

Multi-Temporal Relationship Inference in Urban Areas

Shuangli Li*

School of Computer Science and
Technology, University of Science and
Technology of China
Baidu Research
lsl1997@mail.ustc.edu.cn

Jingbo Zhou†

Business Intelligence Lab,
Baidu Research
zhoujingbo@baidu.com

Ji Liu

Big Data Lab,
Baidu Research
jiliuwork@gmail.com

Tong Xu

School of Computer Science and
Technology, University of Science and
Technology of China & State Key
Laboratory of Cognitive Intelligence
tongxu@ustc.edu.cn

Enhong Chen

Anhui Province Key Laboratory of
Big Data Analysis and Application,
University of Science and Technology
of China & State Key Laboratory of
Cognitive Intelligence
cheneh@ustc.edu.cn

Hui Xiong†

The Thrust of Artificial Intelligence,
The Hong Kong University of Science
and Technology (Guangzhou)
The Department of Computer Science
and Engineering, The Hong Kong
University of Science and Technology
xionghui@ust.hk

ABSTRACT

Finding multiple temporal relationships among locations can benefit a bunch of urban applications, such as dynamic offline advertising and smart public transport planning. While some efforts have been made on finding static relationships among locations, little attention is focused on studying time-aware location relationships. Indeed, abundant location-based human activities are time-varying and the availability of these data enables a new paradigm for understanding the dynamic relationships in a period among connective locations. To this end, we propose to study a new problem, namely *multi-Temporal relationship inference among locations* (Trial for short), where the major challenge is how to integrate dynamic and geographical influence under the relationship sparsity constraint. Specifically, we propose a solution to Trial with a graph learning scheme, which includes a spatially evolving graph neural network (SEENet) with two collaborative components: spatially evolving graph convolution module (SECONV) and spatially evolving self-supervised learning strategy (SE-SSL). SECONV performs the intra-time aggregation and inter-time propagation to capture the multifaceted spatially evolving contexts from the view of location message passing. In addition, SE-SSL designs time-aware self-supervised learning tasks in a global-local manner with additional evolving constraint to enhance the location representation

learning and further handle the relationship sparsity. Finally, experiments on four real-world datasets demonstrate the superiority of our method over several state-of-the-art approaches.

CCS CONCEPTS

• Information systems → Spatial-temporal systems.

KEYWORDS

Relationship Inference, Graph Neural Networks, Spatial Graph

ACM Reference Format:

Shuangli Li, Jingbo Zhou, Ji Liu, Tong Xu, Enhong Chen, and Hui Xiong. 2023. Multi-Temporal Relationship Inference in Urban Areas. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*, August 6–10, 2023, Long Beach, CA, USA. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3580305.3599440>

1 INTRODUCTION

With the widespread availability of human mobility data, discovering location relationships in urban areas has received significant research interest in recent years [3, 13, 20, 28, 48]. Finding the location relationship can help to reveal urban patterns, which definitely benefits intelligent urban management [37] and finally promotes the urban business economy [20]. However, most existing works focus on analyzing static relationships among locations from a spatial perspective, and few known methods study time-aware location relationships from a temporal perspective. Whereas, the human mobility data is always time-correlated [6, 14, 45, 49], e.g., *visiting restaurants at noon and visiting bars at night*. Hence, here we present to investigate a new paradigm for understanding such dynamic multiple relationships between locations in a period, which has been largely overlooked in previous studies.

To this end, we propose to study a problem of *multi-Temporal relationship inference among locations* (we call Trial for short), which is of great importance in many urban application scenarios. The goal of Trial is to recover plenty of missing relationships across multiple time segments from the constructed location graph in an

*This work was done when the first author was an intern at Baidu Research under the supervision of Jingbo Zhou.

†Corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).
KDD '23, August 6–10, 2023, Long Beach, CA, USA

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0103-0/23/08...\$15.00
<https://doi.org/10.1145/3580305.3599440>

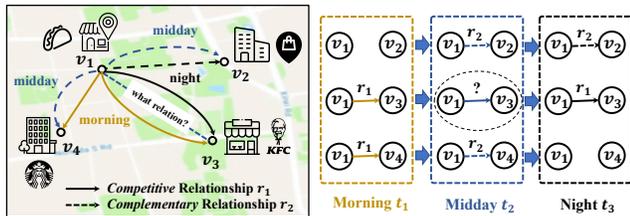


Figure 1: An illustration of multi-temporal relationships around the fast-food restaurant location v_1 . The pair (v_1, v_2) with complementary relationships tend to be visited by common users at night, while another pair (v_1, v_3) with competitive relationships provide similar service at the specific time.

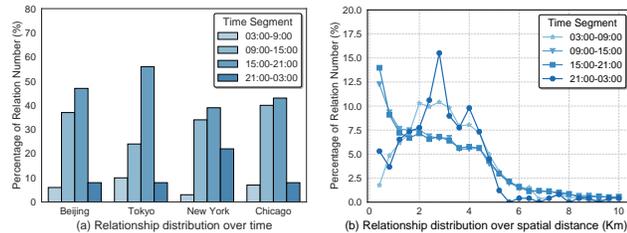


Figure 2: Characteristics of multi-temporal relationships.

urban area. In other words, we aim at mining location connections at different time segments (e.g., morning or evening). Solving Trial can facilitate urban intelligence in many application domains, such as dynamic business advertisements[46], urban resource planning [5, 19], and knowledge-enhanced location recommendation [22, 31].

Here we further present two examples to demonstrate our motivation for Trial. At first, given a target region location (e.g., a residential district), the most relevant regions for this district may be different from morning to evening. After inferring multi-temporal relationships for this district, the urban manager can adaptively plan public transport at different times in a day. Second, let us consider another scenario of urban business as illustrated in Figure 1. On the one hand, the relationship in an urban area may only exist at specific times due to the evolving daily activities of users [45]. In this case, focusing on time-specific correlated locations can help business owners to dynamically optimize their advertisement strategy [23] at different times. On the other hand, we find the relationship can be diverse in a day. Taking Figure 1 as an example, the cafe v_1 (i.e., Starbucks) is competitive with the restaurant v_2 (i.e., Taco Bell) since both provide breakfast in the morning, while they become complementary at midday due to different services. Therefore, the user experience can be also improved with the discovered multi-temporal location pairs to enhance the time-aware location recommendation [2, 15, 32]. Hence, it is critical to effectively understand multi-temporal relationships among locations.

Existing methods for relationship mining cannot handle Trial problem effectively since the temporal and geographical factors are rarely considered in a unified manner. Extensive studies have been conducted for relationship mining in other domains, e.g., business analysis [13, 47] or e-commerce relationship inference [18, 21, 24]. These methods usually cannot be directly used for location relationship mining. Recently, remarkable advancements in graph neural networks (GNNs) [8, 29, 36] have shown the powerful capacity for relationship graph learning and achieved promising results. These GNN-based relationship prediction studies attempt to extend the

effective graph message-passing procedure by learning relational correlations [18, 21] or spatial dependencies [13] for location context understanding. The most recent method [3] further adopts attentive aggregation to handle multiple relationships in the urban area. However, all previous works do not fully consider unique attributes of our Trial, which fail to examine two research challenges in modeling multi-temporal location relationships as follows.

CHALLENGE 1: How to capture the relational dynamics under temporal and geographical contexts? According to the statistical analysis in Figure 2(a), the distribution of the location relationship varies in a day, indicating the dynamic influence plays a crucial role. Moreover, further investigation in Figure 2(b) shows that the relational dynamics is correlated with geography information in the urban area. The geographical context may have different influences on locations at different times. Although some dynamic GNNs [25, 44] for link prediction are designed for multi-temporal graph structures, they fail to incorporate the relational and spatial evolving patterns. How to communicate the dynamic correlations with geographical factors in an effective way remains a unique research challenge. Note that spatial-temporal GNN focusing on graph-based time series analysis (e.g., traffic forecasting) is another substantially different research problem [27, 33], which is not applicable to the linking problem of Trial focusing on dynamic relationships.

CHALLENGE 2: How to deal with the relationship sparsity with spatial and dynamic influences? Since the precious relationship between locations is derived from limited and conditional user behavior data, the multi-temporal relationship is more scarce [18]. Therefore, the learning process of GNN suffers from data sparsity with insufficient context information in aggregation. Most existing graph self-supervised learning methods [11, 30, 50] only pay attention to learning from the augmented graph structures without considering spatial and dynamic influences for relational modeling, thus leading to suboptimal performance. How to take advantage of sparse relationships in the urban area is another notable challenge.

In this paper, we propose a Spatially Evolving graph nEural Network (SEENet) tailored for inferring multi-temporal relationships among locations. To address the above challenges, we design the framework from two perspectives: spatially evolving graph convolution (SECONV) for message passing-level modeling, and 2) spatially evolving self-supervised learning (SE-SSL) for training-level modeling. Firstly, aiming at the **first challenge**, the proposed graph learning procedure SECONV is equipped with intra-time aggregation and inter-time propagation. The key idea is to identify the spatial and evolving influence from multifaceted locations with considering non-local and cross-time neighbors. Specifically, the intra-time learning process performs the second-order aggregation to preserve non-local geographical and relational dependencies at each specific time. On the other hand, the inter-time learning process further propagates the multi-temporal information to capture the spatially evolving context across adjacent time segments. Moreover, SE-SSL is devised to deal with the **second challenge** of relationship sparsity, which adopts the spatial information maximum strategy from a global regional view and the additional evolving constraint from a local relational view. By this means, SEENet can enhance the representation learning for locations with incorporating both spatial distribution and dynamic patterns in a self-supervised manner. The major contributions of this paper are summarized as follows.

- To the best of our knowledge, this is the first work to investigate the problem of *multi-temporal relationship inference* among locations for various valuable scenarios, which studies time-specific relationships in urban areas at a fine-grained level.
- We propose a novel spatially evolving graph neural network named SEENet with collaborative designs for relationship learning among locations, which can capture the geographical and dynamic influence through an intra-time and inter-time spatially evolving graph convolution as well as an effective evolutionary self-supervised learning task.
- We conduct extensive experiments on four real-world datasets, which demonstrates the superiority of SEENet.

