendation, we run each model on 2 datasets with 10 random initializations and report the average results. The experimental results are summarized in Tables 3–6.

Table 3. Model performance on NYC in terms of Precision.

Model	Precision@5	Precision@10	Precision@15	Precision@20
TAD-FPMC	0.218	0.256	0.283	0.332
ARNN	0.256	0.293	0.312	0.367
PLSPL	0.292	0.356	0.391	0.423
STKGRec	0.289	0.349	0.379	0.419
STP-UDGAT	0.295	0.378	0.412	0.463
HAGCN	0.315	0.386	0.434	0.487

Table 4. Model performance on NYC in terms of MAP.

Model	MAP@5	MAP@10	MAP@15	MAP@20
TAD-FPMC	0.142	0.156	0.165	0.173
ARNN	0.154	0.162	0.176	0.184
PLSPL	0.191	0.201	0.211	0.222
STKGRec	0.183	0.196	0.204	0.212
STP-UDGAT	0.199	0.207	0.217	0.231
HAGCN	0.208	0.219	0.225	0.243

Table 5. Model performance on TKY terms of Precision.

Model	Precision@5	Precision@10	Precision@15	Precision@20
TAD-FPMC	0.223	0.263	0.296	0.342
ARNN	0.262	0.303	0.326	0.378
PLSPL	0.301	0.364	0.404	0.443
STKGRc	0.297	0.351	0.392	0.438
STP-UDGAT	0.308	0.385	0.426	0.477
HAGCN	0.321	0.399	0.449	0.492

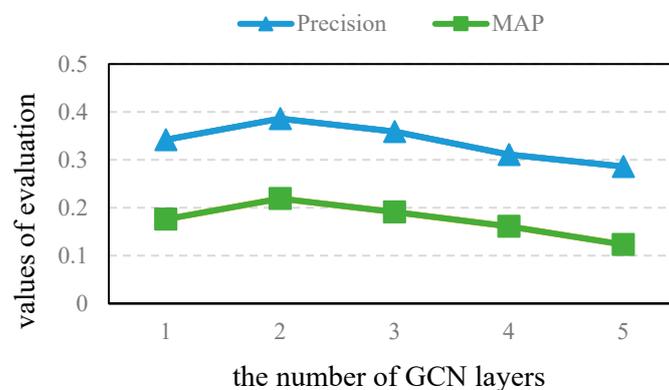
Table 6. Model performance on TKY terms of MAP.

Model	MAP@5	MAP@10	MAP@15	MAP@20
TAD-FPMC	0.146	0.164	0.175	0.186
ARNN	0.159	0.169	0.186	0.197
PLSPL	0.196	0.206	0.219	0.228
STKGRc	0.188	0.203	0.211	0.217
STP-UDGAT	0.206	0.215	0.226	0.240
HAGCN	0.213	0.229	0.239	0.252

Based on the above experimental results, we can make the following observations: (1) HAGCN achieves the best performance on two datasets, which indicates that extracting model performance from different time patterns can improve the model performance; (2) graph-based models have obtained better performance than other models, showing that transforming the users' check-in records into graph structural data is helpful for capturing the check-in habits of users; (3) TAD-FPMC performs the worst on two datasets, which demonstrates that introducing multi-contextual information is crucial to learning user preferences, especially since TAD-FPMC only considers the influence of temporal factors.

5.5. Parameter Analysis

In this section, we provide additional experiments for investigating the influence of two core parameters, the number of GCN layers and the number of hidden dimensions, on the model's performance. We fix the length of recommendation list to 10 and vary the number of layers and hidden dimensions to observe the changes of the model performance on NYC dataset. Specifically, we vary the number of GCN layers in $\{1, 2, \dots, 5\}$ and the number of hidden dimensions in $\{32, 64, \dots, 512\}$. The experimental results are shown in Figures 2 and 3.

**Figure 2.** The influence of GCN layers on NYC dataset.

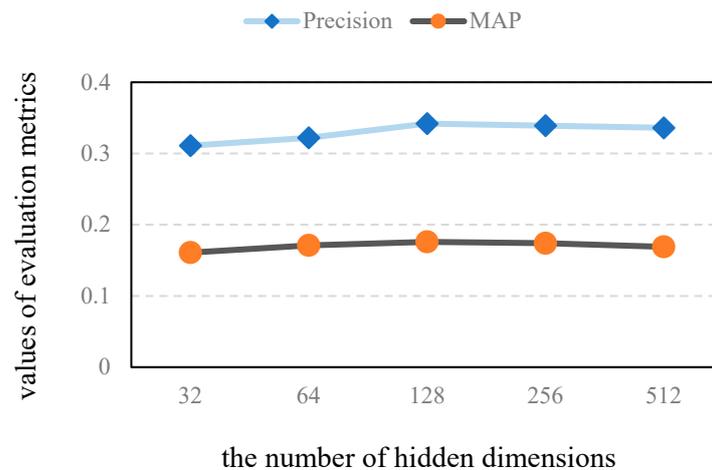


Figure 3. The influence of hidden dimensions on NYC dataset.

