Spatial-Temporal Cross-View Contrastive Pre-training for Check-in Sequence Representation Learning

Letian Gong, Huaiyu Wan, Shengnan Guo*, Xiucheng Li, Yan Lin
Erwen Zheng, Tianyi Wang, Zeyu Zhou, Youfang Lin

Abstract—The rapid growth of location-based services (LBS) has yielded massive amounts of data on human mobility. Effectively extracting meaningful representations for user-generated check-in sequences is pivotal for facilitating various downstream services. However, the user-generated check-in data are simultaneously influenced by the surrounding objective circumstances and the user's subjective intention. Specifically, the temporal uncertainty and spatial diversity exhibited in check-in data make it difficult to capture the macroscopic spatial-temporal patterns of users and to understand the semantics of user mobility activities. Furthermore, the distinct characteristics of the temporal and spatial information in check-in sequences call for an effective fusion method to incorporate these two types of information. In this paper, we propose a novel Spatial-Temporal Cross-view Contrastive Representation (STCCR) framework for check-in sequence representation learning. Specifically, STCCR addresses the above challenges by employing self-supervision from "spatial topic" and "temporal intention" views, facilitating effective fusion of spatial and temporal information at the semantic level. Besides, STCCR leverages contrastive clustering to uncover users' shared spatial topics from diverse mobility activities, while employing angular momentum contrast to mitigate the impact of temporal uncertainty and noise. We extensively evaluate STCCR on three real-world datasets and demonstrate its superior performance across three downstream tasks.

Index Terms—check-in sequence, representation learning, spatial-temporal cross-view, contrastive cluster.

1 Introduction

Location-based services (LBS), such as Gowalla, Weeplace, and Yelp, have experienced significant development over the past decade. These platforms enable users to share and discover location information and surrounding services, resulting in the accumulation of extensive data on human mobility behavior, *e.g.* check-in sequences at points of interest (POIs). This offers prospects for analyzing and comprehending human mobility patterns for various practical applications, such as predicting the next check-in location or time for personalized recommendations, linking trajectories to users, and detecting abnormal trajectories for safety control purposes *etc.*

Learning accurate and universal representations for check-in sequences is a crucial task in human mobility data mining. However, the existing excellent end-to-end models for check-in sequences modeling, such as those designed for location prediction [1], [2], [3], time prediction [4], and trajectory user link [5], [6], [7], often struggle to learn generalized representations for check-in sequences and fail to comprehensively describe the spatial-temporal patterns and high-level semantics of human mobility, since the supervision signals of these models usually rely on limited single-type labels. Therefore, the learned representations are task-specific and poorly generalized. To facilitate the generalization ability for the check-in sequence's representations, pre-training check-in sequence representation via self-supervised learning has been widely studied and proven to be an effective way to fully exploit massive unlabeled check-in data to boost the performance of the downstream tasks.

Representation learning is always one of the hot research topics in deep learning. And recently, contrastive pre-training with self-supervised signals [8] has emerged as the most effective approach for sequence modeling. In particular, some representative works [9], [10], [11] in the spatial-temporal data mining (STDM) domain have proven their effectiveness in learning the representations of check-in sequences. However, the unique spatial and temporal characteristics of check-in sequences raise challenges for these contrastive pre-training based models, meanwhile weakening their ability to capture the macroscopic spatial-temporal mobility patterns and to understand the high-level semantics of user mobility activities. Specifically, we identify three key challenges:

(1) **Temporal uncertainty**: Understanding the *temporal intention* of users' ability from the check-in sequence with uncertain temporal information is challenging. As shown in Fig. 1, based on the user's historical sequence and the user's historical spatial-temporal behavior patterns, the user is most likely to go for dinner next, with the strongest temporal intention being at 17:00. However, the exact time

- *L. Gong, H. Wan, S. Guo, Y. Lin, E. Zheng, T. Wang, Z.Zhou, Y. Lin are with the Key Laboratory of Big Data & Artificial Intelligence in Transportation, Ministry of Education, Beijing Jiaotong University, Beijing 100044, China, and the Key Laboratory of Intelligent Passenger Service of Civil Aviation, CAAC, Beijing, 101318, China.*
  E-mail: gonglt@bjtu.edu.cn; hywan@bjtu.edu.cn; guoshn@bjtu.edu.cn; ylincs@bjtu.edu.cn; zhengerwen@bjtu.edu.cn; wangtianyi@bjtu.edu.cn; zhouzeyu@bjtu.edu.cn; yflin@bjtu.edu.cn.
- *X. Li is with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, 518055, China.*
  E-mail: lixiucheng@hit.edu.cn.

(Corresponding author: Shengnan Guo.)

Fig. 1. shows the temporal uncertainty is influenced by the subjective intention and objective factors. Users' arrival times tend to be in a range of intervals rather than a precise planned time.

of arrival is simultaneously influenced by his/her subjective decisions such as the today's choice of restaurant, and the surrounding objective factors such as traffic and weather. This leads to the temporal uncertainty, *i.e.*, the user may arrive at the restaurant for dinner around 16:45 to 17:15, rather than the precise time of 17:00. Besides, unexpected bugs on service platforms may also bring noise to the recorded check-in time. The uncertainty and noise make it difficult to extract the user's temporal intention from the raw temporal context of the check-in sequences. Most existing check-in sequence representation learning methods capture the temporal patterns by embedding the precise time of check-ins, overlooking the inherent uncertainty presented in the temporal dimension of check-in sequences. Therefore, it is difficult for these methods to explore the periodic patterns and to understand the temporal intention of the users' mobility relying on noisy time.

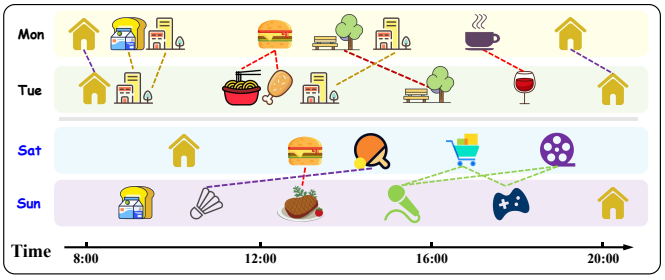

Fig. 2. shows the check-in sequences of a user on working days and weekends, and distinct icons indicate different POIs. Although the check-in POIs on the four days differ, the semantics reflected by the check-in sequences on the two working days (or the two weekends) are similar.

(2) **Spatial diversity**: Users' mobility has a high degree of diversity over POIs. Fig. 2 shows the check-in sequences of a user on consecutive working days and weekends. It can be seen that there are obvious differences between the user's *spatial topic* of working days and weekends. Specifically, the mobility on working days is mainly centered around office-related POIs, while that on weekends is mainly centered around leisure and entertainment POIs. Meanwhile, although the semantics reflected by check-in sequences on the two working days (or the two weekends) are similar, the specific POIs on the two working days (or the two weekends) may be diverse and rarely repeated. We term this phenomenon spatial diversity, which prevents us from effectively capturing the shared mobility patterns between check-in sequences with similar semantics but different POIs. Existing prevalent check-in sequence representation learning methods often adopt the most straightforward word embedding strategy borrowed from the natural language processing (NLP) domain to represent the POIs, *i.e.*, learning embeddings for POIs and retrieving them using indices. That is, these methods treat each POI individually and fail to capture the shared semantics reflected by the POI sequences, resulting in the inability of these models to capture the high-level spatial-temporal semantics of users' mobility.

(3) As discussed above, the raw check-in sequence is discretized and diverse w.r.t. the spatial information, but continuous and uncertain in w.r.t. the temporal information. Consequently, the distinct properties of spatial and temporal information make it challenging to effectively fuse them together. Some existing models adopt a "fuse prior to modeling" approach to model the temporal and spatial information of check-in sequences. However, these models encounter difficulties in simultaneously capturing spatial and temporal semantics within a unified encoder, since they ignore the difference between temporal and spatial information. Furthermore, raw data are at a fine-grained level and barely contain worthwhile semantic information. Besides, there are some disturbances in the raw data, such as temporal noise and spatial diversity. Therefore, combining the fine-grained level properties of temporal and spatial at an early phase would disrupt the performance of the model in relation to each other [12]. Other models explore using separate encoders to learn temporal and spatial representations before fusion. However, the learned temporal and spatial representations reside in separate spaces and lack alignment [13]. They usually use direct collocation or gating mechanisms, ignoring the relevance between spatial topic and temporal intention.

To overcome the aforementioned limitations, we propose the Spatial-Temporal Cross-view Contrastive Representation (STCCR), a pre-training framework for learning the representations of check-in sequences. Our method aims to achieve an effective fusion manner that preserves potential spatial-temporal cross-view correlations at the macroscopic semantic level. By treating one view as the reference, our cross-view contrastive strategy facilitates spatial-temporal information interaction, generating numerous high-quality self-supervisions. Additionally, STCCR learns spatial topics from all check-in sequences using contrastive clustering, capturing topics by exploring the shared mobility pattern from diverse human behavior. In the temporal intention view, we employ an angular margin manner. This leverages angular margin self-supervised signals to mitigate the effects of temporal uncertainty and noise on the angular margin. In summary, our contributions are as follows.

- We propose a novel spatial-temporal cross-view contrastive framework for check-in sequence representation learning from the spatial topic and temporal intention views. To the best of our knowledge, this is the first study to leverage a cross-view contrastive manner to explore the spatial-temporal correlation of human mobility at the macroscopic semantic level.
- We propose an angular margin contrast-based method to exploit the inherent uncertainty of the time information in check-ins. By adding a soft in-

terval to the contrast learning training, the temporal noise information can be filtered so that the model can effectively capture the user's temporal intention.
- We perform contrastive clustering in the spatial dimension. To address the diversity of POIs, we explore shared spatial topics by clustering high-level semantic information from check-in sequences.
- We evaluate STCCR on three real-world datasets for three downstream tasks. The experimental results prove the superiority and versatility of our model.

