

# One Model, Many Cities: A Transferable Social Relationship Inference Framework for Human Mobility Data

Chen Chu

University of Southern California  
Los Angeles, California, USA  
chenchu@usc.edu

Emmanuel Tung

Novateur Research Solutions  
Ashburn, Virginia, USA  
etung@novateur.ai

Cyrus Shahabi

University of Southern California  
Los Angeles, California, USA  
shahabi@usc.edu

Khurram Shafique

Novateur Research Solutions  
Ashburn, Virginia, USA  
kshafique@novateur.ai

## Abstract

Inferring social relationships from mobility data is crucial for many applications because it reflects real-world connections among people. However, large-scale trajectory datasets with ground-truth social ties are exceedingly scarce, making it difficult to train deep models for relationship inference. To address this gap, we propose a transferable social relationship inference framework that can be trained on one high-quality, labeled dataset and then generalized to new datasets, even from different cities.

Our framework rests on the key insight that social bonds depend largely on the frequency of individual meetings and the popularity of those meeting locations, both of which can be inferred statistically from raw trajectory data, irrespective of the underlying geographic semantics. It comprises two main modules: 1) Universal Social Relationship Classifier (USRC): A model trained to infer social relationships from trajectory data, and 2) Spatial Embedding Transfer (SET): A location embedding alignment technique that adapts new datasets to the pre-trained USRC model.

By aligning location embeddings, SET module enables the pre-trained USRC to interpret previously unseen datasets without extra supervision. Experiments on five public datasets demonstrate that our method achieves state-of-the-art performance in zero-shot social relationship inference, surpassing other unsupervised, and in some cases, even supervised, approaches. Additionally, the SET module significantly improves location embedding alignment, outperforming existing baseline methods. The source code and data are available at <https://github.com/chuchen2017/SET>.

## CCS Concepts

- Information systems → Geographic information systems.

---

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

SIGSPATIAL '25, Minneapolis, MN, USA

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 978-1-4503-XXXX-X/2018/06  
<https://doi.org/10.1145/3748636.3762710>

## Keywords

Social Relationship Inference, Human Mobility Modeling, Embedding Alignment, Transfer Learning, Domain Adaptation

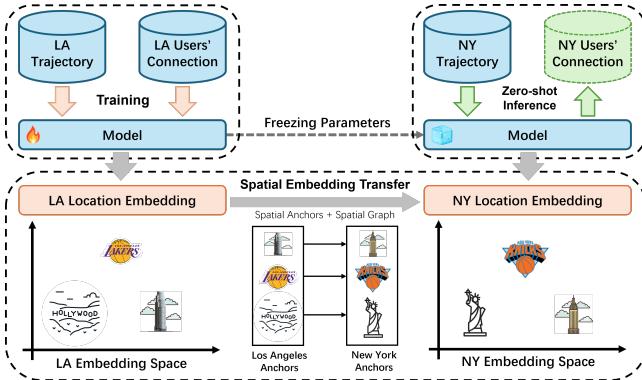
### ACM Reference Format:

Chen Chu, Cyrus Shahabi, Emmanuel Tung, and Khurram Shafique. 2025. One Model, Many Cities: A Transferable Social Relationship Inference Framework for Human Mobility Data. In *The 33rd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '25), November 3–6, 2025, Minneapolis, MN, USA*. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3748636.3762710>

## 1 Introduction

Individuals' social activities greatly influence their mobility behaviors, for example, friends often visit the same places together [6, 8, 10, 33]. As previous studies show [3, 16, 24–26], social ties between users can be accurately inferred using mobility data. Inferring social relationships from mobility data is critical because it captures the physical connections people form in the real world. This knowledge not only supports traditional online social network applications, such as targeted advertising, recommendation systems, and social or cultural studies, but also enables unique use cases. For example, they can help identify unknown members of criminal or terrorist organizations or model disease transmission through human contact in epidemiology. Additionally, they have been studied in the context of location privacy, where social ties can be inferred from co-location data, potentially compromising user anonymity [3].

However, large-scale datasets that include both mobility data and ground-truth social connections for the same individuals are extremely rare. This scarcity is partly due to privacy constraints limiting access to mobility data and partly because social network graphs and mobility records are rarely available for the same set of people. Consequently, only a small number of organizations can access both datasets, making it difficult to train deep models for relationship inference. Most existing methods[14, 25, 26, 30] either rely on supervised learning with explicit social connection labels or adopt unsupervised approaches based on manually defined features or location semantics, thereby limiting their generalizability. Yet, as shown in [23, 24], social ties often hinge on meeting frequency and location popularity—both of which can be derived from raw trajectory data independently of the underlying geography. Despite this, many existing approaches remain tailored to the specific datasets



**Figure 1: An illustration of how the spatial embedding transfer framework enables transferability across datasets.**

on which they were trained, inhibiting their ability to generalize across different regions [16–18].

While various deep learning strategies—such as contrastive learning [4], transfer learning [12], and meta-learning [29]—have been explored to enhance the generalizability of human mobility models [7, 15, 35], to the best of our knowledge, no work has specifically addressed generalizability for social relationship inference from trajectory data.

Beyond model-based transfer learning, embedding space alignment—a foundational technique for the rapidly evolving field of multimodal learning—offers a more generalizable framework for various downstream models and tasks [13, 27].

Therefore, in this paper, we employ spatial embedding alignment to generalize social inference models. To illustrate our high-level idea, consider the example in Figure 1. First, a model is trained on a labeled dataset, termed a source dataset, following a standard supervised learning manner; in the figure, the source dataset is from Los Angeles (LA). To enable cross-dataset transferability, spatial embedding transfer methods are applied to align the location embedding distribution of a target dataset, New York (NY), in our example, with that of LA. Since the model is trained on LA’s embedding space, an effective embedding transfer method ensures that the transformed NY embeddings remain interpretable by the pre-trained LA model. Intuitively, landmarks in NY (e.g., the Statue of Liberty) are mapped to locations in the LA embedding space where similar landmarks (e.g., the Hollywood sign) exist. Since these landmarks have comparable popularity (and perhaps similar semantics for the purposes of social inference) in the two datasets, they are expected to contribute similarly to the inference of social ties. Our spatial embedding transfer enables pre-trained models to be transferable across different geographic regions by matching key spatial and contextual relationships in the location embedding spaces irrespective of the underlying geographic semantics.

To realize the above idea, we introduce a transferable social relationship inference framework that is initially trained on a source dataset (e.g., LA) and then generalized to a target dataset (e.g., NY) by aligning the target location embeddings with those of the source. Our framework comprises two main components—a Universal Social Relationship Classifier (USRC) model and a Spatial Embedding

Transfer (SET) module—and is designed to enable zero-shot inference on target datasets without social relationship labels. USRC is a deep learning inference model that takes two trajectories—each belonging to a different individual—and predicts their social relationship. After USRC is trained on the source dataset, its location embedding layer serves as the “source embedding distribution” for transferability across datasets. Our SET module then aligns the location embedding features of target (unlabeled) datasets with this distribution.

