Spatiotemporal Representation Learning for Translation-Based POI Recommendation

TIEYUN QIAN and BEI LIU, Wuhan University, China
QUOC VIET HUNG NGUYEN, Griffith University, Australia
HONGZHI YIN, The University of Queensland, Australia

The increasing proliferation of location-based social networks brings about a huge volume of user check-in data, which facilitates the recommendation of points of interest (POIs). Time and location are the two most important contextual factors in the user's decision-making for choosing a POI to visit. In this article, we focus on the *spatiotemporal context-aware* POI recommendation, which considers the joint effect of time and location for POI recommendation. Inspired by the recent advances in knowledge graph embedding, we propose a *spatiotemporal context-aware* and translation-based recommender framework (STA) to model the third-order relationship among users, POIs, and spatiotemporal contexts for large-scale POI recommendation. Specifically, we embed both users and POIs into a "transition space" where spatiotemporal contexts (i.e., a <*time, location*> pair) are modeled as *translation vectors* operating on users and POIs. We further develop a series of strategies to exploit various correlation information to address the data sparsity and cold-start issues for new spatiotemporal contexts, new users, and new POIs. We conduct extensive experiments on two real-world datasets. The experimental results demonstrate that our STA framework achieves the superior performance in terms of high recommendation accuracy, robustness to data sparsity, and effectiveness in handling the cold-start problem.

CCS Concepts: • **Information systems** → *Social recommendation*;

Additional Key Words and Phrases: POI recommendation, location-based social networks, spatiotemporal aware, contextual modeling

ACM Reference format:
Tieyun Qian, Bei Liu, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2019. Spatiotemporal Representation Learning for Translation-Based POI Recommendation. *ACM Trans. Inf. Syst.* 37, 2, Article 18 (January 2019), 24 pages.
https://doi.org/10.1145/3295499

1 INTRODUCTION

Location-based social networks (LBSN) like Foursquare, Yelp, and Facebook Places have proliferated over the last one or two decades. Many users on LBSN like to share their experiences with

T. Qian and H. Yin contributed equally to this work.

The work has been supported in part by the NSFC projects (61572376), the 111 project(B07037), ARC Discovery Early Career Researcher Award (Grant No. DE160100308), and ARC Discovery Project (Grant No. DP170103954).

Authors' addresses: T. Qian and B. Liu, School of Computer Science, Wuhan University, 430072, China; emails: {qty, beiliu}@whu.edu.cn; Q. Viet Hung Nguyen, School of Information and Communication Technology, Griffith University, QLD 4215, Australia; email: quocviethung1@gmail.com; H. Yin (corresponding author), School of Information Technology and Electrical Engineering, University of Queensland, QLD 4072, Australia; email: h.yin1@uq.edu.au.

1046-8188/2019/01-ART18 $15.00
https://doi.org/10.1145/3295499

their friends about points of interest (POIs), e.g., restaurants and museums. The huge volume of user check-in data has been collected by the providers of location-based services. This greatly facilitates the recommendation of POIs to help users to explore new places. The POI recommendation is of high value to both the users and businesses, and thus has attracted much attention from researchers in both academia and industry in recent years (Chang et al. 2018; Chen et al. 2015; Cheng et al. 2016; Gao et al. 2015b; Griesner et al. 2015; Hu and Ester 2014; Liu et al. 2011, 2016; Qiao et al. 2018; Wang et al. 2018; Zhu et al. 2015).

The task of POI recommendation is to provide personalized recommendations of places of interest to users at a certain time and location. Different from the traditional recommendation tasks like music or movie recommendation, *POI recommendation is highly spatiotemporal dependent.* For example, a student may go to a school cafeteria or a food court in a shopping mall at lunch time depending on whether he/she is on campus or shopping. This shows the joint spatiotemporal effect on a user's visiting behaviors. Similar results can be observed for the cases with the same location but different times. For example, on weekends, a user may go to a restaurant at noon and visit a bar at night near his/her home. Therefore, the time and location together play an important role in POI recommendation. However, modeling the joint effects of time and location faces a severe challenge from the extreme sparsity of users' check-in data. Usually, there are millions of POIs in the location-based social networks, but a user can only visit a very small number of the POIs. A statistic about the Foursquare dataset (Liu et al. 2013a) shows that the user-POI check-in count matrix has a sparsity of 99.87%. It has been proven that the data sparsity has been the most significant factor that limits the performance of recommender systems (Ye et al. 2010, 2011b).

Some existing studies on POI recommendation have exploited spatial or temporal influences with memory-based collaborative filtering (CF) (Ye et al. 2011b; Yuan et al. 2013) and Markov transition approaches (Cheng et al. 2013). Due to the sparsity of user generated check-in records at a specific spatiotemporal context, it is hard to find similar users with exactly the same spatiotemporal context or calculate the transition probability. Hence, the memory-based CF and Markov transition approaches are not suitable for modeling the spatiotemporal effects in POI recommendation. As model-based CF techniques such as Matrix factorization (MF) methods are able to overcome the sparsity of user-POI matrix to some extent, they have been extended for POI recommendation by integrating the temporal effect and geographical influence (Cheng et al. 2012; Griesner et al. 2015; Lian et al. 2014; Liu et al. 2014b). However, all of them adopted the dot product as the similarity measure and are not metric learning approaches because dot product does not satisfy the crucial triangle inequality. Therefore, these MF-based methods cannot further alleviate the data sparsity issue due to their limited generalization (He et al. 2017a; Hsieh et al. 2017).

Cold start is another challenge in POI recommendation. Cold start is a critical problem in the domain of recommendation systems, and consists of the cold-start item problem and the cold-start user problem. Items that have not received any rating or interaction are called cold-start items. Similarly, users who have not rated any item are called cold-start users. Being different from the traditional recommendation problem, there is a new type of cold-start problem emerging in our problem, called *cold-start spatiotemporal contexts*, which refer to the spatiotemporal contexts where there is not any check-in record. Although both cold-start user and cold-start item problems have been studied in some recent POI recommendation works (Gao et al. 2012, 2015a; Xie et al. 2016; Ye et al. 2011b), we are the first to propose and explore the problem of cold-start spatiotemporal context.

The problem of POI recommendation in the spatiotemporal context can be viewed as a special type of context-aware recommendation (CAR) (Adomavicius et al. 2011; Liu et al. 2013b), where the recommendation process takes one of the following three forms, contextual prefiltering, contextual postfiltering, or contextual modeling. Essentially, our POI recommendation model can be classified into the category of contextual modeling, i.e., the spatiotemporal contextual information is used directly in the modeling process.

Although a few recent studies used the embedding techniques for POI recommendation (Chang et al. 2018; Feng et al. 2015; Hang et al. 2018; He et al. 2017a; Liu et al. 2016; Xie et al. 2016), these methods either do not model the joint spatiotemporal effects (Chang et al. 2018; Feng et al. 2015; He et al. 2017a), or cannot address the sparsity or cold-start problems (Feng et al. 2015; He et al. 2017a; Liu et al. 2016; Xie et al. 2016). In this article, we propose a novel spatiotemporal context-aware and translation-based recommender framework (STA) to overcome these issues, inspired by the outstanding performance of the translation-based knowledge graph embedding models in overcoming the data sparsity and scalability as well as their most powerful expressive abilities (Bordes et al. 2014; Lin et al. 2015; Wang et al. 2014). Generally, the translation-based embedding models perform metric learning, which meets the condition of the triangle inequality that is the most crucial to alleviating the data sparsity issue. Specifically, our STA model takes the location and time as a whole context <time, location> to determine a user's choices of POIs. Both users and POIs are embedded as vectors in a latent "translation space," and each spatiotemporal context is represented as a "translation vector" in the same space. Then, a user's check-in behavior is modeled a spatiotemporal translation operation on the embeddings of the user and the corresponding POI. Furthermore, we present a series of strategies to exploit and integrate various correlation information into the translation model so as to learn the approximate embeddings of new spatiotemporal contexts, new users, and new items.

The main contributions of this article are summarized below:

- To the best of our knowledge, we are the first to adopt the translation-based knowledge graph embedding techniques to model the spatiotemporal effects in POI recommendation. The joint modeling of spatiotemporal information also distinguishes our work from existing studies, which consider this information in a separate way.
- We propose a new type of cold-start problem, cold-start spatiotemporal contexts, and develop effective methods to exploit and integrate various correlation information into the representation learning of cold-start users, items, and spatiotemporal contexts to address cold-start problems.
- We conduct comprehensive experiments to evaluate the performance of the proposed STA model over two real-world datasets. The results show the superiority of our STA model in POI recommendation by comparing with the state-of-the-art techniques.

The rest of this article is organized as follows. Section 2 reviews the related work. Section 3 introduces the preliminary and problem definition. Section 4 presents our model in detail. Section 5 gives the experimental results. Finally Section 6 concludes the article.

2 RELATED WORK

In this section, we introduce the related work, organized by the approaches to utilizing the geospatial, temporal information, and embedding learning technique.

2.1 Leveraging Geo-spatial Information

POI recommendation has been an important topic in location-based services. Most existing studies mainly focused on leveraging spatial information due to the well-known strong correlation between users' activities and geographical distance (Cho et al. 2011; Liu et al. 2014a; Ye et al. 2010; Zheng et al. 2009). For example, Ye et al. (2011b) proposed a Bayesian CF algorithm to explore the geographical influence. Cheng et al. (2012) captured the geographical influence by modeling the probability of a user's check-in on a location as a multi-center Gaussian model and then combined it into a generalized matrix factorization model. Liu et al. (2014b) modeled the geographical neighborhood of a location from both the instance-level and the region-level. Lian et al. (2014) adopted a weighted matrix factorization framework to incorporate the spatial clustering phenomenon. Wang et al. (2017) exploited the co-occurrence patterns and content of spatial items at a given time for spatial item recommendation. Wang et al. (2018) investigated the POI-specific geographical influence by incorporating the geo-influence of POI, the geo-susceptibility of POI, and their physical distance into the learning framework.

2.2 Exploring Temporal Effects

Time is another important factor in POI recommendation. Ye et al. (2011a) found the periodic temporal property that people usually went to restaurants around noon and visit clubs at night. Yuan et al. (2013) developed a CF-based model to integrate temporal cyclic patterns. Gao et al. (2013) combined different temporal states into a user's check-in preferences. Cheng et al. (2013) explored the temporal sequential patterns by using the transition probability of two successive check-ins of a user. Zhao et al. (2016) designed a time indexing scheme to capture the specific temporal characteristics for successive POI recommendation.

