

Article

MTPR: A Multi-Task Learning Based POI Recommendation Considering Temporal Check-Ins and Geographical Locations

Bin Xia ^{1,*} , Yuxuan Bai ¹, Junjie Yin ², Qi Li ¹ and Lijie Xu ¹

¹ Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing 210023, China; 1018041124@njupt.edu.cn (Y.B.); liqics@njupt.edu.cn (Q.L.); ljxu@njupt.edu.cn (L.X.)

² Zhongxing Telecommunication Equipment Corporation, Nanjing 210000, China; junjieyin112358@163.com

* Correspondence: bxia@njupt.edu.cn

Received: 29 June 2020; Accepted: 21 September 2020; Published: 23 September 2020



Abstract: The rapid development of location-based social networks (LBSNs) produces the increasing number of check-in records and corresponding heterogeneous information which bring big challenges of points-of-interest (POIs) recommendation in our daily lives. The emergence of various recommender techniques bridges the gap between the numerous heterogeneous check-ins and the personalized POI recommendation. However, due to the differences between LBSNs and conventional recommendation tasks, besides the user feedback, the spatio-temporal information is also significant to precisely capture the user preferences. In this paper, we propose a multi-task learning model based POI recommender system which exploits a structure of generative adversarial networks (GAN) simultaneously considering temporal check-ins and geographical locations. The GAN-based model is capable of relieving the sparsity of check-in data in POI recommender systems. The temporal check-ins not only present the preference but also show the lifestyle of an individual while the geographical locations describe the active region of users which further filters POIs far from the feasible region. The multi-task learning strategy is capable of combining the information of temporal check-ins and geographical locations to improve the performance of personalized POI recommendation. We conduct the experiments on two real-world LBSNs datasets and the experimental results show the effectiveness of our proposed approach.

Keywords: recommender systems; sequential analysis; attention; generative adversarial networks

1. Introduction

The evolution of mobile phones and wearable devices facilitates the development of location-based social networks (LBSNs), which bridges the gap between the virtual cyberspace and the physical world. LBSNs, such as Foursquare and Gowalla, provide detailed information on massive POIs in our physical world, and users can share their check-in experience based on reviews (e.g., texts and photographs) in the virtual cyberspace. To improve the user satisfaction, the point-of-interest (POI) recommendation is proposed to capture the user preferences based on sufficient information (e.g., check-ins and ratings) and push potential desired POIs to individuals. Therefore, the research of POI recommender systems becomes increasingly popular, and researchers make many efforts on the effectiveness and efficiency of POI recommendation [1–3].

However, the characteristics of location-based social networks also bring significant challenges to current techniques of recommender systems. First, different from the explicit feedback (i.e., ratings) in traditional recommender systems (i.e., books and movies), the check-in records in LBSNs are the

implicit feedback that reflect the user preferences. In other words, high ratings mean definite user preferences to a specific item; however, the amount of check-ins is not relevant to user preferences. Second, although check-in records are much more than ratings due to the characteristics of user behaviors in LBSNs, check-in data is still extremely sparse, which may cause a cold-start problem and make it difficult for models to capture user preferences. Third, it is necessary to simultaneously consider the user preference and geographical influence in POI recommendations, which should be a multi-objective optimization to balancing the trade-off between the user preference and the physical distance. Therefore, there still exist some exciting and challenging tasks that should be resolved in POI recommender systems.

To overcome the challenges mentioned above, many researchers have made lots of efforts on POI recommendations based on innovative techniques. Xia et al. proposed a strategy for relaxing the count of check-ins to several labels distinguishing user preferences and built a classification model for recommending POIs to individuals [4]. Although the problem of implicit feedback is relieved based on the relaxation of preference representation, the extreme situation (e.g., a favorite POI has been visited only once) cannot be considered. To this end, they also proposed an attention-based recurrent neural network for concentrating on the sequential check-in behavior to further relieve the problem of implicit feedback instead of focusing on the count of check-ins on a specific POI [5]. However, due to the sparsity of check-ins, the proposed model in [5] is hard to recommend for POIs that the user has never been to. In addition, there exists much deep learning research applied to the mechanism of word embedding for representing each POI using a continuous vector simultaneously considering the sequential check-ins and the characteristics of POI to improve the performance of POI recommendation [6,7]. However, the sparsity of check-in records also makes these deep learning models challenging for capturing the relevance among massive POIs and users. Therefore, it is significant to relieve the influence of sparse check-in data on the effectiveness of POI recommendations.

To this end, in this paper, we propose a multi-task learning based POI recommendation simultaneously considering the temporal check-ins and the geographical locations. First, the proposed framework exploits a generative adversarial network (GAN) that constructs an adversarial game between a generator (G) and a discriminator (D). Based on the mechanism of adversarial learning, G aims to fit the real data and learn the actual distribution of datasets while D tries to distinguish whether the incoming sample is real or fake (i.e., generated by G). In other words, with the convergence of GAN, the generator can synthesize plausible instances (i.e., check-in records), which is capable of relieving the problem of sparse check-in data. Second, to capture user preferences from implicit feedback, both G and D are designed based on long-short term memory networks (LSTM) capable of mining important information from sequential check-in data instead of constructing a regression task based on the count of check-ins. The personalized temporal check-ins not only present the preference but also show the lifestyle of an individual. In other words, the context of visiting a specific POI is also significant for demonstrating the user preference which can be captured using the recurrent neural network based model. Third, the generator and the discriminator leverage the LSTM to construct a multi-task learning model to fit the next check-in POI and the personalized visiting path based on the longitude and latitude for considering the geographical influence. Due to the generative adversarial network based framework, the extension of check-in records will relieve the problems caused by implicit feedback while the multi-task setup can bring the geographical influence to the POI recommendation. Our contribution can be summarized as below:

- A multi-task learning based GAN is proposed to overcome the limitation of sparse check-in records to the performance of POI recommendations.
- The proposed approach leverages the sequential learning (i.e., POI and physical coordinate) to relieve the problem of implicit feedback and geographical influence simultaneously.
- Extensive experiments are conducted to evaluate the effectiveness of our proposed framework based on two real-world LBSN datasets.

The rest of paper is organized as below: Section 2 introduces the significant work related to our research. Section 3 describes our proposed model in detail. Section 4 shows the experimental results on the two real-world datasets and analyze the comparison between the baselines and our proposed model. Section 5 concludes this paper and shows the interesting future work related to our work.

2. Related Work

POI Recommendation: The difference between the problem of POI recommendation and the recommendation of other types of items is that for the recommendation of POIs the geographical location of the POIs becomes an essential part of the user preferences. Traditional recommender systems (i.e., book recommendation) can provide every item that hits the user preference if the item is not out of stock. However, when a user is looking for a place for dinner after watching a film, it is ridiculous if a POI recommender system pushes a restaurant thousands of miles away from the user ignoring the geographical influence. Therefore, simultaneously considering the geographical influence and preference becomes a popular area that has attracted a lot of research attention. Feng et al. proposed a personalized ranking metric embedding approach to integrating sequential check-ins, preference, and geographical influence. Different from the self-learning POI embedding, the authors utilized the metric embedding to representing each POI in a K-dimensional latent space and leveraged the Euclidean distance to describing the potential transition within the following information [8]. In the following research of Feng et al., they proposed a latent representation model POI2Vec inspired by Word2Vec [6]. Instead of initializing embeddings using the Gaussian distribution, POI2Vec initializes the POI vectors incorporating the geographical influence. Different from their previous work [8], they utilized the relationship between the distance and visiting probability to further represent the POI embeddings. A joint model is then proposed to combine the geographical influence with the user preference and predict the potential visitors for a specific POI. However, in these works, the geographical influence is considered as the physical distance among POIs. Wang et al. redefined the geographical influence as the geo-influence of POI, the susceptibility of POI, and the physical distance, where the geo-influence describes the capacity of POI and the susceptibility describes the propensity of POI [9]. In other words, they do not only consider the physical distance between POIs as the geographical influence but also consider the influence of visiting users. Although the research of geographical influence improves the performance of current POI recommendations, the sparsity of check-ins is also a big challenge where negative sampling can alleviate this issue effectively. Liu et al. proposed geographical information based adversarial learning model named Geo-ALM, which utilized a game setting to generate a few but critical negative samples (i.e., POIs that users have never been to) for improving the effectiveness of POI recommendation [10]. Fu et al. focused on the geographical influence and proposed a region embedding strategy considering the distance and mobility connectivity using the multi-view learning model [11]. Sun et al. combined a nonlocal network for the long-term preference and a geo-dilated recurrent neural network for the short-term preference to address the bias of current techniques in the long-term or short-term preference [12]. Zhou et al. proposed a general adversarial learning based POI recommender system which was inspired by IRGAN using the technique of reinforcement learning to update the generator (i.e., recommender) [13]. Chang et al. was the first attempt to utilize the content-aware embedding to address the problem of representing the characteristics of POIs where the check-in layer and the content layer are combined to capture the embedding of POIs [14]. Bobadilla et al. proposed a generic deep learning architecture to optimizing the collaborative filtering recommender system which is also an effective solution in the task of the POI recommendation [15]. Nguyen et al. also proposed a generic recommender system using the cognitive similarity-based collaborative filtering technique which is capable of addressing the general task of POI recommendation [16]. Compared to the current POI recommendation approaches, our proposed model aims to address the sparsity of check-in records using the mechanism of adversarial learning and taking the geographical influence into account for improving the effectiveness of user and POI embeddings. In addition, the user's preference is fluctuating according to the current situation