Our key contribution lies in the alignment step, which integrates both SET and USRC to transfer knowledge across locations. First, SET identifies anchor points in LA and NY by matching locations based on the similarity of their ranked location popularity. For instance, NY’s most popular location is paired with LA’s, inheriting the pre-trained USRC embedding from LA. This reflects the insight that the likelihood of social connection depends on where people meet—co-visit at a popular landmark like the Empire State Building is generally less indicative of friendship than co-visit at a private residence.

Next, we feed sequences of NY embeddings, derived from user co-visits, pairwise and masked, into a USRC-based contrastive learning framework, freezing the USRC parameters from LA. The goal is to fine-tune the embeddings so they also capture co-visit patterns, leveraging USRC’s pre-trained knowledge from LA. In sum, this alignment transfers both the “popularity” and “co-visit” features of each location (captured through its embedding) from LA to NY.

Across five datasets, our framework demonstrates exceptional generalization ability. Notably, when trained on a richer source dataset and tested on a different target dataset, it outperforms a model trained and tested on target dataset itself—improving the evaluation metric by 14.4%. This underscores the value of leveraging more comprehensive training data for transfer learning.

Our contributions in this study can be summarized as follows:

- A transferable social relationship inference framework that infers social relationships between users based on their trajectory data and can perform zero-shot inference on unlabeled datasets.
- A novel spatial embedding transfer method. The method is designed to align location embedding distributions of two cities based on their human mobility patterns. The aligned embedding space is interpretable by the pre-trained model while preserving the unique characters of the target dataset.
- Through experiments on five real-world datasets, we demonstrated the strong generalizability of our social relationship inference model. Our SET module outperformed other embedding alignment baselines, and our zero-shot inference results also surpassed those of other unsupervised friendship inference models.

## 2 Related Work

**Social relationship inference:** Discovering real-world connections among people by analyzing their mobility data is a critical aspect of modern trajectory data mining. EBM [23, 24], among the earliest methods, introduced an entropy-based framework that uses Rényi entropy to capture co-visit patterns and Shannon entropy to capture location popularity. It showed that popularity and co-visit

patterns alone are sufficient for revealing social ties, making additional location semantics unnecessary for improving relationship inference. Walk2friend [3] applies graph embedding techniques to infer social relationships. Followed by a series of graph-based methods such as MRGAN [25], Heter-GCN [30], SRINet [26], and HMGCL [17]. These methods generally combine social network and location graphs together to form a heterogeneous graph and apply graph neural networks (GNN) to learn the embedding features of each user and location. By computing the similarity between users, the relationship can be scored and inferred. However, these studies typically require evaluation datasets with ground truth friendship data to initiate their heterogeneous graph approaches, which significantly limits their generalizability.

**Transfer Learning:** Transfer learning encompasses a wide range of sub-fields, including domain adaptation [36], domain generalization [34], embedding alignment [1], and meta-learning [20]. Although each subfield has its own assumptions and focus areas, they all share the common goal of transferring knowledge from one domain or task to another. In the context of spatial data, which often suffers from uneven label distribution across regions, transferring knowledge from one area to another is a promising approach to address this imbalance and enable generalization to unseen regions.

Currently, most spatial transfer learning methods focus on domain adaptation and meta-learning. For example, AdaTM [36] and DastNet [28] utilize domain adaptation techniques to enhance predictions in regions with limited data. Similarly, CHAML [5] and MetaStore [20] apply metalearning strategies to tasks such as the next POI recommendation and the placement of the stores. While these methods are grounded in transfer learning principles, they still rely on labeled data from the target region. Furthermore, the models used in these tasks, such as traffic prediction and POI recommendation, are often auto-regressive, meaning the transferred knowledge serves more as an augmentation to existing models rather than enabling training a transferable model from scratch. In this research, we propose a novel spatial transfer learning framework based on embedding alignment. Our method does not require labeled data from the target region and supports various alignment techniques, making it flexible and broadly applicable.

**Embedding alignment:** Embedding space alignment plays key roles in tasks like cross-lingual word translation [1], knowledge graph entities alignment [32], and graph analytics [9]. The fundamental concept is to find a mapping function that gives the optimal matching solution to map one distribution to another [13]. Unsupervised learning methods are commonly used for alignment, with the Wasserstein Generative Adversarial Network (GAN) being one of the foundational approaches [2]. For graph alignment, WAlign [11] employs graph neural networks as graph encoders and uses a Multilayer Perceptron as the discriminator in a Wasserstein GAN to learn to estimate the Wasserstein distance between two distributions. Once the discriminator is trained, it guides the adversarial training of the graph encoders, enabling them to map the target distribution to the source distribution effectively. SANA improves the method by involving the graph augmentation methods in the training process [21]. HyperAlign also involves the graph augmentation method and introduces contrastive learning to obtain more compatible graph structures [9].

Unlike well-studied graph alignment and cross-lingual word alignment algorithms, which are typically applied to target and source datasets with ground truth alignment labels, alignment between different geographical regions lacks such labels. As a result, it must be learned in a fully unsupervised manner, and its effectiveness can only be evaluated through the performance of downstream tasks. For spatial alignment, Takahiro et al. propose a spatial embedding alignment method based on anchor points. The hierarchical batch anchoring method is proposed to select anchor points according to the total number of visits in each urban area, and they then utilize an affine transformation matrix to learn the mapping function between distributions [31]. However, their anchor-based alignment relies solely on popularity, ignoring other insights from trajectory data. Moreover, affine transformation only guarantees alignment among anchor points, leaving its generalization to non-anchor points uncertain.

### 3 Preliminary

**Trajectory:** In this paper, we denote the trajectory of a user  $u$  in length  $l$  as a sequence of visit points  $Traj_u = \{pt_i\}_{i=1}^l$ , where  $pt_i = (loc, t)$ , with  $loc$  is represented by a place id and  $t$  indicates the timestamp of the visit.

**Co-visit sequence:** When two users meet at the same place and time, i.e., their trajectories intersect at a set of points  $pt_i$ , we represent the sequence of these intersections as their co-visit sequence, the time threshold of the intersection is denoted as  $\tau$ . We can formulate the co-visit sequence of user  $n$  and user  $m$  in length  $l$  as  $CV_{n,m} = \{pt_i | pt_i.loc \in Traj_{u_m}, |pt_i.t_{u_n} - pt_i.t_{u_m}| \leq \tau\}_{i=1}^l$ .

**Co-location sequence:** Beyond shared visits at the same time, individuals who visit the same location at different times can also indicate a potential social connection. In [8], this is referred to as a “fellowship” relationship, where one user’s visit influences the other to go to the same location later. Therefore, we formulate the co-location sequence of user  $n$  and user  $m$  in length  $l$  as  $CL_{n,m} = \{(loc, fn, fm)_i | loc_i \in Traj_{u_n}, loc_i \in Traj_{u_m}\}_{i=1}^l$ , where  $fn = Count(loc_i, Traj_{u_n})$  denote the visit frequency of  $loc_i$  of user  $n$ .