2.3 Joint Spatiotemporal Effects

Several works (Griesner et al. 2015; Wang et al. 2017; Yin et al. 2016a, 2016b, 2015; Zhao et al. 2016) examined the joint effects of temporal and spatial information. Griesner et al. (2015) extended geographical matrix factorization with temporal dependencies by integrating both geographical and temporal influences into matrix factorization. Zhao et al. (2016) proposed a ranking-based pairwise tensor factorization framework for successive POI recommendation. Yin et al. (2016a, 2016b, 2015) exploited the heterogenous semantic, temporal, and spatial information for the problem of real-time or out-of-town POI recommendation. The ST-SAGE (Wang et al. 2017) presented an additive generative model to integrate personal interests of the users and the preferences of the crowd in the target region at the given time. All the successive, real-time, and out-of-town POI problems are different from the general POI recommendation investigated in this article. More importantly, the CF, MF, Markov transition, and generative models in these studies are ineffective in dealing with the extreme sparsity in POI recommendation, especially when considering the joint spatial and temporal effects.

2.4 Embedding Learning

Several recent works (Chang et al. 2018; Feng et al. 2015; Hang et al. 2018; He et al. 2017a; Liu et al. 2016; Xie et al. 2016; Yin et al. 2017) investigated how to embed items into a low-dimension space. Most of these works are based on inner product rather than Euclidian distance, and they do not follow the triangle inequality, which is critical in addressing data sparsity. For example, the PRME (Feng et al. 2015), SG-CWARP (Liu et al. 2016), and GE (Xie et al. 2016) methods embedded both users and POIs in a common latent space, and users' preference is inferred based on the distance/similarity between a user and a POI. The SH-CDL model (Yin et al. 2017) performed deep representation learning for POIs and additive representation learning for users' spatial-aware personal preferences. Both PRME (Feng et al. 2015) and SH-CDL model (Yin et al. 2017) ignored to investigate the impact of time. Although GE (Xie et al. 2016) and EDHG (Hang et al. 2018) included temporal information, they were modeled with a separate POI-time bipartite graph different from the POI-Region graph indicating that the spatial and temporal embeddings are in different spaces.

The most recent work related with ours is TransRec (He et al. 2017a), which represents the user as a relation vector to capture the transition from the previous item to the next item, and makes recommendations via the nearest neighbor search between the recommended item and the candidates. Although both TransRec (He et al. 2017a) and our model are based on the translation model in knowledge graph completion (Bordes et al. 2014; Lin et al. 2015; Wang et al. 2014), our model differs from TransRec in the following two aspects. Firstly, we view spatial-temporal effects as the translation from user to POI. This modeling is critical to POI recommendation since users' activities are usually influenced by time and location (Gao et al. 2013; Li et al. 2015; Xie et al. 2016; Ye et al. 2011b; Yuan et al. 2013). In contrast, TransRec does not take the spatial and temporal effects into consideration. Secondly, we extend the translation model and develop a series of approaches to explore various correlation information. These approaches are effective in addressing the data sparsity and cold-start problems, which are not tackled in TransRec.

We also notice several new attempts to integrate other types of information like text contents (Chang et al. 2018) and daily activities (Hang et al. 2018). However, these types of information are not always available in POI recommendation and thus are not investigated in our article.

3 PROBLEM DEFINITION AND PRELIMINARY

In this section, we introduce the definitions and preliminaries.

3.1 Problem Definition

*Definition 1 (**POI**).* A POI v is defined as a uniquely identified geographical site with some functions (e.g., a museum or a hotel), and we use V to denote a set of POIs, i.e., $V = \{v\}$.

*Definition 2 (**Check-in activity**).* A check-in activity is a quadruple (u, t, l, v), which means a user u visiting a POI v at geographical region/location l and time t.

*Definition 3 (**Spatiotemporal context**).* A spatiotemporal context, denoted as tl, is a combination of a time slot t and a location l, e.g., <11:00 a.m., Chicago>. Please note that we discretize timestamps associated with check-in records into time slots, e.g., 24 hours in a day, as many other works have done. Time and time slot are used interchangeably in this article. Similarly, we divide the whole spatial space into many geographical regions based on some predefined criteria. We will not distinguish between locations and regions unless strictly necessary. After the discretization, the number of spatiotemporal contexts becomes limited.

*Definition 4 (**TL-translation**).* We define a TL-translation as a relation between user entity u and POI entity v, and the "relation" here has the same meaning as in knowledge graphs. More specifically, a TL-translation means in this context (time u and location l) u tends to visit v.

For ease of presentation, we summarize the notations in Table 1. The POI recommendation problem investigated in this article has the same settings as that in Xie et al. (2016). The formal problem definition is given as follows.

Problem Definition (Location-based Recommendation) Given a dataset $D = \{d | d = (u, t, l, v)\}$ recording a set of users' activities, and a query $q = (u_q, t_q, l_q)$, we aim to recommend top-k POIs in V that the query user u_q would be interested in.

3.2 Preliminary

In this subsection, we introduce the background knowledge on translation-based knowledge graph embedding.

Knowledge graphs encode structured information of entities and their relations in the form of triples (head_entity, relation, tail_entity). For simplicity, we use (h, r, t) to denote a fact in a knowledge graph. Although a knowledge graph may contain millions of entities and billions of relational facts between these entities, it is usually far from complete. Recently, the task of knowledge graph completion has attracted much attention, and its objective is to model and predict

Table 1. Notations Used in This Article

Variable	Interpretation
u	a user
v	a POI
t	a time slot
l	a location/region
tl	a spatiotemporal context <t, l>
$\vec{u}, \vec{tl}, \vec{v}$	embeddings of u, tl, and v
u_q, t_q, l_q	query user u_q, his/her current time t_q, and location l_q
v_q	the POI that query user u_q will visit
D	a collection of user activity records $D = \{d \| d = (u, t, l, v)\}$
T	a collection of user activity triples $T = \{d \| d = (u, tl, v)\}$
U	a set of users
V	a set of POIs
TL	a set of spatiotemporal contexts

relations between pairs of entities. A promising approach for the task is embedding a knowledge graph into a continuous vector space while preserving structured information of the graph (Bordes et al. 2014; Lin et al. 2015; Wang et al. 2014). Bordes et al. (2014) presented a simple yet effective embedding model TransE, which embeds entities as points in a low-dimensional latent space and relations as translation vectors such that the relationship between two entities is captured by the corresponding translation operation. For example, given the fact that Paris is the capital of France ("capital_of" is the relation between "Paris" and "France"), if we represent "Paris", "France", and "capital_of" with vectors $\vec{h}$, $\vec{t}$, and $\vec{r}$, respectively, then this fact or triple is captured by a translation operation: the embedding of head entity h, plus the translation vector of relation r, determine (approximately) the embedding of the tail entity t, i.e., $\vec{h} + \vec{r} \approx \vec{t}$. As TransE is problematic in modeling "1-to-N", "N-to-1", and "N-to-N" relations, TransH was proposed to enable an entity having different representations when involved in different relations (Lin et al. 2015).

Both TransE and TransH project all entities and relations into the same space. However, some entities may have multiple aspects, and various relations focus on different aspects of entities, which makes a common space insufficient for modeling. It is intuitive that two entities are similar in some aspects and thus close to each other in these relation spaces, but are different in other aspects and thus should be far away from each other in the corresponding relation spaces. In light of this, Lin et al. (2015) proposed the TransR model to build entity and relation embeddings in separate entity space and relation spaces. Specifically, TransR learns embeddings by first projecting entities from entity space ($\vec{h}$ and $\vec{t}$) to a relation-specific space ($\vec{h_r}$ and $\vec{t_r}$) with relation-specific projection matrix M_r, and then building translations between projected entities, i.e., $\vec{h_r} + \vec{r} \approx \vec{t_r}$.

4 TRANSLATION-BASED POI RECOMMENDER MODEL

In this section, we first intuitively describe our STA model and then formally present the model optimization. After that, we detail how to provide top-k online POI recommendation based on the trained STA.

4.1 Model Description and Optimization

We aim to build a model that (1) naturally captures spatiotemporal context-aware user check-in behavior, and (2) is able to address the issues of data sparsity and cold start. Inspired by the

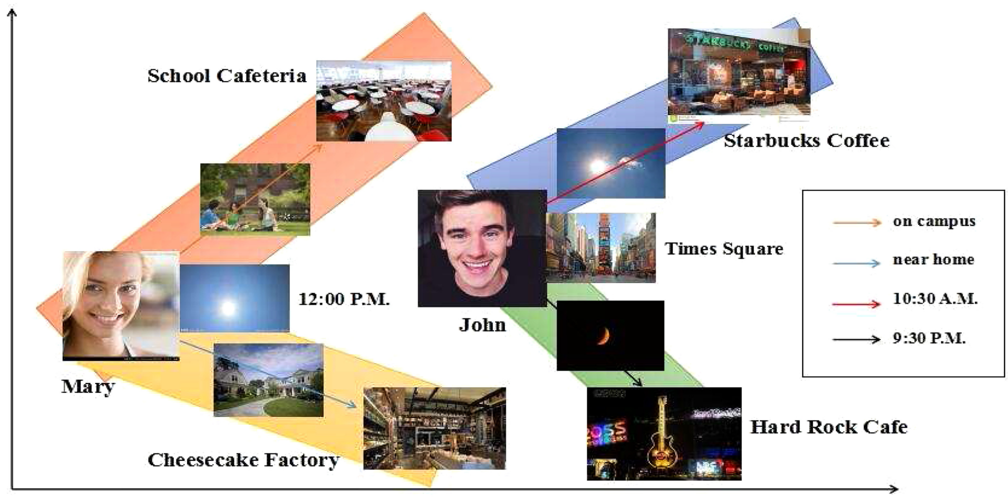


Fig. 1. Impacts of spatiotemporal patterns.

outstanding performance of translation-based knowledge graph embedding models in the task of sparse knowledge graph completion (Bordes et al. 2014; Lin et al. 2015; Wang et al. 2014), we propose a *spatiotemporal context-aware* and translation-based POI recommender framework, called STA. We first combine each time slot and location as a spatiotemporal context $< t, l >$ (tl for short), and convert the quadruples $(u, t, l, v) \in D$ into triples $(u, <t, l>, v)$ in T, which are analogous to fact triples (h, r, t) in a knowledge graph.

To model the spatiotemporal context-aware check-in behaviors, we represent each spatiotemporal context with a translation vector $\vec{tl}$ to capture the spatiotemporal effect that influences users to make choices of POIs to visit. Intuitively, if a POI v is often chosen by users under the spatiotemporal context tl, the probability of a query user u_q visiting v with the same spatiotemporal context will be high. On the other hand, users may visit different POIs under different spatiotemporal contexts. Figure 1 illustrates the spatiotemporal effects.