2 RELATED WORK

In this section, we review the previous literature from two perspectives: topic-related *relationship mining and inference*, and closely technology-related *graph neural networks*.

Relationship Mining and Inference. Mining valuable relationships has attracted increasing attention from both academia and industry [24, 35]. In the early stage, most of the previous works aim at inferring the precious relationship from content information [28] (e.g., textual reviews and descriptions on the web), while expert knowledge is required to design linguistic rules [7, 10] or graphical analysis models [43]. Another line of work seeks to apply deep learning-based techniques to analyze various relationships. Some explore applying the linked auto-encoder [26] and graphlet mining [34, 48] for product-oriented or competitor-oriented applications. Considering the natural inadequacy in learning relational graph structures of these domain-specific methods, some recent works further propose to develop powerful graph learning approaches for relationship discovery. From the perspective of relational dependency learning, DecGCN [21] designs the graph structural integration mechanism for decoupled representation learning, which can detect the mutual influence between different relationships. The recent IRGNN [18] further incorporates the multi-hop complex relationships to alleviate the sparsity issue. However, spatial dependencies between nodes are omitted which encourages researchers to propose geography-based graph methods. Therefore, from the perspective of spatial context modeling in urban area, the fine-grained distance distribution is captured in DeepR [13] with spatial adaptive graph convolutions. More recently, PRIM [3] combines self-attentive spatial context extractor for multiple relation types. However, the important relational dynamics is neglected all along. In this paper, we focus on the challenging multi-temporal relationship inference among locations to fill the research gap.

Graph Neural Networks. Recent years have witnessed the rapid growth of graph neural networks (GNNs), which exhibit a strong ability in learning structural relationships [12, 13, 16, 21, 38, 39]. According to the unique challenging properties of location relationships as introduced before, the technically corresponding GNN models fall into three mainstreams. Firstly, a number of GNNs perform diverse message-passing schemes to capture rich context information from graph structures, such as considering edge types for multiple relational semantics [29] and spatial attributes [3, 13]. Moreover, some efforts have been devoted to studying expressive high-order GNN methods, including mixing neighboring features at

various distances in Mixhop [1], distinguishing non-local topological structures with the random walk [4] or attention-guided sorting [17]. Secondly, graph self-supervised learning methods focus on designing augmented strategies to tackle data scarcity, which utilizes contrastive learning [50] or meaningful tasks [30] on graphs. The most recent RGRL [9] also leverages the relationship information with preserving global and local similarity. Nevertheless, these methods tend to lose effectiveness without considering the spatial and evolving characteristics of relationships. Finally, integrating relational dynamics is also important for snapshot-based graph learning. Along this line, EvolveGCN [25] is proposed to recurrently updates the GNN weights for dynamic link prediction. Although the recent ROLAND [44] takes a further step to involve hierarchical states over time, one-sided dynamic information is still not sufficient. It is noteworthy that the irrelevant spatial-temporal GNNs can not be applied in location relationship learning for comparison. Because most of them are basically designed for time series forecasting with specific sequential values (e.g., traffic flow or weather conditions) [33], which is intrinsically different from our target of multi-time link prediction. Therefore, we aim to develop an adaptive graph neural network to preserve both spatial and dynamic dependencies simultaneously for location relationship inference.

3 PRELIMINARIES

In this section, we first present the concept of the dynamic location graph, and then formalize the problem of multi-temporal relationship inference among locations (Trial for short).

Definition 3.1. (Time-specific Location Relationship). As stated above, relationships between locations are dynamic in a day. Thus, the complex location relationships should be naturally decomposed into multi-temporal segments to meet the unique daily dynamic characteristic of locations. Formally, the pre-defined set of time segments is denoted as $\mathbb{T} = \{t_1, t_2, \dots, t_T\}$, where all T time segments form a whole day. Accordingly, the location relationship set \mathcal{R} contains multiple time-specific relationships among locations. In the following sections, the mentioned *relationship* also stands for *location relationship*. We use $t \in \mathbb{T}$ to generally represent a certain time, while $r \in \mathcal{R}$ refers to a given relationship.

Definition 3.2. (Dynamic Location Graph). In this work, the multi-temporal relationships are organized as a fine-grained dynamic location graph $\mathcal{G} = \{G^{(t)} | G^{(t)} = (\mathcal{V}, \mathcal{E}^{(t)}, \mathbf{L}), t \in \mathbb{T}\}$, where $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ is the set of location nodes with the spatial coordinates $\mathbf{L} \in \mathbb{R}^{2 \times N}$. The dynamic graph \mathcal{G} is composed of T time-specific graphs and contains different relationship edges at different time segments, which is denoted as $\mathcal{E} = \mathcal{E}^{(t_1)} \cup \dots \cup \mathcal{E}^{(t_T)}$. The relational edge $e_{i,j,r}^{(t)} = (v_i, v_j, r, t)$ represents there exists the relationship $r \in \mathcal{R}$ between v_i and v_j at time t .

Since the relational location graph \mathcal{G} is usually sparse and most valuable relationships are absent, our target is to learn from \mathcal{G} and discover all meaningful relationships at different time segments. We formally define the problem as below:

Definition 3.3. (Multi-Temporal Relationship Inference). For the dynamic location graph $\mathcal{G} = \{G^{(t_1)}, \dots, G^{(t_T)}\}$ associated with spatial locations \mathbf{L} , the problem of Trial aims to jointly learn a

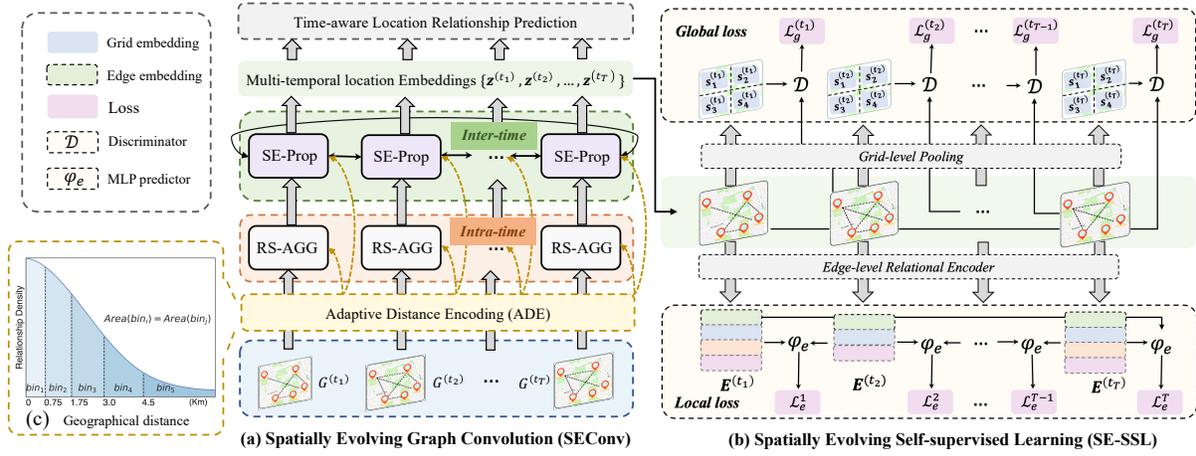


Figure 3: Illustration of the proposed SEENet framework for dynamic location graphs.

model $\mathcal{F}(\{G^{(t_1)}, \dots, G^{(t_r)}\})$ to map the location node set \mathcal{V} into multiple embeddings $\{z^{(t_1)}, \dots, z^{(t_r)}\}$ at multi-temporal segments. Then, given a location pair $(v_i, v_j) \in \mathcal{V} \times \mathcal{V}$ at the time $t \in \mathbb{T}$, the possibility score for each relationship r can be estimated by the prediction function $p_r^{(t)}(z_i^{(t)}, z_j^{(t)})$. Therefore, we can discover all potential location relationships at multiple times.

4 MODEL FRAMEWORK

In this section, we present the Spatially Evolving graph Neural Network (SEENet) model, which learns from multi-temporal dynamic and geographical correlations in an end-to-end manner. As illustrated in Figure 3, the overall framework first takes the dynamic location graph \mathcal{G} in the temporal format as input. Our proposed SEENet is equipped with a spatially evolving graph convolution module (SEConv) to incorporate the evolving context along with the comprehensive spatial influence. After obtaining the representations of locations by SEConv, the spatially evolving self-supervised learning module (SE-SSL) is devised to enhance the model’s capability of learning dynamic relational patterns through well-designed training tasks on sparse location graphs. Finally, we utilize the pre-trained model after SE-SSL for our problem of Trial.

4.1 Spatially Evolving Graph Convolution

The intrinsic evolving correlations associated with spatial contexts are critical to time-specific location representation learning. In past years, GNNs [8] have shown the superiority on processing relational graph structures for Points-of-Interest [3, 13] or items [18, 21]. These GNN methods mainly focus on topological structures with learning spatial dependencies or relational semantics on the single static graph, which fails to deal with multi-temporal relations.

To this end, we develop the Spatially Evolving graph Convolution (SEConv) to capture both spatial location context and dynamic correlations across time. The key idea of SEConv derives from two perspectives for complex context modeling: *intra-time* non-local relational interactions and *inter-time* spatially evolving interactions.