**Source and Target dataset:** This research focuses on creating a generalizable method for inferring social relationships by training a model on one (source) dataset and applying it to another (target) dataset. The *labeled* dataset used for training is referred to as the **source**, denoted as  $D_S$ , while the *unlabeled* dataset to which the model is later applied is called the **target**, denoted as the  $D_T$ . Accordingly, the corresponding embedding spaces are termed the source and target location embedding spaces, respectively.

**Social relationship inference:** The social relationship inference based on trajectory data is defined as follows. We are given a set of  $N$  users, each associated with a trajectory  $Traj_{u_i}$  in a dataset  $D = \{Traj_{u_i}\}_{i=1}^N$ . We also have a social connection matrix  $Y_{N \times N} \in \{0, 1\}$  where  $Y_{n,m} = 1$  indicates that users  $u_n$  and  $u_m$  are friends, and  $Y_{n,m} = 0$  otherwise. The task is to learn a inference function  $\mathcal{F}(Traj_{u_n}, Traj_{u_m}) \rightarrow \hat{y}_{n,m}$ , which takes the trajectories of two users as input and outputs a prediction  $\hat{y}_{n,m}$  indicating the social connectivity of two people.

**Transferable social relationship inference:** Suppose we have two datasets,  $S$  and  $T$ , each containing user trajectories, denoted as

$D_S$  and  $D_T$ . We also have a social network matrix  $Y_S$  for dataset  $S$  that indicates which users are friends. However, no such ground-truth social network information is available for dataset  $T$ . The goal is to leverage the labeled dataset (with  $D_S$  and  $Y_S$ ) to infer the social network  $\hat{Y}_T$  in dataset  $T$  (where  $D_T$  only is provided).

Our Approach is to first learn a model  $\mathcal{F}_S$  that maps from trajectories in  $D_S$  to social connections:  $\mathcal{F}_S : D_S \rightarrow Y_S$ . Next, we fine-tune the trained model  $\mathcal{F}_S$  on the unlabeled trajectories  $D_T$  to obtain a new model  $\bar{\mathcal{F}}_T$ . Finally, we apply  $\bar{\mathcal{F}}_T$  to  $D_T$  to predict the social network  $\hat{Y}_T$ .  $\bar{\mathcal{F}}_T : D_T \rightarrow \hat{Y}_T$  where  $\hat{Y}_T$  represents the inferred social connections among users in the dataset  $T$ .

## 4 Methodology

In this section, we first propose a social connection inference model named Universal Social Relationship Classifier (USRC) and then propose the Spatial Embedding Transfer (SET) module to make USRC transferable across different cities and datasets.

### 4.1 Universal Social Relationship Classifier

To make the model transferable across different regions, we design a model that only uses location data to infer the social relationship between two users.

Firstly, the model takes the trajectory of user  $n$  and  $m$  as input, then the model preprocesses their trajectories to extract the Co-visit sequence and Co-location sequence of two users. After that we employed three independent encoders to learn from their trajectories, co-visit and co-location. The USRC model can be formulated as follows,

$$\text{USRC}(\text{Traj}_{u_n}, \text{Traj}_{u_m}, \text{CL}_{n,m}, \text{CV}_{n,m}) \rightarrow \hat{y}_{n,m} \quad (1)$$

Before learning from the sequence data, an embedding module is employed to map location IDs into a continuous feature space.

$$e_i = \text{Emb}(loc_i) \quad (2)$$

$$\text{Emb}(loc_i) = \theta_{\text{Emb}}^T \cdot \text{onehot}(loc_i) \quad (3)$$

where  $loc_i$  denotes the location id to be embedded,  $e_i$  denotes the feature representation of location  $i$ .  $\text{onehot}$  is the one-hot function, which maps the location id to a one-hot representation  $\text{onehot}(loc_i) \in R^{|l|}$ , where  $|l|$  is the number of locations in the current dataset.  $\theta_{\text{Emb}} \in R^{|l| \times d}$  is the trainable parameter of the location embedding function, where  $d$  is the location embedding feature dimension. The location embedding function is trainable by downstream tasks, which makes the embedding feature of each location contains both the essential information of the current location (e.g., its popularity) and relationships with other locations.

After mapping the locations into the feature space, we can use a deep neural network to learn from a sequence of locations. Since the order of co-visit and co-location locations does not matter, we employ two Transformer Encoder networks without positional encoding to encode  $CL_{n,m}$  and  $CV_{n,m}$  and one additional Transformer Encoder to encode trajectories of two users.

$$cv = \text{max}||\text{Trans}_{cv}(\text{Emb}(CV_{n,m}))|| \quad (4)$$

$$cl = \text{mean}||\text{Trans}_{cl}(\text{Emb}(CL_{n,m}))|| \quad (5)$$

$$traj_n = \text{mean}||\text{Trans}_{traj}(\text{Emb}(\text{Traj}_{u_n}))|| \quad (6)$$

$$traj_m = \text{mean}||\text{Trans}_{traj}(\text{Emb}(\text{Traj}_{u_m}))|| \quad (7)$$

$$\hat{y}_{n,m} = \theta_y(\theta_{cl}cl + \theta_{cv}cv + \theta_t traj_n + \theta_t traj_m) \quad (8)$$

where  $\text{Trans}_{cl}$ ,  $\text{Trans}_{cv}$  and  $\text{Trans}_{traj}$  denote the Transformer Encoder for co-location, co-visit and trajectory, respectively,  $cl \in R^d$ ,  $cv \in R^d$  and  $traj_n \in R^d$  denote the feature representation of the co-location, co-visit sequence and trajectory of user  $n$ .  $\theta_{cl} \in R^{d' \times d}$ ,  $\theta_{cv} \in R^{d' \times d}$ ,  $\theta_t \in R^{d' \times d}$ ,  $\theta_y \in R^{1 \times d'}$  denote the weighting parameters of the model. The co-visit sequence between two users captures the most critical information about their social relationships. Therefore, we apply max pooling to extract the most salient features from the co-visit sequence for representation. On the other hand, while the co-location information and trajectory of each user serve as valuable supplements to co-visit data, especially given the general sparsity of co-visit data, the co-location and user trajectory sequence often contains irrelevant and redundant information. To address this, we employ mean pooling to aggregate and smooth the information within both the co-location sequence and trajectories of two users.

After modeling the co-visit, co-location, and trajectories of two users, we can now have a feature to represent the social relationship of two users  $\hat{y}_{n,m}$ . According to the label data that we have in the training dataset, we have  $y_{n,m} \in \{0, 1\}$ . We use an MSE loss function, treating social connection inference as a regression problem. Predictions  $\hat{y}_{n,m}$  near 1 indicate a higher likelihood of connection, while values near 0 suggest the opposite. Because there's no need to make a binary classification for evaluation, no threshold is required. Moreover, the smooth gradients from MSE help the frozen model adapt more effectively during the following contrastive learning.

Incorporating the temporal information (e.g., time or duration) for each co-visit would increase the complexity of the social inference model and complicate subsequent transfer learning. We did not include temporal info in the USRC, as its impact is suspected to be low.