(e.g., the current location and the previous check-in POI). Therefore, the mechanism of attention is utilized to adjust the embedding based on the complex contextual information.

Generative Adversarial Networks: Due to the effectiveness of addressing various problems and improving the performance of models, generative adversarial networks (GAN) have been attracting a lot of research attention in recent years. The principle of GAN is to jointly train a generator and a discriminator based on a game setting using the adversarial learning, and one or both of the generators and the discriminator will be leveraged to complete proposed tasks [17]. The utilization of GAN in the task of POI recommendation is limited, and Geo-ALM is a simple try to generate critical negative samples using GAN [10]. However, GAN is a simple form of adversarial learning where the framework of GAN is evolving with the development of machine learning models. GAN was proposed to address the problems in the image processing domain due to the advantage of handling continuous data [17]. The goal of recommender systems is to push the items which hit the preference of users where the items are discrete data. In other words, the gradient between discrete data cannot be calculated. Therefore, traditional GAN cannot be leveraged in the domain of recommender systems. To this end, Wang et al. proposed a GAN-based information retrieval model named IRGAN for transforming the traditional gradient into the policy gradient based reinforcement learning [18]. Although IRGAN is capable of generating recommendations (i.e., discrete data), the contradiction between real and fake data will confuse the discriminator and restrict the performance of the generator. To alleviate this problem, Chae et al. proposed a generic collaborative filtering framework based on GAN named CFGAN, which discarded the softmax layer and transformed the output back to the continuous form [19]. On the one hand, it can calculate the gradient of the continuous data simply. On the other hand, GAN is designed to generate continuous data, which means GAN's performance should be improved. In addition, the goal of the generator is to learn the distribution of real data based on the Kullback–Leibler divergence, which may cause the gradient vanishing and the gradient exploding. To address this issue, Arjovsky et al. proposed the Wasserstein distance, which describes the difference between distributions better than the Kullback–Leibler divergence [20]. Wang et al. proposed a deep adversarial substructured learning framework to learning the representation of mobile user profiling where the proposed framework maintained two components preserved the entire graph and the sub graph of users [21]. Zhang et al. designed a spatial embedding strategy to unifying the inter- and intra-region autocorrelations using a collective graph-regularized dual adversarial learning framework [22]. In addition, Thanh-Tung et al. proposed a zero-centered gradient penalty for improving the stability of GANs and the generalization of the discriminator by pushing it toward the optimal discriminator [23].

Attention: The mechanism of attention, which was called memory networks at the beginning, was proposed to address problems in the domain of natural language processing [24]. For example, when we deal with the task of reading comprehension, we are more concerned about the key sentences, even words to our questions than the irrelevant ones. However, the recurrent neural networks (i.e., long-short term memory networks) consider that the latest words are more significant than any other, and the convolutional neural networks treat each word fairly. Both of their mechanisms restrict the further improvement of performance in similar tasks. To this end, Sukhbaatar et al. proposed an end-to-end memory network with a recurrent attention model, which is an updated version of their previous work [24] to address the task of question answering in the reading comprehension [25]. In the end-to-end memory network, the external memory (i.e., the component of attention) is implemented using a weight vector, which is the standard form of attention. Inspired by the end-to-end memory networks, Xia et al. proposed an attention-based recurrent neural network, which exploited the external memory for matching the user preference and historical check-in data considering the geographical influence overcoming the problem of sequential POI recommendations [5]. However, the form of attention is evolving with the improvement of deep learning theory. Vaswani et al. proposed a generic transformer based solely on a self-attention mechanism and also introduced a new form of attention named the multi-head attention, which was inspired by the multiple kernels in convolutional

neural networks [26]. In addition, Tay et al. proposed a novel architecture using the densely connected attention propagation for the reading comprehension, where the design of the bidirectional attention connector was inspired by ResNet [27,28]. Liu et al. proposed a geographical-temporal awareness hierarchical attention network to capture the subtle POI–POI interactions for improving the performance of models only considering the sequential check-ins [29]. Although the attention mechanisms are changing, the goal of attention is to guide models to concentrate on critical information in data.

3. Methodology

To overcome the aforementioned limitations of current approaches, in this paper, we propose a multi-task learning based POI recommender system simultaneously considering sequential check-ins learning and geographical influence. Figure 1 illustrates the overview of our proposed POI recommender system. The training and predicting process can be summarized as below: (1) The historical check-in records are grouped by individuals where each personal check-in sequence is described using POI ids; (2) The sliding window is applied to split the individual sequential check-ins as real training samples for the train of GAN; (3) The features (i.e., user ID, temporal check-ins, and geographical location) of each sample are extracted from the original dataset and integrated as an input hub; (4) Based on the features in the input hub, an adversarial game is construed where the LSTM-based generator generates fake check-in samples while the LSTM-based discriminator tries to distinguish the real and fake samples; (5) the converged generator is capable of capturing the actual user preferences and provides POI recommendations. In the following sections, we will introduce the module of data preprocessing and generative adversarial networks in detail.

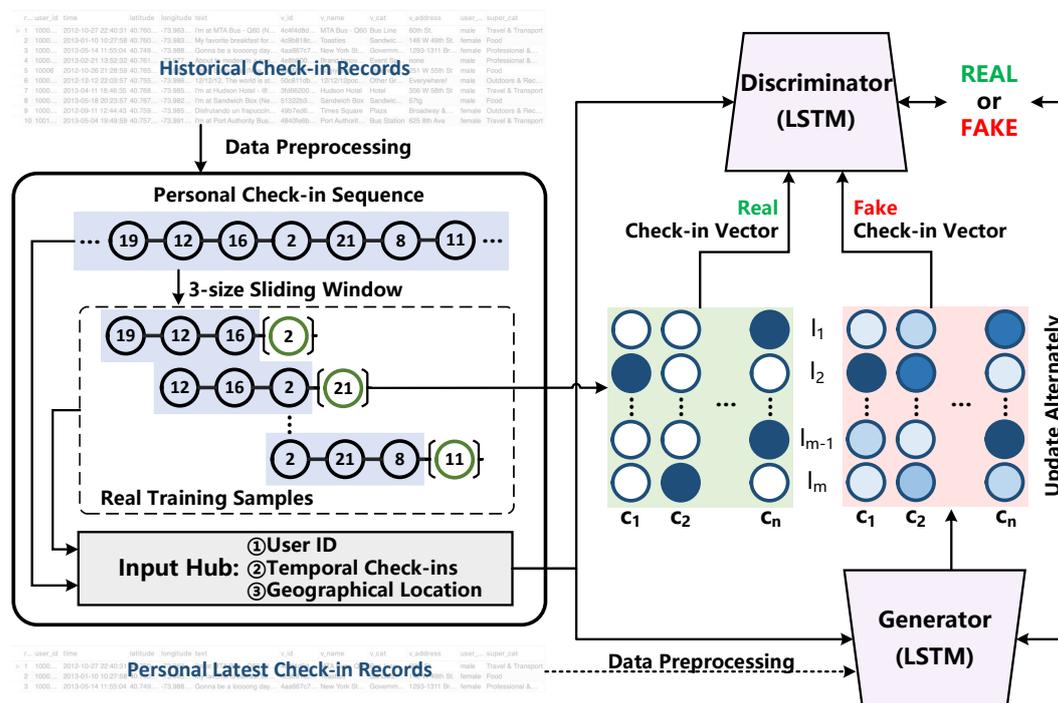


Figure 1. The overview of the proposed GAN based POI recommender system.