Since the geography information plays a crucial role in location relationship, before introducing the main components of SEConv, we first manage to project the scalar distances into informative spatial representations. In view of the varying spatial distribution of

different datasets or cities, we adopt the Adaptive Distance Encoder (ADE) with distribution-aware embedding mechanism [40]. We first calculate all distances based on location’s coordinate matrix L in the relational graph \mathcal{G} , then the statistical distance distribution $P(x)$ is obtained in Figure 3(c). Since the scalar distance value only has limited one-dimension information without learning ability, we build the embedding layer to extract the discrete representations. As illustrated in Figure 3(c), the $P(x)$ distribution is uniformly decomposed into N_b consistent distance-space bins with the constraint of the equal area size under each bin’s curve. The boundary list $B(N_b)$ for distance bins is calculated as:

$$B(N_b) = [b_1, b_2, \dots, b_{N_b}] \quad \text{s.t.} \quad \int_0^{b_k} P(x) dx = \frac{k}{N_b}, \quad (1)$$

Given a pair of locations (v_i, v_j) , we further map the distance $\|L_i - L_j\|$ to the bin index k based on the uniform distance boundaries. The discrete representation $\mathbf{d}_{i,j}$ with the bin index k is obtained through the embedding layer:

$$\mathbf{d}_{i,j} = \text{Embedding}(k) \quad \text{s.t.} \quad b_k \leq \|L_i - L_j\| < b_{k+1}, \quad (2)$$

where L_i is the location coordinate for v_i . The generated distribution-aware distance representations can adaptively imply the overall spatial context from the view of statistical analysis.

After that, the spatial distance embedding is integrated into the *Relational Spatial AGGregation* (RS-AGG) for time-specific non-local dependencies modeling and the *Spatially Evolving contextual Propagation* (SE-Prop) for multifaceted context modeling, which will be introduced later. As shown in Figure 3(a), the overall intra- and inter-time convolutional process at time t is defined as:

$$\begin{aligned} \mathbf{h}_{i,intra}^{(t)} &= \text{AGG}_t \left(\{(\mathbf{h}_j, \mathbf{h}_k, \mathbf{d}_{i,k}) \mid \forall v_j \in \mathcal{N}_i^{(t)}, v_k \in \mathcal{N}_j^{(t)}\} \right), \\ \mathbf{h}_{i,inter}^{(t)} &= \text{Prop}_t \left(\{(\mathbf{h}_{j,intra}^{(\tau)}, \mathbf{d}_{i,j}) \mid \forall v_j \in \mathcal{N}_i^{(\tau)}, \tau \in T(t)\} \right), \end{aligned} \quad (3)$$

where \mathbf{h}_i is the input location embedding, $\mathbf{h}_{i,intra}^{(t)}$ and $\mathbf{h}_{i,inter}^{(t)}$ are the intra- and inter-time embeddings for location v_i , AGG_t and Prop_t are the aggregation and propagation functions at the time t for RS-AGG and SE-Prop, respectively. $\mathcal{N}_i^{(t)}$ is the neighboring set of v_i at time t . $T(t)$ represents multiple adjacent time segments (from T_{t_1} to T_{t_2}) around t for the inter-time propagation process.

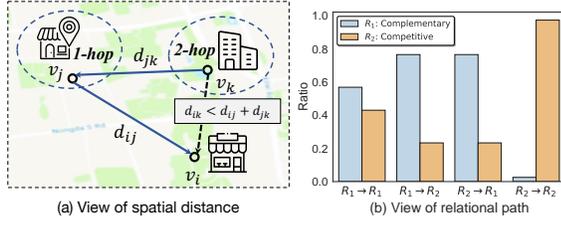


Figure 4: Two views of second-order dependencies in RS-AGG.

4.1.1 Intra-time Interaction: Relational Spatial Aggregation (RS-AGG). In the scenario of location relationships, the abundant second-order dependency can potentially imply the undiscovered relationships. To meet the needs of such unique location graph learning, the proposed RS-AGG adopts the second-order message passing architecture with considering the non-local environment in the first stage of intra-time interaction modeling. Different from previous GNN-based relationship learning methods aggregating information from 1-hop local neighbors, the basic intuition of RS-AGG derives from both spatial and relational views.

From the spatial view. As shown in Figure 4(a), although the 2-hop neighboring locations are not directly connected with the target location v_i , they may be geographically close to v_i . As we have investigated in Figure 2(b), the stronger geographical connection (i.e., closer distance) indicates the higher possibility of relationship existence. However, the ordinary 1-hop message passing process can only deliver the spatial distance between neighboring locations. If we simply stack two GNN layers with propagating distance information (d_{ij}, d_{jk}), the model can not capture the real 2-hop distance d_{ik} in the geographical space (usually $d_{ij} + d_{jk} \neq d_{ik}$ unless three nodes (v_i, v_j, v_k) are spatially collinear).

From the relational view. The 2-hop relational path can reflect the meaningful relationship patterns, which is denoted as $v_i \xrightarrow{R_{i,j}} v_j \xrightarrow{R_{j,k}} v_k$. As shown in Figure 4(b), different types of relational combinations can imply different potential relationships, where the relational dependence $R_{i,j} \rightarrow R_{j,k}$ is helpful to reveal the relationship $R_{i,k}$. Distinguishing such path semantics can greatly improve the model’s ability to discover more hidden relationships.

To utilize the above two characteristics among locations, we incorporate the spatial influence and relational dependence into a unified aggregation network RS-AGG. We first define the second-order neighbors with the mixed relational division strategy:

$$\mathcal{N}_t^2(v_i, r_1 \rightarrow r_2) = \{(v_j, v_k) \mid \underbrace{v_i \xrightarrow{R_{i,j}} v_j \xrightarrow{R_{j,k}} v_k}_{\text{Relational path constraint}} \models r_1 \rightarrow r_2\}, \quad (4)$$

where r_1 and r_2 denote a certain relation pair, \models means that the relational path determined by the triple nodes (v_i, v_j, v_k) satisfy the 2-hop relational pattern $r_1 \rightarrow r_2$ (i.e., $R_{i,j} = r_1$ and $R_{j,k} = r_2$). The neighboring set $\mathcal{N}_t^2(v_i, r_1 \rightarrow r_2)$ defines the associated location pair (v_j, v_k) within the second-order range, where the 2-hop neighbor v_k is reachable via the middle 1-hop neighbor v_j on the graph at time t . Only the pairs connected by at least two different relational paths are included for efficiency. In this way, we can provide the most crucial evidence for inferring relationships.

Then, we further propose the geography-aware relational graph convolutions to preserve multiple dependencies from the complex

intra-time connections. This non-local learning scheme is devised to handle the second-order correlations for each pattern, which can simultaneously aggregate the 1-hop and 2-hop interaction features with the spatial gating mechanism. In general, the path-specific location representation for v_i is generated as follows:

$$\mathbf{h}_{i,r_1 \rightarrow r_2}^{(t)} = \sum_{(v_j, v_k) \in (r_1 \rightarrow r_2)} \left(\mathbf{W}_{r_1 \rightarrow r_2}^{(t)} \mathbf{h}_j^{(t)} + \Phi_t(i, j, k) \cdot \mathbf{W}_{r_1 \rightarrow r_2}^{(t)} \mathbf{h}_k^{(t)} \right), \quad (5)$$

where $(r_1 \rightarrow r_2)$ is simply short for the constructed second-order neighboring set $\mathcal{N}_t^2(v_i, r_1 \rightarrow r_2)$, $\mathbf{h}_j^{(t)}$ and $\mathbf{h}_k^{(t)}$ are the input embeddings of 1-hop neighbor v_j and 2-hop neighbor v_k respectively, $\mathbf{W}_{r_1 \rightarrow r_2}^{(t)}$ is the weight matrix for the specific relational path pattern $r_1 \rightarrow r_2$ at time t . The proposed operator Φ_t represents the spatial gating function to determine the distinctive influence of the second-order information, which is formulated as:

$$\Phi_t(i, j, k) = \text{sigmoid}(\mathbf{a}_{t,r_1,r_2}^T \cdot (\mathbf{s}_{t,r_1,r_2}^{spa} + \mathbf{s}_{t,r_1,r_2}^{rel})), \quad (6)$$

where \mathbf{a}_{t,r_1,r_2}^T is the trainable parameter for importance weight calculation. Here we take the 1-hop relation semantics and 2-hop geographical impact into account. Since the calculated gating score is devised to reflect the relative significance between 1-hop and 2-hop information, it combines the pairwise relational-based vector $\mathbf{s}_{t,r_1,r_2}^{rel}$ and spatial-based vector $\mathbf{s}_{t,r_1,r_2}^{spa}$ from two domain spaces:

$$\mathbf{s}_{t,r_1,r_2}^{spa} = \mathbf{W}_{t,r_1,r_2}^{spa} [\mathbf{G}_{t,r_1,r_2} \mathbf{d}_{i,j} \oplus \mathbf{W}_{r_1 \rightarrow r_2}^{(t)} \mathbf{h}_k^{(t)}], \quad (7)$$

$$\mathbf{s}_{t,r_1,r_2}^{rel} = \mathbf{W}_{t,r_1,r_2}^{rel} [\mathbf{W}_{r_1 \rightarrow r_2}^{(t)} \mathbf{h}_j^{(t)} \oplus \mathbf{W}_{r_1 \rightarrow r_2}^{(t)} \mathbf{h}_k^{(t)}], \quad (8)$$

where $\mathbf{W}_{t,r_1,r_2}^{spa}$, $\mathbf{W}_{t,r_1,r_2}^{rel}$, and \mathbf{G}_{t,r_1,r_2} denote learnable weighted matrices, \oplus represent the concatenation operation. \mathbf{G}_{t,r_1,r_2} transforms the spatial distance representation $\mathbf{d}_{i,j}$ in the relational path-specific latent space. Under the guidance of dual-factor gated mechanism, the informative second-order aggregation process involves both relational and spatial signals.

After the relational spatial graph convolution scheme is performed for all hybrid paired patterns (r_1, r_2) , we combine all path-specific location representations obtained from Eq. (5) with mean pooling to strengthen the intra-time dependency learning.

$$\mathbf{h}_{i,intra}^{(t)} = \sum_{(r_1, r_2) \in \mathcal{R} \times \mathcal{R}} \frac{1}{|\mathcal{R} \times \mathcal{R}|} \cdot \mathbf{h}_{i,r_1 \rightarrow r_2}^{(t)}, \quad (9)$$

where \mathcal{R} is the relationship set, $|\mathcal{R} \times \mathcal{R}|$ is the number of paths.