2 Related work

2.1 Mobility Data Mining

Location-based services have given rise to a new and promising research topic known as mobility data mining, which has led to the emergence of three significant tasks that contribute to enhancing the quality of services: next location prediction (LP), next time prediction (TP), and trajectory user link (TUL). Recent studies have confirmed that deep learning techniques, specifically recurrent neural networks (RNNs) and attention mechanisms, are highly effective in capturing sequential and periodic patterns of human mobility. By combining deep learning techniques, researchers have made significant advancements in capturing both the sequential and periodic patterns of human mobility. The core of these models is the modeling of check-in sequences, which leads to improved accuracy in location prediction and trajectory analysis.

LP aims to anticipate a user's future location based on their historical movement. Several notable models have emerged as prominent approaches in LP. DeepMove [1] leverages RNNs and attention mechanisms to capture the spatial-temporal intentions in users' location data and predict their next destination. STAN [14] introduces a spatial-temporal attention network that incorporates spatial and temporal contexts for accurate prediction. LSTPM [2] focuses on long and short-term patterns in user trajectory using an attention-based LSTM [15] model. SERM [3] utilizes an encoder-decoder architecture with a spatial-temporal residual network to capture user preferences and predict future locations. PLSPL [16] trains two LSTM models for location and category based sequence to capture the user's preference. LightMove [17] designs neural ordinary differential equations to enhance robustness against sparse or incorrect inputs. HMT-GRN [18] alleviates the data sparsity problem by learning different User-Region matrices of lower sparsities in a multitask setting. Graph-Flashback [19] constructs a spatial-temporal knowledge graph to enhance the representation of POIs. GETNext [20] introduces a user-agnostic global trajectory flow map as a means to leverage the abundant collaborative signals.

TUL is a significant task that focuses on establishing connections between different trajectories, facilitating the analysis of user movement patterns, and uncovering valuable insights about their behavior. Notable models have been specifically developed to address the challenge of predicting trajectory links. TULER [6] takes advantage of advanced algorithms to establish links between trajectories, allowing for a comprehensive understanding of user movement patterns. DeepTUL [5] utilizes deep learning techniques to extract representations from trajectory data and facilitate the prediction of trajectory links. S2TUL [21] utilizes graph convolutional networks and sequential neural networks to capture trajectory relationships and intra-trajectory information. GNNTUL [22] employs graph neural networks for human mobility and associates the traces with users on social networks.

TP focuses on estimating the time at which a user is likely to visit their next location. To accomplish this, it is common practice to use intensity functions to represent the rate or density of event occurrences, various models have been developed to model the intensity function and make accurate time predictions effectively. Modeling the intensity function using RNNs or attention mechanisms is a common approach for predicting the occurrence of events. RMTPP [23] utilizes RNNs to model the intensity function. SAHP [24] combines the Hawkes process with self-attention mechanisms to capture the temporal dependencies and spatial influences in event sequences. THP [25] combines the Hawkes process with transformer-based architectures to capture temporal dependencies in event sequences. NSTPP [26] utilizes neural ODEs to model discrete events in continuous time and space, enabling the learning of complex distributions in spatial and temporal domains. IMTPP [27] models the generative processes of observed and missing events and utilizes unsupervised modeling and inference methods for time prediction. DSTPP [28] purposes a novel parameterization framework that uses diffusion models to learn complex joint distributions.

It is important to note that these end-to-end supervised methods designed for specific tasks are not universal. These models do not have a good grasp of the macroscopic semantics of check-in sequences. Thus, learning the universal representation of check-in sequences to improve the model's ability and understand high-level semantics is critical.

2.2 Pretraining and Contrastive Learning

The essence of mobility mining tasks lies in learning the representation of check-in sequences. Numerous studies have demonstrated the effectiveness of employing the pre-training paradigm to achieve check-in sequence representation learning. For instance, TULVAE [7] and MoveSim [29] utilize Variational Auto-Encoder and Generative Adversarial Network, respectively, to capture the movement patterns of check-in sequences through pre-training. TALE [30] proposes a pre-training representation scheme for trajectory point location embedding that incorporates temporal semantics, which effectively improves the performance of next location prediction and location traffic prediction. CTLE [31] proposes a location pre-training representation model that incorporates domain features, which dynamically generates feature representations of the domain environment of the target location so that the model can better capture the macroscopic higher-order semantic information in the latitude and longitude.

As a kind of advanced SSL technology, contrastive learning-based pre-training techniques have demonstrated great potential in the field of Natural Language Processing (NLP). It utilizes self-supervised training by comparing positive and negative pairs generated through data

augmentation., contrastive pre-training models employ a variety of data augmentation strategies. For example, SimCSE [32] utilizes dropout operations for data augmentation. ConSERT [33] disrupts, slices, and deletes representations in the hidden space. VaSCL [34] enhances the discriminative power by introducing challenging negative samples. CLAPS [35] introduces adversarial perturbations to generate indistinguishable augmented samples, thus significantly improving the robustness and discrimination ability.

In the domain of mobility mining, the first model to adopt contrastive learning is SML [10], which applies commonly used data augmentation strategies such as cropping or replacement to check-in sequences. ReMVC [9] learns distinct region embeddings and constraints embedding parameters while transferring knowledge across multiple views. DRAN [36] exploits disentangled representations to capture distinct aspects and corresponding influences for a more precise representation of POIs. CACSR [11] proposes a contrastive pre-training model for learning check-in sequence representations with adversarial perturbations. However, these models are not designed separately for spatial and temporal properties and do not specifically consider spatial-temporal information fusion.

In summary, learning about check-in sequence representation is crucial for mobility mining tasks. Pre-training methods, especially contrastive learning, have demonstrated effectiveness in capturing underlying patterns. It is essential to develop tailored techniques that can effectively extract human spatial-temporal patterns of check-in sequences to improve the performance of contrastive pre-training models in this domain.

3 PRELIMINARIES

3.1 Definitions

Definition 1. **POI Visiting Record.** In location-based services datasets, a user's visit to a certain place is represented by a POI visiting record $r = (u, l, t)$, where u represents the user, l indicates the visited location, and t denotes the timestamp of the visit. The location l is represented by (lid, lon, lat, c), comprising lid as a POI index or a grid index, and accurate longitude lon and latitude lat. c denotes the category of the visited location (*e.g.*, hospital or restaurant).

Definition 2. **Check-in Sequence.** The movement of a user during a specific period can be represented by a list of sequential POI visiting records, which we refer to as a check-in sequence. We denote a check-in sequence as $\mathcal{T} =< r_1, r_2, \cdots, r_s >$, where the POI visit records are ordered by their visited time, and s is the length of the sequence.

3.2 Problem Statement

Pre-training Representation for check-in Sequence. The goal of this paper is to pre-train a parameterized encoder G capable of generating a contextual representation for a given check-in sequence $\mathcal{T}$, denoted as $G(\mathcal{T})$. Specifically, the encoder G is first trained within a spatial-temporal cross-view framework using a contrastive manner, without task-specific objectives. Then, it can be applied to various downstream tasks, such as next Location Prediction (LP), Trajectory User Link (TUL), and Time Prediction (TP), among others. We expect that the parameterized encoder G can be widely used to enhance the performance of these downstream tasks.

4 SPATIAL-TEMPORAL CROSS-VIEW CONTRASTIVE FRAMEWORK

As illustrated in Fig. 3, we propose a **S**patial **T**emporal **C**ross-view **C**ontrastive **R**epresentation (STCCR) model that leverages self-supervision to capture high-level semantics, i.e., the spatial topic and temporal intention of check-in sequences separately and then fuse them at a macroscopic level. To extract the shared spatial topic of the check-in sequence, we introduce the **S**patial **T**opic **M**odule (STM). This module employs contrastive clustering to encode the spatial information of check-in sequences, forcing check-in sequences with similar spatial topics to have similar representations. Additionally, we combine time and user information in the **T**emporal **I**ntention **M**odule (TIM) during pre-training to extract the temporal intention of users. Specifically, we adopt a contrastive learning scheme with an angular margin to model noisy temporal information. Finally, the ST Cross-View Contrastive Module aligns the high-level spatial and temporal semantics into a unified semantic space using project heads, facilitating the integration of spatial-temporal information at the macroscopic semantic level. Next, we provide a detailed explanation of our proposed model in the following sections.

4.1 Spatial Topic Module

The Spatial Topic Module comprises a geohash layer, a transformer layer, and a spatial cluster contrastive block. The geohash layer and transformer layer work together to embed geographical location information into the embedding space. The contrastive spatial cluster block leverages contrastive clustering to capture spatial topics of users' mobility.

4.1.1 Geographical Location Information Encoding

The key advantage of Geohash [1] encoding is its ability to convert geographic coordinates into a string of characters, enabling efficient storage, retrieval, and analysis of location-based data. Geohash represents latitude and longitude information in the following three steps. First, the latitude and longitude are converted into two binary sequences, e_{lat} and e_{lon}. These sequences are obtained by recursively dividing the latitude and longitude ranges. For the latitude value, the range $(-90°, 90°)$ is divided into two sub-ranges: $(-90°, 0)$ and $(0, 90°)$. If the latitude falls within the lower sub-range, a '0' is appended to e_{lat}, otherwise a '1' is appended. The same process is applied to e_{lon} using the initial range $(-180°, 180°)$. Next, the even bits of e_{geo} are set to e_{lat} and the odd bits are set to e_{lon} to create the concatenated binary sequence e_{geo}, where $i = \{0, 1, 2, \cdots, 15\}$.

$$\begin{aligned} e_{(geo,2i)} &= e_{(lat,i)} \\ e_{(geo,2i+1)} &= e_{(lon,i)} \end{aligned} \tag{1}$$

Finally, e_{geo} is converted to Base32 encoding to produce the geohash representation.

