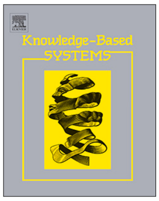
Knowledge-Based Systems 228 (2021) 107214

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



Self-supervised human mobility learning for next location prediction and trajectory classification

Fan Zhou [a], Yurou Dai [a], Qiang Gao [b], Pengyu Wang [a], Ting Zhong [a],*

[a] *University of Electronic Science and Technology of China, China*
[b] *Southwestern University of Finance and Economics, China*

ARTICLE INFO

Article history:
Received 9 February 2021
Received in revised form 2 June 2021
Accepted 9 June 2021
Available online 15 June 2021

Keywords:
Human mobility learning
Self-supervised learning
Location prediction
Contrastive learning
Trajectory classification

ABSTRACT

Massive digital mobility data are accumulated nowadays due to the proliferation of location-based service (LBS), which provides the opportunity of learning knowledge from human traces that can benefit a range of business and management applications, such as location recommendation, anomaly trajectory detection, crime discrimination, and epidemic tracing. However, human mobility data is usually sporadically updated since people may not frequently access mobile apps or publish the geo-tagged contents. Consequently, distilling meaningful supervised signals from sparse and noisy human mobility is the main challenge of existing models. This work presents a Self-supervised Mobility Learning (SML) framework to encode human mobility semantics and facilitate the downstream location-based tasks. SML is designed for modeling sparse and noisy human mobility trajectories, focusing on leveraging rich spatio-temporal contexts and augmented traces to improve the trajectory representations. It provides a principled way to characterize the inherent movement correlations while tackling the implicit feedback and weak supervision problems in existing model-based approaches. Besides, contrastive instance discrimination is first introduced for spatio-temporal data training by explicitly distinguishing the real user check-ins from the negative samples that tend to be wrongly predicted. Extensive experiments on two practical applications, i.e., location prediction and trajectory classification, demonstrate that our method can significantly improve the location-based services over the state-of-the-art baselines.

1. Introduction

The availability of massive digital traces of human whereabouts such as call detail records, GPS trajectories, and social media footprints have enabled numerous studies on learning human mobility patterns for a range of downstream tasks, such as location prediction [1], point-of-interest (POI) recommendation [2,3], route planning [4], tracking the COVID-19 pandemic [5,6], human trace identification [7], traffic forecasting [8], emergency management [9], etc. Human mobility prediction, which aims to predict where a user will arrive in the near future, is a fundamental task that can benefit many areas, such as controlling the spread of infectious diseases, urban planning, and crime identification [10].

Existing studies have found that personalized movement is highly predictable [1,11], even though the activities and mobility patterns of different individuals might be very diverse. The cornerstone of modeling human dynamics lies in exploring the spatio-temporal characteristics and underlying patterns of people's trajectories. Early efforts focus on discovering the underlying laws governing human motion (e.g., random walk and Lévy flight) [12,13] or modeling human mobility with data-driven methods such as Markov-based models [14,15]. Nevertheless, traditional approaches rely on presumed parameters or manually designed features that cannot easily be generalized to different scenarios and may lead to erroneous interpretations. Meanwhile, typical machine learning models may fail to capture spatial and temporal nonlinearity and are therefore inapplicable to handle large-scale, low-cost, noisy, and sparse geo-tagged trajectory data.

Due to the ability of capturing complex and non-linear spatio-temporal relationships, deep learning techniques become increasingly important in a wide range of mobility learning tasks. Most of the existing works resort to recurrent neural networks (RNNs) for encoding the sequential, spatio-temporal, and semantic knowledge associated with the trajectories. Earlier works in this line directly utilizes RNNs to model human mobility that benefits downstream tasks, e.g., next location prediction [16] and human trace discrimination [17]. Meanwhile, techniques for spatial

* Corresponding author.
E-mail addresses: fan.zhou@uestc.edu.cn (F. Zhou), yurou@std.uestc.edu.cn (Y. Dai), qianggao@swufe.edu.cn (Q. Gao), p.y.wang@std.uestc.edu.cn (P. Wang), zhongting@uestc.edu.cn (T. Zhong).

https://doi.org/10.1016/j.knosys.2021.107214
0950-7051/© 2021 Elsevier B.V. All rights reserved.

item embedding [18] and attention mechanisms for footprint weighting [19] have been employed to improve the trajectory representation learning. Previous human mobility learning methods generally extract spatio-temporal features, combined with the recursive hidden state of RNNs, to iteratively ascertain the transition patterns and successive location dependencies using parameterized matrices and the gate mechanism [20,21]. Some recent works have extended the RNN-based approach to account for various spatio-temporal aspects of human mobility and/or improve location prediction from different perspectives. For example, Gao et al. [22] replace RNN with convolutional networks, coupled with a generative probabilistic model, to learn individual behaviors and periodical patterns. It learns the attention on the historical trajectories, from which the latent representation of a recent trajectory is leveraged to match the most similar moving patterns in the past. More recently, Yang et al. [23] employs a flashback operation on hidden states of RNNs, and explicitly queries historical motion episode similar to recent trajectory for improving prediction performance.

Despite achieving effective performance to some extent, current deep human mobility learning models suffer from several major drawbacks. First, the fundamental limit in location-based social networks such as Foursquare and Yelp is the casual and sporadic locations published by users. Due to the sparse nature of APP use, there exist sampling biases in the trajectory data that only contains incomplete traces of individuals. Though some studies [16,20] have assimilated spatio-temporal factors into the gate mechanism in RNN towards alleviating the sparsity issue, their methods rely on simple spatio-temporal features (e.g., dynamic time intervals and geographical distances) and/or memory-style subsequence similarity matching [19,23]. However, neither spatio-temporal factors nor attention-based matching is sufficient for inferring real travel preference from extremely sparse user–location interactions, and may introduce uncontrolled variance in space and time for users' visiting intentions and moving patterns estimate. Besides, the most common strategy for training mobility learning models employs the next location as the single supervision, taking a historical trajectory as input. This learning paradigm follows the idea of training natural language processing (NLP) models but can easily lead to biased context encoding due to the implicit, noisy, and incomplete feedback in the trajectory data. Furthermore, the next immediate updated location is a weak signal and sometimes even irrelevant to her past trajectories. For example, a user may visit a restaurant but find the food does not meet her appetite, and her preferred cafeteria is nearby yet not in the training data. As a result, approaches that merely guided by the loss of the predicted next location may not capture the real intention and rich contextual features associated with the historical mobility that nonetheless are absent in the data sources.

To address the above issues, we borrow the idea of self-supervised learning (SSL) for modeling human mobility. SSL aims to design auxiliary objectives supervised by the labels distilled from the data itself, which recently achieved remarkable success in computer vision (CV) and NLP domains [24] but has not been studied in human dynamics learning. The main reason is that in CV tasks one can easily design auxiliary tasks such as image rotation, distortion, and cropping, which are not applicable in mobility data restricted by spatio-temporal factors. Though word masking and context predicting that are widely used in NLP can provide supplementary supervision in spatio-temporal item encoding [22], the effect is limited due to the intrinsic natures of trajectory data. For example, they fail to provide much additional information about user intentions beyond the published check-ins.

To this end, we take the first step towards self-supervised human mobility learning by incorporating the geographical and temporal knowledge into contrastive motion representation learning. Complementary to the typical predictive loss, e.g., minimizing the distance between predicted and the recorded next locations, our method encourages the trajectory representation to capture implicit mobility intentions through discriminating the true favorable location from a few negative locations. In addition, we present a trajectory augmentation method, which takes the spatial and temporal contexts as prior and generates synthesis trajectories subjecting to the spatio-temporal constraints. In this way, it provides multi-views of a trajectory and allows us to learn mobility representation maximizing the mutual information between the past observed trajectories and the future mobility, which captures more fluent and complete movement intentions of users. In summary, we make the following contributions:

- We introduce Self-supervised Mobility Learning (SML), a new framework for improving sparse and noisy trajectory representation in a self-supervised learning manner.
- We propose a spatio-temporal data augmentation method by leveraging the properties of observed trajectories to enhance the multi-view of trajectories, which can enrich the sparse and sporadic human check-in data with semantic-aware trajectory permutation.
- Extensive comparisons to state-of-the-art methods demonstrate that our method not only significantly improves the next location prediction performance but also can be extended to other LBS applications such as trajectory classification, which highlights the advantages of SML on learning human mobility.

In the rest of this paper, we formalize the problem and present the necessary background in Section 2. Subsequently, we provide the details of the proposed human mobility learning framework in Section 3. The results of the experimental evaluations quantifying the benefits of our approach are presented in Section 4. We review the relevant studies and position our work in that context in Section 5. Finally, this article is concluded in Section 6 with remarks on our future work.

2. Preliminaries

In this section, we first formally define two mobility learning problems investigated in this paper, and then provide the necessary background regarding mutual information and Contrastive Loss in self-supervised learning.

2.1. Problem definition

The SML framework is proposed to solve the human mobility problems on Location Prediction (LP) and Trajectory–User Linking (TUL). Our goal is to learn a model $\mathcal{M}$ to predict the next point of interest or classify the trajectory to the user who generated it. Table 1 summarizes the notations frequently used throughout this paper.

Let $c = <id, lo, la, ca>$ be a tuple representing a POI with identity id and the corresponding geographical context (e.g., longitude lo and latitude la), as well as its category ca (e.g., home, restaurant, or shopping mall). For a user $u \in \mathcal{U}$, we denote its M trajectories as $\mathcal{T} = \{T_1, T_2, \ldots, T_M\}$, where each trajectory T consists of a sequence of POIs, i.e., $T = \{c_1, c_2, \ldots, c_t\}$ where the subscript denotes visiting time t.

Definition 1 (*Location Prediction (LP)*)**.** Given her full historical trajectory $\mathcal{T}$, and the recently visited POI sequence T. The location prediction problem is to learn a model f_{LP} to predict u's next location c_{t+1}, i.e., $f_{\text{LP}}(\mathcal{T}, T) \mapsto c_{t+1}$.

Table 1
Frequently used notations.

Notation	Description
$\mathcal{U}/\mathbf{C}/\mathcal{T}$	A set of users/POIs/trajectories
N/M	The number of users/trajectories
$T_j \in \mathcal{T}$	A trajectory generated by user u_j
c_t	The check-in at time t
$\mathbf{h}_t$	The hidden state of RNN
α_i^t	The attention weight t
$\mathcal{L}$	The loss function
H	Information entropy
KL	KL-divergence
$\mathrm{MI}(\cdot;\cdot)$	Mutual information
f_*	The problem-specific model
$g()$	The embedding function
$p(c_{t+1}\|c_{1:t})$	The conditional distribution of user check-ins
z_{t+k}, $\tilde{z}_{t+k}^j$	A positive sample/the jth negative sample
$\mathrm{S}(\cdot,\cdot)$	The similarity function
K, J	The number of prediction steps and negative samples

Definition 2 (*Trajectory–User Linking (TUL)*)**.** A trajectory $\widetilde{T}$ for which we do not know the user who generated it is called unlinked. Suppose we have a number of unlinked trajectories $\widetilde{\mathcal{T}} = \{\widetilde{T}_1, \widetilde{T}_2, \dots, \widetilde{T}_M\}$ produced by a set of users $\mathcal{U} = \{u_1, \dots, u_N\}$ ($M \gg N$). The TUL problem is to learn a mapping function f_{TUL} that classifies the unlinked trajectories to their corresponding users: $f_{\mathrm{TUL}}(\widetilde{\mathcal{T}}) \mapsto \mathcal{U}$.

Following previous works [17,19], we consider LP and TUL as the classification problems which are similar but different. The LP is to predict the next location c_{t+1} while the TUL tries to link trajectories to users who generate them in the Location-based Social Network (LBSN).