### 4.2 Spatial Embedding Transfer (SET)

To enable USRC to transfer across datasets, we first freeze all its parameters once training on source data is complete. We then use our SET module to align the target location embeddings with those of the source, and finally replace USRC's frozen location embedding layer with these newly aligned embeddings. The key to transferability is how to map the embedding feature distribution of the target dataset to the source dataset properly. The SET module not only needs to align the target embedding distribution to the source space to be compatible with the frozen USRC, but it also needs to capture the unique co-location/visit patterns shown in the target dataset in a way the USRC can interpret. We propose the SET module to solve these two challenges in three steps: Spatial Initiation, Structural Matching, and Global Finetuning. The first two steps adapt the source embeddings to capture location popularity from the target dataset, while the final step incorporates the target's co-location and visit patterns. The framework of the module is shown in Figure 2.

**Spatial Initiation:** Instead of randomly initiating the location embedding of the target dataset, we propose a Spatial Initiation method to meet the assumption required by our Structural Matching method. We define spatial anchor points for this purpose. Prior

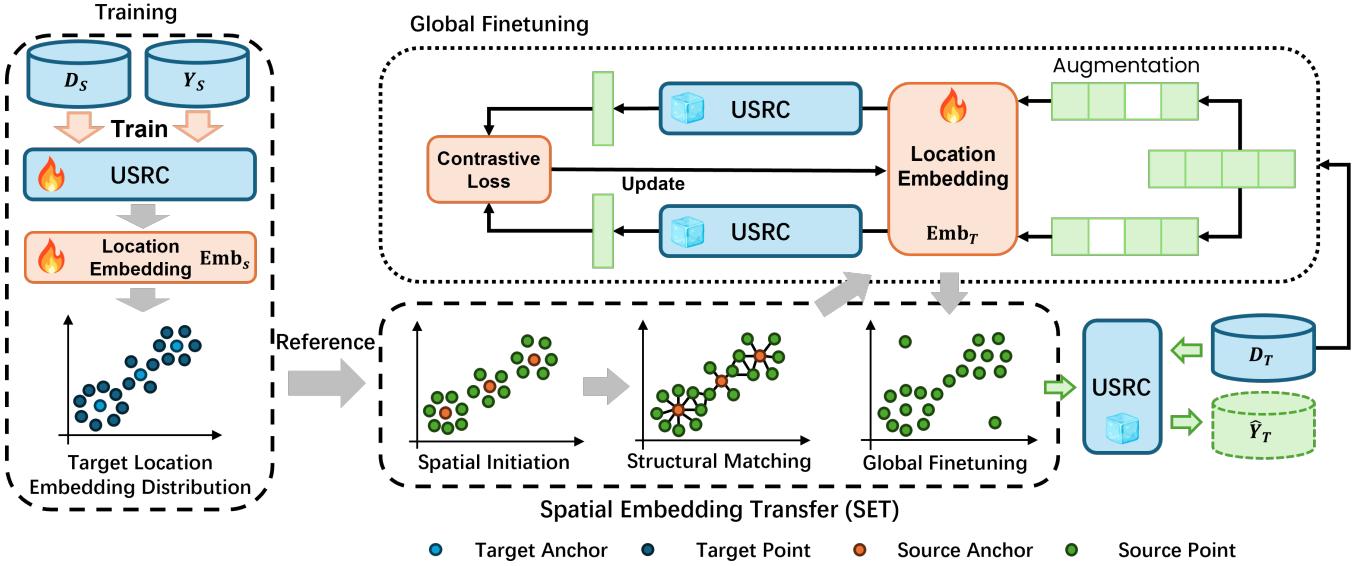


Figure 2: An illustration of the Spatial Embedding Transfer (SET) module.

research shows that different cities share similar visit-frequency patterns, meaning their most and least popular places maintain a consistent proportion across cities [31]. Therefore, sampling locations from two popularity distributions and pairing them as anchor points can effectively capture the structural similarities between cities. We use location entropy to compute popularity [22, 24]. By linear sampling from the ranked location entropy distribution, anchor points of each city are automatically selected. After pairing the ordered anchor points of the source and target cities, we obtain the correspondence relationship of location popularity between the two cities.

$$E_l = - \sum_{u, P_{u,l} \neq 0} P_{u,l} \log P_{u,l} \quad (9)$$

$$\text{Anchors}_S \xrightarrow{\mathbf{P}} \text{Rank}(E_{l_S}) \quad (10)$$

$$\text{Anchor}_T \xrightarrow{\mathbf{P}} \text{Rank}(E_{l_T}) \quad (11)$$

$$\text{Anchor}_{T \rightarrow S} = \text{zip}(\text{Anchor}_T, \text{Anchors}_S) \quad (12)$$

where  $P_{u,l}$  denotes the probability that a randomly picked check-in at location  $l$  belongs to user  $u$ .  $\text{Rank}(E_{l_S})$ ,  $\text{Rank}(E_{l_T})$  denote the ranked location entropy of source and target datasets  $S$  and  $T$ , respectively.  $\mathbf{P} = \alpha \cdot \min(|l_S|, |l_T|)$  denotes the number of locations sampled as anchor points; it represents  $\alpha$  percent of the locations from the dataset with fewer locations.  $\text{Anchors}_S$  and  $\text{Anchor}_T$  denote the anchor points set of datasets  $S$  and  $T$ , which are sampled sequentially in linear order based on their location entropy distributions. By pairing the corresponding anchor points between the two datasets, we obtain the dictionary  $\text{Anchor}_{T \rightarrow S}$ . The matched anchor points between two datasets indicate which places share similar popularity. We then use this information to initialize  $\text{Emb}_T$  by matching each location in the target dataset to a comparable location in the source dataset.

$$\text{Emb}_T[\text{loc}_i \in \text{Anchor}_t] = \text{Emb}_S[\text{Anchor}_{T \rightarrow S}[\text{loc}_i]] \quad (13)$$

Most of the time, the number of locations in two different datasets does not match exactly, and we also want to maintain flexibility during this phase. This means we don't want our initialized target embedding space to be exactly the same as the source one. Therefore, we select only a subset of locations as anchors and set the embeddings of non-anchor locations according to the following rules.

$$\begin{aligned} \text{Emb}_T[\text{loc}_i \notin \text{Anchor}_t] &= \\ \text{Emb}_S[\text{Anchor}_{T \rightarrow S}[\arg \min \text{dis}(\text{loc}_i, \text{Anchor}_T || \text{Graph}_T)] + \delta \quad (14) \end{aligned}$$

For each place that is not selected as an anchor in the initiation process, the  $\arg \min \text{dis}$  function finds its closest anchor point according to  $\text{Graph}_T$  distance and assigns its embedding according to that anchor point. Random noise  $\delta \sim \mathcal{N}(0, \sigma)$  is added to their embeddings to enhance diversity and promote regularization.  $\text{Graph}_T$  is essentially the OD (Origin-Destination) Matrix of  $T$ , a common tool in transportation planning, logistics, and urban mobility that shows the number of trips from each origin to each destination. Here, we derive these flows directly from trajectory data.