4.1.2 Inter-time Interaction: Spatially Evolving Contextual Propagation (SE-Prop). The diversified correlations between locations are also heavily dependent on inter-time interactions. We further propose the spatially evolving propagation layer to capture complex contextual messages, which complements the intra-time spatial aggregation from a dynamic perspective. After updating the location embedding via the inter-time fusion layer, the well-designed SE-Prop explores to integrate the spatially evolving context among multi-temporal location neighbors for better location relationship learning, since the spatial dynamics at different times can play a great role in relationship mining as introduced before.

Specifically, for each location in dynamic graphs, the semantics of embedding at different segments are distinct. Location embeddings at adjacent time segments $T(t)$ can provide the sequential latent information, which has the potential effect on the current

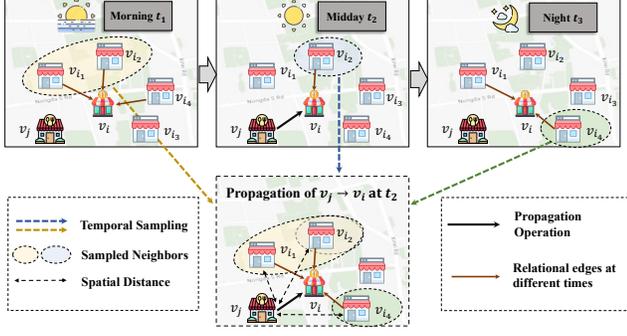


Figure 5: An illustrated example of spatially evolving context construction with random temporal sampling in *SE-Prop*.

location relationship at t due to the time continuity. Thus, the inter-time fusion layer over multiple times is first adopted to update the temporal-enhanced location representation :

$$\tilde{\mathbf{h}}_i^{(t)} = \sum_{\tau \in [T_{t_1}, T_{t_2}]} \frac{1}{T_{t_2} - T_{t_1}} \cdot \mathbf{W}_t \mathbf{h}_{i,intra}^{(\tau)}, \quad T_{t_1} \leq t \leq T_{t_2}, \quad (10)$$

where \mathbf{W}_t is the time-specific transformation matrix, the prior time T_{t_1} and the later time T_{t_2} define the duration of aggregated time segments $T(t)$ (we set T_{t_1} and T_{t_2} as $t - 1$ and $t + 1$ in practice).

Then we further simultaneously leverage the spatial and evolving characteristics among locations in the propagation. As illustrated in Figure 5, different from the classic message passing scheme focusing on pairwise local interactions, the *SE-Prop* additionally considers the spatially evolving context when performing the propagation from each neighboring location v_j to the target v_i . The relationship structure is evolving over time, and a location in the time-specific graph may contain limited connections. Supplying cross-time neighboring nodes can enrich the relational environment in the propagation of $v_j \rightarrow v_i$. Therefore, we first utilize the temporal sampling strategy to gather abundant neighbors, which provides the critical evolving information for the message $v_j \rightarrow v_i$:

$$\mathcal{N}_K(v_i, v_j) = \{v_{i_1}, \dots, v_{i_K} \mid v_k = \text{Sampling}(\cup_{T_{t_1}}^{T_{t_2}} \mathcal{N}_i^{(t)} \setminus \{v_j\})\}, \quad (11)$$

where Sampling stands for the random sampling process across time segments. The set \mathcal{N}_K collects K -size nearby neighbors of v_i with the repeated sampling from multi-temporal neighboring views, which builds the bridge between temporal-correlated locations from time T_{t_1} to T_{t_2} for evolving context construction (e.g., picking up $\{v_{i_1}, v_{i_2}, v_{i_3}\}$ from t_1 to t_3 in Figure 5).

Moreover, the sampled locations are spatially distributed around the target interactive pair $v_j \rightarrow v_i$. The geographical knowledge under the evolving influence can offer multifaceted contexts from a comprehensive view. We introduce an informative vector $\bar{\mathbf{C}}_{i,j}^{(t)}$ to extract such spatially evolving context among the edge $v_j \rightarrow v_i$ with considering the geographical distribution:

$$\bar{\mathbf{C}}_{i,j}^{(t)} = \text{Pooling}(\{\mathbf{G}^{(t)} \mathbf{d}_{j,k} \odot \tilde{\mathbf{h}}_k^{(t)}, v_k \in \mathcal{N}_K(v_i, v_j)\}), \quad (12)$$

where $\mathbf{G}^{(t)}$ is the matrix for distance transformation, $\mathbf{d}_{j,k}$ denotes the internal spatial embedding between two neighbors v_j and v_k , \odot is the Hadamard product for element-wise multiplication. In practice, we utilize the mean pooling with the all-round distance

integration to enable the model to comprehend how far other neighbors in \mathcal{N}_K away from the interacted pair $v_j \rightarrow v_i$. As a result, every message from the neighbor v_j to v_i can carry the evolving context in \mathcal{N}_K together with the spatial context $(\mathbf{d}_{i,j}, \mathbf{d}_{j,i_1}, \dots, \mathbf{d}_{j,i_K})$, rather than only the partial pairwise $\mathbf{d}_{i,j}$.

Finally, we present the contextual propagation module to combine each local neighboring location with the spatially evolving context. This procedure both considers a series of dynamic graph structures and the detailed contexts, which could be formulated in an interactive manner between adjacent time segments:

$$\mathbf{h}_{i,inter}^{(t)} = \sum_{\tau \in [T_{t_1}, T_{t_2}]} \sum_{v_j \in \mathcal{N}_i^{(\tau)}} \mathbf{W}_{\text{Prop}}^{(t)} [(\mathbf{G}^{(t)} \mathbf{d}_{i,j} \odot \tilde{\mathbf{h}}_j^{(\tau)}) \oplus \bar{\mathbf{C}}_{i,j}^{(t)}], \quad (13)$$

where $\mathbf{W}_{\text{Prop}}^{(t)}$ is shared learning matrix for contextual propagation.

4.2 Spatially Evolving Self-Supervised Learning

With the proposed two components *RS-AGG* and *SE-Prop*, we finally obtain the multi-temporal location embeddings $\{\mathbf{h}_{i,inter}^{(t_1)}, \dots, \mathbf{h}_{i,inter}^{(t_r)}\}$. In the following sections, we use symbols $\{z_i^{(t_1)}, \dots, z_i^{(t_r)}\}$ to represent these time-specific embeddings for simplicity.

As depicted in Figure 3(b), we intend to design two essential self-supervised learning tasks to deal with the issue of the sparse relationship labels between locations, which enhances the representation learning process of SECONV beyond spatially intra-time and dynamically inter-time structure modeling. Although it has been proved that applying the graph Self-Supervised Learning (SSL) strategy is effective for the general issue of scarce labeled data [9, 30, 50], current SSL frameworks always fail to capture the complicated evolving patterns among locations under the spatially distributed environment. To this end, the learning procedure component, i.e., SE-SSL, is proposed for multi-temporal location relationships in a self-supervised manner, which contains the global spatial information maximum \mathcal{L}_{global} with the additional local evolving constraint \mathcal{L}_{local} . As a whole, we have the joint learning objective:

$$\mathcal{L}_{ssl} = \lambda_1 \mathcal{L}_{global} + \lambda_2 \mathcal{L}_{local} + \|\Theta\|_2^2, \quad (14)$$

where λ_1 and λ_2 are the hyper-parameters to balance the contributions of local-global loss functions, $\|\Theta\|_2^2$ is the L2 regularization.

4.2.1 Global Spatial Information Maximum. Inspired by the success of Deep Graph Infomax (DGI) [30], we first develop the spatial information maximum objective to capture the global evolving patterns with preserving the gridding spatial dynamics. For time-evolving location graphs, the latent semantics of individual nodes do not stay unchanged over time segments. It is reasonable to treat the relational evolution as a generally smooth process since the human behaviors are gradually varying between adjacent times, indicating that the global surroundings around locations remain partially similar and complementary at the next time segment.

In practical scenarios, the location graph of a city can be partitioned into multiple urban grids [13]. Each urban grid u gathers a cluster of spatially correlated location nodes $\{v_i | v_i \in u\}$ and provides the global surrounding information. All nodes located in an urban grid tend to share the similar spatial environment, which motivates us to introduce such knowledge into the model training.

Different from DGI designed for a single graph, we perform the contrastive learning across the successive time-varying graph structures to fuse the global evolving information into the location representation. Specifically, we first utilize the grid-level pooling function to summarize the location representations to obtain the gridding vector $\mathbf{s}_u^{(t)} = \frac{1}{N_u} \sum_{v_i \in u} \mathbf{z}_i^{(t)}$, where N_u is the number of locations in grid u . After that, the goal is to maximize the cross-time mutual information between location-level representations and urban grid-level representations with the following loss function:

$$\begin{aligned} \mathcal{L}_{global} = & \sum_{t \in \mathbb{T}} \left(\sum_{v_i \in \mathcal{V}} (\mathbb{E}_{pos} [\log \mathcal{D}(\mathbf{z}_i^{(t)}, \mathbf{s}_{u(i)}^{(t-1)})]) \right. \\ & \left. + \frac{1}{|S_i|} \sum_{v_j \in S_i} \mathbb{E}_{neg} [\log(1 - \mathcal{D}(\mathbf{z}_j^{(t)}, \mathbf{s}_{u(i)}^{(t-1)})]) \right], \end{aligned} \quad (15)$$

where $u(i)$ denotes the urban grid where the location v_i belongs to, the bilinear function $\mathcal{D}(\cdot, \cdot)$ is the discriminator to calculate the probability scores which estimates whether v_i is located in the grid u . Note that the natural positive sample $(\mathbf{z}_i^{(t)}, \mathbf{s}_{u(i)}^{(t-1)})$ is extracted from temporal views, while the negative pairs are generated from the gridding sampler S_i . In particular, the location v_i at the t -th time segment and the corresponding urban grid $u(i)$ at the last segment $t - 1$ are regarded as a positive pair. The heuristic sampler aims to distinguish effective negative pairs based on geographical information rather than random sampling.

$$S_i = \{v_j | d_1 < \text{ManhattanDist}[u(i), u(j)] < d_2\}, \quad (16)$$

where the function $\text{ManhattanDist}[\cdot, \cdot]$ over the gridding city map returns the Manhattan distance between the two urban grids $u(i)$ and $u(j)$. For the sake of avoiding inadequate locations which are too close or too far away, we use d_1 and d_2 to define the appropriate sampling scope of spatial areas (we empirically set d_1 and d_2 as 2 and 6 in practice) for high-quality negative samples generation.