2.2. Mutual information & contrastive loss

Mutual information (MI) is a useful information measure in random variable dependencies, which can be regarded as the information contained in a random variable w.r.t. another random variable, i.e., it refers to the correlation between two variables. The mutual information for the two set of variables X and Y is defined as:

$$\begin{aligned} \mathrm{MI}(X;Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X), \end{aligned} \tag{1}$$

where $H(X)$ denotes the information entropy of X: $H(X) = -\sum_{x\in X} p(x)\log p(x)$. The mutual information between X and Y can be perceived as the decrement of the uncertainty of X given Y (i.e., $H(X \mid Y)$), or vice versa.

Suppose the joint distribution of two random variables is $p(x,y)$, and the marginal distributions are $p(x)$ and $p(y)$. The mutual information $\mathrm{MI}(X;Y)$ is actually the Kullback–Leibler (KL) divergence between the joint distribution $p(x,y)$ and the marginal distribution $p(x)p(y)$:

$$\begin{aligned} \mathrm{MI}(X;Y) &= \sum_{x\in X, y\in Y} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \\ &= \mathrm{KL}(P(X,Y) \parallel P(X)P(Y)). \end{aligned} \tag{2}$$

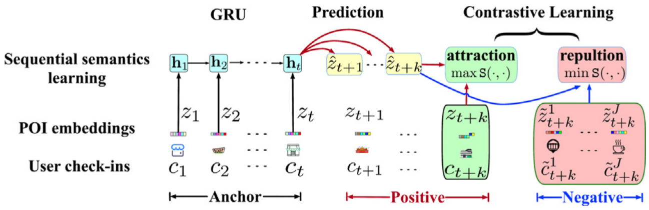


Fig. 1. Illustration of self-supervised mobility learning.

Since KL-divergence is unbounded, it is difficult to directly maximize the mutual information. In practice, we can reduce its lower bound by minimizing the noise contrastive estimation (NCE) loss [25,26], which has been widely used in existing SSL models [27–29]:

$$\mathcal{L}_{\mathrm{NCE}} = \mathbb{E}\left[-\log\left(\frac{e^{g(x)^{\intercal}g(x^{+})}}{e^{g(x)^{\intercal}g(x^{+})} + \sum_{j=1}^{J} e^{g(x)^{\intercal}g(x_j^{-})}}\right)\right], \tag{3}$$

where x, x^{+}, and x^{-} denote the *anchor*, *positive*, and *negative* samples, i.e., x^{+} is *similar* to x and x^{-} is *dissimilar* to x. The similarity measure $e^{g(x)^{\intercal}g(x^{+})}$ (or $e^{g(x)^{\intercal}g(x_j^{-})}$) and embedding methods $g(\cdot)$ may vary from task to task, but the contrastive learning framework is almost the same. More details of SSL and contrastive learning are referred to a recent comprehensive review [24].

3. Methodologies

In this section, we first outline the architecture of the proposed self-supervised mobility learning. Then, we describe the base framework for spatio-temporal factor learning. Finally, we introduce how to pre-train and fine-tune our model using contrastive trajectory learning for downstream tasks

3.1. Overview

Most of deep human mobility prediction methods [16,17,19,23] generally use RNNs to model the spatio-temporal factors associated with users' consecutive check-ins and movement periodicity captured by historical trajectory matching. Essentially, these approaches rely on the ability of RNNs to learn the conditional distribution $p(c_{t+1}|c_{1:t})$ in an auto-regressive manner, i.e., the probability of the next location $p(c_{t+1})$ is dependent on the previous check-ins $c_{1:t}$. The rich knowledge such as mobility contexts and spatio-temporal constraints have not been explicitly considered as the supervision signals for training meaningful mobility representations, although these features have been modeled as auxiliary information.

Inspired by recent advances of self-supervised learning in CV and NLP domains, we propose to learn the predictive representations from the user mobility itself constrained by the spatio-temporal factors. We introduce Self-supervised Mobility Learning (SML), which is designed for modeling sparse and noisy human mobility trajectories, focusing on leveraging rich spatio-temporal contexts and multi-view traces augmentation for self-supervised human mobility learning. Instead of relying solely on historical observations and spatio-temporal factors, we explicitly generate synthetic trajectories with realistic mobility constraints to enhance the trajectory representations and thus the downstream tasks, e.g., next location prediction and trace classification.

As shown in Fig. 1, a trajectory is encoded into latent representations using an embedding method, while taking the spatio-temporal factors (i.e., the distance and time intervals between the successive check-ins) into account [18]. Then, we use a gated recurrent units (GRUs) to model the trajectory $T = \{c_1, c_2, \dots, c_t\}$ that encodes the user's mobility before time $t+1$ into a contextual latent representation $\mathbf{h}_t$. Instead of relying on $\mathbf{h}_t$ to predict the

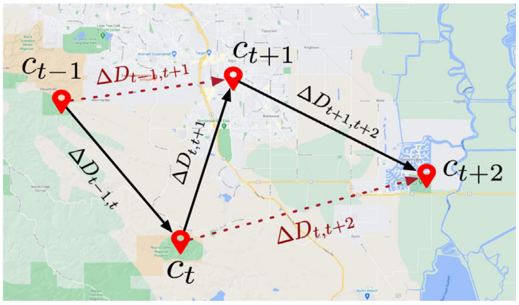


Fig. 2. Multi-step spatio-temporal intervals.

next location c_{t+1} as existing models did, we predict K future visits $\{c_{t+1}, \dots, c_{t+K}\}$ that contains more semantic transitions and motion intentions than a single POI. Furthermore, we aim to discriminate the real future mobility from the noise samples, e.g., the confounding check-ins and distanced POIs, which allows us to extract persistent visiting purposes by maximizing the MI between past trajectories and future visits.

3.2. Spatio-temporal context learning

We first present a basic model to capture the spatio-temporal contexts of human mobility, including POI embedding, spatio-temporal factors and mobility periodicity.

3.2.1. POI embedding

The real user trajectory is high-dimensional data, which requires a way to embed them into low-dimensional dense vectors using some point embedding methods. Inspired by previous works [19,22], we use word2vec [30] to embed the POIs. Specifically, considering the spatio-temporal and sequential relationship of user check-ins, we choose the continuous bag-of-words (CBOW) model to train the trajectory data by predicting the current POI from a window of surrounding check-ins.

Specifically, by predicting each POI c given a user trajectory, the POI is embedded into a latent vector $z \in \mathbb{R}^d$ – d denotes the dimensionality. Here we use negative sampling during training which allows one training sample to update only part of the weight and therefore reduce the amount of training time. Note that the POI embedding is performed once as pre-training mobility data, which is also in a self-supervised learning manner.

3.2.2. Spatio-temporal factors and sequential model

As demonstrated in most existing works [19,22,23], the nearby locations often have underlying relationships and semantic interactions. Besides, the spatio-temporal factors restrict user movement and are predictive signals that should be considered in learning human dynamics. In particular, the travel distance $\Delta \mathrm{D}_{t-1,t}$ and time interval $\Delta \mathrm{T}_{t-1,t}$ between successive check-ins are incorporated as spatio-temporal influencing factors into the RNN model. Instead of only relying on successive check-ins c_{t-1} and c_t, we also take the constraints $\Delta \mathrm{D}_{t-1,t+1}$ and $\Delta \mathrm{T}_{t-1,t+1}$ into account, as illustrated in Fig. 2. The basic idea is to consider the future movement (e.g., the check-ins at and after $t+1$ times) that can help understand the spatio-temporal context of a user. To this end, we use two GRUs [31] to model the iterative spatio-temporal factors as:

$$\begin{aligned} \mathbf{h}'_t &= \mathrm{GRU}\left(c_t, \mathbf{h}_{t-1}, \Delta \mathrm{T}_{t-1,t}, \Delta \mathrm{D}_{t-1,t}\right), \\ \mathbf{h}''_t &= \mathrm{GRU}\left(c_t, \mathbf{h}_{t-1}, \Delta \mathrm{T}_{t-1,t+1}, \Delta \mathrm{D}_{t-1,t+1}\right), \end{aligned} \tag{4}$$

where $\mathbf{h}'_t$ and $\mathbf{h}''_t$ are the hidden states of two GRUs and would be further concatenated as $\mathbf{h}_t = (\mathbf{h}'_t; \mathbf{h}''_t)$.

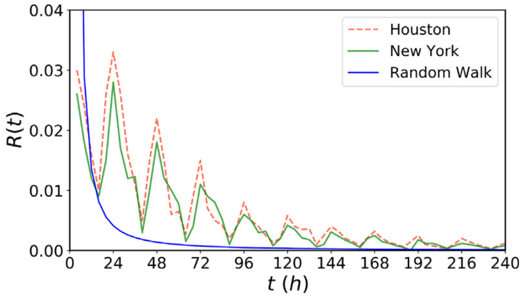


Fig. 3. Return probability distribution $R(t)$ of users in Houston [32] and New York [33] datasets. We randomly select 100 users from each dataset and plot the averaged return probability after t hours.

3.2.3. Movement periodicity

Relying solely on spatio-temporal influence and sequential patterns is insufficient [19,23], since the model would be biased and prefer to nearby POIs that are frequently visited. On the other hand, people's movements are usually periodical, i.e., people tend to visit the same or similar places [22], more formally, with high return probabilities in a set of POIs [13,23,34]. For example, Fig. 3 illustrates the averaged return probability of users from two datasets [32,33], i.e., the probability of revisiting the POIs where the users are first observed after t hours. We can observe regular peaks that demonstrate the tendency of humans to locations they visited before, in contrast with the smooth asymptotic behavior of random walk.

Here we use another GRU to encode historical trajectory $\mathcal{T}$ and retain the intermediate hidden states $\mathbf{o}_h$ as the candidate vectors for trajectory matching. To alleviate this issue and account for the periodicity of movement, we use an attention network to catch the routines in human mobility through querying a trajectory episode in historical records $\mathcal{T}$ that is most similar to the current trajectory T:

$$\alpha_i^t = \mathrm{softmax}(\boldsymbol{F}(\mathbf{h}_t, \mathbf{o}_i)), \tag{5}$$

$$\delta_t = \sum \alpha_i^t \mathbf{o}_i, \tag{6}$$

$$\boldsymbol{F}\left(\mathbf{h}_t, \mathbf{o}\right) = \tanh\left(\mathbf{h}_t \boldsymbol{W} \mathbf{o}\right), \tag{7}$$

where α_i^t is the attention weight that is calculated by the similarity between $\mathbf{h}_t$ and the historical episode o_i. The vector δ_t computes the periodicity in $\mathcal{T}$ related to the user's current mobility. $\boldsymbol{F}$ is the score function calculated by the current mobility ($\mathbf{h}_t$), the learnable parameters $\boldsymbol{W}$, and the historical mobility semantics $\mathbf{o}$ [19].

We note that other more complex schemes, such as variational attention [22], hierarchical attention [19], and hidden state flashback [23], can be used instead. Meanwhile, the POI embeddings can be improved with the extra semantic information as suggested in [35]. We are very interested in including the extra information for POI embedding that would definitely capture more semantic contexts and user preference over POIs in the representations and therefore benefit the downstream tasks, which, however, is beyond the scope of this work and left as our future work.

Algorithm 1 summarizes the spatio-temporal context learning in the proposed model.