3.1. Data Preprocessing

The goal of data preprocessing is to convert the original check-in records to the data, which can be calculated using the machine learning technique. Although lots of useful features exist (e.g., category and review of POI) in each check-in record, the crucial information including user ID, POI ID, check-in timestamps, and location (i.e., latitude and longitude) is considered as the input features to the

following modules. Similar to the majority of previous research, in this paper, the technique of Word2Vec is applied to represent users and POIs using vectors that are capable of capturing the differences among users and POIs [5]. Based on the mechanism of Word2Vec, each word which has a unique id ranged from 0 to the number of words in the bag will be converted to a specific vector according to the id. The vectors, which are the embeddings representing the corresponding words, are usually initialized based on the Gaussian distribution and updated iteratively within the training processing. Similar to Word2Vec, the users and POIs are given unique ids that link to the corresponding embeddings (i.e., vectors) where the User/POI embeddings will be updated within the processing of fitting individual sequential check-ins. In general, the vectors are stored in a matrix, and the line number of the matrix is utilized to extract the corresponding vector (i.e., the representation of the user or POI). Therefore, in this paper, each user or POI is labeled as a unique number starting from 0 instead of using a complex character string (e.g., 4c4f4d8d24edc9b62a5a77bb). The individual temporal check-in records are described as a numerical sequence which is shown in Figure 1.

Due to the consideration of **sequential check-ins** and **geographical influence**, the temporal check-ins and the related geographical location should be extracted from the original datasets. For the influence of **sequential check-ins**, the timestamps are used to sort the corresponding check-in records for constructing the individual sequential check-ins of each user instead of being an explicit feature input to the model. Different from traditional recommender systems (i.e., book and movie), the next POI visit is not only dependent on the user preference but also considering the latest visited POIs. In other words, a user cannot visit multiple restaurants in a short time while it is normal for a user to go shopping among several malls. Therefore, the visiting pattern (i.e., the latest temporal visited POIs) not only describes the typical lifestyle but also represents individual preferences. To this end, the 1-stride fixed-size sliding window is utilized to split the individual check-in sequences into independent samples where the last POI of samples is the actual next visited POI. The previous POIs are considered as a visiting pattern. These samples considered as the real training instances will be transformed to the GAN. For the **geographical influence**, the distance between the last visited POI and the next POI is also a significant reference for users to accept the recommendations. In other words, there exists a trade-off between the distance and the user preference if the time is limited. Therefore, we consider the latitude and longitude as a two-dimensional vector to capture the significance of distance and preference to a specific user. In our previous work [30], we conduct experiments to show that, if the POI embedding is updated without the geographical coordinate information, the POI embeddings cannot be related to the geographical context. Therefore, the geographical influence is significant to the performance of POI recommendations.

Based on the aforementioned steps, the output of data preprocessing can be categorized into two parts: (1) the real training samples for the following adversarial learning; (2) the input of generator and discriminator. For the input of generator and discriminator, the features of each sample are integrated into an input hub which is a module to structuring data, including the user ID, temporal check-ins (i.e., visiting pattern), and geographical locations to the corresponding POI.

3.2. Generative Adversarial Network

The generative adversarial network is the crucial module of our proposed POI recommender system, where the goal of GAN is to relieve the sparsity of check-in data and improve the performance of POI recommendations. To relieving the sparsity of check-in data, the generator is capable of capturing the actual distribution of individual sequential check-in records and synthesizing plausible samples, which can supplement training data. For improving the performance of POI recommendations, compared to the single model, the adversarial learning between a generator and a discriminator can further promote the generalization ability of each model and improve the overall performance. In the following paragraphs, we will introduce the generator, the discriminator, and the adversarial learning in detail.

Generator: Figure 2 illustrates the structure of proposed generator in the framework where the generator comprises an embedding layer, an attention layer, and an output layer. The input of the generator consists of User ID, temporal check-ins, and geographical locations. The User ID is used to obtain the embedding of a specific user. The temporal check-ins are a list of visited POI IDs for the corresponding user using a three-size sliding window. The geographical locations are the latitudes and longitudes of the corresponding POIs in the temporal check-ins. In detail, the embedding layer is utilized to describe the set of users $U = \{u_1, u_2, \dots, u_m\}$, and represent the set of POIs $V = \{v_1, v_2, \dots, v_n\}$ using the corresponding vectors where the embeddings describe the preferences of users and the characteristics of POIs. Furthermore, an LSTM is leveraged to fold the sequential check-ins V_{u_i} of user u_i while another LSTM is used to fit the corresponding sequential geographical locations L_{u_i} . Therefore, the output of the embedding layer is the embedding of a user (i.e., u_i), the latent feature of corresponding temporal check-ins h_v , and the latent feature of corresponding geographical location h_l . The attention layer is leveraged to dynamically adjust the representation of user embeddings based on the latest temporal check-ins and geographical locations. In general, the user embedding u_i is capable of describing the preference of i -th user based on her historical check-ins. However, in POI recommender systems, the final choice of a user is not only dependent on the preference but also relevant to the previously visited POI and the current geographical location. In addition, the preference is more specific to a user than the sequential check-ins and geographical locations. Therefore, instead of concatenating the user embedding and the latent features of temporal check-ins and geographical influence (i.e., contextual information), we utilize the latent features of contextual information to calculate the parameter of attention vector to finely tuning the user embedding that is capable of describing the actual preference based on the contexts. The output of the attention layer is based on the following equation:

$$o_a = \text{softmax}([h_v, h_l] \times w_a + b_a) \times u_i. \quad (1)$$

The output layer (i.e., a fully connected layer) is leveraged to transform the latent features of the previous layer into meaningful outputs. In the proposed generator, the multi-task strategy is used to improve the performance of fitting actual distribution of observed check-in data. Besides the predicted check-in vector, which is calculated based on a two-fully-connected layer, the next POI location is predicted based on a two-fully-connected layer and the latent feature of geographical location h_l . The predicted check-in vector and the next POI location can be calculated based on the following equations, respectively:

$$o_c = \text{softmax}(\text{ReLU}(o_a \times w_{c_1} + b_{c_1}) \times w_{c_2} + b_{c_2}), \quad (2)$$

$$o_l = \text{sigmoid}(\text{ReLU}(h_l \times w_{l_1} + b_{l_1}) \times w_{l_2} + b_{l_2}), \quad (3)$$

where o_c is considered as a fake check-in preference (i.e., a fake sample) for the user u_i according to the current temporal check-ins and geographical location.