1. https://geohash.co

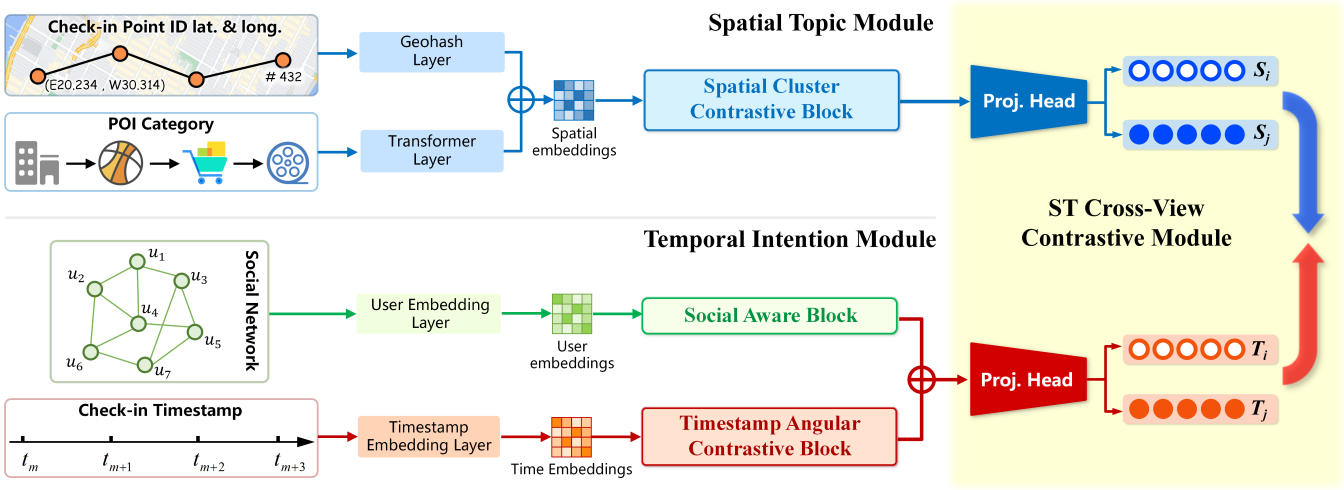

Fig. 3. The model architecture of STCCR. (a)The Spatial Topic Module employs contrast clustering to encode the POIs' id, latitude, longitude, and category. (b)The Temporal Intention Module combines user and time information to obtain the temporal intention patterns of users. (c)The ST Cross-View Contrastive Module aligns spatial and temporal information into a unified semantic space using project heads, facilitating the integration of spatial-temporal information at the macroscopic semantic level.

4.1.2 POI Category representation

To represent the category description of a POI, we treat the description as words and directly utilize a public pre-trained BERT model[2] for sequence representation. We use a variant of BERT in which the final output of the [CLS] token is selected as the representation of the description. The representation of a POI description is denoted as e_{cat}.

4.1.3 Spatial Cluster Contrastive Block

To gain a more comprehensive understanding of user mobility patterns, we propose a spatial cluster contrastive block to capture the underlying shared spatial topics of users' mobility. As discussed in Section 1, we can find a great deal of diversity among the POIs of check-in sequences. Thus, treating each individual check-in sequence separately without considering their common mobility patterns makes it hard to share statistical strength across sequences. We find that users tend to exhibit movement patterns centered on specific spatial topics during different periods. For example, users tend to move around work areas, dining areas, and residential areas during the working days, while focusing on leisure activities such as travel areas, shopping centers, and dining areas during the weekends. Therefore, extracting spatial topics from diverse check-in sequences is crucial to effectively learning the shared mobility patterns of users' movement.

That is, spatial topics refer to the check-in sequences that are generated over different locations but have similar, relative, and shared patterns in terms of spatial movement. To explore the shared spatial topics among sequences, we introduce the "clustering consistency" and "reweighted contrastive" strategies into our model. To represent different shared spatial topics, we define a prototype $\boldsymbol{C}$ which is a set of k cluster centers $\boldsymbol{C} = \{\boldsymbol{c}_1, \cdots, \boldsymbol{c}_K\}$. Meanwhile, we assume that check-in sequences with the same spatial topics fall into a similar semantic space. We use a Bi-GRU as the spatial encoder to encode the spatial information of check-in sequences combined by e_{geo} and e_{cat}. Given a representation of the check-in sequence through the spatial encoder $\boldsymbol{z}_s^n$ as the anchor, we use the dropout augmentation manner as SimCSE [32] to obtain its augmentation $\boldsymbol{z}_s^m$. We calculate each prototype assignment $\boldsymbol{q}_i$ by assessing the similarity of the representation to the prototype as follows:

$$\boldsymbol{q}_i^{(k)} = \frac{\exp \frac{\boldsymbol{z}_s^{j\top} \boldsymbol{c}_k}{\tau}}{\sum_{k' \neq k} \exp \frac{\boldsymbol{z}_s^{j\top} \boldsymbol{c}_{k'}}{\tau}}, \tag{2}$$

where each $\boldsymbol{q}_i = [\boldsymbol{q}_i^{(1)}, \boldsymbol{q}_i^{(2)}, \cdots, \boldsymbol{q}_i^{(k)}, \cdots, \boldsymbol{q}_i^{(K)}]$, $i = \{1, 2, \cdots, N\}$, N is the total number of check-in sequences, τ is a temperature parameter. To ensure the consistency of class attribution between the anchor and augmented sample, we define the clustering consistency loss function as:

$$\mathcal{L}_{\mathrm{C}}\left(\boldsymbol{z}_s^n, \boldsymbol{z}_s^m\right) = \ell\left(\boldsymbol{z}_s^m, \boldsymbol{q}_n\right) + \ell\left(\boldsymbol{z}_s^n, \boldsymbol{q}_m\right), \tag{3}$$

where $\ell(\boldsymbol{z}_s, \boldsymbol{q})$ measures the fit between representation $\boldsymbol{z}_s$ and assignment $\boldsymbol{q}$. We compare the representations $\boldsymbol{z}_s^n$ and $\boldsymbol{z}_s^m$ using their prototype assignments $\boldsymbol{q}_n$ and $\boldsymbol{q}_m$. Each term in Eq. 3 represents the cross-entropy loss between $\boldsymbol{q}$ and the probability obtained by taking a softmax of the dot products of $\boldsymbol{z}_s$ and all columns in $\boldsymbol{C}$, i.e.,

$$\ell\left(\boldsymbol{z}_s^m, \boldsymbol{q}_n\right) = -\sum_k \boldsymbol{q}_n^{(k)} \log \boldsymbol{q}_m^{(k)} \tag{4}$$

Consistency loss makes the anchor and its corresponding sample belong to a similar assignment as much as possible. But this may lead to a plain solution (i.e., the model assigns all samples inside one cluster). To avoid this, we propose a reweighting strategy that assigns larger weights to meaningful negative samples with a moderate prototype distance to the anchor, and smaller weights to negative samples that are easily distinguished. This assigns the samples in the batch and queue to the K classes according to $\boldsymbol{q}$ while satisfying the inter-cluster balance constraint. This makes samples that have similar prototype assignments grouped together as

2. https://huggingface.co

much as possible and avoids the plain-solution problem. We define the reweighting strategy denoted $\mathcal{L}_R$ as:

$$\mathcal{L}_R = -\sum_{n=1}^{N} \log \frac{\phi(n,m)}{\phi(n,m) + M_n \sum_{j\in S} \boldsymbol{w}_{nj}\phi(n,j)}, \quad (5)$$

where $\phi(n,m) = \exp\left(\boldsymbol{z}_s^{n\top}\boldsymbol{z}_s^m/\tau\right)$, $\boldsymbol{w}_{nj}$ is the weight of negative pairs $\left(\boldsymbol{z}_s^n, \boldsymbol{z}_s^j\right)$. $M_n = 2\beta/(\sum_{j\in S}\boldsymbol{w}_{nj})$ is the normalization factor, β is the number of the set S. $S = \{j|\boldsymbol{c}^j \neq \boldsymbol{c}^n\}$, where $\boldsymbol{c}^j$ and $\boldsymbol{c}^n$ are the most probable prototypes of the check-in sequence $\boldsymbol{z}_s^j$ and $\boldsymbol{z}_s^n$, respectively.

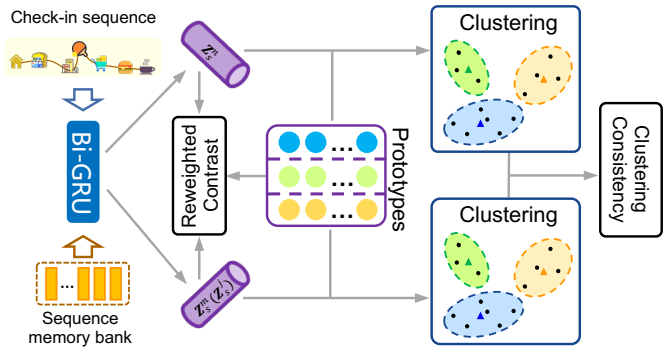

Fig. 4. Spatial cluster contrastive block. We capture spatial topic of user activity through reweighted contrast and cluster consistency manners. In order to improve the clustering effect, we maintain a queue of historical sequences that participate in the computation of the current batch.

We utilize the cosine distance to measure the distance between two assignments $\boldsymbol{q}_n$ and $\boldsymbol{q}_j$ as: $\mathcal{D}(\boldsymbol{q}_n, \boldsymbol{q}_j) = 1 - (\boldsymbol{q}_n \cdot \boldsymbol{q}_j) / \left(\|\boldsymbol{q}_n\|_2 \|\boldsymbol{q}_j\|_2\right)$. Then, we define the weight based on the above assignment distance with the format of the Gaussian function as:

$$w_{nj} = \exp\left\{-\frac{\left[\mathcal{D}(\boldsymbol{q}_n, \boldsymbol{q}_j) - \mu_n\right]^2}{2\sigma_n^2}\right\}, \quad (6)$$

where μ_n and σ_n are the mean and standard deviation of $\mathcal{D}(\boldsymbol{q}_n, \boldsymbol{q}_j)$ for anchor $\boldsymbol{z}_s^n$, respectively. In this way, selected negative samples can enjoy desirable semantic differences from the anchor, and those similar ones are "masked" out in the objective.

Since different clusters represent distinct underlying semantics, such a sampling strategy can ensure a distinguishable semantic difference between the anchor and its negatives. The final training objective is the combination of $\mathcal{L}_R$ and $\mathcal{L}_C$ to jointly optimise the spatial topic, formulated as:

$$\mathcal{L}_{Spatial} = \eta\mathcal{L}_C + \mathcal{L}_R \quad (7)$$

where the constant η balances the clustering consistency loss $\mathcal{L}_C$ and the reweighted contrastive loss $\mathcal{L}_R$. This loss function is jointly minimized concerning the prototype $\boldsymbol{C}$ and the parameters θ of the spatial encoder used to produce the spatial representation $\boldsymbol{z}_s$.