3.2.4. Trajectory augmentation

Since users' check-in behaviors are sporadic, the observed trajectories only reveal people's partial movements and visiting interest preference. This is one of the main reasons that causes the inaccurate human mobility prediction in existing methods. On the other hand, data augmentation methods are the core of

Algorithm 1: Spatio-temporal context learning.

```
Input: User set $\mathcal{U}$; POI set $\mathbf{C}$; Historical trajectories $\mathcal{T}$;
       Current trajectory $T$.
Output: Trained parameters for $f_{\mathrm{LP}}$ and $f_{\mathrm{TUL}}$.
Compute POI embeddings using word2vec;
foreach $T$ do
    Model spatio-temporal influence using two GRUs
    (Eq. (4));
    Concatenate $\mathbf{h}_t = (\mathbf{h}'_t; \mathbf{h}''_t)$;
    Calculate the most similar episode to $T$ in $\mathcal{T}$ via
    Eq. (5)–(7);
end
Pre-train trajectories via SML (cf. Algorithm 2);
Fine-tune the model for $f_{\mathrm{LP}}$ and $f_{\mathrm{TUL}}$.
```

many self-supervised learning studies [24], which could provide a multi-view of the data and help design the auxiliary self-supervised learning tasks [24,36]. These facts motivate us to learn meaningful mobility representations that encode spatio-temporal semantics beyond the sparse and sporadic observations while providing extra signals for enriching the self-supervised mobility learning.

Here we propose to augment the sparse trajectory data with realistic trajectories that match the spatial and temporal patterns of personalized mobility. Intuitively, longer trajectories could reflect the relatively continuous, accurate, and complete user visiting preference, e.g., the users preferred to publish their locations. In contrast, the observed mobility for people with fewer check-ins is full of noisy and implicit feedback. Therefore, we only augment the trajectories no longer than η, i.e., $|T| \leq \eta$. Specifically, we augment the mobility data in three ways, i.e., *trajectory sub-sampling* (TSS), *spatial augmentation* (SA) and *temporal augmentation* (TA).

Trajectory sub-sampling. The simplest way is to sample the subsequences from an original trajectory, which is very similar to image cropping in computer vision. For a N-length trajectory, we can sample at most $N!$ different subtrajectories preserving the visiting orders of the observed trajectory. The generated trajectories can be considered as different views and the unbiased samples from the original data distribution that maintain the local semantics of motion patterns. Besides, subsequences usually represent users' fine-grained activities and coherent preferences that are indicative signals for next location prediction [21].

Though sub-sampling is a straightforward method of augmenting sparse human mobility, neither extra knowledge w.r.t. user preferences nor unobserved motion patterns have been introduced. To address this issue, we generate synthesis trajectories for each user that account for the spatial and temporal semantics.

Spatial augmentation. For a user, a trajectory merely reveals the motion patterns in a period of time such as one day. Due to the sparse check-in behavior, her complete transition patterns may not be fully observed. As shown in Fig. 4(a), there are two observed mobility, i.e., $T_1 = (c_1, c_2, c_3, c_4)$ and $T_2 = (c'_1, c'_2, c'_3)$. If two check-ins such as c_2 and c'_2 belong to the same category (e.g., restaurant) and are very close in geographic distance, the two locations can be considered as "interchangeable". This is intuitive and reasonable to infer the invisible but realistic transition patterns. For example, this user has lunch in Subway (c_2), picks a cup of coffee in Starbucks (c_3), and then goes back to office (c_4). On another day, she takes some food from Taco Bell (c'_2) and then goes to the gym (c'_3) for exercise. Accordingly, the trajectories (c_1, c'_2, c_3, c_4) (i.e., having lunch in Taco Bell and drinking coffee in Starbucks) and (c'_1, c_2, c'_3) (i.e., taking some food from Subway and do exercise in the gym) are realistic transitions, though they are not revealed in the data. Note that the spatial augmentation subjects to two criteria, i.e., within a distance threshold δ (e.g., 5 km) and in the same POI category. Therefore, trajectories such as (c'_1, c_2, c_3), (c'_1, c_2, c_3, c_4), (c_1, c'_2, c_3), and (c_1, c'_2, c'_3) still satisfy the data augmentation criteria.

Temporal augmentation. The close check-ins allow us to explore possible mobility satisfying geographical constraints, but it overlooks users' temporal preference over particular POIs. To this end, we also augment a user's trajectories according to her temporal visiting preference. As illustrated in Fig. 4(b), the user may go back to office (c_4) in workday or go to gym (c'_3) in weekend around the same time (e.g., 2:00 pm). Therefore, it is possible to augment the original trajectories according to the temporal visiting preference if two subsequent check-ins are within a particular time period (e.g., 1 h), even though the two check-ins are not in the same POI category. We note that this type of augmentation can relax the spatial constraint threshold δ to a larger value.

3.3. Self-supervised mobility learning

Now we describe the details of contrastive trajectory learning in SML.

3.3.1. Contrastive mobility learning

Given a sequence of observations $T_{\leq t} = \{c_1, c_2, \ldots, c_t\}$, we have used the GRU-based sequence-to-sequence neural networks to encode $T_{\leq t}$ into a compact hidden state $\mathbf{h}_t$ and decode it as consecutive future visits $T_{>t} = \{c_{t+1}, c_{t+2}, \ldots, c_{t+K}\}$, where K is a small number, as illustrated in Fig. 1. We perform self-supervision in the latent space and predict the latent vectors $\{\hat{z}_{t+1}, \ldots, \hat{z}_{t+K}\}$ corresponding to the K future successive check-ins. Our goal here is to maximize the MI between the context of previous mobility $\mathbf{h}_t$ and future visits $T_{>t}$:

$$\mathrm{MI}(\mathbf{h}_t; T_{>t}) = \sum p(\mathbf{h}_t, T_{>t}) \log(p(T_{>t}|\mathbf{h}_t) - p(T_{>t})), \tag{8}$$

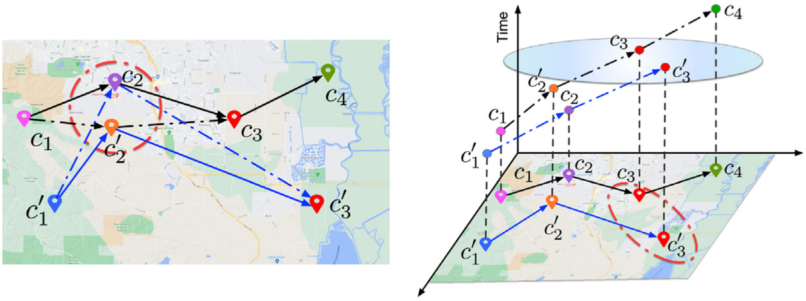
where $\mathrm{MI}(\cdot;\cdot)$ is a sample-based estimator for the real mutual information. Our goal is to learn the underlying *continuous* motion semantics of a user rather than the element-wise consistency of next check-ins prediction. That is, it encodes the fluent mobility context that could reflect the (historical) continuous intentions of a user while being able to predict her future visits.

For each prediction $\hat{z}_{t+k}$, we consider the ground truth z_{t+k} (corresponding to real check-in c_{t+k}) as the *positive* sample, and then randomly sampled J check-ins $\{\tilde{c}^j_{t+k}\}^J_{j=1}$ as the *negative* samples from the trajectories generated by *other* users. To evaluate the prediction performance during training, the NCE loss [25,26] is used:

$$\mathcal{L} = -\sum_{k=1}^{K} \frac{\mathrm{S}\left(\hat{z}_{t+k}, z_{t+k}\right)}{\mathrm{S}\left(\hat{z}_{t+k}, z_{t+k}\right) + \sum_{j=1}^{J} \mathrm{S}\left(\hat{z}_{t+k}, \tilde{z}^j_{t+k}\right)}, \tag{9}$$

where $\mathrm{S}(\cdot,\cdot)$ is the function evaluating the similarity between two vectors, and $\tilde{z}^j_{t+k}$ denotes the embedding of the jth negative sample. The objective here is to maximize the similarity between the predicted results and the ground-truths while minimizing the similarity between the predictions and negative samples. In practice, we use the dot product followed by a nonlinearity transformation to assess the similarities, i.e. , $\mathrm{S}\left(\hat{z}_{t+k}, z_{t+k}\right) = \sigma(\hat{z}_{t+k} \cdot z_{t+k})$, where σ is the `sigmoid` function.

During training, the encoded context $\mathbf{h}_t$ is forced to preserve the knowledge regarding a user's historical movement and provide the opportunity for contrastive representation learning in the latent space. In this period, the future visiting locations can be considered as self-supervision signals for training the model.



(a) Spatial augmentation. (b) Temporal augmentation.

Fig. 4. The illustration of trajectory augmentation with spatial and temporal constraints.

In this way, our model encodes the mobility representations of the underlying information associated with user motion preference and patterns while distinguishing the real visits from the negative candidate POIs. The rationale behind this optimization is to estimate the density ratio of noise samples (i.e., check-ins) by distinguishing between check-ins from the user's real mobility and those from other users known as the noise distribution. As the noise samples increase, Eq. (9) essentially approaches the maximum likelihood estimation.

By optimizing Eq. (9), we essentially maximize the mutual information between historical context $\mathbf{h}_t$ and the (predicted) future visits z_{t+k}. Let $Z = \{z_1, \ldots, z_j, \ldots, z_{J+1}\}$ be $J+1$ random samples containing one positive sample from $p(z_{t+k}|\mathbf{h}_t)$ and J negative samples Z_{neg} from a proposal distribution $p(z_{t+k})$. The optimal value of similarity measurement $\mathrm{S}\left(\hat{z}_{t+k}, z_{t+k}\right)$ is given by $p(\frac{p(z_{t+k}|\mathbf{h}_t)}{p(z_{t+k})})$, i.e., the optimal $\mathcal{L}_{\text{opt}}$ can be decomposed by:

$$
\begin{aligned}
\mathcal{L}_{\text{opt}} = -\mathop{\mathbb{E}}_{Z} \log &\left[\frac{\frac{p(z_{t+k}|\mathbf{h}_t)}{p(z_{t+k})}}{\frac{p(z_{t+k}|\mathbf{h}_t)}{p(z_{t+k})} + \sum_{Z_{\text{neg}}} \frac{p\left(\tilde{z}^j_{t+k}|\mathbf{h}_t\right)}{p\left(\tilde{z}^j_{t+k}\right)}}\right] \\
=\mathop{\mathbb{E}}_{Z} \log &\left[1 + \frac{p\left(z_{t+k}\right)}{p\left(z_{t+k} \mid \mathbf{h}_t\right)} \sum_{Z_{\text{neg}}} \frac{p\left(\tilde{z}^j_{t+k} \mid \mathbf{h}_t\right)}{p\left(\tilde{z}^j_{t+k}\right)}\right] \\
\approx\mathop{\mathbb{E}}_{Z} \log &\left[1 + \frac{p\left(z_{t+k}\right)}{p\left(z_{t+k} \mid \mathbf{h}_t\right)} \mathop{\mathbb{E}}_{\tilde{z}^j_{t+k}} \frac{p\left(\tilde{z}^j_{t+k} \mid \mathbf{h}_t\right)}{p\left(\tilde{z}^j_{t+k}\right)} J\right] \qquad (10) \\
=\mathop{\mathbb{E}}_{Z} \log &\left[1 + \frac{p\left(z_{t+k}\right)}{p\left(z_{t+k} \mid \mathbf{h}_t\right)} J\right] \\
\geq\mathop{\mathbb{E}}_{Z} \log &\left[\frac{p\left(z_{t+k}\right)}{p\left(z_{t+k} \mid \mathbf{h}_t\right)} (J+1)\right] \\
= -\,&\mathrm{MI}\left(\mathbf{h}_t; z_{t+k}\right) + \log(J+1), \qquad (11)
\end{aligned}
$$

where Eq. (10) becomes more accurate as J increases, which explains why more negative samples are desirable in many SSL tasks. Eq. (11) says that the mutual information between the previous semantics $\mathbf{h}_t$ and the real future visits z_{t+k} result in the optimal training. In practice, the model converges when the predicted locations $\hat{z}_{t+k}$ is accurate while maximizing the distance between the latent representations of the real check-ins and the negative samples.