Discriminator: Figure 3 illustrates the proposed discriminator in the framework where the discriminator also comprises an embedding layer, an attention layer, and an output layer. Similar to the structure of the proposed generator, the discriminator also exploits the user ID, the temporal check-ins, and the geographical location to generate the dynamical user preference (i.e., an embedding vector) based on the latest historical check-in records. However, the goal of the discriminator is to distinguish the real samples (i.e., actual check-in records) and the fake samples (i.e., predicted check-in vectors) for helping the generator improve the performance of capturing the actual user preference based on the contextual information. Therefore, besides the user ID, the temporal check-ins, the geographical location, and the check-in vectors, including the actual records and the generated ones, are transformed into a latent vector using a fully connected layer in the embedding layer. The transformed vector, which has the same dimension with the dynamical user preference, represents the user preference based on

the current check-in vector. In the attention layer, the generated and transformed user preferences are concatenated into a long vector, which is considered as the output of the attention layer. For the output layer, the discriminator transforms the output of attention layer into a single value where the value is equal to 0 if the generator produces the check-in vector while the value is equal to 1 if the check-in vector is an actual check-in record:

$$o_d = \text{sigmoid}(\text{ReLU}(o_a \times w_{d_1} + b_{d_1}) \times w_{d_2} + b_{d_2}), \tag{4}$$

where o_a is the output of attention layer in the discriminator.

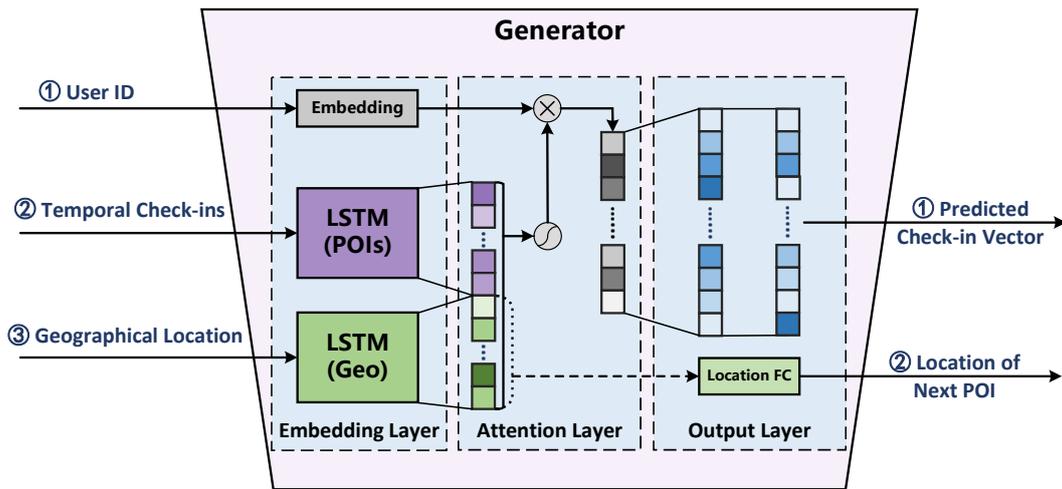


Figure 2. The structure of generator in the framework.

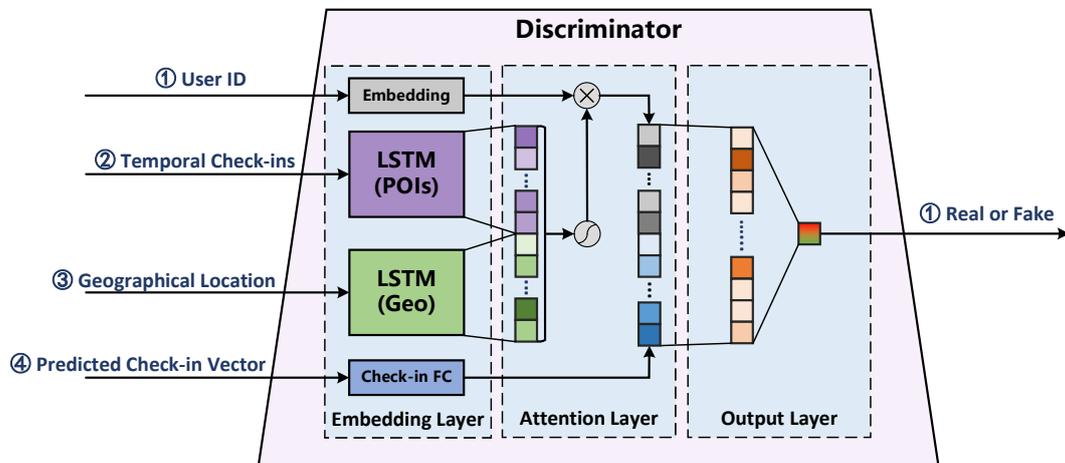


Figure 3. The structure of discriminator in the framework.

Adversarial Learning: The core of the generative adversarial network is the adversarial game between the generator and the discriminator. In other words, promoting the quality of generated check-in records will improve the performance of distinguishing real or fake samples to the discriminator while the improvement of discriminator will also promote the performance of capturing the actual distribution of user preference to the generator. Therefore, different from traditional models, the generator and the discriminator have individual objective functions. The objective function of the generator consists of the prediction loss, the geographical location loss, and the generation loss. The prediction loss is calculated based on the following equation:

$$loss_{pred} = - \sum_{j=1}^n (v_j \log(\hat{v}_j) + (1 - v_j) \log(1 - \hat{v}_j)), \tag{5}$$

where the binary variables $v_j \in \{0, 1\}$ is the actual check-in record on the j -th POI in V for the current user (i.e., User ID), where 0 means that the user has never been to the j -th POI and 1 means that the user has been to the j -th POI. The variables $\hat{v}_j \in [0, 1]$ are the predicted check-in record on the j -th POI in V . The geographical location loss is calculated as below:

$$loss_{loc} = \frac{1}{2} ((l^{lat} - \hat{l}^{lat})^2 + (l^{lon} - \hat{l}^{lon})^2), \tag{6}$$

where l^{lat} and l^{lon} are the normalized latitude and longitude of actual check-in POI while \hat{l}^{lat} and \hat{l}^{lon} are the normalized latitude and longitude of predicted check-in POI. The generation loss is based on the performance of distinguishing the generated check-in records to the discriminator:

$$loss_{gen} = \log(1 - D(\hat{\mathbf{v}}|u, V_u)), \tag{7}$$

where $D(\hat{\mathbf{v}}|u, V_u)$ represents the result of discriminator whether the predicted check-in vector $\hat{\mathbf{v}}$ (i.e., the fake sample) is considered as a real or fake sample given a specific user u and a temporal check-in records V_u . In general, the generator is updated based on $loss_{gen}$, which directly shows the quality of produced fake samples. In this paper, due to the sparsity of actual check-in vector based on an individual and specific temporal check-in records, $loss_{pred}$ is used to accelerate the convergence of capturing the actual distribution to the generator. In addition, due to the significance of geographical influence in POI recommender systems, $loss_{loc}$ is utilized to take the distance from the previous visited POI to the next recommended POI into account. Therefore, the objective function of the generator is calculated based on the following equation:

$$J^G = \min_{\theta} \sum_{i=1}^m (\alpha loss_{pred} + \beta loss_{loc} + (1 - \alpha - \beta) loss_{gen}), \tag{8}$$

where θ is the parameter of generator while $\alpha \in (0, 1)$ and $\beta \in (0, 1)$ is the predefined weight of $loss_{pred}$ and $loss_{loc}$, respectively.

The objective function of discriminator is similar to that of general binary classification, which can be calculated as below:

$$J^D = \min_{\phi} - \sum_{i=1}^m (\log(D(\mathbf{v}|u, V_u)) + \log(1 - D(\hat{\mathbf{v}}|u, V_u))), \tag{9}$$

where \mathbf{v} and $\hat{\mathbf{v}}$ are the real and predicted check-in vector given a specific user u and temporal check-in records V_u , respectively. Note that we exploit the general adversarial setting in this paper where the generator and the discriminator will be alternatively updated once based on the corresponding objective function until both of them converge.

The GANs are trained to be good at understanding the known information. In our proposed model, the user and POI embedding are used to capture the characteristics of each user and POI. Due to the sparsity of check-in records, it is hard for the model to train the embeddings sufficiently. Therefore, the GANs are useful for the model to capture the embeddings of users and POIs well. In addition, the GANs can be good at providing appropriate prediction. Even without the consideration of $loss_{pred}$ and $loss_{loc}$, the generator in this paper is trained to provide the next POI. In detail, the input of the generator is the latest three visited POI (i.e., 3-size sliding window) which describe the current situation of the corresponding user, while the output of the generator is the next POI which would be interested for the user in the current situation (i.e., pattern). Therefore, the structure of GANs provides the effective strategy for our model to capture the user’s preference based on the contextual check-ins.