4.2 Temporal Intention Module

The Temporal Intention Module aims at analyzing users' temporal intentions. It includes a timestamp angular contrastive block and a social aware block. They leverage the angular margin to mitigate the effects of temporal uncertainty and noise.

4.2.1 Timestamp Embedding

The timestamp embedding layers convert the original temporal features into dense vectors. Specifically, we first discretize each timestamp t into hourly intervals and then represent it as a T-dimensional one-hot vector (T = 48). To distinguish weekends from weekdays, we treat weekends as an additional 24 hours. Then we learn the embedding for each time interval, denoted by $E_t \in \mathbb{R}^{T\times d_t}$.

4.2.2 Temporal Angular Contrastive Block

This block leverages the angular margin scheme to enable contrastive learning to mitigate the effects of temporal noise and extract users' temporal intention. To model the positive and negative pairwise relations between sequences, we first generate sequence representations and group them into positive and negative pairs. We then use these pairs as input to a training objective for optimization.

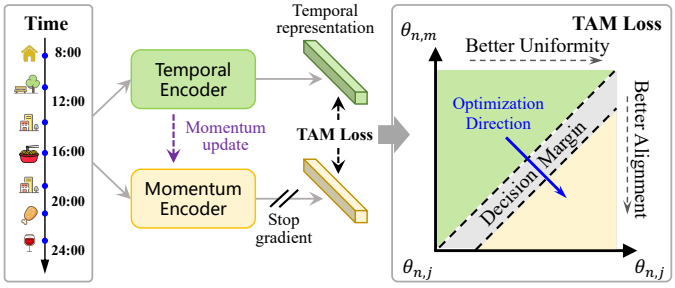

Fig. 5. Angular Margin. Better Uniformity refers to the ability of a model to learn shared representations among similar samples, resulting in improved consistency within the feature space. Better Alignment signifies the model's capability to map different views or variations of the same sample to nearby positions in the feature space, achieving enhanced alignment.

To generate temporal representations from check-in sequences in the temporal dimension, we employ two Bi-GRU encoders, denoted as $\mathcal{M}_t$ and $\mathcal{M}_t'$. Similarly to the approach in ESimCSE [37], we use momentum contrast as the data augmentation method. In particular, we use the momentum-updated encoder to encode the enqueued sequence representation. Formally, denoting the parameters of the encoder $\mathcal{M}_t$ as θ_t and those of the momentum-updated encoder $\mathcal{M}_t'$ as θ_t', we update θ_t' in the following way:

$$\theta_t' \leftarrow \eta\theta_t' + (1-\eta)\theta_t, \quad (8)$$

where $\eta \in [0,1)$ is a momentum coefficient parameter. Note that only the parameters θ_t are updated by back-propagation. To generate temporal representations, we introduce a new set of parameters denoted as θ_t', which are updated using momentum to ensure a smoother evolution than θ_t. We obtain two different temporal representations, denoted as the anchor $\boldsymbol{z}_t^n$ and the augmentation $\boldsymbol{z}_t^m$, by passing it through the models $\mathcal{M}_t$ and $\mathcal{M}_t'$, respectively. These representations share the same semantics and form a positive pair, while negative pairs are obtained by comparing representations from different samples in the same batch. The most widely adopted training objective is the NT-Xent loss, which is formulated as follows:

$$\mathcal{L}_{\text{NT-Xent}} = -\log \frac{e^{\text{sim}(\boldsymbol{z}_t^n, \boldsymbol{z}_t^m)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}\left(\boldsymbol{z}_t^n, \boldsymbol{z}_t^j\right)/\tau}}, \quad (9)$$

where $\mathrm{sim}(\boldsymbol{z}_t^n, \boldsymbol{z}_t^m)$ is the cosine similarity $\frac{\boldsymbol{z}_t^{n\top}\boldsymbol{z}_t^m}{\|\boldsymbol{z}_t^n\|*\|\boldsymbol{z}_t^m\|}$, τ is a temperature hyper-parameter and n is the number of sequences within a batch.

Although the training objective tries to pull representations with similar semantics closer and push dissimilar ones away from each other, these representations may still not be sufficiently discriminative and not be very robust to temporal noise. To demonstrate this, let us first denote angular $\theta_{n,m}$ as follows:

$$\theta_{n,m} = \arccos\left(\frac{\boldsymbol{z}_t^{n\top}\boldsymbol{z}_t^m}{\|\boldsymbol{z}_t^n\| * \|\boldsymbol{z}_t^m\|}\right) \tag{10}$$

The angular margin for $\boldsymbol{z}_t^n$ in NT-Xent is $\theta_{n,m} = \theta_{n,j}$, as show in Fig. 5. Due to the lack of a decision margin, a tiny time perturbation around the angular margin may lead to an incorrect decision. To overcome this problem, we propose a new training objective for temporal representation learning by adding an additive angular margin σ between positive pairs $\boldsymbol{z}_t^n$ and $\boldsymbol{z}_t^m$. This means that we want to keep some interval between the positive samples and do not force them exactly the same. We named it Time Angular Margin contrastive loss (TAM Loss), which can be formulated as follows:

$$\mathcal{L}_{\text{TAM}} = -\log \frac{e^{\cos(\theta_{n,m}+\sigma)/\tau}}{e^{\cos(\theta_{n,m}+\sigma)/\tau} + \sum_{j\neq n} e^{\cos(\theta_{n,j})/\tau}} \tag{11}$$

The TAM loss introduces a angular margin for $\boldsymbol{z}_t^n$ that is defined as $\theta_{n,m} + \sigma = \theta_{n,j}$, as shown in Fig. 5. Compared to the NT-Xent loss, the TAM loss further encourages $\boldsymbol{z}_t^n$ to move towards the region where $\theta_{n,m}$ is smaller and $\theta_{n,j}$ is larger. It increases the similarity of temporal representations with similar semantics and enlarges the discrepancy between different semantic representations. This enhances the alignment and uniformity properties, which are the two key measurements of representation quality related to contrastive learning [32]. Moreover, the angular margin provides an extra margin σ to $\theta_{n,m} = \theta_{n,j}$, which is often utilized during inference, making the loss more tolerant to temporal noise and better at capturing the underlying semantic intentions of the user. Overall, these properties make the TAM loss a more effective training objective than traditional alternatives for extracting users' temporal intentions.

4.2.3 *Social Aware Block*

To capture the intrinsic spatial-temporal movement patterns of different users more effectively, we propose the Social Aware Block, which generates a unique representation for each user. Specifically, we utilize Graph Attention Networks (GATs) to aggregate the neighbor features of each user and employ an adaptive adjacency matrix to aggregate higher-order neighbor features.

Given the social network $\mathcal{G} = \{\mathcal{U}, \mathcal{E}\}$, where $\mathcal{U}$ and $\mathcal{E}$ denote the user and link sets respectively, the i-th user is denoted as u_i. We denote the matrix $E_u \in \mathbb{R}^{|\mathcal{U}|\times D_u}$ as the lookup table of user embedding. Then we leverage GAT to aggregate neighbors' representations and update its embedding for each user, the new embedding is computed as:

$$\boldsymbol{h}_i^{(l)} = \sigma\left(\sum\nolimits_{j\in\mathcal{N}_i} \alpha_{ij}\mathbf{W}\boldsymbol{h}_j^{(l-1)}\right), \tag{12}$$

where $\boldsymbol{h}_i^{(l-1)} \in \mathbb{R}^{D_u}$ denotes the embedding of u_i in the $(l-1)$-th layer. $\mathcal{N}_i$ is the neighbour set of user i. The initial embedding of user u_i, denoted by $\boldsymbol{h}_i^{(0)}$, is simply the i-th row of E_u. σ is the sigmoid activation function. To compute the attention weight $\boldsymbol{a}_{ij}$ between users u_i and u_j, we use the following formula:

$$\alpha_{ij} = \frac{\exp\left(\varphi\left(\boldsymbol{a}^T\left[\boldsymbol{h}_i \| \boldsymbol{h}_j\right]\right)\right)}{\sum_{c\in\mathcal{N}_i}\exp\left(\varphi\left(\boldsymbol{a}^T\left[\boldsymbol{h}_i \| \boldsymbol{h}_c\right]\right)\right)} \tag{13}$$

where φ is the LeakyReLu activation function. Finally, we concat the final user embedding $\boldsymbol{h}_u$ with $\boldsymbol{z}_t$ to update the user's temporal representation $\boldsymbol{z}_t$.

4.3 ST Cross-View Contrastive Module

The temporal intention and the spatial topic are strongly correlated with users and highly susceptible to each other's impacts. This gives rise to a wealth of high-quality self-supervised signals. For example, most users engage in activities such as eating three meals, working, and exercising, but their schedules vary between weekdays and weekends and have different spatial topics. Different professions exhibit diverse spatial topics during the week due to their unique requirements. For instance, doctors, service workers, and students have distinct temporal intentions owing to their professional requirements.

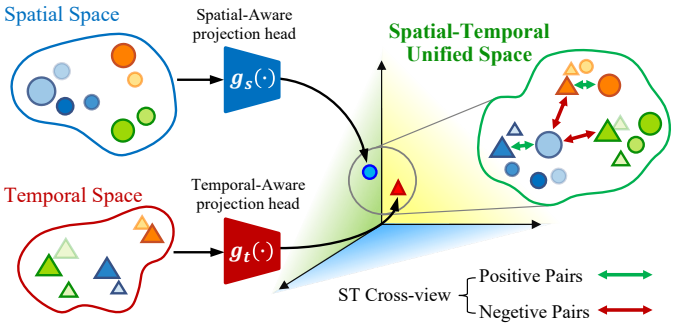

Fig. 6. The architecture of Spatial-Temporal Cross-view Module. The circle represents the spatial representation and the triangle represents the temporal representation. The similar color represents the representations from the same check-in sequence that the deep one is the anchor and the other is its augmentation.