In Figure 1, the left user Mary may visit different restaurants at noon depending on her location, and the right user John, near Times Square, would like to have a cup of coffee at Starbucks Coffee in the morning and go to the bar at night. This indicates that the time and location jointly determine the user's choice. In other words, it is the spatiotemporal context that connects a user to a POI. Similar to the relation "Paris + Capitalof ⇒ France" in the knowledge graph, we represent the spatiotemporal context $< t, l >$ as a type of relation such that it captures the user's check-in behaviors at the specific time and location. Thus, based on the figure, we have:

"Mary + 12:00 P.M. on campus ⇒ School Cafeteria"
"Mary + 12:00 P.M. near home ⇒ Cheesecake Factory"
and
"John + 10:30 A.M. Times Square ⇒ Starbucks Coffee"
"John + 9:30 P.M. Times Square ⇒ Hard Rock Cafe"

The above examples illustrate the basic idea of our proposed STA: a user u will reach an interested POI v_q via a translation edge tl, i.e., $\vec{u} + \vec{tl} \approx \vec{v_q}$. As a user has multiple interests and tends to show different interests in different spatiotemporal contexts (Yin et al. 2017, 2016b), this makes a common space insufficient for modeling. Following the idea in TransR model (Lin et al. 2015), we choose to model user/POI entities and spatiotemporal contexts in distinct spaces, that is one common entity space and multiple spatiotemporal context spaces (i.e., spatiotemporal context-specific entity spaces), and performs translation in the corresponding spatiotemporal context space. Specifically, for each triple (u, <t, l>, v) in T, the embeddings of user u and POI v are set as $\vec{u}, \vec{v} \in \Re^d$, and the embedding of spatiotemporal context tl is set as $\vec{tl} \in \Re^m$. Note that, the dimensions of user and POI embeddings and spatiotemporal context embeddings are not necessarily identical. For each spatiotemporal context tl, we assign a projection matrix $M_{tl} \in \Re^{d \times m}$, which projects users and POIs from the original common entity space to the spatiotemporal context-specific embedding space. With the mapping matrix, we define the projected vectors of users and POIs as $\vec{u}_{tl} = \vec{u}M_{tl}$ and $\vec{v}_{tl} = \vec{v}M_{tl}$ in the spatiotemporal context space, and then perform translation in the corresponding spatiotemporal context space as in $\vec{u}_{tl} + \vec{tl} \approx \vec{v}_{tl}$. This indicates that a POI embedding $\vec{v}_{tl}$ should be the nearest neighbor of $\vec{u}_{tl} + \vec{tl}$. The score function is then defined as:

$$s_{tl}(u, v) = \left\| \vec{u}_{tl} + \vec{tl} - \vec{v}_{tl} \right\|_2^2 \tag{1}$$

To avoid overfitting and learning model parameter values that are too large, we adopt the practice in Lin et al. (2015) to add constraints on the norms of the embeddings u, v, tl, and the projection matrices, i.e., for $\forall u, v, th$, we have $\| \vec{u} \|_2 \leq 1$, $\| \vec{v} \|_2 \leq 1$, $\| \vec{tl} \|_2 \leq 1$, $\| \vec{u}_{tl} \|_2 \leq 1$ and $\| \vec{v}_{tl} \|_2 \leq 1$.

Given the score function defined in Equation (1) for a triple (u, tl, v), the entire objective function for model optimization is as follows.

$$L = \sum_{(u,tl,v) \in S} \sum_{(u',tl,v') \in S'} max(0, s_{tl}(u, v) + \gamma - s_{tl}(u', v')), \tag{2}$$

where $max(a, b)$ is used to getting the maximum between a and b, γ is the margin, and S and S' are the sets of positive and corrupted triples, respectively. The corrupted triples are generated by replacing the head and tail entities in positive triples with the dissimilar user and POI.

We adopt stochastic gradient descent (SGD) (in mini-batch mode) to minimize the objective function in Equation (2). A small set of triples are first sampled from the training data. For each sampled positive triple, we generate the corresponding corrupted triples. All the positive and corrupted triples are put into a mini-batch. We compute the gradient and update the parameters after each mini-batch. When the iteration reaches a predefined number, we learn all the embedding for users, POIs, and spatiotemporal contexts.

Algorithm We now describe the optimization procedure of our STA model in Algorithm 1. Lines 1–4 initialize the embeddings for users, POIs, spatiotemporal contexts, and projection matrix using the random procedure as that in Glorot and Bengio (2010). In each main iteration, lines 5–16 are used for updating parameters in the model. Specifically, lines 6–8 normalize the embedding vectors. Line 9 samples a small set of triplets from the training set. These triplets are used as the positive training triplets of the mini-batch. Lines 11–13 sample a single corrupted (negative) triplet for each positive triplet and compose a pair of (positive, negative) samples. Line 15 updates the parameters by taking a gradient step.

Discussion about the advantages of our STA model As the score in Equation (1) is computed using the Euclidean distance, our STA model learns a metric space where *neighborhood* of a point p (i.e., the set of points whose distances to p are less than a predefined threshold) captures the notion of similarity, and *translation* encapsulates the spatiotemporal transition relation between users and POIs.

There are a number of distance metrics such as discrete metric, Euclidean metric, and Graph metric, and we adopt Euclidean distance. In the Euclidean space $\Re^k$, the Euclidean distance d

ALGORITHM 1: Learning STA

Require: Training set $S = \{(u, tl, v)\}$, user, POI, and spatiotemporal context sets U, V, and TL, projection matrix M_{tl}, margin γ, embeddings dimension k.

initialize $tl \leftarrow uniform(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each $tl \in TL$
  $u \leftarrow uniform(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each user $u \in U$
  $v \leftarrow uniform(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each POI $v \in V$
  $M_{tl} \leftarrow$ identity matrix
loop
  $tl \leftarrow tl/\|tl\|$ for each $tl \in TL$
  $u \leftarrow u/\|u\|$ for each entity $u \in U$
  $v \leftarrow v/\|v\|$ for each entity $v \in V$
  $S_{batch} \leftarrow sample(S, b)$ // sample a mini-batch of size b
  $T_{batch} \leftarrow \emptyset$ // initialize the set of pairs of triplets
  for each $(u, tl, v) \in S_{batch}$ **do**
    $(u', tl, v') \leftarrow sample(S'_{(u,tl,v)})$ // sample a corrupted triplet
    $T_{batch} \leftarrow T_{batch} \bigcup \{((u, tl, v), (u', tl, v'))\}$
  end for
  Update embeddings w.r.t $\sum_{((u,tl,v),(u',tl,v')) \in T_{batch}} \nabla[s_{tl}(u, v) + \gamma - s_{tl}(u', v')]_+$
end loop

between two objects u and v is defined as:

$$d(u, v) = \| \vec{u} - \vec{v} \|_2 = \left(\sum_{i=1}^{k} |u_i - v_i|^2 \right)^{1/2} \tag{3}$$

To avoid computing the square root, we further adopt the squared Euclidean distance as shown in Equation (1). Unlike the dot product, which has been used in many previous studies, the Euclidean distance satisfies the triangle inequality. Dot product cares about the magnitudes and angles of two vectors. In some cases, it has the disadvantages that two similar objects represented by two vectors have to be positioned far away from each other. Please refer to the illustration example in He et al. (2017b). In contrast, if two similar objects are treated as two points in the same space and their distance is measured by the Euclidean distance, then the triangle inequality will ensure the closeness of these two objects.

The inherent triangle inequality assumption plays a key role in overcoming the data sparsity and improving model generalization, as it does in canonical metric learning scenarios (He et al. 2017a). For instance, if a spatiotemporal context tl tends to transition from users to two POIs v and v' (which means that many users tend to visit both v and v' in the spatiotemporal context tl), then our model STA will also put v close to v' in the tl-specific space. This is a desirable property for addressing the data sparsity since our model STA can infer that a user u who often visits v in the spatiotemporal context tl will be also most likely to visit v' in the same spatiotemporal context, which significantly improves the model generalization.

4.2 Recommendation Using STA

Once we have learned the embeddings, given a query user u_q with the query time t_q and location l_q, i.e., $q = (u_q, t_q, l_q)$, we first combine t_q and l_q as a joint spatiotemporal context tl_q, and then we can get the embedding of the ideal POI v_q using Equation (4).

$$\vec{v_q} = \vec{u_q} M_{tl} + \vec{tl}_q \tag{4}$$

For each POI $v \in V$, we compute its distance to the ideal POI v_q in the tl-specific space as defined in Equation (5), and then select the k POIs with the smallest distances as recommendations.

$$d(v, v_q) = \| \vec{v} M_{tl} - \vec{v}_q \|_2^2 \tag{5}$$

It should be noted that the above method for producing top-k online recommendation is different from the conventional recommendation methods such as Lin et al. (2015) and Xie et al. (2016). First, we can generate an explicit representation of an ideal POI v_q in the latent space through the specific spatiotemporal translation of the user's embedding. Second, since the embeddings for POIs in V are also from the same space, we can choose the ones that are the closest neighbors of v_q in this space. This indicates that our recommended POIs are semantically consistent with the ideal POI v_q.

5 COLD-START POI RECOMMENDATION

Cold-start problems create severe challenges for POI recommendation. Recent studies incorporated the side information into CF, MF, and graph embedding approaches to address both cold-start users and cold-start items (Gao et al. 2012; Lian et al. 2014; Xie et al. 2016; Ye et al. 2011b). However, no effort has been made along the direction of the translation-based model in POI recommendation. Besides, there does not exist any work on cold-start spatiotemporal contexts. In this section, we present a set of strategies to exploit various correlations between cold-start users, POIs, spatiotemporal contexts, and their warm-start counterparts. Then, for these cold-start users, POIs, and spatiotemporal contexts, we leverage the check-in records associated with their most similar/dissimilar warm-start counterparts to generate additional training data to learn their embeddings.

5.1 Exploiting Spatiotemporal Correlation

The cold-start spatialtemporal contexts refer to those new time-location pairs that have never appeared in the training dataset. By investigating multiple check-in datasets, we find that almost all individual spatial and temporal contexts are not new although their combinations are cold start.