3.3. POI Recommendation

With the convergence of GAN, the generator is used to predict the check-in vector (i.e., POI recommendation) given a specific user and the corresponding latest sequential check-in records. In this paper, TopN recommendation is leveraged to provide appropriate POIs where the TopN list is generated and ordered by the visiting probability of each POI based on \hat{v} . Within the stage of POI recommendation: (1) the latest check-in records and the corresponding geographical location are extracted based on a specific user; (2) the user and the latest check-in records are transformed into the embeddings; (3) the user embedding, the sequential POI embeddings, and the corresponding geographical locations are considered as the input of generator and the next POI recommendations are predicted ordered by the probability based on the generated n -dimensional vector.

4. Experiments

In this section, we conduct several experiments on two real-world datasets (i.e., Foursquare and Gowalla) to evaluate the effectiveness of our proposed approach. Based on the experiments, we aim to answer the following issues:

- The influence of the embedding and hidden layer on the performance of MTPR;
- The effectiveness of geographical influence and adversarial learning on the POI recommendation;
- The performance of MTPR compared to the baselines on the real-world datasets.

4.1. Experimental Setup

Datasets: The experiments are conducted on two real-world datasets Foursquare and Gowalla (<https://www.ntu.edu.sg/home/gaocong/datacode.htm>). Foursquare is the most popular POI recommender system in the world where users can query interesting POIs (i.e., hotels and restaurants). In addition, different from traditional search engines, Foursquare provides APIs for users to share their check-in records and reviews with others as a social network. Therefore, there are numerous check-in records generated each day. The dataset of Foursquare used in this paper is collected from August 2010 to July 2011 in Singapore, including 342,850 check-ins. Gowalla is also a famous POI recommender system where the dataset used in this paper is collected from February 2009 to October 2010 in Virginia and Nevada USA, including 736,148 check-in records. Table 1 shows the fields of each check-in record and a check-in sample.

Table 1. The fields of each check-in record.

Sample	Field (Description)
USER_42	User_ID
LOC_11903	POL_ID
(37.4191867667, 122.2122015167)	coordinate (the lat. and long. of POI)
04:12	checkin_time (hour:min)
0	date_id (the representation of date)

To relieve the influence of long-tail data to the robustness of the proposed model and baselines, we eliminated the users who have fewer than 10 check-ins, and the POIs which have been visited by fewer than 10 users (POI2Vec removed the users who have fewer than five check-ins, and the POIs which have been visited by fewer than five users). Table 2 shows the statistics of used Foursquare and Gowalla datasets in this paper.

Table 2. The overview of Foursquare and Gowalla.

Statistics	Foursquare	Gowalla
Count of Users	1895	5951
Count of POIs	2109	7449
Check-in records	115,114	270,765
Records per user	60.76	45.50
Records per POI	54.60	36.35

Baselines: To evaluate the effectiveness of MTPR, the proposed approach is compared with several state-of-art methods considering the geographical influence including PRME [8], POI2Vec [6], and GeoIE [9]. In addition, to validate the effectiveness of adversarial learning and geographical influence, we also compare MTPR to its variants which eliminate the adversarial loss or the geographical LSTM. These baselines are briefly introduced as below:

- **PRME [8]:** transforms each POI into a point in a K-dimensional Euler space using the metric embedding and the distance between the transformed points is considered as the similarity between the corresponding POIs.
- **POI2Vec [6]:** is an approach to learning the latent representation (i.e., embedding) of users and POIs based on the individual sequential check-in records simultaneously considering the geographical influence and the user preference. The geographical location is split into blocks and the geographical relationship between POIs is measured using the Huffman tree.
- **GeoIE [9]:** is a method to providing POI recommendations considering the POI-specific geographical influence including the geographical influence of POI, the geographical susceptibility of POI, and their physical distance.
- **Geo-ALM [10]:** is a geographical information based adversarial learning model which utilized a game setting to generate a few but critical negative samples (i.e., POIs that users have never been to) for improving the effectiveness of POI recommendation.
- **MTPR-NoGeo:** is a variant of MTPR which eliminates the geographical location loss $loss_{loc}$ and the corresponding geo-LSTM module.
- **MTPR-NoGen:** is a variant of MTPR which eliminates the generation loss $loss_{gen}$ and trains the generator only based on the prediction loss $loss_{pred}$ and geographical location loss $loss_{loc}$.

Metrics and Parameters: In this paper, Precision, Recall, and F1-score are calculated based on the following equations:

$$Precision@k = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}, \quad (10)$$

$$Recall@k = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}, \quad (11)$$

$$F1 - score@k = \frac{2Recall@k \times Precision@k}{Recall@k + Precision@k}, \quad (12)$$

where k is the length of recommendation list (i.e., top- k recommendation), $R(u)$ is the set of POIs which are recommended to the user in the recommendation list, and $T(u)$ is the set of POIs which are actually visited by the user in the test dataset. For a given user, the corresponding check-in records are sorted by the timestamps and the latest 15% check-ins are considered as the test dataset to evaluate the performance of models.

The experimental setup and the used parameters are introduced in detail as below. The segmentation of datasets is 70%/15%/15% based on the temporal check-in records where the first 70% is the training set for learning the models, the second 15% is the validation set for optimizing the parameters, and the third 15% is the testing set for evaluating the performance. The sliding window, which is used to split the contextual POIs, is set as 3. The learning rate of discriminator and generator is set as 0.01. The weights of loss in the generator are set as $\alpha = 0.5$ and $\beta = 0.3$. The dimension of

user embedding and POI embedding is 300, while the dimension of the hidden layer in LSTM is set as 100. The training epoch is defined as 20. The technique of simultaneous gradient descent is used to update the model. The rate of training epochs of the generator compared with the discriminator is 1:1. To evaluate the topN POI recommendation of baselines and proposed approach, the precision, recall, and F1-score are calculated at Top 5, 10, 15, and 20. In addition, the losses (i.e., $loss_{pred}$, $loss_{gen}$, and $loss_{loc}$) demonstrate the difference between predicted and actual ones. Therefore, the fluctuation of loss is considered as an effective way to determine that whether the model is converged or not in this domain.

4.2. Discussion and Results

First, we evaluate the influence of the embedding and hidden layer on the performance of MTPR. Figures 4 and 5 illustrate the performance of MTPR using different combinations of embeddings and hidden layers on top-10 recommendations of Foursquare and Gowalla, respectively. Note that, except for the 200-dimensional hidden layer, the performance (i.e., precision) is improved with the increasing dimension of embeddings on the datasets. In addition, the performance is better when the dimension of embeddings is higher than the dimension of hidden layers. In summary, the performance (i.e., precision) is not improved monotonously with the increasing dimension of embeddings or hidden layers. It would be better to select the parameters based on specific scenarios.

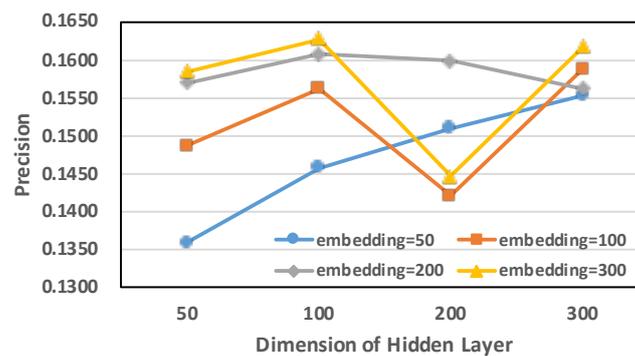


Figure 4. The performance using different dimensions of embedding and hidden layers on Foursquare.