To maximize the mutual information of Eq. (8), we actually conduct a pre-training pretext task to estimate the similarity degree of different trajectory sequences, which is equivalent to minimizing a related combined loss function. According to Eq. (11), we can evaluate the mutual information between the representation vectors $\hat{z}_{t+k}$ and z_{t+k} as follows:

$$
\mathrm{MI}\left(z_{t+k}, \hat{z}_{t+k}\right) \geq \log(J+1) - \mathcal{L}_{\text{opt}}, \qquad (12)
$$

which becomes tighter as J increases. We sampled J negative POIs that enhance the association between the trajectory sequence z_{t+k} and the hidden states $\hat{z}_{t+k}$ of contexts. We can observe that minimizing the loss $\mathcal{L}_{\text{opt}}$ maximizes a lower bound on mutual information.

The full procedure of training SML is outlined in Algorithm 2.

Algorithm 2: Self-supervised mobility learning.

```
Input: The historical context $\mathbf{h}_t$; POI embeddings; The
       prediction steps: $K$; The number of negative samples
       $J$.
Output: Pre-trained trajectory representation.
Augment $\mathcal{T}$ with spatio-temporal and categorical POI
constraints;
foreach $T$ do
    Predict future locations $\hat{z}_{t+1}, \cdots, \hat{z}_{t+K}$;
    for $k = 1 \cdots K$ do
        Compute similarity $\mathrm{S}\left(\hat{z}_{t+k}, z_{t+k}\right)$;
        Sample $J$ negative POIs $\tilde{z}_{t+1}, \cdots, \tilde{z}_{t+K}$;
        Compute similarity $\mathrm{S}\left(\tilde{z}_{t+k}, z_{t+k}\right)$;
        Minimize the loss in Eq. (9);
    end
end
```

3.3.2. Why does SML work?

In our SML, the conditional distribution corresponding to the previous mobility $\mathbf{h}_t$ can be defined as follow:

$$
P_\theta(z_{t+k}) = \frac{\exp\left(s_\theta(z_{t+k}, \mathbf{h}_t)\right)}{\sum_{j=1}^{J} \exp\left(s_\theta\left(\tilde{z}^j_{t+k}, \mathbf{h}_t\right)\right)}, \qquad (13)
$$

where $s_\theta(z_{t+k}, \mathbf{h}_t)$ symbolizes a function with parameters θ quantifying the compatibility of POI z_{t+k} with the previous mobility $\mathbf{h}_t$. The basic idea of NCE is to reduce the density ratio estimation in binary classification, i.e., discriminating between samples from the data distribution P_d and those from a known noise distribution P_n. In the human mobility setting, $P_\theta(z_{t+k})$ refers to the distribution of POIs that will be visited after a previous mobility $\mathbf{h}_t$, and $P_n(\tilde{z}^j_{t+k})$ denotes the noise distribution. The objective function can be therefore written as:

$$
\begin{aligned}
\mathrm{Obj}(\theta) =&\mathbb{E}_{P_d}\left[\log \frac{P_\theta(z_{t+k})}{P_\theta(z_{t+k}) + kP_n(\tilde{z}^j_{t+k})}\right] + \\
&J\mathbb{E}_{P_n}\left[\log \frac{kP_n(\tilde{z}^j_{t+k})}{P_\theta(z_{t+k}) + J\dot{P}_n(\tilde{z}^j_{t+k})}\right], \qquad (14)
\end{aligned}
$$

whose gradient is computed as:

$$\frac{\partial}{\partial\theta}\mathrm{Obj}(\theta) = (P_d(z_{t+k}) - P_\theta(z_{t+k}))\frac{\partial}{\partial\theta}\log P_\theta(z_{t+k}) \times \sum_k \frac{JP_n(\tilde{z}^j_{t+k})}{P_\theta(z_{t+k}) + JP_n(\tilde{z}^j_{t+k})}. \tag{15}$$

As J increase to $+\infty$, Eq. (15) becomes:

$$\frac{\partial}{\partial\theta}\mathrm{Obj}(\theta) \approx \sum_k (P_d(z_{t+k}) - P_\theta(z_{t+k}))\frac{\partial}{\partial\theta}\log P_\theta(z_{t+k}), \tag{16}$$

which is the maximum likelihood gradient. Thus as the number J of negative samples increases, i.e., the ratio of noise POIs to real check-ins increases, the loss function optimization gradually approaches the maximum likelihood gradient [37].

Moreover, there are lots of contrastive learning methods have been proposed in recent years, such as MoCo [27], SimCLR [38], BYOL [39], etc. However, the SOTA contrastive learning methods such as MoCo, SimCLR, and BYOL focus on image recognition tasks, which do not need to consider the temporal and sequential patterns of the data and thus cannot be directly deployed into our model. Besides, most contrastive learning methods require intensive computation, e.g., SimCLR uses very large mini-batches while MoCo needs momentum encoder updating — both are computation intensive [24]. Nevertheless, the basic ideas of different contrastive learning methods are very similar, i.e., minimizing the representations between positive samples while maximizing the distance between different samples. Our method uses InfoNCE as the contrastive loss, which is also widely adopted in contrastive learning models for image recognition, including MoCo and SimCLR.

3.3.3. Fine-tuning for downstream applications

We use both original and augmented training data for pre-training the model for both of the two tasks, i.e., location prediction (LP, Application I) and trajectory–user linking (TUL, Application II). The detailed data statistics of original and augmented training data are shown in Tables 3 and 7. We now present the details of fine-tuning the pre-trained trajectory representations to the specific applications. After training the contrastive trajectory learning model, the model parameters can be used to fine-tune the two tasks. Note that the fine-tuning is performed on the original training set. More specifically, we load all variables except the last softmax layer to fine-tune the model.

Application I — LP: We use the *cross-entropy* to train the model. The primary task is to predict the next location c_{t+1} given current movement T and historical trajectory $\mathcal{T}$. We call this model SML-LP.

Application II — TUL: The TUL problem is slightly different from LP since the labels are the users of the trajectories. This model, called SML-TUL, is also trained with cross-entropy loss.

4. Evaluations

We now present the experimental evaluations which investigate the model performance of ours and baselines on two applications. Specifically, we provide the quantitative results to answer the following research questions:

- **RQ1**: How does SML perform on human mobility prediction and trajectory classification when compared with the state-of-the-art baselines?
- **RQ2**: How do the hyper-parameters affect the performance of SML?
- **RQ3**: How do different components in SML contribute to the overall performance?

Table 2
Statistics of the datasets.

City	Users	POIs	Check-ins	Trajectories
New York	1,083	9,815	1,20,007	76,905
Singapore	2,321	5,082	1,94,108	59,864
Houston	4,627	9,326	3,62,783	67,550
California	3,987	21,358	2,39,493	1,23,935

Table 3
The statistics of the original and augmented training data for location prediction.

Dataset	Original training data	Augmented data
New York	36,182	40,723
Singapore	34,713	25,151
Houston	18,501	49,040
California	66,612	57,323

4.1. Application I: Location prediction

We first evaluate the performance of SML-LP comparing to the methods for identifying trajectories for given users.

4.1.1. Experimental settings

Datasets. We evaluate all methods using the publicly available datasets collected from two LBSNs: Foursquare and Gowalla. In Foursquare, we selected the data from two cities, i.e., New York and Singapore [33]. In Gowalla, we used the data from two states in U.S., i.e., California and Houston [32]. Following previous works [19,22], we filtered out the POIs visited by fewer than five users, and the users whose check-ins are fewer than five. For each user, we concatenate her all chronological check-ins, and divide the whole trajectories into subsequences with the time interval of 6 h each, as it was done in previous related works [16,19,22]. The statistics of the data are summarized in Table 2. In addition, Table 3 shows the number of original and augmented training data used for contrastive trajectory learning.

Baselines. We compare our model with the following baselines:

- *PRME* [40] is a pair-wise metric embedding method that encodes the POIs into a latent Euclidean space with matrix factorization and estimates the movement transitions using the Markov chain.
- *NexT* [1] exploits users' individual and collective mobility and proposes a two-step prediction model. It first exploits sequential mobility patterns and then extracts a set of spatial–temporal features for user mobility prediction using a supervised learning-based decision tree model.
- *ST-RNN* [16] is an RNN-based method that combines spatio-temporal factors when predicting user's next location.
- *POI2Vec* [18] is a POI embedding-based method taking geographical and temporal influence into account, and predicts the next location based on the learned POI representations.
- *HST-LSTM* [20] incorporates spatial–temporal influence into LSTM in a sequence-to-sequence learning manner, and leverages the contextual information to improve the model performance on sparse data prediction.
- *Flashback* [23] is another RNN-based sparse mobility model doing flashbacks queries on hidden states of RNN. The basic idea is to search the periodic motion patterns from historical data which is similar to DeepMove and VaNext except that it performs matching in the hidden states.
- *DeepMove* [19] combines RNN and attention mechanism to model human dynamics. It introduces a trajectory matching network to learn the motion periodicity by querying the same/similar movement in a users' recorded data.

- *VANext* [22] uses the variational attention to encode the recent mobility episode and exploits user's periodical mobility. It utilizes a CNN to capture moving patterns of users instead of RNNs.

In our previous work [22], VANext-S utilized VAE to pretrain all trajectories, including the data in the testing set, in an unsupervised manner. However, the model proposed in this work only used the training data and its augmented trajectories, for contrastive mobility learning. For a fair comparison, we only compare our model with VANext that follows the same settings as the approach proposed in this work as well as the compared baseline models. We respectfully note that we have conducted additional experiments to evaluate our method on pretraining all trajectory data (including the testing data). Although the results showed that our model outperforms VANext-S by a large margin, we only reported the results using training data for contrastive learning for a fair comparison.

Implementations. All experiments were implemented in Python on a server with an Nvidia GTX 2080 Ti GPU. We use Adam [41] to train all deep learning based models. The initial learning rate is 0.001 and decays 10% every 10 epochs. The hidden size of GRUs and attention networks are set to 300. The batch size is 16. The negative samples J and the predicted steps K are set to 10 and 3 respectively unless otherwise specified.

Metrics. Following related studies, we evaluate all methods with three widely used metrics for location prediction: average accuracy (ACC@$*$), where $* = 1, 5, 10$, area under the ROC curve (AUC), and mean average precision (MAP).

4.1.2. Performance comparisons

Tables 4 and 5 report the performance of different methods on the datasets of four cities, where the best is shown in **bold**, and the second-best is shown as <u>underlined</u>. A paired t-test is performed, and $*$ indicates that the performance gain is significant with p-value < 0.001 compared to the most competitive method. After examining the model performance, we have the following observations.

First, our SML-LP consistently and significantly outperforms previous next location prediction methods by a large margin across all datasets. In particular, our approach yields 10.4%, 5.9%, and 13.2% improvement over the best baseline method in terms of ACC@5, AUC, and MAP, respectively, in the New York dataset. These results firmly prove our method's superiority that learns human mobility from the sparse and noisy self-supervision signals. Compared to previous deep mobility models, ours not only distills more meaningful contexts by contrasting the real user check-ins with negative trajectories but also enriches the sparse user trajectories with synthesis but realistic mobility. Furthermore, the superiority of our method indicates that self-supervised architecture proposed in this work is particularly suitable for modeling human mobility data.

Second, the matrix factorization-based embedding approach such as PRME is not competitive since the user check-in behavior is very sparse, making the matrix factorization insufficient to capture user preference and movement contexts. The latent representation model POI2Vec significantly improves the POI embedding performance due to its ability to incorporate the geographical influence that is important for modeling user mobility behavior. However, these methods only consider the mobility contexts but fail to model users' complex sequential transition regularities. As an ensemble learning model, NexT requires significantly intensive manual feature engineering that cannot be generalized to different data. More importantly, it can only capture the linear interactions between users and POIs, resulting in poor prediction performance compared to deep learning-based models.