In order to build the embedding for an unseen check-in $\tau =< t, l >$ at the specific time t and location l, we leverage the contexts for finding its nearest and farthest neighbors. One way to exploit the contextual information between two check-ins is based on their spatial and temporal similarity. More specifically, we propose to define the temporal similarity sim_t based on users' check-in activities on time t_i and t_j.

$$sim_t(t_i, t_j) = \frac{\sum_{u \in U} \frac{\vec{a}_i \cdot \vec{a}_j}{||\vec{a}_i||_2 ||\vec{a}_j||_2}}{|U|}, \tag{6}$$

where $\vec{a}_i$ and $\vec{a}_j$ are the vector of POIs visited by the same user u at time t_i and t_j, respectively. We further define the spatial similarity sim_l as the check-in probability under the geographic distance between l_i and l_j. As pointed out in Ye et al. (2011b), the check-in probability may follow the power-law distribution. Thus, we use power law distribution to model the check-in probability to the distance between two POIs visited by the same user.

$$sim_l(l_i, l_j) = a * (d(l_i, l_j))^b, \tag{7}$$

where the $d(.,.)$ denotes the geographic distance, and a and b ($a > 0, b < 0$) are the coefficients of a power-law distribution, which can be determined by a linear curve fitting method

(Ye et al. 2011b). We then use a linear combination method to combine the spatial and temporal similarities between two spatiotemporal contexts $\tau_i =< t_i, l_i >$ and $\tau_j =< t_j, l_j >$.

$$sim_{tl}(\tau_i, \tau_j) = \alpha_1 \cdot sim_t(t_i, t_j) + \alpha_2 \cdot sim_l(l_i, l_j), \tag{8}$$

5.2 Exploiting User Correlation

Normally, users who have not visited any POIs are called cold-start users. However, different from the data investigated in Liu and Xiong (2013), Yin et al. (2013), and Gao et al. (2015b), which have users' comments or tweets, the two datasets used in our experiments do not contain any additional content information for users.

More formally, the problem of cold-start user is defined as: given a cold-start user u_q, his/her friend set $N(u_q)$, and his/her current time t_q and location l_q, the goal is to recommend the most likely POIs to u_q. We propose two geo-social correlation measures to incorporate a user's social and geographical contexts.

$$sim_u(u_q, u_k) = \lambda_1 \cdot sim_{soc}(u_q, u_k) + \lambda_2 \cdot sim_{geo}(u_q, u_k) \tag{9}$$

We employ the normalized ratio of common friends in two users' social circles as the similarity metric of social influence.

$$sim_{soc}(u_q, u_k) = \frac{|N(u_q) \bigcap N(u_k)| + 1}{\sum_{k=1}^{k=|N(u_q)|}(|N(u_q) \bigcap N(u_k)| + 1)}, \tag{10}$$

The geographical similarity sim_{geo} is similar to the definition in Equation (7). The only difference is that l_i and l_j here denote the current location of user u_q and the home location of u_k. Note the home location is not presented in our data. However, for a normal user u_k who has check-in records, we define it as the average position of check-ins in the 25km by 25km cell with the most check-ins (Scellato et al. 2011).

5.3 Exploiting POI Correlation

The cold-start POIs refer to those containing geographic and content (tag) information but do not have any check-ins (Xie et al. 2016). For evaluation, cold-start POIs are defined as those visited by less than five users (Yin et al. 2016b). The problem of cold-start POI is defined as: given a cold-start POI p_q, its tag set $T(p_q)$, and its location l_q, the goal is to learn the latent vector for p_q, which can be used in making recommendations to users who may prefer it according to its semantic and geographical attributes.

We propose two geo-semantic correlation measures to incorporate the semantic and geographical contexts of a POI.

$$sim_p(p_q, p_k) = \gamma_1 \cdot sim_{sem}(p_q, p_k) + \gamma_2 \cdot sim_{geo}(p_q, p_k) \tag{11}$$

We once again adopt Equation (7) to compute the geographical similarity $sim_{geo}(p_q, p_k)$ between the locations of two POIs. The semantic similarity sim_{sem} is defined as the Jaccard coefficient between the tag set $T(p_q)$ and $T(p_k)$ for POI p_q and p_k.

$$sim_{sem}(p_q, p_k) = \frac{|T(p_q) \bigcap T(p_k)|}{|T(p_q) \bigcup T(p_k)|} \tag{12}$$

5.4 Learning Embedding in Cold-Start Setting

We incorporate the contextual information into representation learning via positive and negative sampling to help address the sparsity and cold-start problems. The intuition is that when we encounter the unseen check-ins or the cold-start problems, we sample from the most similar or dissimilar examples instead. We here take the process for handling new $< t, l >$ spatiotemporal contexts as an example.

Based on the similarity function in Equation (8), we define the nearest and farthest spatiotemporal contexts τ_m^n and τ_m^f of a new spatiotemporal context $\tau_m = tl_m$.

$$\begin{aligned} \tau_m^n &= \max_{\tau_m^i} sim_{tl}(\tau_i, \tau_m), \\ \tau_m^f &= \min_{\tau_m^i} sim_{tl}(\tau_i, \tau_m) \end{aligned} \tag{13}$$

We then extend our basic STA model to learn the embeddings for new spatiotemporal contexts. For a cold-start spatiotemporal context τ_m, we first find its nearest and farthest warm-start spatiotemporal contexts (e.g., τ_m^n and τ_m^f) as well as their associated check-in records, and then use τ_m to replace τ_m^n and τ_m^f in these associated check-in records to form new positive and negative examples in the training dataset, respectively.

We finally train on this new training data using the same method as basic STA, and we can get the embedding for the new spatiotemporal context tl_m. Similar procedure can be applied to building the embedding for the cold-start user u_q and the cold-start POI p_q.

We have six coefficients in Equations (8), (9), and (11), where $\alpha 1$ and $\alpha 2$, $\lambda 1$ and $\lambda 2$, and $\gamma 1$ and $\gamma 2 \in [0, 1]$ control the relative importance of the spatial and temporal, the social and geographic, and the semantic and geographic similarities, respectively. For simplicity, we just set them to the same value 0.5. We support the fine tuning of arbitrary values of these parameters.

For the approaches to dealing with unseen check-ins and cold-start problem, we call them STA-C-TL (STA for cold-start check-ins), STA-C-U (STA for cold-start users), and STA-C-P (STA for cold-start POIs), respectively, and we include the strategy for addressing new spatiotemporal contexts into the STA model due to its generalizability to basic settings.

Our STA-C-TL model can address the problem of cold-start spatiotemporal contexts. This issue has not been tackled in the previous study.

We notice several related works have been proposed to incorporate social network into POI recommendations (Gao et al. 2012; Li et al. 2016; Ye et al. 2010). Our STA-C-U is different from these studies in that the social connections are only used to compute the user's representation in our model, the recommendation of POI is computed based on this user's current time and location. In contrast, existing approaches directly leverage the historical check-ins of their friends. Moreover, while all these methods incorporate geographic information, none of them has considered the time factor for POI recommendation.

Our STA-C-P model proposed for dealing with cold-start POIs can also be applied to the normal POI recommendation problem. However, it requires that those POIs should contain content information. For the recommendation on datasets like Gowalla, STA-C-P is not valid. Hence, we treat it as an extended model. Please also note that it is STA-C-P that uses the same information as GE (Xie et al. 2016) does. Our standard STA model, on the other hand, uses less information than GE as it does not include the contents of POIs.

6 EXPERIMENTS

In this section, we first introduce the experimental setup and then compare our experimental results with the baselines. Finally, we show the performance of our method in addressing the data sparsity and cold-start problem.

Table 2. Statistics of Two Datasets

	Foursquare	**Gowalla**
# of users	114,508	107,092
# of POIs	62,462	1,280,969
# of check-ins	1,434,668	6,442,892
#std time slots	24	24
# of locations	5,846	200
# of $< t, l >$ contexts	28,868	3,636

6.1 Datasets

We evaluate our methods on two real-life LBSN datasets: Foursquare and Gowalla. A number of researchers have conducted experiments on the data collected from these two social networks (Chen et al. 2015; Gao et al. 2015b; Xie et al. 2016; Yin et al. 2016b; Yuan et al. 2013). However, many of them are collected from various regions or in different time spans. In this article, we use the publicly available version[1] provided by the authors of Xie et al. (2016).

The two datasets have different scales such as geographic ranges, the number of users, POIs, and check-ins. Hence, they are good for examining the performance of algorithms on various data types. Their statistics are listed in Table 2.

Each check-in is stored as user-ID, POI-ID, POI-location in the form of latitude and longitude, check-in timestamp, and POI-content (only for Foursquare). In order to get the spatiotemporal contexts $< t, l >$ in Table 2, we use the same discretization method as that in Gao et al. (2013) and Xie et al. (2016), i.e., dividing time into 24 time slots that correspond to 24 hours, and the whole geographical space into a set of regions according to 5,846 administrative divisions (for Foursquare) and 200 regions clustered by a standard k-means method (for Gowalla). We finally get 28,868 and 3,636 $< t, l >$ pairs on Foursquare and Gowalla, respectively.

6.2 Evaluation Metrics

To thoroughly evaluate the models, we use two widely-used metrics, i.e., recall (*Rec@k*) and normalized discounted cumulative gain (*NDCG@k*). While recall indicates the ratio of recovered POIs to the visited locations, NDCG measures the ranking quality, which assigns higher scores to POIs at top position ranks. We do not include the precision into the evaluation scheme because precision is not a suitable performance measure in the field of POI recommendation. As pointed out in Wang et al. (2015), *Precision@k* takes the zero entries into consideration. However, a zero entry may be due to the fact that the user is not interested in the POI, or that the user is totally not aware of its existence. Thus, it is most likely for *Precision@k* to underestimate the real recommendation accuracy. Besides, the measure *Precision@k* naturally favors users with many POIs in the test set, as it is easier for the recommender model to achieve higher precision values for those users with more POIs in the test set than the users with less POIs.

To produce a top-k recommendation list for a query user, we compute a preference score for each POI and sort them by score. The recall@k for each user is defined as:

$$Rec@k = \frac{tp}{tp + tn}, \tag{14}$$

[1] https:/sites.google.com/site/dbhongzhi

where tp is the number of POIs visited by a user u and also in the top-k recommendations, and tn is the number of POIs visited by u but not in the top-k recommendations.

The $NDCG@k$ for each user is defined as:

$$NDCG@k = \frac{DCG@k}{IDCG@k}, \tag{15}$$

where

$$DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{log_2(i+1)}, \tag{16}$$

and rel_i refers to the graded relevance of the result ranked at position i. We use the binary relevance in our work, i.e., $rel_i = 1$ if the result is in the test set, and 0, otherwise. $IDCG@k$ is the $DCG@k$ value when the recommended POIs are ideally ranked.

The average of recall and NDCG values over all users are reported as the final $Rec@k$ and $NDCG@k$ ($k = \{5, 10, 20, 30, 50\}$). These two metrics are both in the range $[0, 1]$ and a higher value means better results.