To model check-in sequences based on these self-supervised cross-view signals, we propose a spatial-temporal cross-view contrastive learning framework. This approach enables the encoder to focus on learning optimal representations in both temporal and spatial views in the early stage and fuse at the semantic level. As learning progresses, we align the temporal and spatial representations by unifying them into a shared semantic space based on spatial-temporal parallel pairs. By tightly fusing the most relevant semantic information from the temporal and spatial views, we fully utilize the diverse check-in sequence data to uncover the spatial-temporal patterns of users.

The Spatial-Temporal Cross-View contrastive module is designed to learn unified check-in sequence representations prior to fusion. It learns a similarity function $s = g_s\left(\boldsymbol{z}_s\right)^\top g_t\left(\boldsymbol{z}_t\right)$, which assigns higher similarity scores to

parallel spatial-temporal pairs. Here, g_s and g_t are spatial and temporal projection heads that map the representation of sequences to a unified semantic space. For each spatial and temporal pair, we calculate the softmax-normalized spatial-to-temporal similarity score as follows:

$$\boldsymbol{p}_m^{\mathrm{s2t}}(\boldsymbol{z}_s) = \frac{\exp\left(s\left(\boldsymbol{z}_s, \boldsymbol{z}_t^m\right)/\tau\right)}{\sum_n \exp\left(s\left(\boldsymbol{z}_s, \boldsymbol{z}_t^n\right)/\tau\right)}, \tag{14}$$

and the temporal-to-spatial similarity as:

$$\boldsymbol{p}_m^{t2s}(\boldsymbol{z}_t) = \frac{\exp\left(s\left(\boldsymbol{z}_t, \boldsymbol{z}_s^m\right)/\tau\right)}{\sum_n \exp\left(s\left(\boldsymbol{z}_t, \boldsymbol{z}_s^n\right)/\tau\right)}, \tag{15}$$

where τ is a learnable temperature parameter. Let $\boldsymbol{y}^{\mathrm{s2t}}(\boldsymbol{z}_s)$ and $\boldsymbol{y}^{t2s}(\boldsymbol{z}_t)$ denote the ground-truth one-hot similarity, where negative pairs have a probability of 0 and the positive pair has a probability of 1. The spatial-temporal contrastive loss is defined as the cross-entropy H between $\boldsymbol{p}$ and $\boldsymbol{y}$:

$$\begin{aligned}\mathcal{L}_{ST} = \frac{1}{2}\mathbb{E}_{(\boldsymbol{z}_s,\boldsymbol{z}_t)\sim D}[\mathrm{H}\left(\boldsymbol{y}^{s2t}(\boldsymbol{z}_s), \boldsymbol{p}^{s2t}(\boldsymbol{z}_s)\right) \\ + \mathrm{H}\left(\boldsymbol{y}^{t2s}(\boldsymbol{z}_t), \boldsymbol{p}^{t2s}(\boldsymbol{z}_t)\right)]\end{aligned} \tag{16}$$

Ultimately, the pre-training loss is represented as:

$$\mathcal{L}_{Pre} = \mathcal{L}_{Spatial} + \mathcal{L}_{TAM} + \mathcal{L}_{ST}. \tag{17}$$

4.4 Fine-tuning for Downstream Applications

We employ the training set to pre-train the STCCR. We combine the spatial representation with the temporal representation as the global human behavior representation during the pre-training stage. Then we use a projection head to fine-tune among next location prediction (**LP**), next time prediction (**TP**), and trajectory user link (**TUL**) tasks, respectively. Firstly, we consider LP and TUL downstream tasks as multiclassification problems, as formulated below. Given a check-in sequence $\mathcal{T}^u$ from a specific user $u \in |\mathcal{U}|$, we feed it to the pre-trained encoder to obtain the check-in sequence representation $G(\mathcal{T}^u)$. Then we use a projection head $f_{\boldsymbol{\theta}}$ to predict the classification y such as the next location where the user will soon arrive or the user who generated this check-in sequence, *i.e.*, $f_{\boldsymbol{\theta}}\left(G(\mathcal{T}^u), \boldsymbol{\theta}\right) \mapsto y$. We maximize the conditional log-likelihood $\log f_{\boldsymbol{\theta}}(y \mid G(\mathcal{T}^u))$ for a given N observations $\{(G(\mathcal{T}^u), y)\}_{i=1}^N$ as follows:

$$\begin{aligned}\mathcal{L}_{MLE}(\boldsymbol{\theta}) &= \sum_{i=1}^{N} \log f_{\boldsymbol{\theta}}\left(y \mid G(\mathcal{T}^u)\right), \\ f_{\boldsymbol{\theta}}\left(y \mid G(\mathcal{T}^u)\right) &= \operatorname{softmax}\left(\boldsymbol{W}G(\mathcal{T}^u) + \boldsymbol{b}\right).\end{aligned} \tag{18}$$

Secondly, for downstream tasks of time prediction, we follow the method of IFLTPP [4] using an intensity-free method to model the interaction time as a mixture distribution. We first gain the mixture weights w, means μ and standard deviations s from the check-in sequence representation $G(\mathcal{T}^u)$. Then we build a mixed distribution function and sample to get the prediction time τ as follows:

$$p\left(\tau \mid w, \mu, s\right) = \sum_{k=1}^{K} \frac{1}{\tau s_k \sqrt{2\pi}} \exp -\frac{(\log \tau - \mu_k)^2}{2{s_k}^2}, \tag{19}$$

where k represents the number of independent Gaussian distributions in the mixed distribution. Then, we can sample from the mixture model in the parsing solution.

$$\tau = \sum_{k=1}^{K} w_k \exp\left(a\mu_k + b + \frac{a^2 s_k}{2}\right), \tag{20}$$

where a denotes the mean of the whole set and b denotes the standard deviation of the whole set.

5 Experiments

To evaluate the performance of our proposed model, we carried out extensive experiments on three real-world check-in sequence datasets, targeting three different types of downstream: Location Prediction (LP), Trajectory User Link (TUL), and Time Prediction (TP). *The code has been released at: https://github.com/LetianGong/STCCR.*

5.1 Datasets

In our experiments, we use three real-world datasets derived from raw WeePlace [3], Gowalla [4] of New York City (NYC), and Tokyo (TKY) check-in data. Our model undergoes a filtering process that selects high-quality trajectory sequences for training. To ensure data consistency, we set a maximum historical time limit of 120 days and filter out users with fewer than 10 records and places visited fewer than 10 times. Table 1 provides statistical information for each proceed dataset. We split the datasets into training, validation, and test sets at a 6:2:2 ratio. The three datasets exhibit distinct spatial-temporal correlations. For example, on workdays, a user may visit a breakfast restaurant at a certain time, while on rest days, the user's eating schedule may shift, making their behavior more time-sensitive. Due to the large number of POIs, sparse data sets, irregular time intervals, and varying user intentions, it is challenging to forecast and extract geographic and temporal information. Through our experiments, we demonstrate the superiority of our STCCR model, which we evaluate on all three datasets.

TABLE 1
Statistics of datasets

	Gowalla-NYC	Gowalla-TKY	WeePlace
#users	2,333	3,583	1,028
#locations	7,690	46,278	9,295
#check-ins	74,022	128,916	104,762

5.2 Baselines

Location Prediction Methods We cover one classic check-in prediction models and three state-of-the-art LP models to demonstrate the superiority of our model.

- **DeepMove** [1] is a classical check-in sequence position prediction model to capture the periodicity of trajectory motion.

3. http://www.yongliu.org/datasets
4. https://snap.stanford.edu/data/loc-Gowalla.html

- **PLSPL** [16] learns the long-term preference of the user by attention and the short-term preference of two LSTMs.
- **LightMove** [17] leverages neural ordinary differential equations to enhance resilience against sparse or incorrect input.
- **HMT-LSTM** [18] addresses data sparsity by learning user region matrices with varying lower sparsities.

Trajectory User Link Methods We select two end-to-end and two pre-trained TUL task models for comparison.

- **TULER** [6] is the first study to propose the trajectory user link and simulate it using RNN.
- **TULVAE** [7] follows the work of TULER and pre-trains the encoder by adding VAE on a prior basis.
- **MoveSim** [29] captures the temporal changes in human motion employing a generative adversarial network for trajectory pre-training.
- **S2TUL** [21] combines homogeneous graphs and sequential neural networks to a greedy method to relink trajectories.

Time Prediction Methods We selected one classic and two SOTA models for comparison.

- **IFLTPP** [4] shows how to overcome the limitations of intensity-based approaches by directly modeling the conditional distribution of inter-event times. It draws on the literature on normalizing flows to design models that are flexible and efficient
- **THP** [25] extends the transformers to include time and mark influences between events to calculate. the conditional intensity function for the arrival of future events in the sequence
- **NSTPP** [26] acquires the ability to retrieve and prioritize a relevant set of continuous-time event sequences based on a given anchor sequence.

Sequence Representation Methods We select four representative baselines for the representation of contrast sequences using contrastive learning. We apply them to learn the representation of the check-in sequence and serve three downstream tasks.

- **ReMVC** [9] designs an intra-inter view contrastive learning module to learn distinct region embeddings and transfer knowledge across multiple views.
- **VaSCL** [34] is suggested to add hard negative samples into the NLP field. It uses smaller batches to get better performance.
- **SML** [10] used self-supervised representation to harmonize noisy and nonuniform length trajectories.
- **CACSR** [11] utilizes adversarial perturbations in contrastive learning for data augmentation to enhance the capability of sequence representation.

5.3 Settings

We standardize all data using the Log Mean-Std approach and feed the normalized data into the model, which is optimized using reverse mode automatic differentiation and Adam [38]. To train the model, historical data of locations and users are first embedded, followed by location features fed into the spatial topic module and user IDs fed into the user embedding layer. We found that the time information worked better without the embedding layer, so we did not embed it. The representation vectors output from the spatial topic module and the temporal intention module are first mapped in the spatial-temporal joint space by the projection head. After that, we do contrastive learning between the outcomes of the two modules. Three separate downstream tasks are subsequently completed by combining the two representations from those two modules.