Third, RNN-based approaches (e.g., ST-RNN, HST-LSTM, DeepMove, etc.) are capable of capturing users' historical sequential behaviors and predicting the immediate next location more accurately compared to previous human mobility models. These approaches rely on the spatio-temporal factors and the sequential regularity of users' historical check-ins, which, however, consists of extremely sparse and implicit user feedback that may not fully reflect users' real preference and continuous check-in behaviors. Among previous deep learning models, the mobility periodicity is a strong indicator for location prediction, which can be proved by the improvements of Flashback, DeepMove, and VANext over the ST-RNN and HST-LSTM methods. As an attention-based mobility model, VANext exhibits better performance than other baselines, implying the effectiveness of learning long-term historical mobility periodicity with convolutional networks. The additive benefits of VANext are clear, which also indicates the advantages of modeling the uncertainty inherent in user mobility, e.g., due to the privacy issue or inaccurate indoor GPS signals. However, these methods still suffer from the data sparsity problem, which may result in biased user preference learning. Besides, these baselines overlook the rich supervisions inherent in user mobility that can distinguish the positive future visits from a set of negative POIs.

4.1.3. Parameter sensitivity

We now investigate the influence of two important parameters in our SML-LP model, i.e., the prediction steps K and the number of negative samples J.

Fig. 5 reports the parameter sensitivity analysis of SML-LP in terms of three evaluation metrics across the four datasets. Note that the more steps prediction only happens during trajectory pre-training. That is, we follow the next location prediction during the fine-tuning stage. Figs. 5(a)–5(c) demonstrate that the SML performance varies with the K values, indicating that predicting more future steps instead of only the next one helps understand users' motion patterns and intentions. This is reasonable since a few more future check-ins can reflect more fluent user mobility and alleviate the prediction bias. This result also implies that future mobility is beneficial for understanding users' past behaviors and visiting patterns. We also found that a small value of K (e.g., 3 or 4) is sufficient, which would result in performance degradation if we further increase the prediction steps. This happens due to the difficulty of predicting a larger value of future visits, i.e., it may increase the variance in estimating the long-term mobility.

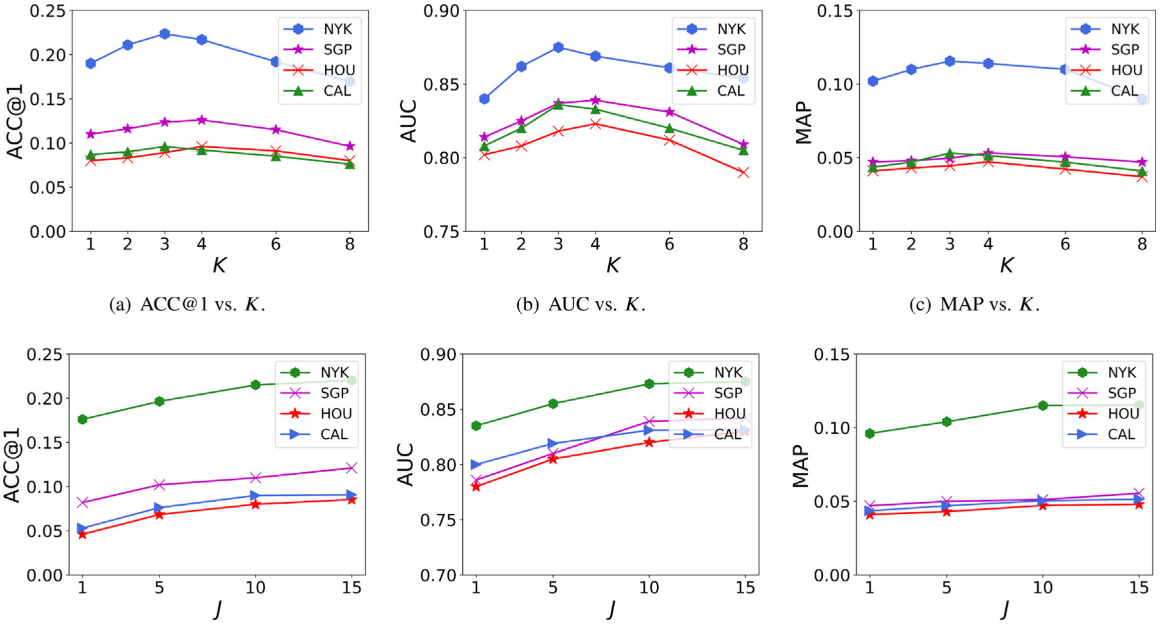
The negative samples play essential roles in discriminating the real user visits from the confounding POIs. As shown in Fig. 5(d)- 5(f), $10 - 15$ negative samples are adequate for our model to achieve the best performance. In other words, more negative samples are unnecessary for location prediction, which is slightly different from the results reported in SSL-based CV tasks. The rationale is that the spatio-temporal factors principally restrict the user mobility, e.g., the next location is usually in surrounding areas of the user's recent check-ins. Therefore, the distanced negative samples would not help improve the model performance. In contrast, the most useful samples in the location prediction task are nearby POIs that may lead to wrong predictions. Consequently, more negative POIs do not necessarily mean *hard* negative POIs — the latter is the key to facilitate better and faster contrastive learning [42]. In this spirit, the hard samples, both positive and negative, should be carefully considered in learning human mobility. However, designing the hard POI samples in contrastive trajectory learning is beyond the scope of this work and hence left as our future work.

Table 4
Next location prediction performance comparisons on New York and Singapore datasets.

Method	New York					Singapore				
	ACC@1	ACC@5	ACC@10	AUC	MAP	ACC@1	ACC@5	ACC@10	AUC	MAP
PRME	12.83	24.38	19.66	73.56	5.62	6.36	6.27	8.29	71.06	2.96
NexT	16.93	26.10	28.48	71.35	5.78	6.65	14.66	17.16	72.10	3.02
ST-RNN	18.64	28.01	30.23	74.86	6.52	6.64	15.01	18.34	73.61	3.32
POI2Vec	18.79	28.85	30.81	75.32	7.95	6.89	14.87	18.27	74.10	3.56
HST-LSTM	18.86	33.21	36.23	79.82	8.14	7.96	16.26	20.27	76.69	3.72
Flashback	19.30	34.56	37.13	80.25	8.21	8.56	17.34	22.62	78.50	3.95
DeepMove	19.29	35.37	38.35	81.11	8.90	8.82	17.93	22.80	80.21	4.16
VANext	21.09	38.22	44.88	84.51	10.21	11.60	24.53	30.25	81.36	4.50
SML-LP	**22.01**	**42.18**	**48.44**	**87.35**	**11.56**	**12.11**	**26.21**	**32.26**	**83.91**	**5.12**

Table 5
Next location prediction performance comparisons on Houston and California datasets.

Method	Houston					California				
	ACC@1	ACC@5	ACC@10	AUC	MAP	ACC@1	ACC@5	ACC@10	AUC	MAP
PRME	2.36	6.27	8.29	69.23	2.72	1.57	4.10	5.91	69.20	2.61
NexT	5.45	10.32	11.27	72.34	3.07	5.15	9.18	10.11	72.23	2.96
ST-RNN	6.32	11.38	13.37	74.30	3.20	5.23	10.30	11.94	74.02	3.26
POI2Vec	6.32	11.21	13.52	74.23	3.21	4.64	8.77	10.23	73.23	3.12
HST-LSTM	6.31	12.53	15.01	76.90	3.32	5.63	11.26	16.25	77.82	3.41
Flashback	6.63	13.02	16.2	78.51	3.40	6.87	12.68	16.95	78.01	3.60
DeepMove	6.65	13.43	15.87	78.30	3.53	6.62	13.79	17.15	80.15	3.84
VANext	7.29	15.78	20.21	80.15	4.16	7.34	14.71	17.78	81.24	4.62
SML-LP	**8.54**	**17.63**	**22.97**	**82.01**	**4.72**	**9.08**	**17.42**	**20.88**	**83.61**	**5.04**



(a) ACC@1 vs. *K*. (b) AUC vs. *K*. (c) MAP vs. *K*.

(d) ACC@1 vs. *J*. (e) AUC vs. *J*. (f) MAP vs. *J*.

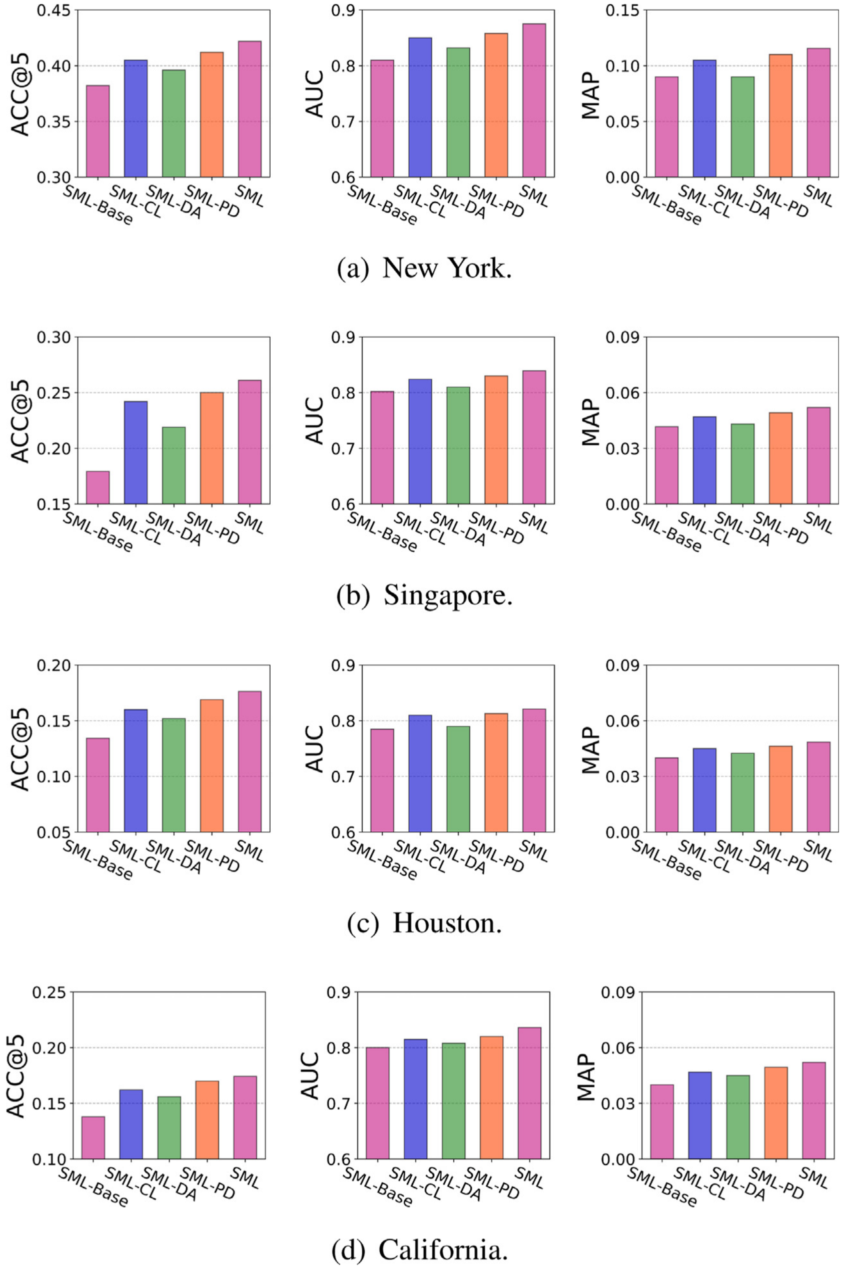
Fig. 5. The influence of the parameters *J* and *K*.