6.3 Comparison Methods

We compare our STA with 10 POI recommendation models in the experiment. They represent the state-of-the-art methods: firstly, they cover four types of popular recommendation techniques, i.e., *collaborative filtering*, *matrix factorization*, *distributed representation*, and *hybrid model*; secondly, they consider six important factors that influence user decision-making for choosing POIs, including *user preference*, *temporal*, *geographical*, *social*, *content*, and *sequential influence*. The baselines are categorized by the recommendation techniques and listed below.

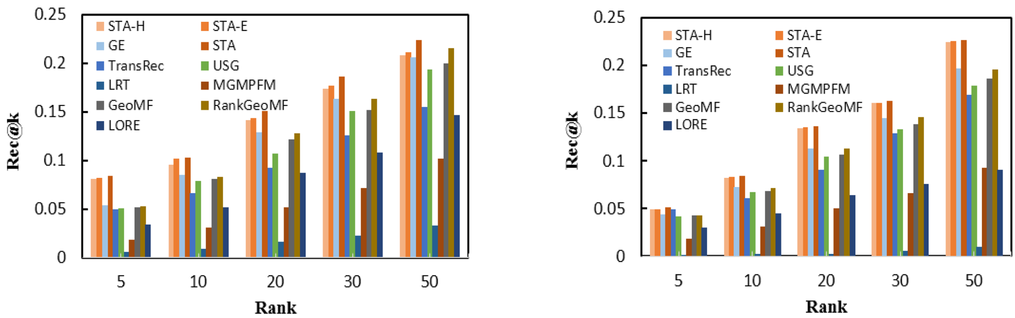
Collaborative Filtering Baseline—{USG} CF is a traditional and widely-adopted technique for recommender systems. USG (Ye et al. 2011b) presents a unified framework to perform collaborative recommendation, which fuses user preference, social influence, and geographical influence in POI recommendation.

Matrix Factorization Baselines—{LRT, GeoMF, RankGeoMF} MF is a widely adopted approach for modeling user preferences to recommendation. LRT (Gao et al. 2013) is selected as a baseline because it is a time-enhanced MF model which utilizes the temporal properties of user movement. GeoMF (Lian et al. 2014) is a weighted MF model that integrates the geographical influence by modeling users' activity regions and the influence propagation on geographical space. RankGeoMF (Li et al. 2015) is a ranking based MF model that fits the users' preference rankings for POIs to learn the latent factors of users and POIs.

Distributed Representation Baselines—{GE, TransRec, STA-E, STA-H} Distributed representation uses low-dimensional dense vectors to represent data points. GE (Xie et al. 2016) adopts a graph-based embedding framework to integrate the sequential, geographical, temporal cyclic, and semantic effect into a shared space. TransRec (He et al. 2017a) embeds items into a transition space where users are modeled as translation vectors operating on item sequences.

Also note that although we choose the TransR technique in knowledge graph embedding to materialize our STA model, the essence of our proposed framework is to model the spatiotemporal context as a translation relation in the embedding space. This indicates that we do not rely on a specific translation model. Hence, we can use TransE (Bordes et al. 2014) or TransH (Wang et al. 2014) to realize STA. We denote the resulting methods as STA-E and STA-H baselines, respectively.

Hybrid Model Baselines—{LORE, MGMPMF} Hybrid models refer to those which combine the outputs of two or more recommendation methods. For example, MGMPFM (Cheng et al. 2012) is a fusion model combining the outputs of Poisson factor model and a geographical modeling



(a) Comparision on Foursquare Dataset

(b) Comparision on Gowalla Dataset

Fig. 2. Comparisons with the baselines on the two datasets in terms of Rec@K.

method. LORE (Zhang et al. 2014) employs additive Markov chain, collaborative filtering, and kernel density estimation to integrate the sequential, social, and geographical influences for POI recommendations.

6.4 Experimental Settings

We first organize the quadruples (u, v, t, l) in each dataset by users to get each user's profile D_u. We then rank the records in D_u according to the check-in timestamps, and finally divide these ordered records into two parts: the first 80% as the training data, and the remaining 20% of the data as the test data. Moreover, the last 10% of the check-in records in the training data are used as a validation set for tuning the hyper-parameters.

We use the default settings in the original TransR (Lin et al. 2015) as the parameter settings for our STA model. Specifically, we set the learning rate $\lambda = 0.0001$, the margin $\gamma = 2$, the mini-batch size $B = 4,800$, and the embedding dimensions $m = d = 100$, and we traverse over all the training data for 1,000 rounds. The same settings are also used for STA-E and STA-H models.

The parameters for other baselines are listed as follows. The notations of these parameters correspond to the definitions, and the values are the same as those in their original papers.

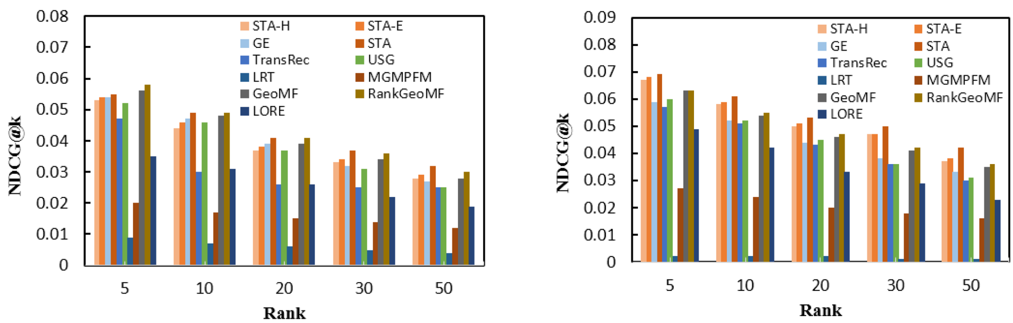
GE: $d = 100$, $N = 150M$
USG: $\alpha = 0.1$, $\beta = 0.1$, $\eta = 0.05$
LRT: $\alpha = 2.0$, $\beta = 2.0$, $\lambda = 1$, $K = 100$, $T = 1day$
LORE: $\alpha = 0.05$, $T = 1day$
TransRec: $\alpha = 0.2$, $K = 10$, $\lambda_\Theta = 0.1$, $\epsilon = 0.05$
MGMPMF: PMF: $\alpha = 20.0$, $\beta = 0.2$, $K = 30$, MGM: $\alpha = 0.2$, $\theta = 0.02$, $d = 15$
GeoMF: $\alpha = 0.01$, $\gamma = 0.01$, $\delta = 15$, $\lambda = 10$, $iters = 10$, $K = 100$, $L = 50 \times 50$
RankGeoMF: $\alpha = 0.2$, $\epsilon = 0.3$, $\gamma = 0.0001$, $K = 100$, $C = 1.0$, $nearestPOIs = 300$

6.5 Comparison with Baselines

We present the comparison results on the two datasets in terms of recall and NDCG in Figure 2 and Figure 3, respectively.

From Figure 2, we can make the following important observations.

— Our proposed STA-style models significantly and consistently outperform all baselines in terms of recall on both datasets. For example, the *Rec*@5 values achieved by STA, STA-H, and STA-E on the Foursquare dataset are 0.084, 0.081, and 0.0.082, respectively, much better than 0.054 for GE, which is the best among other baselines. Our STA model also achieves



(a) Comparision on Foursquare Dataset

(b) Comparision on Gowalla Dataset

Fig. 3. Comparisons with the baselines on the two datasets in terms of NDCG@K.

approximate 38%, 37%, and 39% relative improvements over two state-of-the-art MF-based methods, GeoMF and RankGeoMF, and a CF-based method, USG, respectively.

— Among all MF-based methods, LRT performs worst. The reason may be that LRT only considers temporal information but neglects the geographical and social effects. On the other hand, though the fused method MGMPFM combines geographic and personal interest, its poor results show that separately modeling two types of influencing factors and then fusing their outputs does not contribute much to the performance. Similar results can be observed for LORE on Gowalla dataset. Besides the loss functions used in the matrix factorization models, LRT and MGMPFM were originally designed for rating prediction rather than top-k recommendation, which also partly accounts for their poor performance.
— Although TransRec is a translation-based method, its performance is significantly worse than our STA model. This might be due to their different modeling strategies. TransRec takes the previous item (POI) as the critical factor in determining user's next activity. This may be appropriate for basket recommendation like buying a mouse after a desktop. However, in POI recommendation, time and location play more important roles. Our STA model is preferable than TransRec for the task of POI recommendation due to its ability in joint modeling spatiotemporal contexts.

While our STA model shows drastic improvements over baselines in terms of $Rec@k$ on both Foursquare and Gowalla datasets, its $NDCG@5$ value (0.055) is slightly lower than that of GeoMF (0.056) and RankGeoMF (0.058) on Foursquare in Figure 3(a). (Detailed values are shown in Table 3). However, it still outperforms all other baselines for $NDCG@5$. In addition, for other $NDCG@k$ ($k = 10, 20, 30, 50$), STA model gets the best performance. More importantly, in terms of NDCG metric on Gowalla dataset in Figure 3(b), our model significantly outperforms all baselines by a large margin for all k settings. For instance, in terms of $NDCG@5$, STA gains an improvement of 9%, 13%, and 15% over RankGeoMF (GeoMF), USG, and GE, respectively, which are the top-3 among all baselines.