From Figure 2 we can observe that the model performance is sensitive to the number of GCN layers. This is because the model will suffer from the over-smoothing problem [29,30] when increases the number of GCN layers. Thus, the large number of GCN layer could hurt the model performance.

As for the influence of the hidden dimension, we can observe that the model performance changes slightly when we increase the number of hidden dimensions from Figure 3. This phenomenon indicates that the model performance is not sensitive to the number of hidden dimensions

5.6. Ablation Study

In this section, we study the influence of the different time patterns on the model performance. We propose two variants of our proposed HAGCN, named HAGCN-H and HAGCN-D. In HAGCN-H, the hours pattern is preserved. Additionally, in HAGCN-D, we observe the day pattern. We also set a fixed value 10 to the length of recommendation list and observe the model performance on NYC dataset. The experimental results are summarized in Table 7.

Table 7. The performance of HAGCN and its variants on NYC.

Metric	HAGCN-H	HAGCN-D	HAGCN
Precision	0.373	0.362	0.386
MAP	0.205	0.203	0.219

We can make the following findings from Table 7: (1) HAGCN beats its two variants, showing that learning user preferences through different time patterns is a promising way to increase the model performance; (2) HAGCN-H outperforms HAGCN-D, which is because the hours pattern can preserve more information of users' check-in habits than the day pattern. We can also conclude that our proposed HAGCN can extract more precise user preferences from the check-in records.

6. Conclusions

In this paper, we study one of the most important tasks, next POI recommendation in the location-based social networks. To better understand the check-in habits of different users from the check-in records, we propose a novel model named dynamic temporal-based graph convolutional network (HAGCN). The key idea of HAGCN is to extract user preferences under different time patterns. For this purpose, HAGCN first constructs several interaction graphs to represent the relations of POIs under different contextual information. Next HAGCN leverages GCN-based neural network modules to learn the representations of

POI in these constructed graphs. Then, for different users, HAGCN develops an attention-based fusion module to adaptively learn the user preferences. Moreover, a dynamic preference estimation is proposed to preserve the influence of temporal factors on user preferences. We present the results of extensive experiments conducted on two real-world datasets. The experimental results show that our proposed HAGCN consistently outperforms several competitive methods on next POI recommendation task.

As extensive experimental results from real-world datasets have proved the promising performance of HAGCN on the task of next POI recommendation, HAGCN could be applied in the POI recommendation service of location-based mobile applications in LBSNs, such as Foursquare and Yelp, to help users explore new and interesting POIs.

Author Contributions: Conceptualization, J.W. and S.J.; Data curation, S.J. and L.S.; Formal analysis, L.S.; Funding acquisition, L.S.; Methodology, J.W.; Project administration, S.J.; Software, S.J.; Writing—original draft, J.W. and S.J.; Writing—review and editing, J.W. and L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Key Research and Development Program of China (No.2022YFC3302103), the Fundamental Research Funds for the Central Universities (No. CUC220C011, CUC22GZ038) and Humanities and Social Science Research Project of Hebei Education Department (SZ2023161).

Data Availability Statement: The dataset can be downloaded from the link <https://sites.google.com/site/yangdingqi/home/foursquare-dataset> (accessed on 1 March 2023).

Conflicts of Interest: The authors of this publication declare there is no conflict of interest.