4.2.2 Local Relational Evolving Constraint. Besides the global evolving correlations, the relevance among time-specific relationships is also evolutionary from a local perspective. We further introduce the edge-level relational evolving constraint to complement the global grid-level evolution in SE-SSL. The key idea of this constraint is to explore if the relation $r_{ij} \in \mathcal{R}$ between v_i and v_j still remains at time t when this relation exists at the last $t - 1$ time segment. We observe some specific short-term relations would disappear at the subsequent segment t , while a bundle of long-term influential relations would continue to survive. Therefore, considering the complex evolving pattern is of great importance due to the potential time continuity for location relationships. We first acquire the relational edge representation $\mathbf{e}_{ij}^{(t)} = \mathbf{h}_i^{(t)} \odot \mathbf{h}_j^{(t)}$. To further explicitly preserve the relation-aware evolving patterns between adjacent times, we define the following local objective:

$$\begin{aligned} \mathcal{L}_{local} = & \sum_{t \in \mathbb{T}} \sum_{e_{ij} \in \mathcal{E}^{(t)}} \left(\delta(r_{ij}^{(t)}, r_{ij}^{(t-1)}) \log[\varphi(\mathbf{e}_{ij}^{(t)}, \mathbf{e}_{ij}^{(t-1)})] \right. \\ & \left. + (1 - \delta(r_{ij}^{(t)}, r_{ij}^{(t-1)})) \log[1 - \varphi(\mathbf{e}_{ij}^{(t)}, \mathbf{e}_{ij}^{(t-1)})] \right), \end{aligned} \quad (17)$$

where $\mathcal{E}^{(t)}$ is the relational edge set at the t -th time segment, the Kronecker delta function $\delta(\cdot, \cdot)$ outputs 1 only if the relationship r_{ij} remains the same from time $t - 1$ to t , the MLP function $\varphi(\cdot, \cdot)$ is adopted to calculate the evolving probability.

4.3 Time-aware Relationship Inference

After training the proposed SECONV through the spatially evolving self-supervised learning stage, we then take advantage of the well-trained model to predict the time-aware location relationships. Given a pair of locations (v_i, v_j) , the model can learn the pairwise multi-slot location embeddings $\{(\mathbf{z}_i^{(t)}, \mathbf{z}_j^{(t)}) | t \in \mathbb{T}\}$ over the whole time segments \mathbb{T} . Finally, we adopt the time-specific DistMult factorization [41] as the scoring function for prediction at each time.

$$\hat{y}_{ij,r}^{(t)} = \sigma(\mathbf{z}_i^{(t)T} \mathbf{W}_r^{(t)} \mathbf{z}_j^{(t)}), \quad (18)$$

where the time-aware diagonal matrix $\mathbf{W}_r^{(t)}$ is the learnable parameter for relationship r , σ stands for the sigmoid function. Then the cross entropy loss function between the predicted probability $\hat{y}_{ij,r}^{(t)}$ and the label $y_{ij,r}^{(t)}$ is used to jointly optimize the model under a time aggregated multi-task learning manner:

$$\mathcal{L}_{rel} = \sum_{t \in \mathbb{T}} \sum_{(v_i, r, v_j) \in \mathcal{Y}_{trn}} \left(y_{ij,r}^{(t)} \log \hat{y}_{ij,r}^{(t)} + (1 - y_{ij,r}^{(t)}) \log(1 - \hat{y}_{ij,r}^{(t)}) \right), \quad (19)$$

where \mathcal{Y}_{trn} is the training edge set, $y_{ij,r}^{(t)}$ indicates whether there exists the relationship r between v_i and v_j at the time segment t .

5 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the proposed SEENet compared against the state-of-the-art methods. The code of SEENet is available at <https://github.com/PaddlePaddle/PaddleSpatial/tree/main/research/SEENet>.

5.1 Experiment Settings

5.1.1 Datasets. Our experiments are conducted on four real-world citywide datasets from two distinct relationship learning domains. The first two urban **business-based** datasets (*Beijing* and *Tokyo*) are derived from commercial behaviors of users, while the other two (*New York* and *Chicago*) are urban **mobility-based** datasets constructed from trajectory data to ensure the data diversity.

- **Business-based Relational Data.** (Business-RD for short) In the scene of urban business, it has been studied that there are two significant behavior-driven relationships among locations [3, 13], i.e., *competitive* and *complementary* relationships, which could be generated from session-based location query data or check-in data. Therefore, we follow previous works [3, 13, 21] to construct the Business-RD (**Beijing** and **Tokyo**) from query dataset QueryBJ and check-in dataset Foursquare [42] respectively.
- **Mobility-based Relational Data.** (Mobi-RD for short) Since urban mobility is another important aspect to reflect the dynamic location correlations, we further extend the relationship in the mobile trajectory domain. In specific, we utilize the taxi and bike trajectory datasets NYCTaxi¹ and DivvyBike² to generate the Mobi-RD (**New York** and **Chicago**), which includes *high-flow* and *low-flow* relationships according to the flowing degree.

As aforementioned, the special location relationships are influenced by time in reality since the above user behaviors and trajectories are dependent on different time periods in a day (e.g., two restaurants tend to be competitive at midday instead of midnight) [45]. Thus,

¹<https://nyc.gov/site/tlc/about/tlc-trip-record-data.page>

²<https://ride.divvybikes.com/system-data>

Table 1: Overall performance on Business-RD (BEIJING and TOKYO) and Mobi-RD (NEW YORK and CHICAGO). We conduct experiments with five random seeds and report the average performance together with the standard deviation.

Method	BEIJING		TOKYO		NEW YORK		CHICAGO	
	MRR@10	HR@10	MRR@10	HR@10	MRR@10	HR@10	MRR@10	HR@10
GCN	0.1278±0.005	0.2873±0.010	0.1386±0.003	0.3034±0.012	0.1213±0.005	0.3184±0.012	0.1052±0.002	0.3467±0.008
PathGCN	0.1311±0.006	0.3191±0.015	0.1380±0.004	0.3387±0.014	0.1375±0.005	0.3478±0.012	0.1056±0.006	0.3471±0.025
CompGCN	0.1637±0.001	0.4482±0.005	0.1394±0.004	0.3590±0.005	0.1423±0.012	0.3772±0.021	0.0923±0.002	0.3305±0.002
MixHop	0.1703±0.001	0.4582±0.006	0.1412±0.003	0.3618±0.006	0.1488±0.014	0.3528±0.026	0.1094±0.006	0.3491±0.011
NL-GNN	0.1811±0.004	0.4233±0.009	0.1639±0.003	0.4134±0.009	0.1785±0.006	0.4021±0.009	0.1107±0.001	0.3695±0.008
GCA	0.1561±0.016	0.3416±0.035	0.1650±0.003	0.4010±0.012	0.1188±0.002	0.3449±0.006	0.1116±0.002	0.3736±0.005
DGI	0.1776±0.007	0.3893±0.012	0.1738±0.001	0.3976±0.004	0.1538±0.006	0.3985±0.023	0.1098±0.002	0.3587±0.010
RGRL	0.1952±0.006	0.4216±0.015	<u>0.1775±0.007</u>	<u>0.4253±0.016</u>	0.1624±0.001	0.3939±0.006	0.1107±0.002	0.3676±0.005
EvolveGCN	0.2123±0.003	0.4870±0.001	0.1634±0.003	0.4070±0.010	0.2054±0.002	0.4518±0.010	0.0976±0.001	0.3290±0.001
ROLAND	0.2127±0.013	0.4966±0.030	0.1607±0.004	0.4133±0.006	0.1980±0.004	0.4571±0.009	<u>0.1237±0.002</u>	<u>0.3859±0.008</u>
DecGCN	0.1758±0.001	0.4175±0.002	0.1552±0.000	0.3686±0.007	0.1700±0.002	0.4142±0.005	0.1112±0.002	0.3466±0.001
IRGNN	0.1807±0.012	0.4176±0.025	0.1299±0.001	0.3136±0.010	0.1638±0.005	0.3843±0.006	0.1123±0.003	0.3408±0.005
DeepR	<u>0.2184±0.002</u>	<u>0.5257±0.006</u>	0.1662±0.001	0.3902±0.005	0.1988±0.001	0.4496±0.003	0.1058±0.004	0.3628±0.010
PRIM	0.1973±0.001	0.4992±0.002	0.1454±0.003	0.3990±0.006	<u>0.2229±0.007</u>	<u>0.5008±0.002</u>	0.1021±0.001	0.3634±0.003
SEENet	0.2545±0.003	0.5524±0.007	0.2314±0.003	0.4880±0.009	0.2526±0.003	0.5376±0.009	0.1506±0.002	0.4338±0.012

Table 2: Statistics of four real-world datasets.

Dataset	Beijing	Tokyo	New York	Chicago
Relation Type	Business	Business	Mobility	Mobility
Relation Source	Map Query	Check-in	By Taxi	By Bike
# Nodes	30,114	3,013	1,587	483
# Relations at t_1	3,270	1,820	647	1,059
# Relations at t_2	96,233	3,991	7,016	5,894
# Relations at t_3	97,829	9,419	7,909	6,275
# Relations at t_4	4,155	1,446	4,523	1,322

we reasonably detail the relationships with time-aware refinements. In practice, as suggested by the literature [14] with considering the real-life experience and data analysis, we evenly split a day into four segments t_1 - t_4 , i.e., *morning*, *midday*, *night*, and *midnight*. We construct the above four datasets at each time according to timestamps, and finally obtain the **multi-temporal Business-RD and Mobi-RD** in Table 2. The details of relationship construction are included in Appendix A.1.