**Structural Matching:** We propose a structural fine-tuning module to further align the target location embeddings with the source embedding space. Once each target location is assigned a value from the source distribution,  $\text{Emb}_T$  occupies the same feature space as  $\text{Emb}_S$ . We then refine their alignment by incorporating OD flows derived from the trajectory data, ensuring a more precise matching of the two distributions. Algorithm 1 describes the Structural Matching approach to align target and source embedding distributions using the OD graph of the target dataset. In each iteration, the algorithm updates the embeddings for anchor points and their neighbors, guiding them toward the anchor point's nearest feature in the source embedding  $\text{Emb}_S$ . The flow on the edges in the graph

**Algorithm 1** Structural Matching

---

**Input:**  $Emb_T$ ,  $Emb_S$ ,  $Anchor_T$ ,  $Graph_T$ , Number of iterations  $\tau$ , Threshold  $\epsilon$ , Learning rate  $LR$

**Output:**  $Emb_r$

```

1: for each  $iter \in [1, \tau]$  do
2:   for each  $loc_i \in Anchor_T$  do
3:      $w = Emb_S[\arg \min \text{dis}(loc_i, Emb_S)]$ 
4:      $lr = LR \cdot (1 - \frac{iter}{\tau})$ 
5:     for each  $loc_j \in Graph_T[loc_i]$  do
6:       if  $Graph_T[loc_i, loc_j] > \epsilon$  then
7:          $Emb_T[loc_j] = lr \cdot Graph_T[loc_i, loc_j] \cdot (Emb_T[loc_j] - w)$ 
8:       end if
9:     end for
10:   end for
11: end for
12: return  $Emb_T$ 

```

---

serve as weights, influencing how far each node moves. The learning rate decreases over iterations, allowing for a gradual adjustment of the target embedding distribution.

**Global Finetuning:** After Structural Matching, we have aligned the location embedding distribution of the target dataset to the embedding space on which the USRC has been trained. However, each city or dataset has unique human mobility patterns, and these unique characters often play an important role in location representation. In this section, we propose an unsupervised location embedding finetuning method, which aims to finetune the target location embedding to learn the unique co-location/visit pattern of the target dataset and make it a better fit for the pre-trained model.

The source embedding distribution is trained using gradients passed from the USRC model, making the USRC the only model capable of interpreting the distribution. Therefore, to ensure that the fine-tuned target embedding remains suitable for the pre-trained model, the pre-trained USRC must be involved in the fine-tuning process.

Towards this end, we propose an embedding finetuning method based on contrastive learning. The pre-trained USRC takes augmented user co-location and co-visit sequences as input and outputs hidden representations of these augmented pairs. We only augment co-visit and co-location sequences and mask the trajectory embeddings to ensure the model zeroes in on the target dataset's unique co-location patterns—critical for social connection inference. Full trajectories contain excessive redundant information that isn't needed for this task. According to the similarity between output representations, location embeddings are trained to fit in the input space of USRC. The formal definition of the algorithm is as follows. Firstly, we define our location sequence augmentation function,

$$Aug(\{loc_i\}_{i=1}^l) = \{loc_i \neq loc_k\}_{i=1}^{l-1} \quad (15)$$

which randomly removes  $loc_k$  from  $loc_{(1,l)}$ . The augmentation aims to slightly modify a co-location or co-visit sequence by randomly removing one of the locations in the sequence. We only augment sufficiently long sequences where omitting a single location won't alter their overall meaning, ensuring that most of the original information is preserved. In Algorithm 2, we first replace the location

**Algorithm 2** Global Matching

---

**Input:**  $Emb_T$ , Pretrained USRC, Target Dataset  $D_T$ , Iteration  $\tau$ , Batch size  $bs$ , Temperature  $t$ , Learning rate  $lr$

**Output:**  $Emb_B$

```

1: USRC.Emb  $\leftarrow Emb_T$ 
2: for each  $iter \in [1, \tau]$  do
3:   sample minibatch  $\{(Traj_{u_n}, Traj_{u_m}, CL_{n,m}, CV_{n,m})\}_{i=1}^{bs}$ 
4:   for each  $k \in [1, bs]$  do
5:      $h_{2k-1} = \text{USRC}(\{Traj_{u_n}, Traj_{u_m}, Aug(CL_{n,m}), Aug(CV_{n,m})\}_k)$ 
6:      $h_{2k} = \text{USRC}(\{Traj_{u_n}, Traj_{u_m}, Aug(CL_{n,m}), Aug(CV_{n,m})\}_k)$ 
7:   end for
8:   for each  $i \in [1, 2bs]$  and  $j \in [1, 2bs]$  do
9:      $s_{ij} = \frac{h_i^\top h_j}{\|h_i\| \cdot \|h_j\|}$ 
10:    end for
11:     $l(i, j) = -\log \left( \frac{\exp(s_{ij}/t)}{\sum_{k=1, k \neq i}^{2bs} \exp(s_{ik}/t)} \right)$ 
12:    Loss  $= \frac{1}{2bs} \sum_{k=1}^{bs} [l(2k-1, 2k) + l(2k, 2k-1)]$ 
13:     $\min_{\text{USRC.Emb}} \text{Loss}(\text{USRC.Emb}, D_T)$ 
14:  end for
15: return  $Emb_T = \text{USRC.Emb}$ 

```

---

embedding layer of the pre-trained USRC with the embedding layer of the target dataset, to which we have applied Spatial Initiation and Structural Matching. Subsequently, we generate a contrastive learning pair using the augmentation function to the co-location or co-visit sequence in a training sample. USRC is employed to create the feature representation of the current sample. Next, we compute the contrastive loss, which aims to maximize the cosine similarity between augmented samples derived from the same co-location/visit pair while minimizing the similarity for negative pairs within the same batch. The gradient of the contrastive loss is used to update the embedding layer of USRC, specifically the location embedding corresponding to the target dataset.

Notably, while the training loss flows through USRC's parameters, they remain unchanged, preserving the model's source-domain knowledge. At the same time, the loss updates the embedding layer, teaching USRC to interpret the revised location embeddings. Consequently, the location embedding is fine-tuned into a distribution fully compatible with USRC.

## 5 Experiments

### 5.1 Datasets

For our experiments, we use two real-world publicly available user check-in datasets with labeled friendships, Gowalla and Foursquare [6, 33]. Users with check-in data and their friendship relationships in Los Angeles, New York and Stockholm(ST) are extracted from the datasets, forming five independent datasets. We then infer the relationships across all users within the same dataset.

To refine the quality of the dataset, consistent with prior studies using these datasets [24], we assumed that users without any co-location are unknown to each other and removed those user pairs from the dataset. Table 1 depicts the key statistics for each dataset. Owing to distinct check-in data collection methods, Gowalla and

Foursquare differ significantly, even within the same city—justifying our treatment of them as independent datasets.