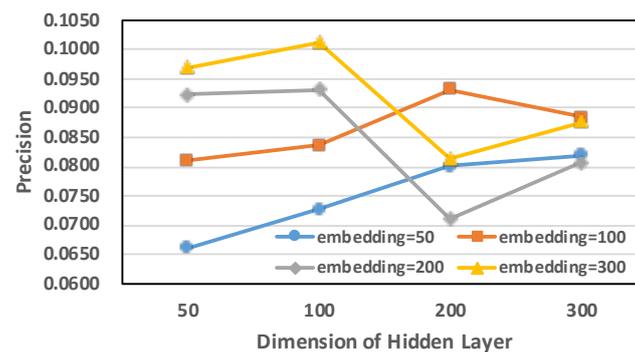


Figure 5. The performance using different dimensions of embedding and hidden layers on Gowalla.

In addition, we evaluate the influence of different combinations of α and β on Foursquare. Figure 6 shows the F1-score of our proposed approach on Foursquare using the possible combination of α and β (i.e., $\alpha + \beta \leq 1$). The value of F1-score reaches the peak near $\alpha = 0.5$ and $\beta = 0.3$. Furthermore, the performance is worse when $loss_{pred}$ is ignored (i.e., $\alpha = 0$); meanwhile, the performance is not stable if $loss_{loc}$ is missed (i.e., $\beta = 0$). The performance is better if the approach simultaneously takes $loss_{pred}$, $loss_{gen}$, and $loss_{loc}$ into account instead of only considering $loss_{pred}$ (i.e., $\alpha = 1$). The experimental results demonstrate that the GAN-based structure cannot address the task of POI recommendations

according to $loss_{gen}$ (i.e., $\alpha = 0$ and $\beta = 0$); however, the GAN-based structure is capable of improving the performance if the proposed approach takes $loss_{pred}$ and $loss_{loc}$ into account (especially $loss_{pred}$).

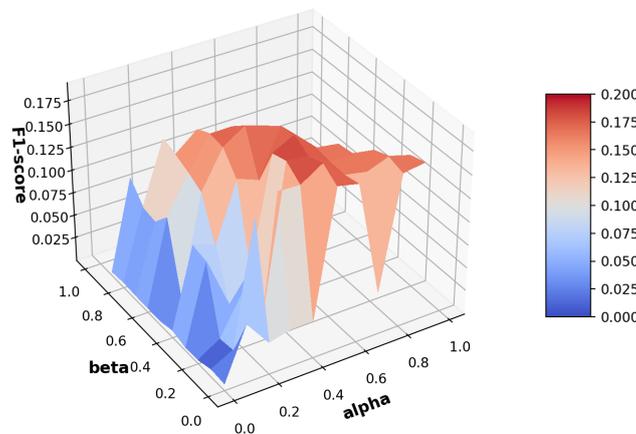


Figure 6. The performance using different combinations of alpha and beta on Foursquare.

To investigate the convergence of our proposed approach, we conduct an experiment which traces the loss of generator (i.e., the convex linear combination J^G of $loss_{pred}$, $loss_{gen}$ and $loss_{loc}$ shown in Equation (8)) and discriminator, J^D , during the process of training each batch (i.e., the batch size is 32) on Foursquare. Figure 7 illustrates that the fluctuation of loss become stable after the 5th epoch. Although there exists a fluctuation from the 15th and 18th epoch, the trend of loss becomes stable again after the 19th epoch. In summary, the stable trend of loss demonstrates the convergence of the proposed approach.

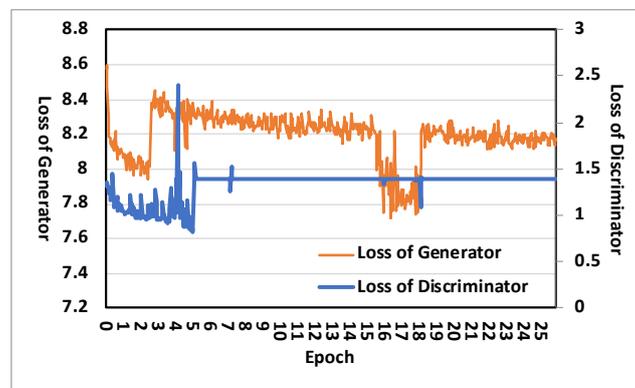


Figure 7. The convergence of the proposed approach on Foursquare.

Second, we investigate the effectiveness of geographical influence and adversarial learning to the POI recommendation. Tables 3 and 4 show the performance of MTPR using different components on Foursquare and Gowalla, respectively. The std error of each metric is given in the bracket next to the corresponding metric. The density of the dataset (i.e., actual check-in records of individuals) causes the low values of Precision, Recall, and F1-measure. As we can observe from Table 2, although the check-in records of each user are about 50, most of the users only have 20–30 check-in records where these records also have many duplicate POIs. In summary, MTPR outperforms MTPR-NoGeo and MTPR-NoGen, which means the components of geographical module and adversarial learning are capable of improving the performance of POI recommendations. In detail, MTPR-NoGen outperforms MTPR-NoGeo, which is a model that consists of LSTM but without the consideration of geographical influence, which demonstrates that the geographical influence is more significant than the mechanism of adversarial learning in the current POI recommendation task. In addition to all the variants of MTPR, the precision is decreasing with the increase of TopN recommendations. In other words,

the approaches rank the relevant recommendation (i.e., the hitting recommendations based on the testing dataset) at the top of the list, which shows the effectiveness of MTPR.

Table 3. The performance of MTPR using different components on Foursquare.

Metric		Model		
		MTPR-NoGeo	MTPR-NoGen	MTPR
Precision	Top5	0.152(0.088)	0.219(0.011)	0.232(0.003)
	Top10	0.113(0.060)	0.157(0.006)	0.166(0.001)
	Top15	0.092(0.045)	0.126(0.004)	0.132(0.001)
	Top20	0.079(0.036)	0.106(0.003)	0.111(0.001)
Recall	Top5	0.105(0.061)	0.152(0.008)	0.161(0.002)
	Top10	0.156(0.083)	0.217(0.009)	0.229(0.002)
	Top15	0.192(0.094)	0.261(0.009)	0.274(0.002)
	Top20	0.219(0.100)	0.292(0.009)	0.307(0.003)
F1-score	Top5	0.124(0.072)	0.179(0.009)	0.190(0.002)
	Top10	0.131(0.069)	0.182(0.007)	0.192(0.002)
	Top15	0.125(0.061)	0.169(0.006)	0.178(0.002)
	Top20	0.116(0.053)	0.155(0.005)	0.163(0.001)

Table 4. The performance of MTPR using different components on Gowalla.

Metric		Model		
		MTPR-NoGeo	MTPR-NoGen	MTPR
Precision	Top5	0.071(0.048)	0.120(0.035)	0.153(0.001)
	Top10	0.050(0.034)	0.083(0.024)	0.106(0.001)
	Top15	0.040(0.027)	0.065(0.018)	0.082(0.000)
	Top20	0.034(0.022)	0.054(0.015)	0.068(0.000)
Recall	Top5	0.058(0.039)	0.098(0.029)	0.125(0.001)
	Top10	0.081(0.055)	0.136(0.039)	0.172(0.001)
	Top15	0.097(0.065)	0.160(0.044)	0.201(0.001)
	Top20	0.110(0.071)	0.177(0.048)	0.222(0.001)
F1-score	Top5	0.064(0.043)	0.108(0.031)	0.138(0.001)
	Top10	0.062(0.042)	0.103(0.029)	0.131(0.001)
	Top15	0.057(0.038)	0.093(0.026)	0.117(0.001)
	Top20	0.051(0.033)	0.083(0.023)	0.104(0.000)

Third, we compare MTPR with the baselines to validate the effectiveness of our proposed approach in the task of TopN recommendations. Table 5 shows the comparison between the generated samples (i.e., recommendations) and the real check-ins. These cases (i.e., POI ids) are sampled from four users where the contextual check-in sequence is ordered by timestamps, the real check-ins are ordered by POI ids, and the Top10 recommendations are ordered by the relevance to the user preferences. The real check-ins are the actual check-in records following the contextual check-in sequence. The first case is familiar in the recommendations where the 896th POI hits the user preference while the remaining recommendations are invalid. The second one is an effective recommendation list where several recommendations (i.e., 904, 1007, 1275, 1642, and 722) hit the user preference. The third one is also a typical case in the dataset where the contextual check-ins consist of the duplicated POIs (i.e., 209), which is difficult for the model to provide appropriate recommendations. The last case shows that the proposed approach may generate POIs users have visited (i.e., 1477). In other words, the diversity of recommendations should be further improved.