To assess the projected values, we retransform them back to the actual values and compare them to the ground truth. The evaluation measures of the location prediction and trajectory user link tasks include Acc@k and mean reciprocal rank (MRR). The evaluation measures of the time prediction task include mean absolute error (MAE), the root mean square error (RMSE), and mean absolute percentage error (MAPE). The STCCR model was constructed using the PyTorch[5]. The loss function is a cross-entropy loss for the LP and TUL tasks and an MAE loss for the TP task. The performance on the validation sets determines the hyperparameters and the best models. All experiments are performed five times, and the means and standard deviations are calculated. To make a fair comparison, for all methods, the embedding dimension d is 256, while the hidden state h has a size of 256. The learning rate is 0.001. Other detailed settings of the STCCR model for the three datasets are described in 1. The is pre-trained for 100 epochs on the training sets with the early-stopping mechanism of 10 patience. All trials have been conducted on Intel Xeon E5-2620 CPUs and NVIDIA RTX A5000 GPUs.

5.4 Comparison and Analysis of Results on the Down-Stream Tasks

Table 2 and Table 3 shows the comparison results of our model with other baseline models on three downstream tasks. The best is shown in **bold**, and the second-best is shown as underlined.

Our representation learning pre-trained models can meet or even exceed the best-performing end-to-end models. Besides, the performance of STCCR far outperforms other sequential representation learning models on three tasks. In the LP task, STCCR improves the sota representation models by 3.9%, 2.5% 7.1% in terms of Acc@5, Acc@20, and MRR average on the three datasets, respectively. In the TUL task, STCCR improves by 4.3%, 4.7%, 5.3% in terms of Acc@5, Acc@20, and MRR average on the three datasets, respectively. In the TP task, STCCR improves by 7.9%, 4.4%, 1.3% in terms of MAE, RMSE, and NLL average on the three datasets, respectively.

These results demonstrate that our model performs very well on these downstream tasks. The strength of our model benefits from our spatial-temporal cross-view contrastive framework, which is able to encourage the spatial topic module and temporal intention module to mine the spatial-temporal patterns from mobility behavior data. The spatial topic module uses contrastive clustering can effectively capture the spatial semantics of human mobility and enhance

5. https://pytorch.org

TABLE 2
Next location prediction (LP) and trajectory user link (TUL) performance comparison between different approaches.

Datasets		Gowalla-NYC			Gowalla-TKY			WeePlace		
Metric		Acc@5 (%)↑	Acc@20 (%)↑	MRR (%)↑	Acc@5 (%)↑	Acc@20 (%)↑	MRR (%)↑	Acc@5 (%)↑	Acc@20 (%)↑	MRR (%)↑
Task	Method									
LP	DeepMove	34.04±0.16	51.70±0.29	28.88±0.41	20.73±0.35	29.01±0.12	15.24±0.37	34.39±0.18	47.37±0.28	25.25±0.23
	LightMove	36.75±0.31	53.78±0.18	30.72±0.17	23.11±0.30	33.45±0.23	17.73±0.34	37.91±0.23	54.49±0.21	28.76±0.22
	PLSPL	36.93±0.31	54.42±0.38	30.98±0.43	23.17±0.37	33.47±0.39	17.49±0.21	38.82±0.44	55.19±0.23	29.47±0.21
	HMT-LSTM	36.57±0.21	53.62±0.38	30.41±0.43	22.48±0.33	32.99±0.19	17.07±0.42	37.84±0.19	54.14±0.25	28.41±0.19
	VaSCL	29.04±0.18	40.70±0.18	22.88±0.46	18.73±0.23	27.01±0.14	14.24±0.23	30.39±0.29	43.37±0.30	23.25±0.41
	ReMVC	37.61±0.19	54.35±0.22	31.07±0.41	23.15±0.22	33.61±0.29	**18.09±0.34**	39.03±0.43	55.21±0.16	28.94±0.29
	SML	35.75±0.15	53.88±0.22	30.15±0.15	21.56±0.35	31.63±0.32	16.82±0.31	37.87±0.26	53.19±0.28	27.75±0.23
	CACSR	33.35±0.44	51.05±0.15	29.24±0.17	23.12±0.37	33.57±0.39	17.89±0.21	39.02±0.41	55.20±0.16	29.64±0.29
	STCCR	**38.38±0.15**	**54.78±0.24**	**31.24±0.25**	**23.62±0.23**	**34.43±0.34**	17.93±0.23	**39.16±0.16**	**55.36±0.32**	**30.07±0.11**
TUL	TULER	59.71±0.23	69.15±0.20	54.17±0.18	70.87±0.28	79.25±0.26	66.74±0.42	73.61±0.29	80.98±0.46	70.88±0.35
	TULVAE	60.94±0.19	69.56±0.19	56.01±0.18	73.19±0.31	79.46±0.31	67.52±0.29	75.78±0.27	86.43±0.18	72.92±0.41
	MoveSim	64.21±0.32	72.12±0.18	59.56±0.12	72.12±0.41	79.78±0.16	67.23±0.22	82.35±0.18	87.32±0.15	74.27±0.21
	S2TUL	65.49±0.25	73.21±0.32	60.19±0.19	75.04±0.21	81.68±0.35	70.64±0.34	83.46±0.19	89.33±0.23	77.82±0.29
	VaSCL	55.43±0.15	65.54±0.19	48.43±0.37	53.94±0.36	57.23±0.29	53.15±0.43	62.15±0.29	70.83±0.21	54.45±0.17
	ReMVC	65.37±0.37	73.23±0.22	60.34±0.33	74.23±0.46	81.67±0.28	69.34±0.45	83.52±0.16	88.72±0.18	76.92±0.36
	SML	64.57±0.21	72.62±0.38	59.41±0.43	72.48±0.33	79.99±0.19	67.07±0.42	82.84±0.19	87.14±0.25	74.41±0.19
	CACSR	63.75±0.15	72.88±0.22	59.15±0.15	73.15±0.22	80.61±0.29	68.29±0.34	82.03±0.43	88.29±0.16	76.94±0.29
	STCCR	**66.17±0.11**	**74.04±0.13**	**61.79±0.08**	**75.48±0.31**	**82.08±0.25**	**70.97±0.29**	**84.58±0.08**	**90.14±0.05**	**78.03±0.36**

TABLE 3
Next time prediction (TP) performance comparison between different approaches.

Datasets		Gowalla-NYC			Gowalla-TKY			WeePlace		
Metric		MAE↓	RMSE↓	NLL(e-2)↓	MAE↓	RMSE↓	NLL(e-2)↓	MAE↓	RMSE↓	NLL(e-2)↓
Task	Method									
TP	IFLTPP	25.34 ±0.19	36.93 ±0.26	66.91 ±0.27	22.98 ±0.36	33.21 ±0.19	60.13 ±0.18	29.21 ±0.28	36.92 ±0.18	**74.92 ±0.32**
	THP	24.77 ±0.23	35.07 ±0.20	66.94 ±0.16	21.97 ±0.31	32.91 ±0.3	**59.09 ±0.32**	27.66 ±0.36	35.30 ±0.27	74.98 ±0.21
	NSTPP	24.32 ±0.18	35.02 ±0.21	66.84 ±0.26	21.37 ±0.26	32.48 ±0.17	61.87 ±0.25	28.24 ±0.29	36.09 ±0.18	78.36 ±0.25
	VaCSL	29.09 ±0.16	37.99 ±0.31	79.38 ±0.15	23.70 ±0.21	35.92 ±0.27	65.23 ±0.33	31.86 ±0.22	38.48 ±0.27	85.63 ±0.15
	ReMVC	25.86 ±0.21	37.14 ±0.23	71.11 ±0.22	22.48 ±0.23	33.60 ±0.19	62.55 ±0.33	28.51 ±0.33	36.17 ±0.27	80.07 ±0.15
	SML	26.28 ±0.17	36.03 ±0.25	71.52 ±0.26	22.78 ±0.31	33.42 ±0.22	62.26 ±0.22	29.47 ±0.31	35.88 ±0.18	77.75 ±0.15
	CACSR	27.05 ±0.24	37.69 ±0.21	73.73 ±0.35	22.67 ±0.32	34.07 ±0.28	64.13 ±0.37	30.71 ±0.15	38.34 ±0.24	82.84 ±0.32
	STCCR	**24.25 ±0.22**	**34.68 ±0.27**	**64.86 ±0.28**	**20.99 ±0.25**	**31.87 ±0.35**	59.16 ±0.24	**27.28 ±0.31**	**35.28 ±0.31**	75.41 ±0.24

the generalization of the model by clustering shared mobility patterns. The temporal intention module uses angular contrastive manner which can force the same sample to be not exactly the same, but within a small region. This is a good way to mitigate the effects of temporal uncertainty and noise on the user's temporal intention. The self-supervised signals from unified space can effectively mitigate the sparsity of check-in data and spur the encoder to learn accurate general representations for check-in sequences. In general, the superiority and versatility of our method show that the spatial-temporal cross-view contrastive framework proposed in this work is well suited for modeling check-in sequences.

ReMVC designs an intra-view and an inter-view contrastive learning module to transfer knowledge across multiple views, but it does not effectively extract the semantic information before the information interaction. SML used a prior fusion manner to learn representation for every check-in sequence but ignoring the diversity of user movement and the temporal noise can't capture the potential patterns of activity. The performance of VaSCL on the check-in dataset conclusively proves that data augmentation techniques in NLP are not directly transferable to human movement trajectory data. Although difficult negative samples are introduced in VaSCL. The contrastive method of similar semantic samples is considered in the contrastive process, but it is still unable to mine the spatial topic and temporal intention of users based on the characteristics of check-in data. Our previous work, CACSR, performs data augmentation by combating perturbations, which can largely combat the noise in the original data, but ignore the shared topic of users moving in spatial.

5.5 Ablation experiments

To further evaluate the effects of different components in STCCR, we conduct ablation experiments and analyze experimental results on the Gowalla-NYC and WeePlace datasets. We compare these four variants with the STCCR model for the three downstream tasks. Fig. 7 shows the results.

- **Basic:** We use contrastive manners like SimCSE and InfoNCE loss to replace the origin manner in Spatial Topic Module and Temporal Intention Module. Simply combine two representation vectors to complete the downstream task.
- **w/o STCV:** We remove the ST Cross-View Module, instead we directly combine the representation from the other two modules. The rest of the settings are the same as STCCR. We use this setting to evaluate the function of the ST Cross-View Module.