6.6 Sensitivity to Data Sparsity

To investigate the sensitivity of STA and other methods to data sparsity, we conduct extensive experiments to evaluate their performance on two datasets by reducing the amount of training data. Specifically, we keep the testing dataset unchanged and reduce the training data randomly by a ratio of 5% to 20% stepped by five. For clarity, we only present the results on Foursquare and Gowalla by reducing 20% training data in Table 3 and Table 4, respectively, and the trends with

Table 3. Sensitivity to Data Sparsity on Foursquare (- for 20% Less Training Data)

Rec \ M k	USG		LORE		GeoMF		RankGeoMF		TransRec		GE		STA	
	USG	USG-	LORE	LORE-	Geo	Geo-	Rank	Rank-	Trans	Trans-	GE	GE-	STA	STA-
5	0.051	0.048	0.034	0.026	0.052	0.048	0.053	0.047	0.047	0.043	0.054	0.051	**0.084**	**0.082***
10	0.079	0.077	0.052	0.045	0.081	0.075	0.083	0.076	0.072	0.065	0.085	0.083	**0.103**	**0.101***
20	0.107	0.105	0.087	0.072	0.122	0.114	0.128	0.115	0.095	0.076	0.129	0.125	**0.151**	**0.148***
30	0.151	0.148	0.108	0.093	0.152	0.144	0.163	0.146	0.144	0.119	0.163	0.160	**0.186**	**0.182***
50	0.193	0.190	0.146	0.124	0.200	0.190	0.215	0.195	0.186	0.159	0.216	0.212	**0.224**	**0.220***
NDCG \ M k	USG		LORE		GeoMF		RankGeoMF		TransRec		GE		STA	
	USG	USG-	LORE	LORE-	Geo	Geo-	Rank	Rank-	Trans	Trans-	GE	GE-	STA	STA-
5	0.052	0.050	0.035	0.028	0.056	0.051	**0.058**	0.050	0.051	0.045	0.054	0.051	0.055	**0.053***
10	0.046	0.046	0.031	0.025	0.048	0.044	**0.049**	0.043	0.044	0.038	0.047	0.043	**0.049**	**0.047***
20	0.037	0.035	0.026	0.022	0.039	0.036	**0.041**	0.036	0.036	0.030	0.039	0.036	**0.041**	**0.040***
30	0.031	0.029	0.022	0.019	0.034	0.032	0.036	0.032	0.030	0.021	0.032	0.029	**0.037**	**0.035***
50	0.025	0.023	0.019	0.016	0.028	0.026	0.030	0.026	0.023	0.015	0.027	0.024	**0.032**	**0.030***

Table 4. Sensitivity to Data Sparsity on Gowalla (- for 20% Less Training Data)

Rec \ M k	USG		LORE		GeoMF		RankGeoMF		TransRec		GE		STA	
	USG	USG-	LORE	LORE-	Geo	Geo-	Rank	Rank-	Trans	Trans-	GE	GE-	STA	STA-
5	0.042	0.040	0.030	0.024	0.043	0.038	0.043	0.037	0.041	0.038	0.044	0.040	**0.051**	**0.050***
10	0.067	0.063	0.045	0.037	0.068	0.063	0.071	0.062	0.063	0.057	0.072	0.068	**0.084**	**0.082***
20	0.104	0.099	0.064	0.052	0.106	0.099	0.113	0.102	0.095	0.084	0.113	0.108	**0.136**	**0.134***
30	0.133	0.129	0.076	0.060	0.138	0.129	0.146	0.133	0.131	0.121	0.145	0.141	**0.163**	**0.162***
50	0.179	0.173	0.090	0.070	0.186	0.175	0.195	0.180	0.175	0.152	0.197	0.193	**0.226**	**0.224***
NDCG \ M k	USG		LORE		GeoMF		RankGeoMF		TransRec		GE		STA	
	USG	USG-	LORE	LORE-	Geo	Geo-	Rank	Rank-	Trans	Trans-	GE	GE-	STA	STA-
5	0.060	0.058	0.049	0.039	0.063	0.055	0.063	0.052	0.057	0.052	0.059	0.056	**0.069**	**0.067***
10	0.052	0.050	0.042	0.033	0.054	0.048	0.055	0.047	0.051	0.045	0.052	0.049	**0.061**	**0.060***
20	0.045	0.043	0.033	0.027	0.046	0.041	0.047	0.041	0.043	0.037	0.044	0.041	**0.053**	**0.051***
30	0.036	0.034	0.029	0.023	0.041	0.037	0.042	0.037	0.036	0.031	0.038	0.034	**0.050**	**0.048***
50	0.031	0.030	0.023	0.018	0.035	0.032	0.036	0.032	0.030	0.025	0.033	0.030	**0.042**	**0.040***

other ratios are quite similar. We also omit the results for LRT and MGMPFM, which are two of the worst baselines, as well as those for our own STA-E and STA-H variants.

The best results for the original and reduced datasets are presented in **bold** and **bold*** in Tables 3 and 4, respectively. We then have the follow interesting observations.

- With the reduction of training data, the *Rec@k* and *NDCG@k* values for all approaches decrease. However, our STA model always achieves the best results on the two datasets and in both metrics. In particular, on the original Foursquare, STA achieved the third best performance in terms of *NDCG*@5, but it becomes the best one on the small training dataset. Furthermore, our STA model gains more significant improvements over baselines in terms of *Rec@k* on the reduced data than on the original data.
- Although RankGeoMF is the best for *NDCG*@5 on the original Foursquare dataset, its performance drops quickly when the data becomes sparse. The reason may be that RankGeoMF

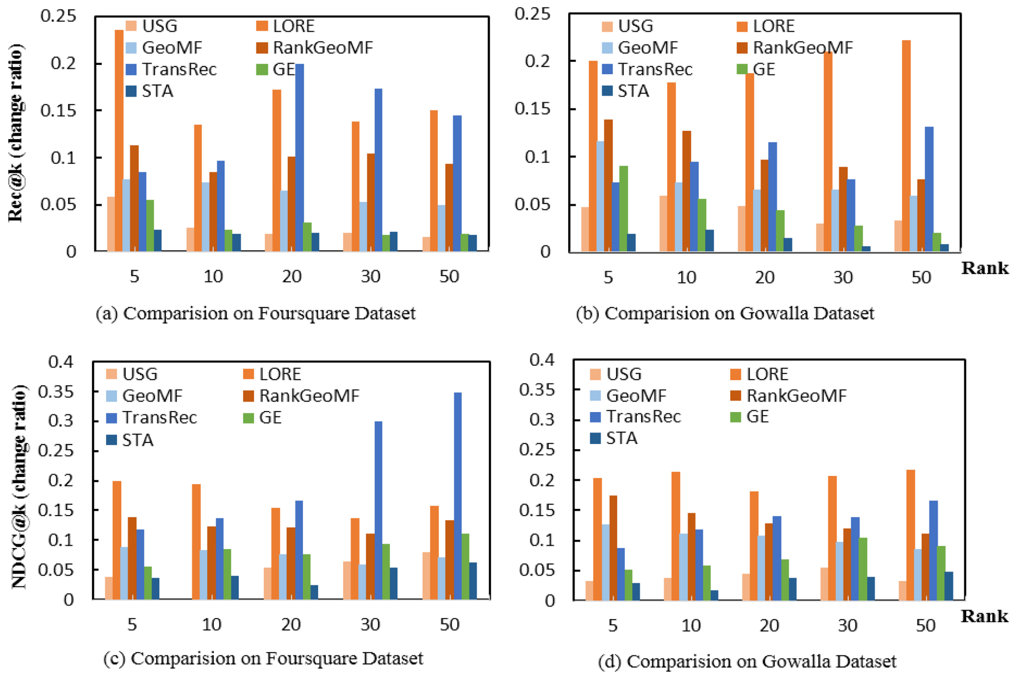


Fig. 4. Changes with the reduction of training data (smaller change ratios are better).

ranks the positive samples higher than the negative ones. However, in the reduced sparse dataset, there are not enough positive examples for RankGeoMF to learn the correct rankings and this consequently deteriorates the performance.

To have a close look at the trends with the reduction of training data, we further present the change ratios in Figure 4. Obviously, the decrease of recall and NDCG values for our STA model are significantly smaller than any other approaches. For instance, the *Rec*@5 values of GE, RankGeoMF, and GeoMF show a 10%, 14%, and 12% drop, respectively. In contrast, our STA model only has a 2% change. This strongly demonstrates that our model is most robust to the data sparsity.

On the other hand, we note that LORE and TransRec are quite sensitive to the number of training data, while USG is the second robust. This is because USG well explores the rich side information, while TransRec does not utilize any social or geographic information and LORE does not directly model user preference. Moreover, both LORE and TransRec rely heavily on the sequential information. When check-in data becomes sparse with a low sampling rate, their performances will significantly decrease.

6.7 Test for Cold-Start Problems

Among the baselines, very few methods can deal with cold-start problems. Specifically, only USG (Ye et al. 2011b) and GE (Xie et al. 2016) are developed to handle cold-start users and cold-start POIs, respectively. Hence, we compare our STA-C models with these two baselines. Note that no existing work is developed for addressing the cold-start spatiotemporal context problem. Since GE (Xie et al. 2016) also contains the time and location information in two graphs, we modify it and take this modified version as the baseline for the cold-start spatiotemporal context problem.

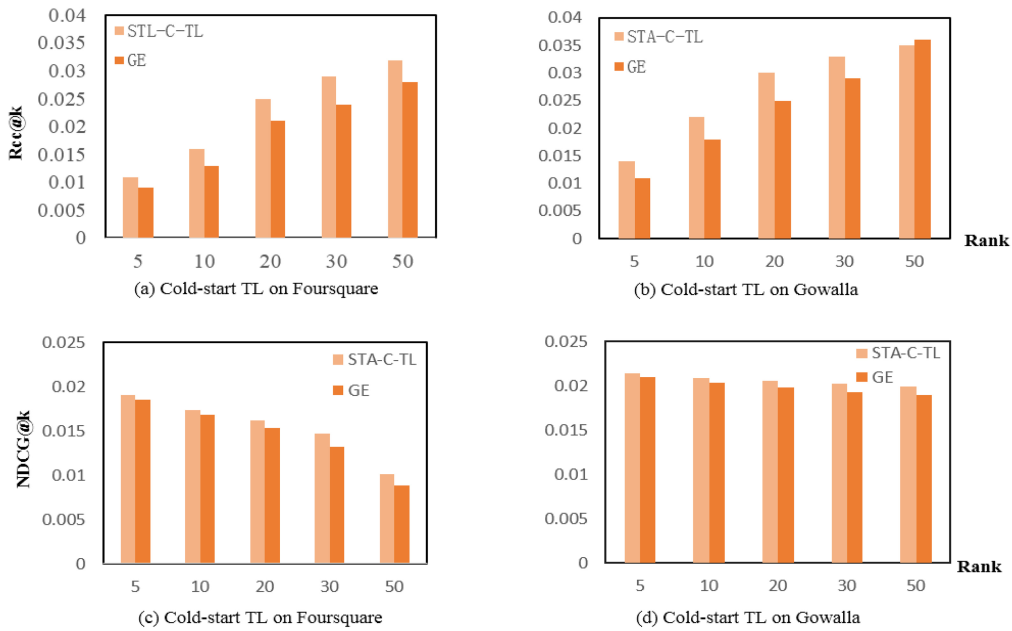
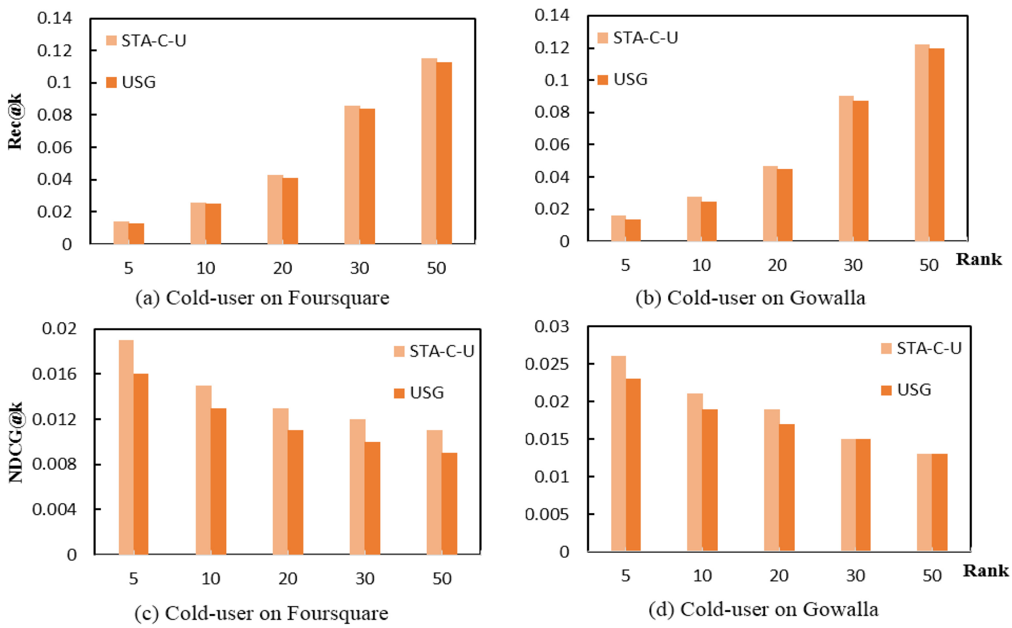