4.1.4. Ablation study

To scrutinize the efficacy of different components of SML-LP, we compare its variants from several aspects. We study the model's performance by ablating three important components. Towards that, we derive the following variants of SML-LP:

- SML-Base is a basic model that removes both trajectory data augmentation and contrastive mobility learning. The resulted model can be considered as a basic RNN-based mobility learning model such as DeepMove and Flashback.
- SML-CL is a model incorporating contrastive mobility learning and trajectory pre-training into the SML-Base model. Compared to the full SML-LP model, this one performs pre-training only on the original trajectory data, i.e., without trajectory augmentation.
- SML-DA is a model only considering data augmentation without contrastive trajectory learning, i.e., it augments the trajectory data and uses SML-Base for training the location prediction model.
- SML-PD is a variant of SML that does not consider movement periodicity, i.e., it removes the historical trajectory matching described in Section 3.2.3.

Fig. 6 illustrates the performance of variants on four datasets, from which we have the following findings. First, removing any



(a) New York.

(b) Singapore.

(c) Houston.

(d) California.

Fig. 6. Ablation study of SML-LP on four datasets.

sub-module would lead to performance degradation, indicating that both of the two main building blocks in SML are useful to improve the location prediction performance.

Second, contrastive trajectory learning that maximizes the mutual information between users' past movement and future check-ins contributes significantly to learning mobility semantics. This result proves the primary motivation of this work, i.e., leveraging the supervision signals extracted from the mobility semantics can significantly improve location prediction performance.

Third, the gap between SML and SML-DA indicates the importance of augmenting the sparse and incomplete trajectories with the synthesis trajectories constrained by the spatio-temporal factors. This result suggests that the generated trajectory data can enhance the model's ability to infer possible movements out of the data distribution. Though this kind of augmentation may introduce bias w.r.t. user real check-ins, it also increases the model robustness by reducing a user's check-in uncertainty and dislike check-ins. Meanwhile, this promising result provides a new perspective of improving next movement prediction performance. We respectively note that the trajectory permutation methods offered in this work are different from the previous trajectory generation methods [43,44] that mostly focus on protecting geo-privacy of the trajectory data.

Finally, we can observe a significant performance drop of SML-PD, which suggests the importance of modeling movement periodicity as has been widely observed in the literature [19,23].

4.2. Application II: Trajectory-user linking

To explore the generalization of our SML to other LBSN problems, we turn to address another important mobility task — trajectory classification (or trajectory–user linking [17]), which is an essential application for mining human individual mobility patterns. Specifically, we also leverage the similar trajectory augmentation modules to alleviate the training trajectories' sparsity

Table 6
The statistics of datasets.

Dataset	M[a]	$\|D_l\|/\|D'\|$[b]	$\|\mathcal{K}\|$[c]	$\bar{\mathcal{R}}$[d]	$\mathcal{T}_{max}$[e]
Gowalla	201	9,920/10,048	10,958	219	131
Brightkite	92	9,920/9,984	2,123	471	184
Foursquare	300	13,181/13,129	6,146	162	33

[a] M: the number of users.

[b] $|D_l|/|D'|$: the trajectories for training and testing.

[c] $|\mathcal{K}|$: the number of different check-ins.

[d] $\bar{\mathcal{R}}$: the average trajectory length.

[e] $\mathcal{T}_{\max}$: the maximum trajectory length.

Table 7
The statistics of the original and augmented training data for trajectory–user linking.

Dataset	Original training data	Augmented data
Gowalla	9,920	7,841
Brightkite	9,920	12,634
Foursquare	13,181	11,362

issue by integrating the spatio-temporal influence. Besides, TUL aims to maximize the difference among those trajectories generated by different users while minimizing the difference among those trajectories generated by the same users. To this end, given a sub-trajectory from the training dataset, we consider sampling a trajectory from the same user as a positive instance and randomly choose the negative instances generated by other users. Consequently, we tackle the TUL problem based on the contrastive trajectory learning and call the model SML-TUL. Theoretically, this method maximizes the mutual information between the given anchor trajectory and its positive instance. For comparison, we name the model using the trajectory data augmentation but without contrastive mobility learning as TUL-DA.

4.2.1. Experimental settings

Datasets. For fairly comparison, we evaluate all methods on three real-world LBSN datasets: Gowalla [45], Brightkite [45], and Foursquare [46]. Moreover, we select 201 and 92 users and corresponding trajectories from Gowalla and Brightkite following previous works [17,22]. For Foursquare, we use the data from the most popular city – New York – for evaluation, which consists of 300 users and 26,310 trajectories. Table 6 shows the statistics of the three datasets, and Table 7 summarizes the original and augmented training data for pre-training.

Baselines. We select several representative benchmarks addressing the TUL problem as baselines. These methods are deep human mobility learning models, including:

- *RNN-based TUL* [17], including **TULER-LSTM**, **TULER-GRU**, and **Bi-TULER**, are the basic TUL solutions that explore human mobility patterns for trajectory classification using various RNN models.
- *GAN-based TUL* [47] methods utilize an adversarial network to augment the trajectory data by approximating the inherent distribution of the user mobility trajectories. In this paper, we model **TGAN-G** and **TGAN-L** which use GRU and LSTM as the building models, respectively.
- *MoveSim* [48] is a self-attention based sequential modeling network for capturing the temporal transitions in human mobility. It incorporates the prior knowledge regarding human mobility and utilizes generative adversarial learning for trajectory pre-training.
- *Variational TUL* [7] is coupled with variational inference, which aims at learning a low-dimensional space for each sparse trajectory. We accordingly choose recent **TULVAE** [7] to learn the latent mobility representation and classify the trajectories.

In addition to the full version of our method SML-TUL, we also propose a simplified method called **TUL-DA**, which leverages the proposed trajectory augmentation to integrate the spatio-temporal influence from sparse trajectories without using contrastive trajectory pre-training. As a result, we can focus on evaluating the impact of the trajectory augmentation module for the TUL problem.

Metrics & hyper-parameters. Following previous works [7,17], we choose *ACC*@1, *ACC*@5, macro-*P*, macro-*R*, and macro-F_1 as metrics to evaluate the TUL performance of all approaches. As for the hyper-parameter settings, the learning rate of all methods is initialized with 0.00095. The dropout rate is set as 0.5, and the batch size is 64. Besides, we follow previous work [7] to embed each POI in a 250-dimensional vector using the word2vec method and use 300 units to construct the RNN architecture. In the end, the batch size of the contrastive mobility learning is 128, and both the positive samples and the negative samples in each training iteration are set to 10.

4.2.2. Performance comparison

Table 8 summarizes the model performance comparisons between our approaches and the baselines, where, similarly, the best result is shown in **bold** and the second best is underlined.

First, TUL-DA, which only leverages the trajectory augmentation, outperforms the benchmarks in most scenarios, demonstrating that trajectory augmentation is capable of providing more meaningful trajectories to enhance the model performance by alleviating the data sparsity problem. TULVAE is a standard semi-supervised method that needs to use both training and testing trajectories for mobility distribution estimation, which requires significantly more computational overhead for stochastic inference. In contrast, TUL-DA only needs existing training data to make trajectory augmentation, which can considerably reduce the risk of data leakage and computational cost. Interestingly, we can find that TUL-DA shows superiority over TULVAE in most cases, which indicates that using trajectory augmentation instead of variational inference provides a new perspective to approximate and fit the inherent distribution of mobility data.

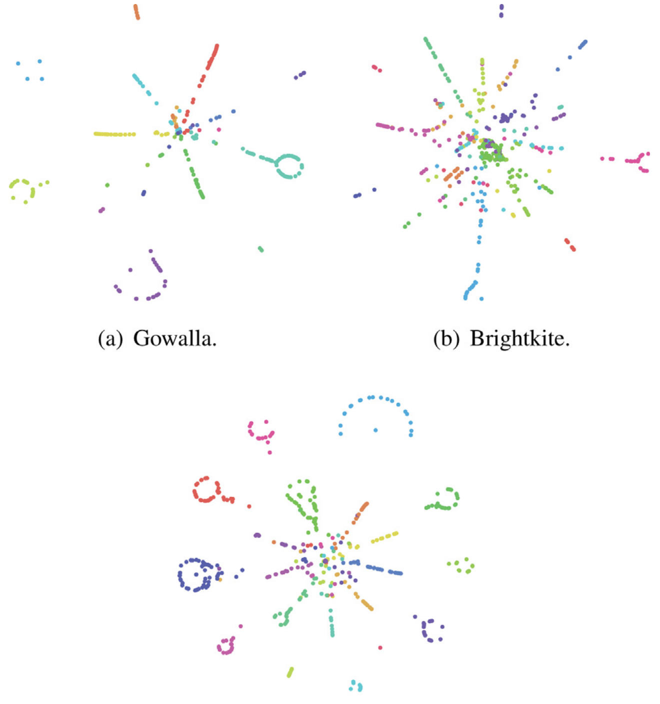
Next, our contrastive mobility learning model SML-TUL consistently performs better than previous methods in terms of all metrics across all datasets, which indicates that our approach can successfully learn to discriminate the difference of trajectories from different users. The performance advantage is achieved by the contrastive mobility learning that captures the underlying similarity of trajectories from the same user and the mobility difference between different people. Compared to previous TUL methods, SML-TUL applies self-supervised signals to enhance human mobility representations for the TUL task, which incorporates auxiliary pre-training objectives to explore intrinsic correlations among trajectories from the same users by maximizing the mutual information. This result also suggests that the self-supervised pre-training effectively improves the performance of the human mobility learning model for location-based services.

4.2.3. Visualization

Even though the human mobility semantics learned by the SML can improve the location-based services, one could argue that the improvements are achieved by the essential spatio-temporal influence learning, rather than the mobility representation captured by the contrastive trajectory learning. However, we posit that our SML encourages the mobility representations from different users to be distinguished that emphasizes the advantages of the proposed model.

Table 8
TUL performance comparison among different methods on three datasets.

Dataset	Method	Metric				
		ACC@1	*ACC*@5	macro-*P*	macro-*R*	macro-F_1
Gowalla	TULER-LSTM	41.24%	56.88%	31.70%	28.60%	30.07%
	TULER-GRU	40.85%	57.41%	29.52%	27.80%	28.64%
	Bi-TULER	41.95%	57.58%	32.15%	31.66%	31.90%
	TGAN-G	43.61%	60.99%	33.81%	32.93%	33.36%
	TGAN-L	43.79%	60.98%	33.25%	32.88%	33.06%
	MoveSim	43.62%	60.25%	33.06%	32.37%	32.71%
	TULVAE	45.40%	62.39%	36.13%	34.71%	35.41%
	TUL-DA	45.64%	61.10%	35.89%	35.54%	35.71%
	SML-TUL	**45.71%**	**63.98%**	**36.47%**	**35.83%**	**36.15%**
Brightkite	TULER-LSTM	43.01%	59.84%	38.45%	35.81%	37.08%
	TULER-GRU	44.03%	61.36%	38.86%	36.47%	37.62%
	Bi-TULER	43.54%	60.68%	38.20%	36.47%	37.31%
	TGAN-G	45.80%	64.08%	40.22%	37.57%	38.85%
	TGAN-L	45.69%	63.43%	41.88%	37.82%	39.74%
	MoveSim	45.75%	63.95%	42.63%	38.39%	40.40%
	TULVAE	45.98%	64.84%	43.15%	39.65%	41.33%
	TUL-DA	46.19%	64.39%	43.11%	39.92%	41.46%
	SML-TUL	**47.19%**	**64.98%**	**44.85%**	**40.55%**	**42.60%**
Foursquare	TULER-LSTM	51.22%	59.11%	47.19%	44.23%	45.66%
	TULER-GRU	50.91%	58.87%	47.18%	44.12%	45.60%
	Bi-TULER	53.88%	61.44%	50.35%	47.22%	48.73%
	TGAN-G	52.84%	60.63%	49.41%	46.04%	47,67%
	TGAN-L	53.00%	60.68%	49.45%	46.19%	47.76%
	MoveSim	53.62%	61.25%	49.88%	46.97%	48.38%
	TULVAE	54.28%	61.96%	50.06%	48.42%	49.22%
	TUL-DA	55.27%	65.59%	52.19%	50.07%	51.10%
	SML-TUL	**57.23%**	**66.07%**	**53.38%**	**51.95%**	**52.66%**


(a) Gowalla.