5.1.2 Setup. Following the previous relationship-based works [13, 18], we randomly sample 10% of relational edges for testing and 10% of edges as the validation set at each time segment, while the remaining 80% of relations are utilized to construct the dynamic location graph for training. We also replace the destination node of each edge with other random locations for negative sampling.

5.1.3 Baselines and Evaluation Metrics. We compare SEENet with a variety of advanced GNN methods for dynamic location relationship inference: (i) The relational-based GNNs (GCN [8], PathGCN [4], CompGCN [29], MixHop [1], and NL-GNN [17]) are typical graph structure learning models considering node correlations and contexts. (ii, iii) We also select recent graph self-supervised learning models (DGI [30], GCA [50], and RGRL [9]) and snapshot-based dynamic GNNs for link prediction (EvolveGCN [25] and ROLAND [44]). (iv) Moreover, SEENet is compared with state-of-the-art relationship prediction methods (DecGCN [21], IRGNN [18], DeepR [13], and PRIM [3]) which are designed for learning relational and spatial dependencies. We conduct time-specific link prediction experiments to rank candidate locations at each time. As introduced in [21] for relationship inference, we

adopt Mean Reciprocal Ranking ($MRR@k$) and Hit Rate ($HR@k$) as evaluation metrics. The baseline descriptions, parameter settings, and more experimental details are introduced in Appendix A.2.

5.2 Overall Performance Comparison

We first compare the overall experimental results on four real-world relationship datasets, where the evaluation metrics are calculated over all time periods to reflect the general prediction performance. As presented in Table 1, we report the metrics MRR@10 and HR@10 in the relationship prediction results following the previous work [21]. The **values in boldface** indicate the best results, while the **underlined values** signify the second-best results. On the whole, it is observed that our SEENet consistently outperforms all different baseline methods by an obvious margin on each dataset. In specific, compared with the second-best model on four datasets, SEENet improves the MRR@10 by 16.5%, 30.4%, 13.3%, and 21.7% respectively. We further have the following observations and findings.

As we can see, the relational-based GNNs roughly perform worse than other types of baselines since they only leverage the graph topological structures with learning node relations. It is not surprising that the recently proposed PathGCN just achieves comparable results with GCN because the learned spatial operators with random paths are uncertain and sometimes may go against the relationship prediction. Moreover, CompGCN performs slightly better because of the ability to distinguish multiple relations, while the non-local message-passing models (Mixhop and NL-GNN) can explicitly learn multi-hop relational contexts and further improve the performance. In general, the above GNNs are not ideal without spatial and dynamic learning schemes, which verifies that simply aggregating relational edges is not adequate for complex dynamic location graphs. As to self-supervised learning methods, RGRL outperforms the other two baselines, as it can enable Self-Supervised Learning (SSL) with capturing the augmentation-invariant relationship at the same time. Note that although both DGI and RGRL adopt the global-local SSL architecture on graphs, our SEENet incorporates the spatial distribution and evolving pattern into SSL beyond the traditional framework and performs much better.

Table 3: Ablation studies with the metric MRR@10.

Variants	Beijing	Tokyo	New York	Chicago
SEENet-SEC	0.1780	0.1636	0.1548	0.1218
SEENet-RS	0.1792	0.1901	0.1744	0.1323
SEENet-LD	0.2149	0.169	0.1951	0.1193
SEENet-C	0.2469	0.1718	0.2169	0.1215
SEENet-SSL	0.2321	0.2042	0.1937	0.1371
SEENet-L	0.2466	0.2157	0.2116	0.1441
SEENet-G	0.2463	0.2215	0.2493	0.1429
SEENet	0.2545	0.2314	0.2526	0.1506

From the perspective of dynamic relationship modeling, since EvolveGCN and ROLAND can take advantage of dynamic propagation cross times to preserve time-aware relational characteristics, these models exhibit considerable improvement over classic GNNs. Furthermore, we can see that the latest ROLAND can not always outperform EvolveGCN due to the failure of recurrent designs on some datasets, indicating dynamic GNNs are still not powerful enough in the domain of location graph learning. From the perspective of special relationship prediction approaches, we notice that spatial-oriented models (DeepR and PRIM) tend to perform better than DecGCN and IRGNN as a result of considering essential geographical information and adapting the relationship learning in the scenario of location graphs. One exception is on the Chicago dataset with the fewest relations. The potential reason is that DeepR and PRIM requiring enough neighbors lose their effectiveness when learning spatial contexts on small graphs. By contrast, our model can fully capture both spatial correlations and evolving dynamics with graph convolution-level and SSL-level enhancements. Therefore, SEENet is much more effective for multi-temporal relationship inference among locations.

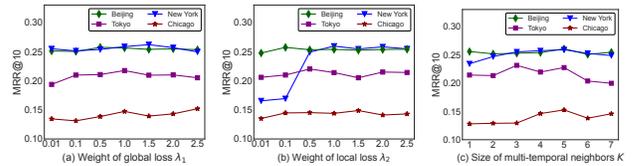
5.3 Impact of Spatial and Dynamic Designs

5.3.1 How SEENet Architecture Design Helps (Model Analysis). To investigate the contribution of each component in our designed multi-slot spatial graph convolutions SEConv, we compare SEENet with the following variants on four datasets in Table 3.

- **SEENet-SEC** replace the whole SEConv with a classic GCN.
- **SEENet-RS** removes spatial *RS-AGG* for intra-time learning.
- **SEENet-LD** removes dynamic *SE-Prop* for inter-time learning.
- **SEENet-C** drops the evolving context of *SE-Prop* in Eq. (13).

It is obvious that other variants of SEENet outperform SEENet-SEC and the performance generally decreases when we gradually remove the graph learning components. In particular, if we replace the time-aware contextual aggregation with a simple GCN-style function when performing inter-time interactions, we find that SEENet-C gets worse, which confirms that it is beneficial to integrate the spatially evolving context for Trial. Furthermore, SEENet-LD performs better than SEENet-RS on Beijing and New York datasets while the converse is observed on the other two datasets, proving both spatial and dynamic modeling can play significant roles across different scenarios. In summary, the results highlight the importance of designing synergistic SEENet architecture to combine geographical factors and dynamic relationships.

5.3.2 How Self-Supervised Learning Design Helps (SE-SSL analysis). We also conduct essential experiments to validate the effectiveness of the well-designed SE-SSL with removing different objectives.

**Figure 6: Parameter analysis on four citywide datasets.**

- **SEENet-SSL** w/o pre-training, i.e., removing the whole SE-SSL.
- **SEENet-L** w/o the loss of local relational evolving constraint.
- **SEENet-G** w/o the loss of global spatial information maximum.

As we can see in Table 3, there is a consistent performance degradation when excluding either global-view loss or local-view loss. SEENet-SSL performs even worse than SEENet-G and SEENet-L when dropping global and local learning objectives at once, showing both of them can contribute to model training. The observation verifies that considering global spatial distribution as well as local evolving relationship patterns in a self-supervised learning manner is critical for location relationship inference.

We further explore the influence of various important parameters in SEENet. More experimental results are in Appendix A.3.3.

5.3.3 Coefficients of global and local loss functions. As depicted in Figure 6(a) and 6(b), we first study the weight of global loss λ_1 and local loss λ_2 respectively. When increasing the coefficient for each loss, the results slightly get better and keep stable in general, with the exception that the performance on New York increases rapidly in the beginning and then remains at high scores. This indicates integrating more local evolving relationship information is necessary for some scenarios. Overall, our collaborative global-local learning makes the model training more expressive and stable.

5.3.4 Scales of spatially evolving context. We also investigate the number of sampled neighbors K for capturing critical multi-temporal spatial factors in *SE-Prop*. Figure 6(c) shows that the performance of SEENet gradually improves at first and then declines with neighbors growing, which achieves the best at $K = 5$. The reason is that communicating more location neighbors in *SE-Prop* can provide more informative spatial contexts, while sampling excessive neighbors may lead to unexpected homogenization of the propagation. To conclude, moderate fine-grained contextual information can help to outperform all baselines in our problem Trial.

6 CONCLUSION

In this paper, we provided a new multi-temporal perspective to understand the location relationship, which is significant in urban intelligence. In detail, we proposed a spatially evolving GNN framework, named SEENet, to effectively discover multi-temporal location relationships in urban areas. The designed spatially evolving convolution can capture intra- and inter-time spatial contexts with dynamic influence. To overcome the issue of data sparsity, we also devised essential self-supervised learning tasks to integrate evolving patterns. Extensive experiments were conducted on four datasets to verify the effectiveness of the proposed model.

ACKNOWLEDGMENTS

The work was partially supported by grants from the National Natural Science Foundation of China (Grant No.61960206008).