- **w/o STM:** We use contrastive manner like SimCSE and InfoNCE loss to replace the origin manner in only Spatial Topic Module. The rest of the settings are the same as STCCR. The rest of the settings are the same as STCCR. We use this setting to evaluate the function of the Spatial Topic Module.
- **w/o TIM:** We use a contrastive manner like SimCSE and InfoNCE loss to replace the origin manner in only the Temporal Intention Module. The rest of the settings are the same as STCCR. The rest of the settings are the same as STCCR. We use this setting to evaluate the function of the Temporal Intention Module.

Among them, the spatial topic module has the greatest improvement for the next location prediction task. This is thanks to the spatial topic module, which combines the advantages of contrastive learning and clustering methods to effectively capture the shared spatial patterns among different users. This approach avoids learning a representation for each sample and can be valid for mitigating inadequate generalizability. This can motivate the model to more effectively mine higher-level semantic information about users' spatial movements from clustered self-supervised signals.

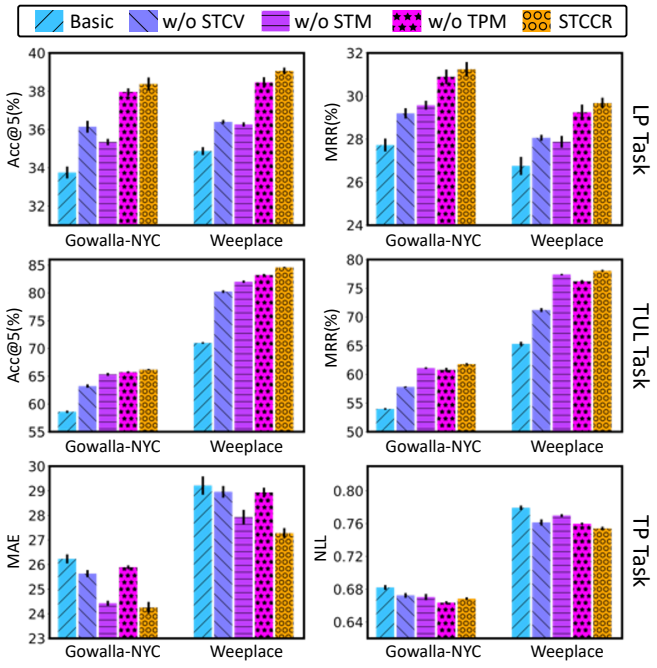

Fig. 7. Component analysis of STCCR

The temporal intentions module has the greatest improvement for the time prediction task. This is because the temporal intention module combines the advantages of contrastive learning and decision boundaries. By introducing decision bounds, the noise in the time series can be well filtered. This alleviates the difficulties in mining users' temporal intentions on check-in datasets with large temporal uncertainty.

The ST cross-view contrastive module has significant improvements for the trajectory user task and the next location prediction task. The ST cross-view contrastive module can exploit the natural self-supervised signals between temporal and spatial. It effectively mines the historical behavioral habits and neighboring travel intentions of users. It can be seen that the introduction of this module has a huge effect on the overall boost of the model.

This result proves the usefulness of three modules in using contrastive clustering, angular margin, and ST cross-view contrastive manner to learn the representation of check-in sequences. Finally, these modules are combined in our final model to produce the best results.

TABLE 4
Settings of the STCCR model in three datasets.

Parameter	Range
Cluster centers	16, 64, 256, 512, 2048
Queue length	0, 128,512,2048,8096
Angular margin	0, 0.03, 0.09, 0.18,0.3
Projection head size	32, 128, 512, 2048

Underline denotes the optimal value.

5.6 Effects of Parameters

In this section, we evaluate the effects of hyper-parameters in three modules: 1) the cluster number and queue length in the STM. 2) The theta margin during angular contrast in the TIM. 3) The loss weight of the ST cross-view module during the pre-training. The experiments are conducted on all the datasets, and while evaluating one of the hyper-parameters, we lock the other ones to optimum. We set the best number of cluster centers for each dataset in Table 4.

5.6.1 Effects of Cluster Center Number

Fig. 8a shows the experimental results on the hyper-parameter tuning of cluster number. From these figures, we observe that the performance first improves as we increase the number of cluster centers, then deteriorates as it exceeds the optimum point. A small number of centers means thousands of users only have a limited number of spatial topics, which fails to capture the diversity of human activities. A large number of centers will cluster too many activity themes. This will make the method fail to extract shared activity themes. This degenerates into a w/o STM among the ablation experiments, i.e., learning a sequence representation for each sample. Reduces the generalization performance of the STM module. A moderate number of cluster centers can better capture semantic information about user movement in spatial.

5.6.2 Effects of Queue Length

Fig. 8b shows the experimental results on the hyperparameter tuning of the queue length. This figure demonstrates that performance improves steadily as we increase the length of the queue in the STM. A longer queue can contain more samples, nourishing the cluster centers in each epoch and helping downstream tasks gain better results. Yet, we observe that the degree of performance improvement is limited when the queue length is longer than eight thousand, and the longer queue will lead to a higher computational expense.

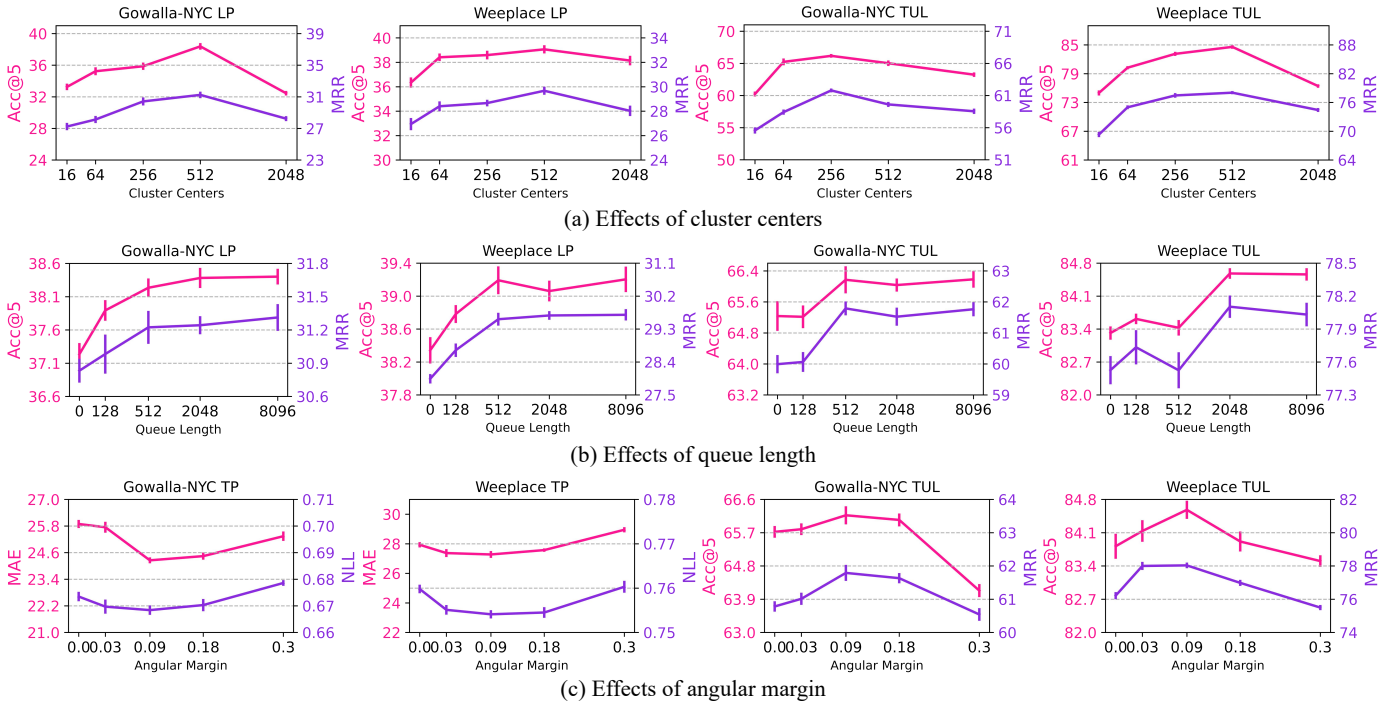

Fig. 8. Effects of hyper-parameters validated on Gowalla-NYC and Weeplace dataset.

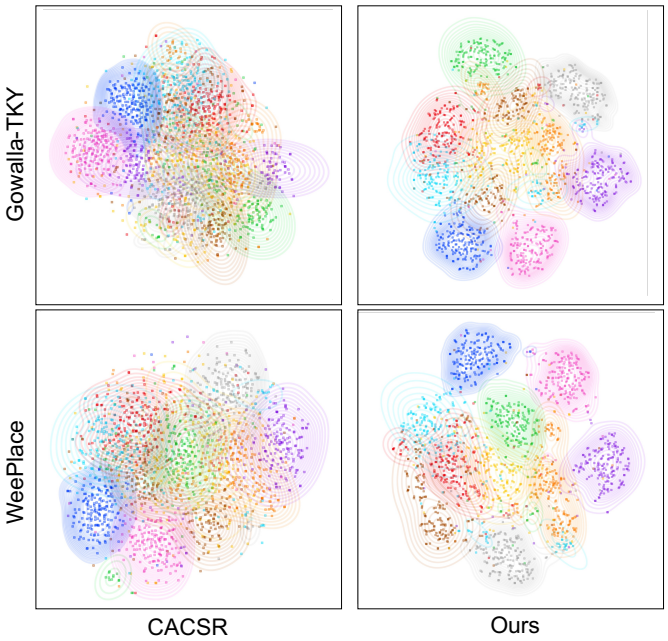

Fig. 9. Case visualization comparison between CACSR and Ours.

5.6.3 *Effects of Theta Margin*

Fig. 8c shows the experimental results on the hyperparameter tuning influence of the theta margin. It can be seen that theta margin has a great impact on the TP task. A too-small margin for noise does not play a filtering role; a large margin will lead to meaningless comparison training. Therefore, a suitable theta margin can effectively filter the noise while capturing the macroscopic temporal intention. In conclusion, we set the theta margin to 0.09 for all datasets.