(b) Brightkite.

(c) Foursquare.

Fig. 7. Visualization of the learned latent space trained by SML. Each dot denotes a trajectory and the same colored trajectories are from the same user. We show 20 users for a better visualization.

Towards that, we use t-SNE [49] to plot the latent space of trajectories. Specifically, we randomly select 20 users and their corresponding trajectories from each dataset. The learned latent embedding of each trajectory is projected to the 2D space. Fig. 7 plots the learned embeddings of trajectories on three datasets, where we can observe an apparent clustering effect of trajectories. This means that our model can effectively disentangle the trajectories generated by different people, which is essential for downstream tasks such as trajectory classification. This result also implies that a better representation with good discriminability is critical for human mobility learning.

5. Related work

There is a large body of works learning human mobility representation and predicting future user mobility. In the sequel, we overview the most related studies in the literature.

5.1. Human mobility learning

Human mobility learning has attracted considerable attention in the community due to its critical impacts on urban planning and our quality of life. Pioneering studies focus on displacement patterns and heavy-tailed distribution of distances [11–13,50].

The scale-free random walks, a.k.a. Lévy flight, is usually used to explain individual motion mechanisms. Existing studies have found that urban human mobility is characterized by a high degree of regularity and hence of predictability [11,51], which inspires many subsequent works on learning and predicting human dynamics [1]. Earlier studies use stochastic methods such as Markov chains (MC) and hidden Markov models (HMM) [15,52,53] or ensemble learning models [1] to estimate the transition probability between consecutive locations and sequential visiting patterns based on the past trajectories.

Recent advances in neural networks have motivated considerable deep human mobility learning models, ranging from location prediction/recommendation [19,22,50,59] and human trace classification [17], to urban flow/traffic forecasting [60–62] and trajectory generation [47]. These approaches are mainly built on recurrent neural networks (RNN) and their variants such as long-short term memory networks (LSTM) [63] and GRU [31]

Table 9
Summary of the main works on next location prediction and trajectory classification in recent years. MF: matrix factorization; MC: Markov chain; CF: collaborative filtering; HMM: hidden Markov model; VAE: variational autoencoder; KG: knowledge graph. ACC: accuracy; RMSE: root mean square error; AUC: Receiver operating characteristic; MRR: mean reciprocal rank; MAP: mean average precision.

Methods	Model	Evaluation	Spatial	Temporal	Periodicity	Sequential	POI category	Contrastive
PMF [37]	MF	RMSE	×	✓	×	×	×	×
Markov Chain [54]	MC	ACC	×	✓	×	✓	×	×
FPMC [55]	MC, MF	ACC, Precision, Recall	×	✓	×	✓	×	×
SLoP [53]	CF	ACC	×	✓	×	✓	×	×
FPMC-LR [52]	MC, MF	ACC, Precision, Recall	✓	✓	×	✓	×	×
HMTP [15]	HMM	AUC	×	✓	×	✓	×	×
PRME [40]	MF	Precision, Recall	✓	✓	×	✓	×	×
ST-RNN [16]	RNN	Precision, F1, AUC, MAP	✓	✓	×	✓	×	×
POI2Vec [18]	word2vec	Precision, Recall	✓	✓	×	✓	×	×
TULER [17]	LSTM, GRU	ACC, Precision, Recall	✓	✓	×	✓	×	×
TULVAE [7]	LSTM, GRU, VAE	ACC, Precision, Recall	✓	✓	×	✓	×	×
TGAN [47]	LSTM, GRU, GAN	ACC, Precision, Recall	✓	✓	×	✓	×	×
HST-LSTM [20]	LSTM	ACC, MRR	✓	✓	×	✓	×	×
ARNN [56]	KG, LSTM, Attention	ACC	✓	✓	×	✓	×	×
Flashback [23]	LSTM, Attention	ACC	✓	✓	✓	✓	×	×
ASPPA [21]	LSTM, Attention	ACC, MRR	✓	✓	×	✓	×	×
CEM [57]	CNN	ACC, MAP	✓	✓	✓	✓	×	×
STGN [58]	LSTM	ACC, MAP	✓	✓	✓	✓	×	×
DeepMove [19]	LSTM, Attention	ACC	✓	✓	✓	✓	×	×
VANext [22]	CNN, Variational Attention	ACC, AUC, MAP	✓	✓	✓	✓	×	×
NEXT [1]	Decision tree	ACC, Precision	✓	✓	✓	✓	×	×
SML-LP (this work)	GRU, Attention, CL	ACC, AUC, MAP	✓	✓	✓	✓	✓	✓
SML-TUL (this work)	GRU, Attention, CL	ACC, Precision, Recall	✓	✓	✓	✓	✓	✓

which are used to capture the movement regularities in people's historical trajectories. In contrast to MC and HMM, RNN-based approaches can capture long-term transition dependencies while being able to handle large-scale mobility data. Recently, Li et al. [64] conducts comprehensive investigation on the predictability of different methods and find that RNN-based models significantly outperform MC-based approaches on human mobility prediction.

Nonetheless, vanilla RNNs are deterministic and ignore spatio-temporal characteristics of user check-ins, which requires auxiliary techniques to improve model performance. For example, DeepMove [19] utilizes an attention network to discriminate the importance of POIs, which can query historically similar movement episodes for predicting the current mobility. VANext [22] addresses the issue of movement uncertainty by introducing variational autoencoder [65] to pre-train user trajectories and alternatively uses the 1-D convolutional networks (CNN) to replace RNN, which can largely accelerate the training efficiency. Kong et al. [20] propose a hierarchical LSTM model that combines spatio-temporal influences into LSTM to alleviate the data sparsity issue. Chen et al. [36] also employ a context-aware RNN to capture mobility regularity and periodicity and design a co-attention scheme for social and temporal context learning. Flashback [23] is a retrospect method that matches similar mobility patterns in historical trajectories, which is very similar to DeepMove and VANext except that Flashback performs subsequence query in the hidden states of RNN. The main results on two human mobility learning tasks studied in this paper are summarized in Table 9.

Despite the promising results on a range of human mobility prediction tasks, previous methods still suffer from the sparse check-in and implicit feedback issues, due to the noisy and weak supervision signals inherent in the people's footprints left in the online social networks. Though spatio-temporal contexts such as displacement distance and time intervals can be explicitly modeled in RNNs, they rely on sequential dependencies to train the model that may not fully reflect the user motion patterns due to the sporadic check-in behaviors. In addition, existing studies consider the next location as the training objective without discriminating the negative POIs, which may not fully explore the spatio-temporal semantics in people's travel data. In contrast, we present a novel human mobility learning model that can distill informative signals from the sparse mobility while explicitly discriminating the positive POIs from the confused negative POIs. Besides, we present a mobility data permutation method allowing us to sample more meaningful but unobserved transitions, which not only enriches the sparse trajectory data but also provide extra supervision for improving movement representation and prediction. Our method borrows the idea from recent advances in self-supervised learning, enabling our model to consider the spatio-temporal constraints and the latent conditional dependencies, and, more importantly, automatically infer individual preference and motion patterns via maximizing the mutual information between movements episodes and the full trajectory.

5.2. Self-supervised learning

Deep learning relies heavily on large human-curated labeled data, which, however, is expensive and requires expert knowledge for semantic annotations. Typical unsupervised learning (e.g., clustering) aims to leverage the amount of unlabeled data, but its capability is limited due to the lack of supervisory signals during the model training. As a form of unsupervised learning, self-supervised learning (SSL) [24] targets at mining knowledge from unlabeled data via addressing proxy objectives supervised by the labels extracted from the data itself. Recent empirical studies demonstrate that the representations learned in a SSL manner can generalize well to downstream tasks even without using the downstream labels, and have achieved remarkable successes in many domains, including image classification [27,38,66], natural language understanding [67], reinforcement learning [68], speech recognition [69], recommender systems [29,70], video understanding [71], etc. For example, recent SSL models such as SimCLR [38], MoCo [27], and BYOL [39] achieve comparable and even better performance in image recognition compared to the supervised counterparts . The basic idea behind these approaches is to design data-specific pretext tasks and learn representations by maximizing positive pairs' similarities while minimizing similarities of negative pairs, rather than training an instance classifier in typical supervised learning models.

Most SSL works follow the contrastive learning paradigm and usually utilize the InfoNCE loss [26] or its variants [27,38] as the training objectives. Contrastive predictive coding [26] is widely adopted for measuring the model performance on discriminating the real data from the negative samples, which is equivalent to maximizing a lower bound of the mutual information between the positive data samples and unrelated data. Our model also utilizes the InfoNCE objective for self-supervised learning. However, we propose a novel method for augmenting the sparse spatio-temporal human mobility data that can efficiently leverage supervisions extracted from the data itself. In this vein, we initiate the attempts to study the spatio-temporal data in a SSL manner.

6. Conclusion

This paper presents a self-supervised mobility learning framework to analyze user movements and check-in behaviors. To our knowledge, it is among the first study that provides a way to predict and classify human mobility by developing self-supervisions from the massive trajectory data and large-scale user activities. As we mentioned before, deep neural networks are efficient for motion pattern mining due to the ability to learn complex relations and sequential dependencies in human check-ins. Still, existing approaches confront data sparsity issues and lack sufficient supervision signals for learning efficient trajectory representation. Our framework can help alleviate such problems to efficiently achieve self-supervised mobility learning objectives, such as predicting users' future behavior by maximizing the mutual information between the user's future movements and her historical trajectories. We conducted extensive experiments on two location-based tasks to verify the effectiveness of the proposed framework. The experimental results demonstrate that our method can significantly improve the mobility prediction and classification performance compared to previous approaches. Overall, our proposed framework is useful and can help the mobility learning model understand the similarity and differences between different trajectories.

As our future work, we plan to further improve the trajectory representations by investigating more mobility contexts such as check-in uncertainty and venues' hierarchy. In addition, designing a better method for choosing the negative samples, especially those hard to be distinguished, is of special interest, which may further improve the robustness of the contrastive trajectory learning. Last but not least, extending our method to other location-based services such as POI recommendation and traffic flow prediction are our ongoing works.

CRediT authorship contribution statement

Fan Zhou: Conceptualization, Methodology, Data curation, Writing. **Yurou Dai:** Experiments, Validation, Visualization. **Qiang Gao:** Conceptualization, Experiments, Resources. **Pengyu Wang:** Methodology, Visualization, Review & editing. **Ting Zhong:** Funding acquisition, Resources, Review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant No. 62072077 and No. 61602097).