Table 5. The comparison between the generated samples (i.e., recommendations) and the real check-ins at Top10.

Contextual Check-in Sequence	Real Check-Ins	Top10 Recommendations
2072, 1194, 1118	169, 289, 462, 486, 628, 759, 767, 839, 896 , 954, 986, 1005, 1076, 1194, 1273, 1312, 1453	1630, 427, 276, 1621, 896 , 538, 1308, 903, 1918, 1329
858, 1242, 2081	119, 265, 352, 473, 489, 722 , 726, 904 , 1007 , 1120, 1132, 1154, 1273, 1275 , 1448, 1498, 1571, 1574, 1610, 1642 , 1654, 1747, 1971	904 , 1007 , 1275 , 83, 1226, 1642 , 1447, 1641, 722 , 1880
209, 209, 209	144, 183, 209, 360, 362, 373, 706, 925, 933, 1058, 1219, 1416, 1432, 1551, 1566, 1629, 1677, 1758, 1947, 1971, 1981	1237, 899, 1180, 1807, 1017, 1758, 561, 1810, 1097, 209
889, 1477 , 1345	42, 264, 628, 701, 722 , 777, 784 , 889, 931, 976, 1325, 1477 , 1921	1477 , 690, 805, 1210, 333, 722 , 1306, 1395, 784 , 427

To reveal the overall performance of our proposed approach, Tables 6 and 7 describe the performance of baselines and MTPR on the two real-world datasets. Observed from the experimental results, MTPR outperforms other baselines on both datasets, which significantly improves the Top5 recommendation. In detail, the GAN-based model Geo-ALM outperforms other baselines that shows that the structure of generative adversarial networks is capable of improving the performance of POI recommendations. In addition, the performance of PRME is fluctuating with the increasing length of recommendation list (i.e., TopN recommendations), which means that PRME cannot capture the user preference exactly but promotes the diversity of recommendations. Compared to the performance of MTPR-NoGeo and MTPR-NoGen in Tables 3 and 4 with the performance of baselines considering the geographical influence in Tables 6 and 7, both MTPR-NoGeo and MTPR-NoGen outperform the baselines which further demonstrates that the structure of geographical LSTM and the mechanism of adversarial learning are capable of improving the performance of POI recommendations.

Table 6. The performance of baselines and MTPR on Foursquare.

Metric		Model				
		PRME	POI2Vec	GeoIE	Geo-ALM	MTPR
Precision	Top5	0.036(0.007)	0.121(0.055)	0.138(0.021)	0.189(0.005)	0.232(0.003)
	Top10	0.034(0.008)	0.123(0.043)	0.120(0.017)	0.152(0.002)	0.166(0.001)
	Top15	0.052(0.004)	0.093(0.012)	0.101(0.015)	0.124(0.003)	0.132(0.001)
	Top20	0.064(0.003)	0.078(0.009)	0.088(0.005)	0.101(0.004)	0.111(0.001)
Recall	Top5	0.025(0.005)	0.084(0.036)	0.099(0.014)	0.143(0.002)	0.161(0.002)
	Top10	0.047(0.004)	0.173(0.021)	0.165(0.007)	0.192(0.003)	0.229(0.002)
	Top15	0.108(0.003)	0.195(0.015)	0.203(0.006)	0.243(0.001)	0.274(0.002)
	Top20	0.180(0.005)	0.213(0.010)	0.224(0.003)	0.274(0.004)	0.307(0.003)
F1-score	Top5	0.029(0.003)	0.099(0.086)	0.115(0.016)	0.163(0.003)	0.190(0.002)
	Top10	0.039(0.007)	0.143(0.057)	0.138(0.013)	0.170(0.001)	0.192(0.002)
	Top15	0.070(0.005)	0.125(0.025)	0.134(0.008)	0.164(0.005)	0.178(0.002)
	Top20	0.094(0.004)	0.114(0.014)	0.126(0.007)	0.148(0.002)	0.163(0.001)

Table 7. The performance of baselines and MTPR on Gowalla.

Metric		Model				
		PRME	POI2Vec	GeoIE	Geo-ALM	MTPR
Precision	Top5	0.053(0.006)	0.098(0.065)	0.115(0.018)	0.123(0.002)	0.153(0.001)
	Top10	0.044(0.007)	0.082(0.049)	0.090(0.020)	0.099(0.001)	0.106(0.001)
	Top15	0.030(0.005)	0.060(0.015)	0.066(0.016)	0.079(0.003)	0.082(0.000)
	Top20	0.400(0.002)	0.053(0.008)	0.062(0.006)	0.065(0.001)	0.068(0.000)
Recall	Top5	0.045(0.004)	0.082(0.034)	0.098(0.012)	0.111(0.002)	0.125(0.001)
	Top10	0.073(0.004)	0.140(0.025)	0.138(0.009)	0.148(0.002)	0.172(0.001)
	Top15	0.079(0.002)	0.143(0.013)	0.161(0.005)	0.173(0.004)	0.201(0.001)
	Top20	0.122(0.003)	0.170(0.012)	0.215(0.002)	0.220(0.001)	0.222(0.001)
F1-score	Top5	0.048(0.003)	0.089(0.080)	0.105(0.015)	0.117(0.001)	0.138(0.001)
	Top10	0.054(0.005)	0.103(0.053)	0.108(0.015)	0.119(0.001)	0.131(0.001)
	Top15	0.043(0.004)	0.084(0.022)	0.093(0.006)	0.108(0.001)	0.117(0.001)
	Top20	0.186(0.004)	0.080(0.013)	0.096(0.004)	0.100(0.002)	0.104(0.000)

However, there still exist some limitations and shortcomings which should be issued to improve the performance of POI recommendations further. First, the setup of geographical influence (i.e., coordinate LSTM) is straightforward and easily causes the problem of overfitting. A smarter strategy should be proposed to fit the path of changing coordinate for considering the geographical influence. Second, the training samples (i.e., real and fake) contain the instances that have the same contextual sequential POIs but are followed by different check-ins where these samples will confuse the models during the training process. It is challenging to relieve the influence caused by the conflicting training samples. Third, the structures of generator and discriminator are similar, which is not beneficial to adversarial learning. The significant difference between generators and discriminators may further improve the performance of the adversarial setup.

5. Conclusions

In this paper, we propose a multi-task learning based POI recommendation approach simultaneously considering the sequential check-in records and the geographical influence. The experimental results show that both the module of geographical LSTM and the setup of adversarial learning are capable of improving the performance of POI recommendations on the real-world datasets (i.e., Foursquare and Gowalla). In addition, the performance of the proposed method outperforms the baselines, which also take the geographical influence into account, especially at the Top5 recommendations, which demonstrates the effectiveness of MTPR to capture the user preference.

In the future, there still exists some work for which it is worth making an effort. First, besides the geographical influence, the category of POI is also a significant factor for the next visited POI in the sequential check-in records. It would be better to simultaneously consider the geographical and the category influence for capturing the user preference. Second, deep learning-based GAN training is time-consuming and hard to use in practical scenarios. An effective online learning approach would make these deep learning-based methods accessible in recommender systems. Third, the TopN recommendation is a traditional strategy for providing delicate items based on the relevance between users and items. However, if the user preference cannot be captured precisely, the ranking list is not meaningful anymore. Therefore, it is necessary to propose a kind of recommendation strategy considering the changing user preference.