5.7 Case Visualization of Sequence Representations

We posit that our STCCR encourages mobility representations from users with different activity topics and intentions to be distinguished. It emphasizes the advantages of the proposed model. Towards that, we compare our pre-trained model with CACSR using t-SNE [39] to plot the latent space of check-in sequences. Specifically, we select 10 locations with the most truth labels in different categories and their corresponding check-in sequences from each dataset. The learned latent representations of each check-in are projected to the 2D space. Fig. 9 plots the learned representations of check-in sequences on two datasets, where we can observe an apparent clustering effect of sequences. This means that our model can effectively capture the high-level semantic purpose of different users, which is essential for downstream tasks such as next location prediction and user trajectory user link. This result also implies that a better representation with good discriminability is critical for human mobility learning.

6 Conclusions

We present STCCR, a Contrastive Spatial-Temporal Cross-view Representation method, to enhance the understanding of human movement patterns in check-in sequences. To address the limitations of existing representation learning approaches, STCCR leverages a cross-view contrastive framework that considers both spatial topic and temporal intention views. This innovative approach enables the exploration of macroscopic semantics and associations from different viewpoints, leading to improved representation learning for check-in sequences. Furthermore, we introduce an angular momentum contrastive method that effectively captures the inherent uncertainty and temporal intention

within the time series data. Through contrastive clustering of spatial topics, we uncover shared spatial activity patterns, enhancing the understanding of human mobility. Experimental evaluations on real-world check-in datasets demonstrate the superiority and versatility of our proposed STCCR model.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No. 62272033) and the China Postdoctoral Science Foundation (Grant No. 2023T160044).

REFERENCES

[1] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin, "Deepmove: Predicting human mobility with attentional recurrent networks," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 1459–1468.

[2] K. Sun, T. Qian, T. Chen, Y. Liang, Q. V. H. Nguyen, and H. Yin, "Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 214–221.

[3] D. Yao, C. Zhang, J. Huang, and J. Bi, "Serm: A recurrent model for next location prediction in semantic trajectories," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 2411–2414.

[4] O. Shchur, M. Biloš, and S. Günnemann, "Intensity-free learning of temporal point processes," *International Conference on Learning Representations (ICLR)*, 2020.

[5] C. Miao, J. Wang, H. Yu, W. Zhang, and Y. Qi, "Trajectory-user linking with attentive recurrent network," in *Proceedings of the 19th international conference on autonomous agents and multiagent systems*, 2020, pp. 878–886.

[6] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, and F. Zhang, "Identifying human mobility via trajectory embeddings." in *IJCAI*, vol. 17, 2017, pp. 1689–1695.

[7] F. Zhou, Q. Gao, G. Trajcevski, K. Zhang, T. Zhong, and F. Zhang, "Trajectory-user linking via variational autoencoder." in *IJCAI*, 2018, pp. 3212–3218.

[8] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020.

[9] L. Zhang, C. Long, and G. Cong, "Region embedding with intra and inter-view contrastive learning," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[10] F. Zhou, Y. Dai, Q. Gao, P. Wang, and T. Zhong, "Self-supervised human mobility learning for next location prediction and trajectory classification," *Knowledge-Based Systems*, vol. 228, p. 107214, 2021.

[11] L. Gong, Y. Lin, S. Guo, Y. Lin, T. Wang, E. Zheng, Z. Zhou, and H. Wan, "Contrastive pre-training with adversarial perturbations for check-in sequence representation learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 4, pp. 4276–4283, Jun. 2023. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/25546

[12] J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, and L. He, "Multi-level feature learning for contrastive multi-view clustering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 051–16 060.

[13] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," *Advances in neural information processing systems*, vol. 34, pp. 9694–9705, 2021.

[14] Y. Luo, Q. Liu, and Z. Liu, "Stan: Spatio-temporal attention network for next location recommendation," in *Proceedings of the Web Conference 2021*, 2021, pp. 2177–2185.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] Y. Wu, K. Li, G. Zhao, and Q. Xueming, "Personalized long-and short-term preference learning for next poi recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[17] J. Jeon, S. Kang, M. Jo, S. Cho, N. Park, S. Kim, and C. Song, "Lightmove: A lightweight next-poi recommendation fortaxicab rooftop advertising," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, ser. CIKM '21. Association for Computing Machinery, 2021, p. 3857–3866.

[18] N. Lim, B. Hooi, S.-K. Ng, Y. L. Goh, R. Weng, and R. Tan, "Hierarchical multi-task graph recurrent network for next poi recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, p. 1133–1143.

[19] X. Rao, L. Chen, Y. Liu, S. Shang, B. Yao, and P. Han, "Graph-flashback network for next location recommendation," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1463–1471.

[20] S. Yang, J. Liu, and K. Zhao, "Getnext: trajectory flow map enhanced transformer for next poi recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on research and development in information retrieval*, 2022, pp. 1144–1153.

[21] L. Deng, H. Sun, Y. Zhao, S. Liu, and K. Zheng, "S2tul: A semi-supervised framework for trajectory-user linking," in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, ser. WSDM '23. Association for Computing Machinery, 2023, p. 375–383. [Online]. Available: https://doi.org/10.1145/3539597.3570410

[22] F. Zhou, S. Chen, J. Wu, C. Cao, and S. Zhang, "Trajectory-user linking via graph neural network," in *ICC 2021 - IEEE International Conference on Communications*, 2021, pp. 1–6.

[23] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, "Recurrent marked temporal point processes: Embedding event history to vector," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. Association for Computing Machinery, 2016, p. 1555–1564.

[24] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz, "Self-attentive Hawkes process," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 11 183–11 193. [Online]. Available: https://proceedings.mlr.press/v119/zhang20q.html

[25] S. Zuo, H. Jiang, Z. Li, T. Zhao, and H. Zha, "Transformer hawkes process," in *Proceedings of the 37th International Conference on Machine Learning*, ser. ICML'20. JMLR.org, 2020.

[26] V. Gupta, S. Bedathur, and A. De, "Learning temporal point processes for efficient retrieval of continuous time event sequences," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, 2022, pp. 4005–4013.

[27] V. Gupta, A. De, S. Bhattacharya, and S. Bedathur, "Learning temporal point processes with intermittent observations," in *Proc. of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2021.

[28] Y. Yuan, J. Ding, C. Shao, D. Jin, and Y. Li, "Spatio-temporal diffusion point processes," *arXiv preprint arXiv:2305.12403*, 2023.

[29] J. Feng, Z. Yang, F. Xu, H. Yu, M. Wang, and Y. Li, "Learning to simulate human mobility," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 3426–3433.

[30] H. Wan, Y. Lin, S. Guo, and Y. Lin, "Pre-training time-aware location embeddings from spatial-temporal trajectories," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[31] Y. Lin, H. Wan, S. Guo, and Y. Lin, "Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 5, 2021, pp. 4241–4248.

[32] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," in *EMNLP (1)*, 2021.

[33] Y. Yan, R. Li, S. Wang, F. Zhang, W. Wu, and W. Xu, "ConSERT: A contrastive framework for self-supervised sentence representation transfer," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 5065–5075. [Online]. Available: https://aclanthology.org/2021.acl-long.393

[34] D. Zhang, W. Xiao, H. Zhu, X. Ma, and A. Arnold, "Virtual augmentation supported contrastive learning of sentence representa-

tions," in *Findings of the Association for Computational Linguistics: ACL 2022*, 2022, pp. 864–876.
[35] S. Lee, D. B. Lee, and S. J. Hwang, "Contrastive learning with adversarial perturbations for conditional text generation," in *International Conference on Learning Representations*, 2021.
[36] Z. Wang, Y. Zhu, H. Liu, and C. Wang, "Learning graph-based disentangled representations for next poi recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1154–1163.
[37] X. Wu, C. Gao, L. Zang, J. Han, Z. Wang, and S. Hu, "Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding," *arXiv preprint arXiv:2109.04380*, 2021.
[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[39] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

Letian Gong received the B.S. degree in computer science from Beijing Jiaotong University, Beijing, China, in 2021.

He is currently working toward the Ph.D. degree in the School of Computer and Information Technology, Beijing Jiaotong University. His interest falls in area of deep learning and data mining, especially their applications in spatial-temporal data mining.

Huaiyu Wan received the Ph.D. degree in computer science and technology from Beijing Jiaotong University, Beijing, China, in 2012.

He is a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. His current research interests focus on spatial-temporal data mining, social network mining, and user behavior analysis.

Shengnan Guo received the Ph.D. degree in computer science from Beijing Jiaotong University, Beijing, China, in 2021.

She is a lecturer at the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests focus on spatial-temporal data mining and intelligent transportation systems.

Xiucheng Li received the Ph.D. student at the School of Computer Science and Engineering, Nanyang Technological University, Singapore.

At present, he is an assistant professor at the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen). He is focused on representation learning, generative models and probabilistic inference, spatial and time series data analysis.

Yan Lin received the B.S. degree in computer science from Beijing Jiaotong University, Beijing, China, in 2019. He is currently working toward the Ph.D. degree from the School of Computer and Information Technology, Beijing Jiaotong University. His research interests include spatial-temporal data mining and graph neural networks.

Erwen Zheng received the B.S. degree in mathematics from Beijing Jiaotong University, Beijing, China, in 2023.

He is currently working toward the M.S. degree in the School of Computer and Information technology, Beijing Jiaotong University. His interest falls in area of deep learning and data mining, especially their applications in spatial-temporal data mining.

Tianyi Wang received the B.S. degree in computer science from Beijing Jiaotong University, Beijing, China, in 2022.

He is currently working toward the M.S. degree in the School of Computer and Information technology, Beijing Jiaotong University. His interest falls in area of deep learning and data mining, especially their applications in spatial-temporal data mining.

Zeyu Zhou received the B.S. degree in mathematics from Beijing Jiaotong University, Beijing, China, in 2022.

He is currently working toward the M.S. degree in the School of Computer and Information technology, Beijing Jiaotong University. His interest falls in area of deep learning and data mining, especially their applications in spatial-temporal data mining.

Youfang Lin received the Ph.D. degree in signal and information processing from Beijing Jiaotong University, Beijing, China, in 2003.

He is a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. His main fields of expertise and current research interests include big data technology, intelligent systems, complex networks, and traffic data mining.