Fig. 5. Test for cold-start spatiotemporal contexts.

6.7.1 Test for Cold-Start Spatiotemporal Contexts. This experiment evaluates the performance of our proposed STA-C-TL model and the modified GE (Xie et al. 2016) for cold-start spatiotemporal contexts. To test the recommendation performance for the cold-start spatiotemporal contexts, we choose the check-ins with less than 10 occurrences in a user's records. These check-ins are removed from the training data and used as the ground truth for test data. Note that the single time slot and location still appear in the training data, and their combinations with other locations and time slots are used as the training set. This also ensures GE can utilize such single time and location information in the POI-Time and POI-Location graphs. The comparison results between our STA-C-TL model and GE are shown in Figure 5.

It is clear that our proposed STA-C-TL model outperforms GE in almost all cases. The only exception is *Rec*@50 on Gowalla dataset, where the score for our model is 0.035 and that for GE is 0.036. This result is reasonable yet not important. On one hand, our STA-C-TL model uses less information than GE as it does not take the semantic tag information into consideration. GE can find more POIs by using the tags and accordingly increases its recall performance. On the other hand, even with the help of additional tag information, all other *Rec@k* (k=5, 10, 20, 30) scores for GE are inferior to those for our model. In reality, the results by smaller k are always more useful to users. Furthermore, our STA-C-TL model performs better than GE in terms of all *NDCG@k* values.

6.7.2 Test for Cold-Start Users. This experiment evaluates the performance of our proposed STA-C-U model and the USG (Ye et al. 2011b) for cold-start users. To test the recommendation performance for cold-start users, we choose users with less than 10 check-ins as cold-start users, following the work of Yin et al. (2017). For those cold-start users, we remove their check-ins from the training dataset and take them as the test set, and use the check-in records generated by the other users as the training set. The comparison results are shown in Figure 6.



(a) Cold-user on Foursquare (b) Cold-user on Gowalla

(c) Cold-user on Foursquare (d) Cold-user on Gowalla

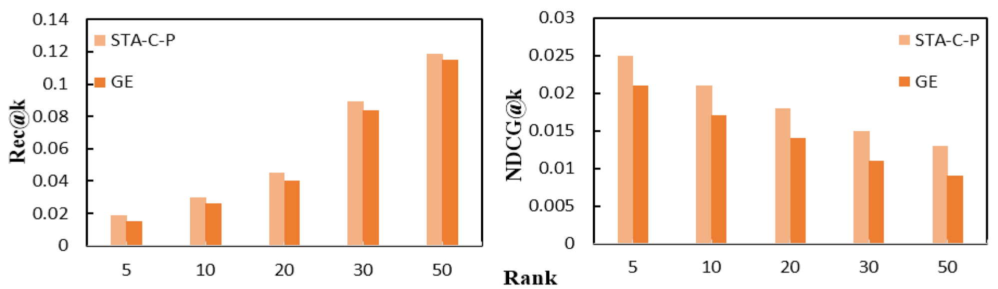
Fig. 6. Test for cold-start users.

We can see that our proposed STA-C-U model outperforms USG in terms of both metrics. For example, the *NDCG*@5 of USG on Foursquare is 0.016 and that of our STA-C-U model is 0.019, showing an 18.75% improvement. Similarly, STA-C-U outperforms USG by 13.04% in terms of *NDCG*@5 on Gowalla.

Overall, we use the same types of social and geographical information as USG does, but our STA model achieves better performance than USG. This demonstrates the superiority of our model in dealing with cold-start users.

6.7.3 Test for Cold-Start POIs. In this experiment, we further compare the effectiveness of our extended STA-C-P model with GE (Yin et al. 2016b) in addressing the cold-start POIs. We first choose POIs with less than 10 check-ins as cold-start POIs, and then select users with at least one cold-start check-in as test users. To simulate a more real cold-start scenario, we remove all check-in records associated with these selected POIs. For each test user, we first choose her/his check-ins associated with cold-start POIs as the test set, and the remaining check-ins as the training set. Our aim is to measure whether the cold-start POIs in the test set can be ranked in the top-k results. Since there is no content information for POIs in Gowalla, we conduct experiments, just as GE did, only on Foursquare. The results for cold-start POI recommendation are shown in Figure 7.

From Figure 7, it is clear that our proposed STA-C-P model consistently beats GE when recommending cold-start POIs in both *Rec*@k or *NDCG*@k metric. The superior performance of the STA-C-P model is due to the nearest/farthest neighbor sampling strategy. As long as an existing POI v shares one tag or is with a short distance to the cold-start POI v_c, our STA-C-P model can get an approximate embedding for v_c. In contrast, GE utilizes the bipartite graphs of POI-Word and POI-Location. The weight of an edge in the graph is calculated by a Term Frequency – Inverse Document Frequency (TF-IDF) value of the word or the frequency of a location. The edge weight is proportional to the probability of edge sampling. Since there are few check-in records for



(a) Cold-POI on Foursquare in terms of Rec@k (b) Cold-POI on Foursquare in terms of NDCG@k

Fig. 7. Test for cold-start POIs on foursquare.

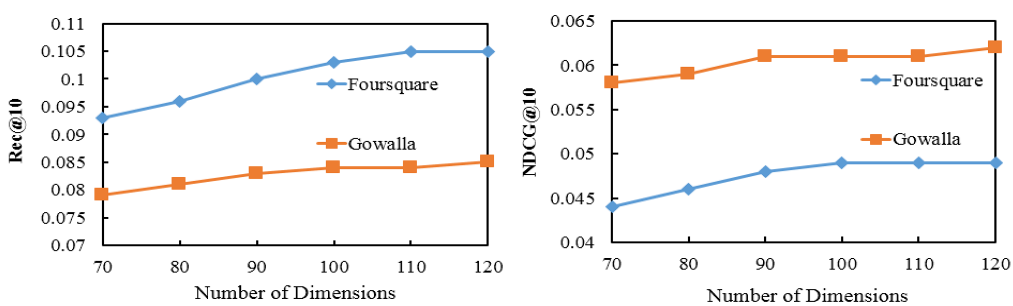


Fig. 8. Effects of dimensionality.

cold-start POIs, a v_c-word and v_c-location edge in GE has an extremely rare chance to be selected and updated. Consequently, the embedding for v_c learned by GE is not accurate and it deteriorates the recommendation performance.

6.8 Effects of Model Parameters

This section investigates the effects of two main parameters involved in our STA model, i.e., the embedding dimension d and the time interval.

We first take *Rec*@10 and *NDCG*@10 as examples to show the effects of embedding dimension d on Foursquare and Gowalla in Figure 8. From the results, we observe that the performance first improves with the increase of the embedding dimension d and then the increment becomes negligible. The reason is that d represents the model complexity. Thus, when d is small, the model has limited ability to describe the data. However, when d exceeds a threshold (say $d = 100$), the model is complex enough to describe the structure in the data. At this point, it is less helpful to improve the model performance by increasing d.

To investigate the effects of time interval, we divide timestamps by three methods, i.e., splitting time into 24, 7, and 2 time slots, corresponding to the daily, weekly, and weekday/weekend patterns, respectively. We depict effects of time interval in Figure 9.

We observe that the impact of the daily patterns is the most significant one on both datasets. In addition, the weekday and weekend patterns also capture different temporal patterns, which contribute to POI recommendation. The combination of all is probably much better than only using a single one. However, we follow the previous studies to divide the time into the same 24 slots as LRT (Gao et al. 2013) and GE (Xie et al. 2016) have done, which are the only two baselines exploiting and integrating temporal effects.

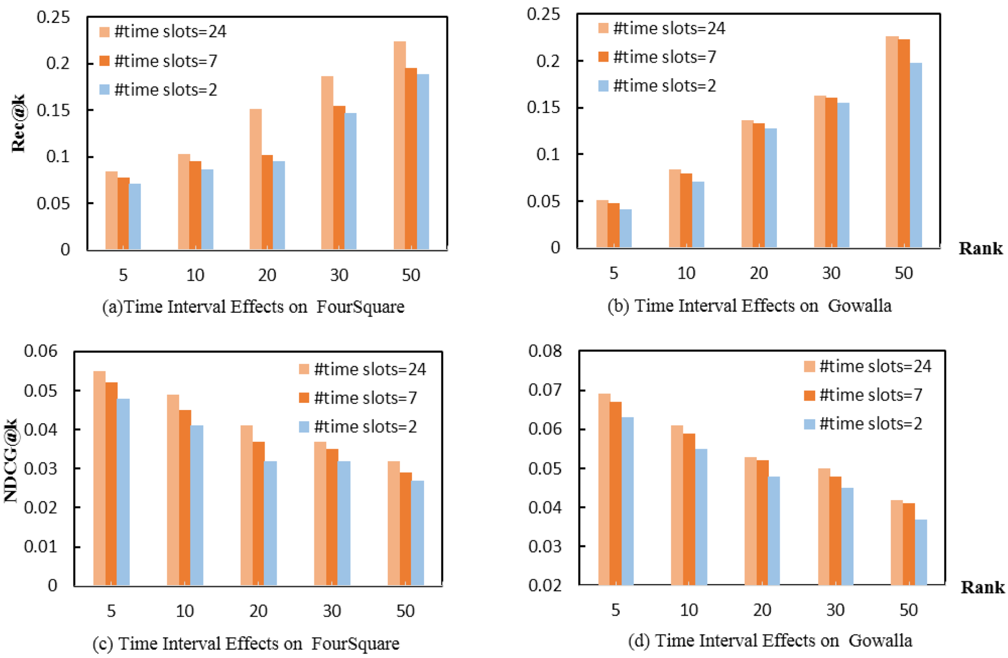


Fig. 9. Effects of time interval.