**Table 1: Statistics of the datasets used**

Dataset	Gowalla_LA	Gowalla_NY	Gowalla_ST	Foursquare_LA	Foursquare_NY
# users	7,498	7,302	10,693	5,736	9,199
# locations	10,146	10,866	23,471	20,166	33,128
# check-ins	367,586	368,027	686,851	1,189,571	1,881,703
# friends	2,270	2,424	6,101	936	1,896
# un-friends	60,893	71,304	104,673	41,370	95,100

## 5.2 Baselines

We first compare the zero-shot inference performance of our method with other unsupervised social relationship inference baselines. Next, we evaluate our spatial embedding transfer method against other embedding alignment approaches.

**Unsupervised inference baselines:** The following baselines do not require label data for their evaluation. To the best of our knowledge, there are few unsupervised friendship inference models at this stage; the included two baselines are the most recent ones we could find.

**Walk2friend** [3]: Walk2Friend constructs a heterogeneous graph of users and locations. Through random walk and skip-gram embedding algorithms, it obtains user embeddings and infers relationships between pairs of users by computing their cosine similarities.

**Heter-GCN** [30]: Heter-GCN also relies on a heterogeneous graph structure, using an unsupervised loss to train a graph convolutional network that captures relationships among both heterogeneous and homogeneous nodes. The inference stage then compares user node embeddings to determine their similarity.

**Embedding alignment baselines:** We compare our SET module with other embedding alignment methods by evaluating how well the inference model uses each alignment for zero-shot prediction. **HBP** [31]: HBP applies a hierarchical batch anchoring method to select anchor points and then applies Procrustes alignment to compute the rotational matrix, which could map the target embedding distribution to that of the source.

**HBA** [31]: HBA applies the same anchoring method of HBP but uses the Affine alignment to align two distributions.

**LSNA** [19]: LSNA is a social network alignment method that utilizes graph convolution layers to learn structural information and then uses cross-network convolution to generate the aligned features of nodes in two graphs.

**SANA** [21]: SANA uses Graph Attention Network as the encoder and utilizes graph augmentation to refine the alignment relationship between nodes of different graphs.

**WAlign** [11]: WAlign uses a GCN to encode graphs and then applies Wasserstein GAN (WGAN) to train a generator to map the target distribution to the source distribution.

**HyperAlign** [9]: HyperAlign employs contrastive learning in graph embedding and uses graph augmentation to create hyperedges to augment the topological relationship. The alignment network is also trained based on WGAN.

LSNA, SANA, and HyperGAN are network alignment methods designed to find the optimal alignment solution for each individual node in the target and source graphs. That is, the model identifies

which node in the source graph is most similar to a given node in the target graph, considering the graph structure and node features. In our experiments, we used these methods to determine node-to-node alignment solutions. Subsequently, we assigned the embedding features of the nodes in the source embedding layer to their corresponding nodes in the target layer. HBP, HBA, SET, WAlign, and our method SET belong to embedding alignment methods, which aim to find a mapping function to directly map the target embedding distribution to that of the source.

## 5.3 Experiment Setting

In our experiments, we trained five USRC models, each on a different dataset, and evaluated each model on its own validation set corresponding to the same training dataset. Thus, these results represent the upper bound of USRC's performance for that dataset under a fully supervised setup. Subsequently, for each model, we apply various embedding alignment methods to map the other four target datasets' embeddings onto the source embedding space where the model was initially trained. The evaluation is based on the model's performance on the transferred datasets. We average each dataset's transfer performance from the other four to compare our zero-shot inference performance with other unsupervised friendship inference methods.

The embedding dimension for location embedding layers for all datasets is set as 256.  $\sigma$  is set as 0.5. For each dataset, the top 500 locations with the highest location entropy are selected as anchor points, followed by a linear sampling of 60% from the remaining locations. The Structural Matching learning rate is set as 0.1, and the number of iterations is set as 10. The global Global Finetuning learning rate is set as 0.001, the batch size is set as 4, and the number of iterations is set as 3.

## 5.4 Evaluation Metrics

To evaluate the prediction performance of all models, we employed two widely used metrics in social relationship inference: the Area Under the Precision-Recall Curve (PR-AUC) and the Area Under the Receiver Operating Characteristic Curve (ROC-AUC). Because the dataset has far more non-friend pairs than friend pairs, thus a significant class imbalance in the dataset, we use PR-AUC as our primary metric to evaluate performance across different datasets.

## 5.5 Experiment Results

In this section, we present the transfer inference result of our method, compare the result with other unsupervised social relationship inference models, and then compare our SET module with other spatial embedding transfer baselines.

Firstly, we present the transfer inference results of our method. Table 2 presents a matrix of PR-AUC results for our model across five datasets. Each row represents the dataset used for training, and each column indicates the dataset used for inference. The diagonal cells with underlines (where the model is trained and tested on the same dataset) correspond to the supervised scenario, while the off-diagonal cells show the zero-shot transfer results (i.e., the model is trained on one dataset but used for inference on another). For example, in the 1st column of Table 2, the 1st cell shows the performance of a USRC model trained and evaluated in a supervised manner

**Table 2: Cross-dataset PR-AUC results of our transferable social relationship inference model.**

		Evaluation datasets				
PR-AUC		Gowalla_LA	Gowalla_NY	Gowalla_ST	Foursquare_LA	Foursquare_NY
Training datasets	Gowalla_LA	<b>0.6548</b>	0.5270	0.7792	<b>0.5253</b>	<b>0.5677</b>
	Gowalla_NY	0.5307	<b>0.5446</b>	0.6891	0.4122	0.3801
	Gowalla_ST	0.5208	0.4193	<b>0.7961</b>	0.4466	0.5174
	Foursquare_LA	0.5834	0.4550	0.7302	<b>0.5007</b>	0.5414
	Foursquare_NY	0.5560	0.5122	0.6803	0.4739	<b>0.4962</b>

**Table 3: Performance comparison of unsupervised social relationship inference models**

Method	Gowalla_LA		Gowalla_NY		Gowalla_ST		Foursquare_LA		Foursquare_NY	
	PR-AUC	ROC-AUC								
Walk2friend	0.1823	0.7125	0.1323	0.6741	0.2433	<b>0.7306</b>	0.1033	0.7718	0.1086	0.7457
Heter-GCN	<b>0.3478</b>	<b>0.7728</b>	<b>0.2773</b>	<b>0.7619</b>	<b>0.3191</b>	0.7155	<b>0.2901</b>	<b>0.7913</b>	<b>0.3097</b>	<b>0.7835</b>
OURS	<b>0.5567</b>	<b>0.8249</b>	<b>0.4980</b>	<b>0.8871</b>	<b>0.7197</b>	<b>0.7736</b>	<b>0.4704</b>	<b>0.8178</b>	<b>0.4964</b>	<b>0.8747</b>