References

1. Shi, L.; Du, J.; Cheng, G.; Liu, X.; Xiong, Z.; Luo, J. Cross-media search method based on complementary attention and generative adversarial network for social networks. *Int. J. Intell. Syst.* **2021**, *37*, 4393–4416. [\[CrossRef\]](#)
2. Hwangbo, Y.; Lee, K.J.; Jeong, B.; Park, K.Y. Recommendation system with minimized transaction data. *Data Sci. Manag.* **2021**, *4*, 40–45. [\[CrossRef\]](#)
3. Shi, L.; Luo, J.; Zhu, C.; Kou, F.; Cheng, G.; Liu, X. A survey on cross-media search based on user intention understanding in social networks. *Inf. Fusion* **2023**, *91*, 566–581. [\[CrossRef\]](#)
4. Ji, W.; Meng, X.; Zhang, Y. SPATM: A social period-aware topic model for personalized venue recommendation. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3997–4010. [\[CrossRef\]](#)
5. Hu, X.; Xu, J.; Wang, W.; Li, Z.; Liu, A. A graph embedding based model for fine-grained POI recommendation. *Neurocomputing* **2021**, *428*, 376–384. [\[CrossRef\]](#)
6. Ji, W.; Meng, X.; Zhang, Y. STARec: Adaptive learning with spatiotemporal and activity influence for POI recommendation. *ACM Trans. Inf. Syst. (TOIS)* **2021**, *40*, 1–40. [\[CrossRef\]](#)
7. Shi, L.; Song, G.; Cheng, G.; Liu, X. A user-based aggregation topic model for understanding user's preference and intention in social network. *Neurocomputing* **2020**, *413*, 1–13. [\[CrossRef\]](#)
8. He, J.; Li, X.; Liao, L.; Song, D.; Cheung, W. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
9. He, J.; Li, X.; Liao, L. Category-aware next point-of-interest recommendation via listwise bayesian personalized ranking. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; Volume 17, pp. 1837–1843.
10. Li, X.; Jiang, M.; Hong, H.; Liao, L. A time-aware personalized point-of-interest recommendation via high-order tensor factorization. *ACM Trans. Inf. Syst. (TOIS)* **2017**, *35*, 1–23. [\[CrossRef\]](#)
11. Liu, Q.; Wu, S.; Wang, L.; Tan, T. Predicting the next location: A recurrent model with spatial and temporal contexts. In Proceedings of the Thirtieth AAAI conference on artificial intelligence, Phoenix, AZ, USA, 12–17 February 2016.
12. Yao, D.; Zhang, C.; Huang, J.; Bi, J. Serm: A recurrent model for next location prediction in semantic trajectories. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 2411–2414.
13. Chen, M.; Li, W.Z.; Qian, L.; Lu, S.L.; Chen, D.X. Next POI recommendation based on location interest mining with recurrent neural networks. *J. Comput. Sci. Technol.* **2020**, *35*, 603–616. [\[CrossRef\]](#)
14. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Guo, Q.; Sun, Z.; Zhang, J.; Theng, Y.-L. An attentional recurrent neural network for personalized next location recommendation. In Proceedings of the AAAI Conference on artificial intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 83–90.
16. Yang, Q.; Hu, S.; Zhang, W.; Zhang, J. Attention mechanism and adaptive convolution actuated fusion network for next POI recommendation. *Int. J. Intell. Syst.* **2022**, *37*, 7888–7908. [\[CrossRef\]](#)

17. Wu, Y.; Li, K.; Zhao, G.; Qian, X. Personalized long-and short-term preference learning for next POI recommendation. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 1944–1957. [[CrossRef](#)]
18. Huang, L.; Ma, Y.; Wang, S.; Liu, Y. An attention-based spatiotemporal lstm network for next poi recommendation. *IEEE Trans. Serv. Comput.* **2019**, *14*, 1585–1597. [[CrossRef](#)]
19. Li, Q.; Xu, X.; Liu, X.; Chen, Q. An Attention-Based Spatiotemporal GGNN for Next POI Recommendation. *IEEE Access* **2022**, *10*, 26471–26480. [[CrossRef](#)]
20. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
21. Chen, W.; Wan, H.; Guo, S.; Huang, H.; Zheng, S.; Li, J.; Lin, S.; Lin, Y. Building and exploiting spatial–temporal knowledge graph for next POI recommendation. *Knowl. Based Syst.* **2022**, *258*, 109951. [[CrossRef](#)]
22. Wang, Z.; Zhu, Y.; Zhang, Q.; Liu, H.; Wang, C.; Liu, T. Graph-enhanced spatial-temporal network for next POI recommendation. *ACM Trans. Knowl. Discov. Data (TKDD)* **2022**, *16*, 1–21. [[CrossRef](#)]
23. Wang, Z.; Zhu, Y.; Liu, H.; Wang, C. Learning Graph-based Disentangled Representations for Next POI Recommendation. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, 11–15 July 2022; pp. 1154–1163.
24. Kim, J.; Jeong, S.; Park, G.; Cha, K.; Suh, I.; Oh, B. Dynaposgnn: Dynamic-positional gnn for next poi recommendation. In Proceedings of the 2021 International Conference on Data Mining Workshops (ICDMW), Auckland, New Zealand, 7–10 December 2021; pp. 36–44.
25. Lim, N.; Hooi, B.; Ng, S.K.; Wang, X.; Goh, Y.L.; Weng, R.; Varadarajan, J. STP-UDGAT: Spatial-temporal-preference user dimensional graph attention network for next POI recommendation. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; pp. 845–854.
26. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. *arXiv* **2012**, arXiv:1205.2618.
27. He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; Wang, M. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, China, 25–30 July 2020; pp. 639–648.
28. Wu, S.; Sun, F.; Zhang, W.; Xie, X.; Cui, B. Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [[CrossRef](#)]
29. Chen, D.; Lin, Y.; Li, W.; Li, P.; Zhou, J.; Sun, X. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In Proceedings of the AAAI conference on artificial intelligence, New York, USA, 7–12 February 2020; Volume 34, pp. 3438–3445.
30. Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; Weinberger, K. Simplifying graph convolutional networks. In Proceedings of the International conference on machine learning, Long Beach, CA, USA, 10–15 June 2019; pp. 6861–6871.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.