REFERENCES

- [1] Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *ICML*. PMLR, 21–29.
- [2] Wei Chen, Huaiyu Wan, Shengnan Guo, Haoyu Huang, Shaojie Zheng, Jiamu Li, Shuohao Lin, and Youfang Lin. 2022. Building and exploiting spatial-temporal knowledge graph for next POI recommendation. *KBS* 258 (2022), 109951.
- [3] Yile Chen, Xiucheng Li, Gao Cong, Cheng Long, Zhifeng Bao, Shang Liu, Wanli Gu, and Fuzheng Zhang. 2022. Points-of-interest relationship inference with spatial-enriched graph neural networks. *PVLDB* 15, 3 (2022), 504–512.
- [4] Moshe Eliasof, Eldad Haber, and Eran Treister. 2022. pathgcn: Learning general graph spatial operators on paths. In *ICML*. PMLR, 5878–5891.
- [5] Qiang Gao, Wei Wang, Li Huang, Xin Yang, Tianrui Li, and Hamido Fujita. 2023. Dual-grained human mobility learning for location-aware trip recommendation with spatial-temporal graph knowledge fusion. *Information Fusion* 92 (2023), 46–63.
- [6] Linlang Jiang, Jingbo Zhou, Tong Xu, Yanyan Li, Hao Chen, and Dejing Dou. 2022. Time-aware Neural Trip Planning Reinforced by Human Mobility. In *IJCNN*. 1–8.
- [7] Nitin Jindal and Bing Liu. 2006. Identifying comparative sentences in text documents. In *SIGIR*. 244–251.
- [8] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.
- [9] Namkyeong Lee, Dongmin Hyun, Junseok Lee, and Chanyoung Park. 2022. Relational self-supervised learning on graphs. In *CIKM*. 1054–1063.
- [10] Rui Li, Shenghua Bao, Jin Wang, Yong Yu, and Yunbo Cao. 2006. Cominer: An effective algorithm for mining competitors from the web. In *ICDM*. IEEE, 948–952.
- [11] Shuangli Li, Jingbo Zhou, Tong Xu, Dejing Dou, and Hui Xiong. 2022. Geomgcl: Geometric graph contrastive learning for molecular property prediction. In *AAAI*, Vol. 36. 4541–4549.
- [12] Shuangli Li, Jingbo Zhou, Tong Xu, Liang Huang, Fan Wang, Haoyi Xiong, Weili Huang, Dejing Dou, and Hui Xiong. 2021. Structure-aware interactive graph neural networks for the prediction of protein-ligand binding affinity. In *SIGKDD*. 975–985.
- [13] Shuangli Li, Jingbo Zhou, Tong Xu, Hao Liu, Xinjiang Lu, and Hui Xiong. 2020. Competitive analysis for points of interest. In *SIGKDD*. 1265–1274.
- [14] Xin Li, Mingming Jiang, Huiting Hong, and Lejian Liao. 2017. A time-aware personalized point-of-interest recommendation via high-order tensor factorization. *TOIS* 35, 4 (2017), 1–23.
- [15] Defu Lian, Yongji Wu, Yong Ge, Xing Xie, and Enhong Chen. 2020. Geography-aware sequential location recommendation. In *SIGKDD*. 2009–2019.
- [16] Hao Liu, Qiyu Wu, Fuzhen Zhuang, Xinjiang Lu, Dejing Dou, and Hui Xiong. 2021. Community-Aware Multi-Task Transportation Demand Prediction. In *AAAI*. 320–327.
- [17] Meng Liu, Zhengyang Wang, and Shuiwang Ji. 2021. Non-local graph neural networks. *TPAMI* (2021).
- [18] Weiwen Liu, Yin Zhang, Jianling Wang, Yun He, James Caverlee, Patrick PK Chan, Daniel S Yeung, and Pheng-Ann Heng. 2021. Item relationship graph neural networks for e-commerce. *TNNLS* (2021).
- [19] Yu Liu, Jingtao Ding, and Yong Li. 2021. Knowledge-driven site selection via urban knowledge graph. *arXiv preprint arXiv:2111.00787* (2021).
- [20] Yu Liu, Jingtao Ding, and Yong Li. 2022. Developing knowledge graph based system for urban computing. In *SIGSPATIAL*. 3–7.
- [21] Yiding Liu, Yulong Gu, Zhuoye Ding, Junchao Gao, Ziyi Guo, Yongjun Bao, and Weipeng Yan. 2020. Decoupled graph convolution network for inferring substitutable and complementary items. In *CIKM*. 2621–2628.
- [22] Hui Luo, Jingbo Zhou, Zhifeng Bao, Shuangli Li, J Shane Culpepper, Haochao Ying, Hao Liu, and Hui Xiong. 2020. Spatial object recommendation with hints: When spatial granularity matters. In *SIGIR*. 781–790.
- [23] Mohammad Mahdian, Okke Schrijvers, and Sergei Vassilvitskii. 2015. Algorithmic cartography: Placing points of interest and ads on maps. In *SIGKDD*. 755–764.
- [24] Julian McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring networks of substitutable and complementary products. In *SIGKDD*. 785–794.
- [25] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao Schardl, and Charles Leiserson. 2020. Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In *AAAI*, Vol. 34. 5363–5370.
- [26] Vineeth Rakesh, Suhang Wang, Kai Shu, and Huan Liu. 2019. Linked variational autoencoders for inferring substitutable and supplementary items. In *WSDM*. 438–446.
- [27] Zahraa Al Sahili and Mariette Awad. 2023. Spatio-Temporal Graph Neural Networks: A Survey. *arXiv preprint arXiv:2301.10569* (2023).
- [28] George Valkanas, Theodoros Lappas, and Dimitrios Gunopoulos. 2017. Mining competitors from large unstructured datasets. *TKDE* 29, 9 (2017), 1971–1984.
- [29] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *ICLR*.
- [30] Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep Graph Infomax. In *ICLR*.
- [31] Huanong Wang, Qiaohong Yu, Yu Liu, Depeng Jin, and Yong Li. 2021. Spatio-Temporal Urban Knowledge Graph Enabled Mobility Prediction. *IMWUT* 5, 4 (2021), 1–24.
- [32] Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. 2019. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In *SIGKDD*. 968–977.
- [33] Senzhang Wang, Jiannong Cao, and Philip Yu. 2020. Deep learning for spatio-temporal data mining: A survey. *TKDE* (2020).
- [34] Zihan Wang, Ziheng Jiang, Zhaochun Ren, Jiliang Tang, and Dawei Yin. 2018. A path-constrained framework for discriminating substitutable and complementary products in e-commerce. In *WSDM*. 619–627.
- [35] Marcel Werle and Sven Laumer. 2022. Competitor identification: A review of use cases, data sources, and algorithms. *IJIM* 65 (2022), 102507.
- [36] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *TNNLS* 32, 1 (2020), 4–24.
- [37] Congxi Xiao, Jingbo Zhou, Jizhou Huang, Hengshu Zhu, Tong Xu, Dejing Dou, and Hui Xiong. 2023. A Contextual Master-Slave Framework on Urban Region Graph for Urban Village Detection. In *ICDE*.
- [38] Congxi Xiao, Jingbo Zhou, Jizhou Huang, An Zhuo, Ji Liu, Haoyi Xiong, and Dejing Dou. 2021. C-Watcher: A Framework for Early Detection of High-Risk Neighborhoods Ahead of COVID-19 Outbreak. In *AAAI*. 4892–4900.
- [39] Derong Xu, Jingbo Zhou, Tong Xu, Yuan Zia, Ji Liu, Enhong Chen, and Dejing Dou. 2023. Multimodal Biological Knowledge Graph Completion via Triple Co-attention Mechanism. In *ICDE*.
- [40] Jia Xu, Fei Xiong, Zulong Chen, Mingyuan Tao, Liangyue Li, and Quan Lu. 2022. G2NET: A General Geography-Aware Representation Network for Hotel Search Ranking. In *SIGKDD*. 4237–4247.
- [41] Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *ICLR*.
- [42] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2014. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45, 1 (2014), 129–142.
- [43] Yang Yang, Jie Tang, Jacklyne Keomany, Yanting Zhao, Juanzi Li, Ying Ding, Tian Li, and Liangwei Wang. 2012. Mining competitive relationships by learning across heterogeneous networks. In *CIKM*. 1432–1441.
- [44] Jiakuan You, Tianyu Du, and Jure Leskovec. 2022. ROLAND: graph learning framework for dynamic graphs. In *SIGKDD*. 2358–2366.
- [45] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-aware point-of-interest recommendation. In *SIGIR*. 363–372.
- [46] Kaichen Zhang, Jingbo Zhou, Donglai Tao, Panagiotis Karras, Qing Li, and Hui Xiong. 2020. Geodemographic influence maximization. In *SIGKDD*. 2764–2774.
- [47] Le Zhang, Tong Xu, Hengshu Zhu, Chuan Qin, Qingxin Meng, Hui Xiong, and Enhong Chen. 2020. Large-scale talent flow embedding for company competitive analysis. In *The WebConf*. 2354–2364.
- [48] Jingbo Zhou, Tao Huang, Shuangli Li, Renjun Hu, Yanchi Liu, Yanjie Fu, and Hui Xiong. 2021. Competitive Relationship Prediction for Points of Interest: A Neural Graphlet Based Approach. *TKDE* (2021).
- [49] Jingbo Zhou, Hongbin Pei, and Haishan Wu. 2018. Early warning of human crowds based on query data from Baidu maps: Analysis based on Shanghai stampede. *Big data support of urban planning and management: The experience in China* (2018), 19–41.
- [50] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In *The WebConf*. 2069–2080.

A APPENDIX

A.1 Dataset Introduction

• **Business-based Relationships.** We first use the QueryBJ dataset collected from Baidu Maps, which contains millions of map search query logs from January 2019 to August 2019 in **Beijing**. Each query log provides the user’s query session with timestamps. Following the works [3, 21], the two types of relations between locations v_i and v_j are defined as:

- (1) Users viewed v_i also viewed v_j within a query session;
- (2) Users viewed v_i then viewed v_j across different sessions.

As for the public Foursquare dataset [42], we collect the check-in data from April 2012 to February 2013 in **Tokyo**. All visited locations of a user are classified into several categorical trips (e.g., all restaurants visited in a day form a categorical trip). Similarly, following [13], we have the two kinds of relationships as follows:

- (3) Users visited v_i also visited v_j within a categorical trip;
 - (4) Users visited v_i then visited v_j across different categories.
- According to these relationship studies [3, 13, 24], we refer to (1) and (3) as *competitive* relationships while the other (2) and (4) are known as *complementary* relationships.

• **Mobility-based Relationships.** For the taxi driving data, the public NYCTaxi includes the order records traveling throughout **New York** from January 2015 to June 2015, while the dataset DivvyBike collects bike riding orders from January 2017 to June 2017 in **Chicago**. In this scene, the location can be a bike station or a region. Each trajectory by taxi or bike includes the pick-up and drop-off locations with timestamps, which connects a pair of locations. After counting the overall records for mobile-based relevancy, we label the two meaningful relationships between location pairs, namely *high-flow* (top 25% high) and *low-flow* (top 50% high) relationships according to the mobility degree.