**Table 4: Zero-shot inference comparison of various spatial embedding transfer methods**

Method	Gowalla_LA		Gowalla_NY		Gowalla_ST		Foursquare_LA		Foursquare_NY	
	PR-AUC	ROC-AUC								
HBP	0.2470	0.6607	0.2614	0.6538	0.1659	0.5904	0.1477	0.5826	0.2108	0.6268
HBA	0.2864	0.7091	0.2539	0.6895	0.1492	0.5867	0.1382	0.5681	0.1928	0.6647
LSNA	0.2336	0.6567	0.3026	0.7131	0.1526	0.5882	0.1680	0.6399	0.1988	0.5917
SANA	0.1693	0.6271	0.1931	0.6415	0.1438	0.5653	0.1259	0.5293	0.1423	0.5638
WAlign	<b>0.3323</b>	<b>0.7503</b>	<b>0.3417</b>	<b>0.7419</b>	0.1541	0.5754	0.1606	0.5869	0.2315	0.6514
HyperGAN	0.2278	0.6672	0.2529	0.6738	<b>0.2005</b>	<b>0.6314</b>	<b>0.2097</b>	<b>0.6561</b>	<b>0.2821</b>	<b>0.6868</b>
SET	<b>0.5400</b>	<b>0.8640</b>	<b>0.4410</b>	<b>0.8154</b>	<b>0.4760</b>	<b>0.8055</b>	<b>0.5266</b>	<b>0.8332</b>	<b>0.5140</b>	<b>0.8254</b>

on the Gowalla\_LA dataset. The 2nd, 3rd, 4th, and 5th cells show the transfer result of our proposed framework: the SET module is applied to adapt the same USRC model, trained on Gowalla\_LA, to the Gowalla\_NY, Gowalla\_ST, Foursquare\_LA, Foursquare\_NY datasets.

As shown in Table 2, our approach provides outstanding transferability across both different cities and datasets. Notably, when trained on a high-quality dataset like Gowalla\_LA, it even outperforms the best supervised results on the Foursquare\_LA and Foursquare\_NY datasets.

Next, we compare our zero-shot inference results against other unsupervised inference baselines. For our method, we report the average zero-shot performance on each dataset, where the model has been transferred from each of the remaining four datasets. As shown in Table 3, our method significantly outperforms other baselines. The performance of walk2friend is highly dependent on random walks, which become inefficient on large datasets, leading to weak results. Heter-GCN relies on positive pairs selected through manually crafted features, causing poor generalization across various datasets. In contrast, our SET module automatically adapts to new datasets without predefined rules. Meanwhile, it avoids the computational overhead of graph embedding methods, resulting in better scalability. Finally, we compare our SET module to other embedding transfer methods. All approaches use the same transfer

framework and the same pre-trained USRC model, so the focus here is on the alignment quality of the embedding space. Table 4 reports each model’s average PR-AUC and ROC-AUC performance, using the dataset indicated in the table’s column for training. The values shown represent the average zero-shot results across the remaining four datasets for each column.

Table 4 shows that our transfer method consistently outperforms all other embedding alignment baselines across every dataset, and it is the only approach that achieves performance on par with supervised methods. Additionally, we observe that embedding mapping strategies (HBP, HBA, WAlign) generally outperform network alignment techniques (LSNA, SANA, HyperGAN), suggesting that an exact match between target and source distributions is not necessary; a more flexible mapping function can yield a more suitable target distribution.

The spatial and temporal granularity of human mobility datasets, as well as the quality of the social relationship network, are two important factors that influence the performance of the proposed transfer method. The Foursquare dataset has a much denser check-in frequency than Gowalla, therefore, the overall transfer performance between these two datasets is lower than that achieved within the same dataset across different cities. Meanwhile, the

**Table 5: Zero-shot inference comparison of variants of the SET module**

Method	Gowalla_LA		Gowalla_NY		Gowalla_ST		Foursquare_LA		Foursquare_NY	
	PR-AUC	ROC-AUC								
Random	0.1934	0.5953	0.1991	0.5805	0.1408	0.5599	0.1251	0.5463	0.1331	0.4969
Structural	0.2859	0.6843	0.2616	0.6551	0.1920	0.5982	0.3081	0.6906	0.2749	0.6734
Global	<u>0.4381</u>	<u>0.8342</u>	<u>0.3831</u>	<u>0.7777</u>	<u>0.2283</u>	<u>0.6202</u>	<u>0.3211</u>	<u>0.6861</u>	<u>0.3068</u>	<u>0.7243</u>
SET	<b>0.5400</b>	<b>0.8640</b>	<b>0.4410</b>	<b>0.8153</b>	<b>0.4760</b>	<b>0.8055</b>	<b>0.5266</b>	<b>0.8332</b>	<b>0.5140</b>	<b>0.8254</b>

Gowalla\_ST dataset contains a dense and high-quality social relationship network, resulting in the best transfer performance when used as either the target or the source city.

## 5.6 Ablation Study

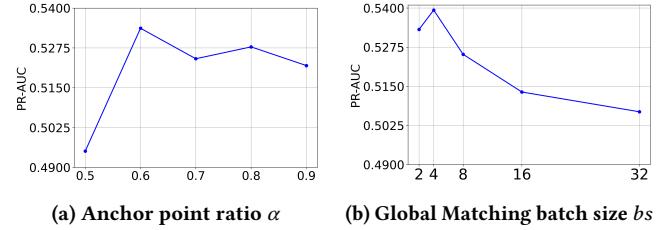
Table 5 presents an ablation study of our SET module components:

- Random – The USRC frozen model uses completely random location embeddings.
- Structural – The Global Finetuning module is removed, so embeddings rely only on Spatial Initiation and Structural Matching, capturing location popularity but not co-location/visit information.
- Global – Instead of using embeddings from Structural Matching, a randomly initialized layer is passed into the Global Finetuning module—representing the opposite scenario in which co-location/visit information is prioritized while location popularity alignment is omitted.

As observed from Table 5, models that take random embeddings as location features cannot perform well in social relationship inference because random embeddings cannot capture location information, such as popularity and connectivity. Therefore, although the frequency of co-visits and co-location contains information about whether two people are friends, where people meet plays a more critical role in inferring their social connectivity. Another observation is that both Structural Matching and Global Finetuning can successfully align location embeddings to the source space. This indicates that both location information (e.g., popularity, connectivity with other locations) and co-location patterns are important for social relationship inference. The high-quality alignment of SET module is achieved by integrating information from both aspects. Structural Matching provides fundamental location information, offering a suitable initialization for Global Finetuning. After the initialized embeddings capture essential mobility patterns shared across datasets, Global Finetuning refines the alignment by learning unique co-location patterns specific to the dataset, achieving the best matching.

## 5.7 Parameter Sensitivity

To analyze the effects of hyperparameters of the SET, we use a USRC model trained on the Gowalla\_LA dataset and transfer it to other datasets with different parameter settings. The results are compared by measuring the average performance of the model on the other datasets. We analyze two important hyperparameters of the SET:  $\alpha$ , the ratio of locations selected as anchor points during Spatial Initiation; and  $bs$ , the batch size used during Global Matching. The results are shown in figure 3.



**Figure 3: Overall model performance on different hyperparameter settings.**

As we can see from Figure 3(a), the alignment quality does not continue to improve as the number of anchor points increases. The best performance is achieved when 60% of the locations are selected as anchors, while the remaining 40% remain flexible. The flexibility preserved during the Spatial Initiation supports our assumption that different datasets have different embedding distributions and improves the effectiveness of the subsequent Structural and Global Matching.