References

[1] C. Comito, NexT: a framework for next-place prediction on location based social networks, Knowl.-Based Syst. 204 (2020) 106205.
[2] Y. Si, F. Zhang, W. Liu, An adaptive point-of-interest recommendation method for location-based social networks based on user activity and spatial features, Knowl.-Based Syst. 163 (2019) 267–282.
[3] G. Zhao, P. Lou, X. Qian, X. Hou, Personalized location recommendation by fusing sentimental and spatial context, Knowl.-Based Syst. 196 (2020) 105849.
[4] C.-Y. Tsai, B.-H. Lai, A location-item-time sequential pattern mining algorithm for route recommendation, Knowl.-Based Syst. 73 (C) (2015) 97–110.
[5] Q. Hao, L. Chen, F. Xu, Y. Li, Understanding the urban pandemic spreading of COVID-19 with real world mobility data, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2020, pp. 3485–3492.
[6] F. Zhou, X. Xu, G. Trajcevski, K. Zhang, A survey of information cascade analysis: Models, predictions, and recent advances, ACM Comput. Surv. 54 (2) (2021) 27:1–27:36.
[7] F. Zhou, Q. Gao, G. Trajcevski, K. Zhang, T. Zhong, F. Zhang, Trajectory-user linking via variational autoencoder, in: International Joint Conference on Artificial Intelligence, 2018, pp. 3212–3218.
[8] Z.E. Abou Elassad, H. Mousannif, H. Al Moatassime, A proactive decision support system for predicting traffic crash events: A critical analysis of imbalanced class distribution, Knowl.-Based Syst. 205 (2020) 106314.
[9] X. Lu, L. Bengtsson, P. Holme, Predictability of population displacement after the 2010 Haiti earthquake, Natl. Acad. Sci. 109 (29) (2012) 11576–11581.
[10] Y. Zhang, P. Siriaraya, Y. Kawai, A. Jatowt, Predicting time and location of future crimes with recommendation methods, Knowl.-Based Syst. 210 (2020) 106503.
[11] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 6 (1) (2017) 12.
[12] D. Brockmann, L. Hufnagel, T. Geisel, The scaling laws of human travel, Nature 439 (7075) (2006) 462–465.
[13] M.C. González, C.A. Hidalgo, A.-L. Barabási, Understanding individual human mobility patterns, Nature 453 (7196) (2008) 779–782.
[14] W. Mathew, R. Raposo, B. Martins, Predicting future locations with hidden Markov models, in: International Conference on Ubiquitous Computing, 2012, pp. 911–918.
[15] S. Qiao, D. Shen, X. Wang, N. Han, W. Zhu, A self-adaptive parameter selection trajectory prediction approach via hidden Markov models, IEEE Trans. Intell. Transp. Syst. 16 (1) (2014) 284–296.
[16] Q. Liu, S. Wu, L. Wang, T. Tan, Predicting the next location: A recurrent model with spatial and temporal contexts, in: International Joint Conference on Artificial Intelligence, 2016, pp. 194–200.
[17] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, F. Zhang, Identifying human mobility via trajectory embeddings, in: International Joint Conference on Artificial Intelligence, 2017, pp. 1689–1695.
[18] S. Feng, G. Cong, B. An, Y.M. Chee, POI2Vec: Geographical latent representation for predicting future visitors, in: AAAI Conference on Artificial Intelligence, 2017, pp. 102–108.
[19] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, D. Jin, DeepMove: Predicting human mobility with attentional recurrent networks, in: The World Wide Web Conference, 2018, pp. 1459–1468.
[20] D. Kong, F. Wu, HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction, in: International Joint Conference on Artificial Intelligence, 2018, pp. 2341–2347.
[21] K. Zhao, Y. Zhang, H. Yin, J. Wang, K. Zheng, X. Zhou, C. Xing, Discovering subsequence patterns for next POI recommendation, in: International Joint Conference on Artificial Intelligence, 2020, pp. 3216–3222.
[22] Q. Gao, F. Zhou, G. Trajcevski, K. Zhang, T. Zhong, F. Zhang, Predicting human mobility via variational attention, in: The World Wide Web Conference, 2019, pp. 2750–2756.
[23] D. Yang, B. Fankhauser, P. Rosso, P. Cudre-Mauroux, Location prediction over sparse user mobility traces using RNNs: Flashback in hidden states! in: International Joint Conference on Artificial Intelligence, 2020, pp. 2184–2190.
[24] X. Liu, F. Zhang, Z. Hou, Z. Wang, L. Mian, J. Zhang, J. Tang, Self-supervised learning: Generative or contrastive, vol. 1, no. 2, 2020, arXiv preprint arXiv:2006.08218.
[25] M.U. Gutmann, A. Hyvärinen, Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics, J. Mach. Learn. Res. 13 (1) (2012) 307–361.
[26] A. van den Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, 2018, arXiv preprint arXiv:1807.03748.
[27] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: International Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.

[28] L. Kong, C. de Masson d'Autume, L. Yu, W. Ling, Z. Dai, D. Yogatama, A mutual information maximization perspective of language representation learning, in: International Conference on Learning Representations, 2020.

[29] K. Zhou, H. Wang, W.X. Zhao, Y. Zhu, S. Wang, F. Zhang, Z. Wang, J.-R. Wen, S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization, in: ACM International Conference on Information & Knowledge Management, 2020, pp. 1893–1902.

[30] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: International Conference on Learning Representations, 2013.

[31] J. Chung, C. Gulcehre, K.H. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014, arXiv preprint arXiv:1412.3555.

[32] X. Liu, Y. Liu, K. Aberer, C. Miao, Personalized point-of-interest recommendation by mining users' preference transition, in: ACM International Conference on Information & Knowledge Management, 2013, pp. 733–738.

[33] Q. Yuan, G. Cong, Z. Ma, A. Sun, N.M. Thalmann, Time-aware point-of-interest recommendation, in: International Conference on Research and Development in Information Retrieval, 2013, pp. 363–372.

[34] L. Pappalardo, F. Simini, S. Rinzivillo, D. Pedreschi, F. Giannotti, A.-L. Barabási, Returners and explorers dichotomy in human mobility, Nature Commun. 6 (1) (2015) 1–8.

[35] D. Yao, C. Zhang, J. Huang, J. Bi, SERM: A recurrent model for next location prediction in semantic trajectories, in: ACM International Conference on Information & Knowledge Management, 2017, pp. 2411–2414.

[36] Y. Chen, C. Long, G. Cong, C. Li, Context-aware deep model for joint mobility and time prediction, in: International Conference on Web Search and Data Mining, 2020, pp. 106–114.

[37] A. Mnih, R.R. Salakhutdinov, Probabilistic matrix factorization, Adv. Neural Inf. Process. Syst. 20 (2007) 1257–1264.

[38] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple frame work for contrastive learning of visual representations, in: International Conference on Machine Learning, 2020, pp. 1597–1607.

[39] J. Grill, F. Strub, F. Altché, C. Tallec, P.H. Richemond, E. Buchatskaya, C. Doersch, B.Á. Pires, Z. Guo, M.G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, M. Valko, Bootstrap your own latent: A new approach to self-supervised learning, in: Advances in Neural Information Processing Systems, 2020.

[40] S. Feng, X. Li, Y. Zeng, G. Cong, Y.M. Chee, Q. Yuan, Personalized ranking metric embedding for next new POI recommendation, in: International Joint Conference on Artificial Intelligence, 2015, pp. 2069–2075.

[41] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations, 2015.

[42] Y. Kalantidis, M.B. Sariyildiz, N. Pion, P. Weinzaepfel, D. Larlus, Hard negative mixing for contrastive learning, in: Advances in Neural Information Processing Systems, 2020.

[43] V. Bindschaedler, R. Shokri, Synthesizing plausible privacy-preserving location traces, in: IEEE Symposium on Security and Privacy, 2016, pp. 546–563.

[44] K. Ouyang, R. Shokri, D.S. Rosenblum, W. Yang, A non-parametric generative model for human trajectories, in: International Joint Conference on Artificial Intelligence, 2018, pp. 3812–3817.

[45] E. Cho, S.A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2011, pp. 1082–1090.

[46] D. Yang, D. Zhang, V.W. Zheng, Z. Yu, Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs, IEEE Trans. Syst. Man Cybern.: Syst. 45 (1) (2015) 129–142.

[47] F. Zhou, R. Yin, G. Trajcevski, K. Zhang, J. Wu, A. Khokhar, Improving human mobility identification with trajectory augmentation, GeoInformatica (2019) 1–31.

[48] J. Feng, Z. Yang, F. Xu, H. Yu, M. Wang, Y. Li, Learning to simulate human mobility, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2020 pp. 3426–3433.

[49] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).

[50] Z. Yao, Y. Fu, B. Liu, W. Hu, H. Xiong, Representing urban functions through zone embedding with human mobility patterns, in: International Joint Conference on Artificial Intelligence, 2018.

[51] M. Luca, G. Barlacchi, B. Lepri, L. Pappalardo, Deep learning for human mobility: a survey on data and models, 2020,

[52] C. Cheng, H. Yang, M.R. Lyu, I. King, Where you like to go next: Successive point-of-interest recommendation, in: International Joint Conference on Artificial Intelligence, 2013, pp. 2605–2611.

[53] D. Lian, V.W. Zheng, X. Xie, Collaborative filtering meets next check-in location prediction, in: International Conference on World Wide Web, 2013, pp. 231–232.

[54] S. Gambs, M.-O. Killijian, M.N.n. del Prado Cortez, Next place prediction using mobility Markov chains, in: Workshop on Measurement, Privacy, and Mobility, 2012.

[55] S. Rendle, C. Freudenthaler, L. Schmidt-Thieme, Factorizing personalized markov chains for next-basket recommendation, in: International Conference on World Wide Web, 2010, pp. 811–820.

[56] Q. Guo, Z. Sun, J. Zhang, Y.-L. Theng, An attentional recurrent neural network for personalized next location recommendation, in: AAAI Conference on Artificial Intelligence, vol. 34, 2020, pp. 83–90.

[57] M. Chen, Y. Zuo, X. Jia, Y. Liu, X. Yu, K. Zheng, CEM: A convolutional embedding model for predicting next locations, IEEE Trans. Intell. Transp. Syst. (2020).

[58] P. Zhao, A. Luo, Y. Liu, F. Zhuang, J. Xu, Z. Li, V.S. Sheng, X. Zhou, Where to go next: A spatio-temporal gated network for next poi recommendation, IEEE Trans. Knowl. Data Eng. (2020).

[59] A. Rossi, G. Barlacchi, M. Bianchini, B. Lepri, Modelling taxi drivers' behaviour for the next destination prediction, IEEE Trans. Intell. Transp. Syst. 21 (7) (2019) 2980–2989.

[60] J. Zhang, Y. Zheng, D. Qi, Deep spatio-temporal residual networks for citywide crowd flows prediction, in; International Joint Conference on Artificial Intelligence, 2017, pp. 1655–1661.

[61] F. Zhou, L. Li, K. Zhang, G. Trajcevski, Urban flow prediction with spatial–temporal neural ODEs, Transp. Res. C 124 (2021) 102912.

[62] F. Zhou, Q. Yang, T. Zhong, D. Chen, N. Zhang, Variational graph neural networks for road traffic prediction in intelligent transportation systems, IEEE Trans. Ind. Inf. 17 (4) (2021) 2802–2812.

[63] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[64] H. Li, F. Lin, X. Lu, C. Xu, G. Huang, J. Zhang, Q. Mei, X. Liu, Systematic analysis of fine-grained mobility prediction with on-device contextual data, IEEE Trans. Mob. Comput. (2020).

[65] D.P. Kingma, M. Welling, Auto-encoding variational bayes, in: International Conference on Learning Representations, 2014.

[66] F. Zhou, C. Cao, T. Zhong, J. Geng, Learning meta-knowledge for few-shot image emotion recognition, Expert Syst. Appl. (2021) 114274.

[67] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the NAACL-HLT, 2019, pp. 4171–4186.

[68] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, G. Brain, Time-contrastive networks: Self-supervised learning from video, in: IEEE International Conference on Robotics and Automation, 2018, pp. 1134–1141.

[69] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, Y. Bengio, Multi-task self-supervised learning for robust speech recognition, in: International Conference on Acoustics, Speech and Signal Processing, 2020, pp. 6989–6993.

[70] J. Ma, C. Zhou, H. Yang, P. Cui, X. Wang, W. Zhu, Disentangled self-supervision in sequential recommenders, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2020, pp. 483–491.

[71] A. Owens, A.A. Efros, Audio-visual scene analysis with self-supervised multisensory features, in: European Conference on Computer Vision, vol. 11210, 2018, pp. 639–658.