A.2 Experiment Details

A.2.1 Implementation and setup. We train the model on 24 Intel CPUs and a group of Tesla P40 GPUs. The unified input location features are initialized with one-hot embedding for generalized representation learning. For data splitting, we guarantee that the testing and validation sets \mathcal{Y}_{pred} are absolutely independent of the training set \mathcal{Y}_{trn} across all times, meaning that every relational edge in \mathcal{Y}_{pred} at a certain time will not appear in \mathcal{Y}_{trn} at any time.

A.2.2 Evaluation Metrics. In inferring location relationships, the model should be able to accurately rank the relational locations instead of only predicting whether the relationships exist or not. Thus, it is important that relevant locations are ranked higher than irrelevant ones with the ranking metrics MRR and HR. Given the test set T , we follow these steps for evaluation at each time segment: (1) For each relational pair (v_i, v_j, r) , we first calculate all pairwise scores for relation r between v_i and the N locations in the candidate set S using the relational-specific prediction function f_r . (2) We sort these N locations in descending order and obtain the ranked list. The ranking index of node v_j is denoted as $rank_j$ (i.e., a certain position in the list). (3) If $rank_j \leq k$, we consider it a hit (successfully discovering the target location v_j from the top- K locations), otherwise, we consider it a miss. The HR@ k metric is defined as the average of total hits over the entire test set T ,

which is denoted as $HR@k = \frac{\sum_{rank_j \leq k} 1}{|T|}$. (4) To calculate the MRR@ k metric, we first compute the reciprocal rank, denoted as $\frac{1}{rank_j}$. Note that the reciprocal rank is 0 when getting a miss (i.e., $rank_j > k$). Thus, the MRR@ k metric is defined as the average of total reciprocal ranks over the entire test set T , which is denoted as $MRR@k = \frac{1}{|T|} \sum_{rank_j \leq k} \frac{1}{rank_j}$.

A.2.3 Parameter Settings. For all models, we set the dimension of embedding and hidden layers to 64 with the two-layer GNN architecture. Since some baselines are not proposed for relationship discovery, the DistMult function [41] is also adopted as the final prediction layer for these models. In our proposed SEENet, the model is trained by Adam optimizer with an initial learning rate of 0.01. The number of negative samplers is set to 5. For the GNN framework of SECONV, we set the number of distance bins to 40, and the size of multi-temporal neighbors to 5 for spatial context learning. For the self-supervised learning of SE-SSL, the related parameters of balancing weights (i.e., λ_1 and λ_2) and gridding size in the global loss are set according to the experimental results on the validation set (The certain range is shown in Figure 6 and Figure 7).

For baseline models, the number of random walk and the path length are both set to 5 in pathGCN, while the composition operator of subtraction is used with the dropout rate of 0.2 in CompGCN. For high-order graph learning methods, the set of integer adjacency powers in MixHop is defined as $P = \{0, 1, 2\}$, and the kernel size in NL-GNN is set to 5 for effective non-local context learning. As to graph self-supervised learning methods, we also use the bilinear scoring function as the discriminator for DGI. The degree centrality function with the dropping probabilities of 0.3 and 0.2 is employed for graph augmentation in GCA. The balancing coefficient for relational self-supervised learning is set to 0.9 in RGRL. For dynamic GNN models, the MLP and GRU modules are adopted for recurrent updater in EvolveGCN and ROLAND. For relationship learning methods (DecGCN and IRGNN), we stack two convolutional layers with two 3-layer MLP functions for complex relational dependencies. For DeepR, The grid size of buckets and the number of sectors are set to 100 (meters) and 4, respectively. The number of attentive heads is set to 2 with the scaling factor of 2 in PRIM.

A.2.4 Baseline Method Description. We compare SEENet with the following methods for multi-temporal relationship inference:

- **GCN** [8] is a well-known graph neural network, which aggregates nodes with topological weights for relational modeling.
- **PathGCN** [4] adopts the point-wise graph convolutions to learn the complex spatial operator from random paths for improving relationship prediction performance.
- **CompGCN** [29] extends the GCN architecture to jointly embeds both nodes and relationships in the graph. It can incorporate multi-relational information with composition operations.
- **MixHop** [1] is a kind of higher-order message passing network, where nodes receive abundant and distant information with mixing feature representations of neighbors at various distances.
- **NL-GNN** [17] is a recent non-local aggregation framework with an attentive sorting to capture global relational structures.
- **DGI** [30] leverages the mutual information maximization for graph self-supervised learning (SSL) through a local-global scheme.

Table 4: Time-specific performance evaluation (MRR@10) over times on two datasets of Business-RD and Mobi-RD.

Dataset	Method	Morning	Midday	Night	Midnight
BEIJING	NL-GNN	0.2302	0.1803	0.1821	0.1340
	RGRL	0.2232	0.1901	0.2018	0.1352
	EvolveGCN	<u>0.2718</u>	0.2072	0.2169	<u>0.1760</u>
	ROLAND	0.2402	0.2080	<u>0.2187</u>	0.1567
	DeepR	0.2152	<u>0.2247</u>	0.2159	0.1334
	PRIM	0.2114	0.2005	0.1962	0.1409
	SEENet	0.3206	0.2442	0.2621	0.2602
NEW YORK	NL-GNN	0.1386	0.2042	0.2029	0.1532
	RGRL	0.1778	0.1419	0.1607	0.1850
	EvolveGCN	0.1463	0.2185	0.2259	0.1576
	ROLAND	0.1799	0.2027	0.2129	<u>0.1993</u>
	DeepR	0.1838	0.2122	0.2080	0.1734
	PRIM	<u>0.2072</u>	<u>0.2276</u>	<u>0.2420</u>	0.1976
	SEENet	0.2722	0.2367	0.2714	0.2415

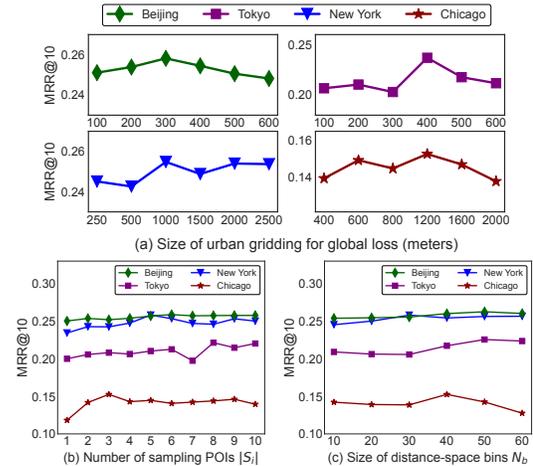
- **GCA** [50] develops the graph contrastive SSL with advanced adaptive augmentations to enhance representation learning.
- **RGRL** [9] is a recent relational-aware SSL framework to alleviate the data scarcity issue by considering the relationship among nodes in both global and local perspectives.
- **EvolveGCN** [25] employs an RNN module to dynamically update weights of internal GNNs for dynamic link predictions.
- **ROLAND** [25] is the latest snapshot-based GNN model to further generalize the relational GNN to a dynamic setting, which can enable the prediction of multi-time relations.
- **DecGCN** [21] generates node embeddings in separated relationship-specific spaces, which can capture the mutual inference between structural and semantic information for relationship discovery.
- **IRGNN** [18] is proposed to discover multiple relationships by incorporating multi-hop relational information on sparse graphs.
- **DeepR** [13] introduces spatial adaptive GNN model to handle the unique spatial attribute of location graphs and achieves great performance for static relationship inference.
- **PRIM** [3] is the current state-of-the-art GNN model for location relationship inference. Both the weighted relational convolution and self-attentive spatial context extraction improve the results.

A.3 Additional Experimental Results

A.3.1 Time-specific Performance Analysis. Table 4 exhibits the multi-temporal relationship prediction results over all times in a day. The proposed SEENet provides stable performance gains over major competitive baselines at all time periods, which consistently verifies the superiority of our multi-slot spatial learning framework in time-aware predictions. We also observe that most baselines perform worse at midnight due to the sparsity of location relationships. We observe that the graph at midnight with fewer people activities only contains limited environmental information for location relationship inference. Moreover, we find that dynamic GNNs (EvolveGCN and ROLAND) can partly alleviate the above issue thanks to the capability of involving the relational knowledge from the previous time, which agrees with the fact that considering dynamic relationships is essential and valuable. However, the performance is still unsatisfactory. By contrast, our model can achieve stable and accurate prediction results at any time, demonstrating the exhaustive effectiveness of SEENet.

Table 5: The efficiency studies on Beijing dataset.

Method	Training (s/epoch)	Inference (s/rank)
RGRL	1.4464	0.1212
ROLAND	2.8150	0.1540
DeepR	2.3462	0.2340
PRIM	1.8416	0.2058
SEENet	2.1631	0.2517

**Figure 7: More parameter analysis on four citywide datasets.**

A.3.2 Efficiency Analysis. We also conduct the efficiency evaluation for the training and inference time. We compare our model with several most competitive baselines on the largest Beijing dataset in Table 5. The results show that SEENet can be comparably efficient with other models and will not sacrifice much computation and training time to trade for performance. It is worth noting that the “inference (s/rank)” refers to the total time of ranking all nodes on the graph for a specific location. When we use SEENet to discover location relationships, inferring the ranking list for one location on a large graph (over 30,000 nodes) takes only 0.25s on average.

A.3.3 Additional Parameters Analysis. As shown in Figure 7, we first present the influence of urban gridding size in the global spatial information maximum loss, which determines the spatial granularity for global region pooling. With the growth of the gridding size (e.g., the larger region area), the performance of our model first increases and then tends to decrease. The reason is that SEENet needs a suitable splitting scale for higher-level aggregation according to different citywide application domains. While the larger region grid can contain more informative spatial locations, additional redundancies may be introduced for relationship learning. Moreover, we change the another important parameter of SE-SSL, i.e., geographical sampling size in S_i , to investigate the affect of heuristic negative sampler in Figure 7(b). The performance improves slightly at first and then keep relatively stable as the number of negative samples grows, verifying that more grid-based negative samples can enhance the global spatial self-supervised learning. Finally, Figure 7(c) presents the effect of the size of distance-space bins N_b . We can observe that our model can stably perform well with changing distance bins due to the adaptive ability to learn spatial distances in SEENet.