7 CONCLUSION

We present a novel spatiotemporal aware STA model for learning representations of users, spatiotemporal patterns, and POIs. The basic idea is to capture the geographic and temporal effects using a <time, location> pair, and then model it as a translation connecting users and POIs. We realize STA using the knowledge graph embedding technique and further extend it to incorporate correlation information for addressing cold-start problem. Our work makes the following contributions. (1) We learned a metric space and encapsulated the spatiotemporal transition relation as a translation between users and POIs. (2) The metric learning process followed the inherent triangle inequality assumption, which helped overcome the data sparsity and improving model generalization. (3) We further developed a set of effective strategies to incorporate the side information into our proposed STA model to address the data sparsity and cold-start problems.

We conduct extensive experiments on two real-life datasets. Our results show that our STA framework achieves the state-of-the-art performance in terms of recall and NDCG metrics. The STA model also significantly outperforms the baselines in terms of the effectiveness in addressing both the data sparsity and cold-start problems.

REFERENCES

Gediminas Adomavicius, Bamshad Mobasher, Francesco Ricci, and Alex Tuzhilin. 2011. Context-aware recommender systems. *AI MAGAZINE* Fall (2011), 67–80.

Antoine Bordes, Xavier Glorot, Jason Weston, and Yoshua Bengio. 2014. A semantic matching energy function for learning with multi-relational data. *Machine Learning* 94, 2 (2014), 233–259.

Buru Chang, Yonggyu Park, Donghyeon Park, Seongsoon Kim, and Jaewoo Kang. 2018. Content-aware hierarchical point-of-interest embedding model for successive POI recommendation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-18)*. 3301–3307.

Xuefeng Chen, Yifeng Zeng, Gao Cong, Shengchao Qin, Yanping Xiang, and Yuanshun Dai. 2015. On information coverage for location category based point-of-interest recommendation. In *Proceedings of AAAI Conference on Artificial Intelligence*. 37–43.

Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. 2012. Fused matrix factorization with geographical and social influence in location-based social networks. In *Proceedings of AAAI Conference on Artificial Intelligence*. 17–23.

Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. 2016. A unified point-of-interest recommendation framework in location-based social networks. *ACM TIST* 8, 1 (10 2016), 1–21.

Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. 2013. Where you like to go next: Successive point-of-interest recommendation. In *Proceedings of International Joint Conference on Artificial Intelligence*. 2605–2611.

Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: User movement in location-based social networks. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1082–1090.

Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. 2015. Personalized ranking metric embedding for next new POI recommendation. In *Proceedings of 24th International Joint Conference on Artificial Intelligence*. 2069–2075.

Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2015b. Content-aware point of interest recommendation on location-based social networks. In *Proceedings of 29th AAAI Conference on Artificial Intelligence*. 1721–1727.

Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2013. Exploring temporal effects for location recommendation on location-based social networks. In *ACM Conference on Recommender Systems*. 93–100.

Huiji Gao, Jiliang Tang, and Huan Liu. 2012. gSCorr: Modeling geo-social correlations for new check-ins on location-based social networks. In *Proceedings of ACM International Conference on Information and Knowledge Management*. 1582–1586.

Huiji Gao, Jiliang Tang, and Huan Liu. 2015a. Addressing the cold-start problem in location recommendation using geo-social correlations. *Data Mining and Knowledge Discovery* 29, 2 (2015), 299–323.

X. Glorot and Y. Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*. 249–256.

Jean-Benoit Griesner, Talel Abdessalem, and Hubert Naacke. 2015. POI recommendation: Towards fused matrix factorization with geographical and temporal influences. In *Proceedings of the 9th ACM Conference on Recommender Systems*. 301–304.

Mengyue Hang, Ian Pytlarz, and Jennifer Neville. 2018. Exploring student check-in behavior for improved point-of-interest prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 321–330.

Ruining He, Wang Cheng Kang, and Julian Mcauley. 2017a. Translation-based recommendation. In *ACM Conference on Recommender Systems*. 161–169.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017b. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*. 173–182.

Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *Proceedings of the 26th International Conference on World Wide Web*. 193–201.

Bo Hu and Martin Ester. 2014. Social topic modeling for point-of-interest recommendation in location-based social networks. In *Proceedings of IEEE International Conference on Data Mining*. 845–850.

Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-interest recommendations: Learning potential check-ins from friends. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 975–984.

Xutao Li, Gao Cong, Xiaoli Li, Tuan Anh Nguyen Pham, and Shonali Krishnaswamy. 2015. Rankgeofm: A ranking based geographical factorization method for point of interest recommendation. In *Proceedings of International Conference on Research on Development in Information Retrieval*. 433–442.

Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 831–840.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of 29th AAAI Conference on Artificial Intelligence*. 2181–2187.

Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. 2013a. Learning geographical preferences for point-of-interest recommendation. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1043–1051.

Bin Liu and Hui Xiong. 2013. Point-of-interest recommendation in location based social networks with topic and location awareness. In *Proceedings of the 2013 SIAM International Conference on Data Mining (SDM)*. 396–404.

Qi Liu, Enhong Chen, Hui Xiong, Yong Ge, Zhongmou Li, and Xiang Wu. 2014a. A cocktail approach for travel package recommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 26, 2 (2014), 278–293.

Qi Liu, Yong Ge, Zhongmou Li, Enhong Chen, and Hui Xiong. 2011. Personalized travel package recommendation. In *Proceedings of the 11th IEEE International Conference on Data Mining*. 407–416.

Qi Liu, Haiping Ma, Enhong Chen, and Hui Xiong. 2013b. A survey of context-aware mobile recommendations. *International Journal of Information Technology and Decision Making (IJITDM)* 12, 1 (2013), 139–172.

Xin Liu, Yong Liu, and Xiaoli Li. 2016. Exploring the context of locations for personalized location recommendations. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*. 1188–1194.

Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. 2014b. Exploiting geographical neighborhood characteristics for location recommendation. In *Proceedings of ACM International Conference on Information and Knowledge Management*. 739–748.

Shaojie Qiao, Nan Han, Jiliu Zhou, Rong-Hua Li, Cheqing Jin, and Louis Alberto Gutierrez. 2018. SocialMix: A familiarity-based and preference-aware location suggestion approach. *Engineering Applications of Artificial Intelligence* 68 (2018), 192–204.

Salvatore Scellato, Anastasios Noulas, Renaud Lambiotte, and Cecilia Mascolo. 2011. Socio-spatial properties of online location-based social networks.. In *Proceedings of International Conference on Web and Social Media (ICWSM)*. 329–336.

Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-specific geographical influence for point-of-interest recommendation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-18)*. 3877–3883.

Hao Wang, Naiyan Wang, and Dit Yan Yeung. 2015. Collaborative deep learning for recommender systems. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1235–1244.

Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Wasim Sadiq, and Xiaofang Zhou. 2017. ST-SAGE: A spatial-temporal sparse additive generative model for spatial item recommendation.*ACM TIST* 8, 48 (3 2017), 1–25.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of 28th AAAI Conference on Artificial Intelligence*. 1112–1119.

Min Xie, Hongzhi Yin, Hao Wang, Fanjiang Xu, Weitong Chen, and Sen Wang. 2016. Learning graph-based POI embedding for location-based recommendation. In *Proceedings of ACM International Conference on Information and Knowledge Management*. 15–24.

Mao Ye, Krzysztof Janowicz, and Wang Chien Lee. 2011a. What you are is when you are: The temporal dimension of feature types in location-based social networks. In *Proceedings of ACM Sigspatial International Conference on Advances in Geographic Information Systems*. 102–111.

Mao Ye, Peifeng Yin, and Wang Chien Lee. 2010. Location recommendation for location-based social networks. In *Proceedings of ACM Sigspatial International Conference on Advances in Geographic Information Systems*. 458–461.

Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. 2011b. Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proceedings of International Conference on Research on Development in Information Retrieval*. 325–334.

Hongzhi Yin, Bin Cui, Xiaofang Zhou, Weiqing Wang, Zi Huang, and Shazia W. Sadiq. 2016a. Joint modeling of user check-in behaviors for real-time point-of-interest recommendation. *ACM Transactions on Information Systems* 35, 2 (2016), 11:1–11:44.

Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. 2013. LCARS: A location-content-aware recommender system. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 221–229.

Hongzhi Yin, Weiqing Wang, Hao Wang, Ling Chen, and Xiaofang Zhou. 2017. Spatial-aware hierarchical collaborative deep learning for POI recommendation. *IEEE Transactions on Knowledge and Data Engineering* 29, 11 (2017), 2537–2551.

Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Nguyen Quoc Viet Hung. 2016b. Adapting to user interest drift for POI recommendation. *IEEE Transactions on Knowledge and Data Engineering* 28, 10 (2016), 2566–2581.

Hongzhi Yin, Xiaofang Zhou, Yingxia Shao, Hao Wang, and Shazia Sadiq. 2015. Joint modeling of user check-in behaviors for point-of-interest recommendation. In *Proceedings of ACM International Conference on Information and Knowledge Management*. 1631–1640.

Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-aware point-of-interest recommendation. In *Proceedings of International Conference on Research Development in Information Retrieval*. 363–372.

Jia Dong Zhang, Yanhua Li, and Yanhua Li. 2014. LORE: Exploiting sequential influence for location recommendations. In *ACM Sigspatial International Conference on Advances in Geographic Information Systems*. 103–112.

Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael R. Lyu, and Irwin King. 2016. STELLAR: Spatial-temporal latent ranking for successive point-of-interest recommendation. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. 316–321.

Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. 2009. Mining interesting locations and travel sequences from GPS trajectories. In *Proceedings of the 18th International World Wide Web Conference*. 791–800.

Wen-Yuan Zhu, Wen-Chih Peng, Ling-Jyh Chen, Kai Zheng, and Xiaofang Zhou.2015. Modeling user mobility for location promotion in location-based social networks. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1573–1582.

Received July 2018; revised October 2018; accepted November 2018