Author Contributions: Conceptualization, B.X., Y.B. and J.Y.; methodology, B.X.; validation, Y.B. and J.Y.; formal analysis, Q.L.; investigation, L.X.; resources, Y.B. and J.Y.; data curation, Y.B. and J.Y.; writing—original draft preparation, B.X.; writing—review and editing, B.X.; visualization, B.X.; supervision, B.X.; project administration, B.X.; funding acquisition, B.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant No. 61802205, the Natural Science Foundation of Jiangsu Higher Education Institutions of China under Grant No. 18KJB520037, and the research funds of Nanjing University of Posts and Telecommunications under Grant No. NY218116.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xia, B.; Li, T.; Li, Q.; Zhang, H. Noise-tolerance matrix completion for location recommendation. *Data Min. Knowl. Discov.* **2018**, *32*, 1–24. [[CrossRef](#)]
2. Bothorel, C.; Lathia, N.; Picot-Clémente, R.; Noulas, A. Location recommendation with social media data. In *Social Information Access: Systems and Technologies*; Brusilovsky, P., He, D., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2018; Volume 10100, pp. 624–653. [[CrossRef](#)]
3. Noulas, A.; Scellato, S.; Lathia, N.; Mascolo, C. A random walk around the city: New venue recommendation in location-based social networks. In Proceedings of the 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing, Amsterdam, The Netherlands, 3–5 September 2012; pp. 144–153. [[CrossRef](#)]
4. Xia, B.; Ni, Z.; Li, T.; Li, Q.; Zhou, Q. VRer: Context-Based Venue Recommendation using embedded space ranking SVM in location-based social network. *Expert Syst. Appl.* **2017**, *83*, 18–29. [[CrossRef](#)]
5. Xia, B.; Li, Y.; Li, Q.; Li, T. Attention-based recurrent neural network for location recommendation. In Proceedings of the 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE 2017), Nanjing, China, 24–26 November 2017; Li, T., Martínez-López, L., Li, Y., Eds.; IEEE: Piscataway, NJ, USA, 2017; pp. 1–6. [[CrossRef](#)]
6. Feng, S.; Cong, G.; An, B.; Chee, Y.M. POI2Vec: Geographical latent representation for predicting future visitors. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 102–108.
7. Lian, D.; Zhang, Z.; Ge, Y.; Zhang, F.; Yuan, N.J.; Xie, X. Regularized content-aware tensor factorization meets temporal-aware location recommendation. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; pp. 1029–1034.
8. Feng, S.; Li, X.; Zeng, Y.; Cong, G.; Chee, Y.M.; Yuan, Q. Personalized ranking metric embedding for next new POI recommendation. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015), Buenos Aires, Argentina, 25–31 July 2015; pp. 2069–2075.
9. Wang, H.; Shen, H.; Ouyang, W.; Cheng, X. Exploiting POI-specific geographical influence for point-of-interest recommendation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI 2018), Stockholm, Sweden, 13–19 July 2018; pp. 3877–3883. [[CrossRef](#)]
10. Liu, W.; Wang, Z.; Yao, B.; Yin, J. Geo-ALM: POI recommendation by fusing geographical information and adversarial learning mechanism. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI 2019), Macao, China, 10–16 August 2019; pp. 1807–1813. [[CrossRef](#)]
11. Fu, Y.; Wang, P.; Du, J.; Wu, L.; Li, X. Efficient region embedding with multi-view spatial networks: A perspective of locality-constrained spatial autocorrelations. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, HI, USA, 27 January–1 February 2019; AAAI Press: Palo Alto, CA, USA, 2019; pp. 906–913. [[CrossRef](#)]
12. Sun, K.; Qian, T.; Chen, T.; Liang, Y.; Nguyen, Q.V.H.; Yin, H. Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, 7–12 February 2020; AAAI Press: Palo Alto, CA, USA, 2020; pp. 214–221.
13. Zhou, F.; Yin, R.; Zhang, K.; Trajcevski, G.; Zhong, T.; Wu, J. Adversarial point-of-interest recommendation. In Proceedings of the World Wide Web Conference (WWW 2019), San Francisco, CA, USA, 13–17 May 2019; Liu, L., White, R.W., Mantrach, A., Silvestri, F., McAuley, J.J., Baeza-Yates, R., Zia, L., Eds.; ACM: New York, NY, USA, 2019; pp. 3462–34618. [[CrossRef](#)]
14. Chang, B.; Park, Y.; Park, D.; Kim, S.; Kang, J. Content-aware hierarchical point-of-interest embedding model for successive POI recommendation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI 2018), Stockholm, Sweden, 13–19 July 2018; Lang, J., Ed.; pp. 3301–3307. [[CrossRef](#)]

15. Bobadilla, J.; Alonso, S.; Hernando, A. Deep Learning Architecture for Collaborative Filtering Recommender Systems. *Appl. Sci.* **2020**, *10*, 2441. [[CrossRef](#)]
16. Nguyen, L.V.; Hong, M.S.; Jung, J.J.; Sohn, B.S. Cognitive Similarity-Based Collaborative Filtering Recommendation System. *Appl. Sci.* **2020**, *10*, 4183. [[CrossRef](#)]
17. Goodfellow, I.J. NIPS 2016 Tutorial: Generative Adversarial Networks. *arXiv* **2017**, arXiv:1701.00160.
18. Wang, J.; Yu, L.; Zhang, W.; Gong, Y.; Xu, Y.; Wang, B.; Zhang, P.; Zhang, D. IRGAN: A minimax game for unifying generative and discriminative information retrieval models. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, 7–11 August 2017; Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A.P., White, R.W., Eds.; ACM: New York, NY, USA, 2017; pp. 515–524. [[CrossRef](#)]
19. Chae, D.; Kang, J.; Kim, S.; Lee, J. CFGAN: A generic collaborative filtering framework based on generative adversarial networks. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM 2018), Torino, Italy, 22–26 October 2018; Cuzzocrea, A., Allan, J., Paton, N.W., Srivastava, D., Agrawal, R., Broder, A.Z., Zaki, M.J., Candan, K.S., Labrinidis, A., Schuster, A., et al., Eds.; ACM: New York, NY, USA, 2018; pp. 137–146. [[CrossRef](#)]
20. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875.
21. Wang, P.; Fu, Y.; Xiong, H.; Li, X. Adversarial substructured representation learning for mobile user profiling. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2019), Anchorage, AK, USA, 4–8 August 2019; Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., Karypis, G., Eds.; ACM: New York, NY, USA, 2019; pp. 130–138. [[CrossRef](#)]
22. Zhang, Y.; Fu, Y.; Wang, P.; Li, X.; Zheng, Y. Unifying inter-region autocorrelation and intra-region structures for spatial embedding via collective adversarial learning. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2019), Anchorage, AK, USA, 4–8 August 2019; Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., Karypis, G., Eds.; ACM: New York, NY, USA, 2019; pp. 1700–1708. [[CrossRef](#)]
23. Thanh-Tung, H.; Tran, T.; Venkatesh, S. Improving generalization and stability of generative adversarial networks. In Proceedings of the 7th International Conference on Learning Representations (ICLR 2019), New Orleans, LA, USA, 6–9 May 2019.
24. Weston, J.; Chopra, S.; Bordes, A. Memory networks. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
25. Sukhbaatar, S.; Szlam, A.; Weston, J.; Fergus, R. End-to-end memory networks. In Proceedings of the Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, Montreal, QC, Canada, 7–12 December 2015; pp. 2440–2448.
26. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
27. Tay, Y.; Luu, A.T.; Hui, S.C.; Su, J. Densely connected attention propagation for reading comprehension. In Proceedings of the Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018 (NeurIPS 2018), Montréal, QC, Canada, 3–8 December 2018; pp. 4911–4922.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]
29. Liu, T.; Liao, J.; Wu, Z.; Wang, Y.; Wang, J. Exploiting geographical-temporal awareness attention for next point-of-interest recommendation. *Neurocomputing* **2020**, *400*, 227–237. [[CrossRef](#)]
30. Yin, J.; Li, Y.; Liu, Z.; Xu, J.; Xia, B.; Li, Q. ADPR: An attention-based deep learning point-of-interest recommendation framework. In Proceedings of the International Joint Conference on Neural Networks (IJCNN 2019), Budapest, Hungary, 14–19 July 2019; pp. 1–8. [[CrossRef](#)]