As shown in Figure 3(b), the Global Matching module produces the most distinguishable location embedding distribution when the batch size is 4. In contrastive learning, a larger batch size typically provides more negative samples, which encourages the model to learn more generalizable embeddings. However, our objective is not to learn general-purpose representations, but rather to fine-tune the model to closely align with the specific embedding distribution of the pretrained USRC model. A smaller batch size facilitates this alignment by limiting the diversity of negative samples, allowing the model to focus on fine-grained distinctions that are more relevant to the USRC model-specific embedding space. In particular, it encourages the model to concentrate on matching the distributional characteristics of the pretrained embeddings rather than being regularized by a large number of uninformative negatives. Consequently, this leads to more specialized and discriminative representations tailored to our target task.

## 5.8 Computational Complexity

In this section, we analyze the computational complexity of SET. The complexity of Spatial Initiation is  $O(NL + L \log L + L^2 \log L)$ , where  $N$  denotes the number of users, and  $L$  denotes the number of locations. The complexity for Structural Matching is  $O(\tau k L^2)$ , where  $k$  denotes the average number of neighbors of a location in OD graph. The complexity of Global Matching is  $O(bs^2 N)$ . Since the number of epochs  $\tau$  and the batch size  $bs$  are generally small compared to the number of users and locations, they can be considered constants in this analysis. Therefore, the overall complexity

is:  $O(NL + L \log L + L^2 \log L + kL^2 + N)$ . Given that the alignment process is a one-time cost and eliminates the need to train a new friendship inference model, the time cost of the SET module is acceptable, especially compared to the time required to train a model from scratch.

We also consider the computational resources required during alignment. Graph-based embedding alignment methods, including LSNA, SANA, WAlign and HyperAlign, typically require the entire graph to be loaded into GPU memory for computation. From the dataset statistics, we know that the size of such a graph is  $L^2$ . Since the number of locations  $L$  is typically large for most datasets, the enormous memory cost of the backward propagation renders these methods difficult to use in practice. In our experiments, we found that it was not feasible to run these methods on a GPU with 80 GB of memory for the Foursquare\_NY dataset, which has 33,128 locations. In contrast, our method does not rely on full-graph computation. By breaking the alignment process into multiple steps, we significantly reduce memory requirements and achieve alignment with much lower GPU usage. In practice, the whole transfer process takes no more than 10 minutes on all experimental datasets.

## 6 Conclusion and Future Directions

In this paper, we proposed a transferable social relationship inference framework, which includes a trajectory-based social relationship inference model (USRC) and a spatial embedding transfer method (SET). The spatial embedding transfer is conducted in an unsupervised manner by leveraging information extracted from trajectory data. It maps the location embedding distribution of the target dataset to the source embedding space, which the relationship inference model has been trained on. This enables the model to transfer the knowledge it learned from a labeled dataset to unlabeled ones. Different from other domain adaptation methods that assume embeddings should be aligned perfectly, we build on our assumption that each city could have its own feature space distribution. This unique distribution can be learned from the human mobility pattern of the dataset. By leveraging the power of contrastive learning, we can approximate this distribution and boost the performance of the downstream model. Through experiments on five publicly available real-world datasets, our method demonstrates exceptional transferability. Notably, the zero-shot inference performance of the model, when trained on a high-quality dataset, surpasses that of the supervised inference methods. Our findings serve as a promising step toward the development of a geospatial foundation model, one that can be trained on large-scale, high-quality data while remaining highly generalizable across a broad range of datasets and downstream tasks.

Due to the sensitivity of location data and social relationship networks, current research is constrained by the limited availability of datasets with diverse granularities, such as CDR data or datasets including the specific type of social relationships (e.g., colleagues, friends, family). Exploring such datasets would be a valuable direction for future research. Moreover, incorporating richer contextual information into spatial transfer learning could enable modeling the correlation between spatial context and human mobility patterns, offering another promising research direction.

## Acknowledgments

This research has been funded in part by the NIH awards R01LM014026, NSF award CNS-2125530 and DMS-2428039, and the Intelligence Advanced Research Projects Activity (IARPA) via the Department of Interior/Interior Business Center (DOI/IBC) contract number 140D0423C0033. The U.S. Government is authorized to reproduce and distribute reprints for the Governmental purposes, notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NIH, NSF, IARPA, or the U.S. Government. In addition, this research project has benefited from the Microsoft Accelerate Foundation Models Research (AFMR) grant program, through which leading foundation models hosted by Microsoft Azure, along with access to Azure credits, were provided to conduct the research.

## References

- [1] David Alvarez-Melis and Tommi S. Jaakkola. 2018. Gromov-Wasserstein Alignment of Word Embedding Spaces. arXiv:1809.00013 [cs.CL] <https://arxiv.org/abs/1809.00013>
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 214–223. <https://proceedings.mlr.press/v70/arjovsky17a.html>
- [3] Michael Backes, Mathias Humbert, Jun Pang, and Yang Zhang. 2017. walk2friends: Inferring Social Links from Mobility Profiles. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (Dallas, Texas, USA) (CCS '17)*. Association for Computing Machinery, New York, NY, USA, 1943–1957. doi:10.1145/3133956.3133972
- [4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 1597–1607. <https://proceedings.mlr.press/v119/chen20j.html>
- [5] Yudong Chen, Xin Wang, Miai Fan, Jizhou Huang, Shengwen Yang, and Wenwu Zhu. 2021. Curriculum Meta-Learning for Next POI Recommendation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (Virtual Event, Singapore) (KDD '21)*. Association for Computing Machinery, New York, NY, USA, 2692–2702. doi:10.1145/3447548.3467132
- [6] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (San Diego, California, USA) (KDD '11)*. Association for Computing Machinery, New York, NY, USA, 1082–1090. doi:10.1145/2020408.2020579
- [7] Chen Chu, Hengcai Zhang, Peixiao Wang, and Feng Lu and. 2024. Simulating human mobility with a trajectory generation framework based on diffusion model. *International Journal of Geographical Information Science* 38, 5 (2024), 847–878. doi:10.1080/13658816.2024.2312199 arXiv:<https://doi.org/10.1080/13658816.2024.2312199>
- [8] David J. Crandall, Lars Backstrom, Dan Cosley, Siddharth Suri, Daniel Huttenlocher, and Jon Kleinberg. 2010. Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences* 107, 52 (2010), 22436–22441. doi:10.1073/pnas.1006155107 arXiv:<https://www.pnas.org/doi/pdf/10.1073/pnas.1006155107>
- [9] Manh Tuan Do and Kijung Shin. 2024. Unsupervised Alignment of Hypergraphs with Different Scales. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (Barcelona, Spain) (KDD '24)*. Association for Computing Machinery, New York, NY, USA, 609–620. doi:10.1145/3637528.3671955
- [10] Nathan Eagle, Alex (Sandy) Pentland, and David Lazer. 2009. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences* 106, 36 (2009), 15274–15278. doi:10.1073/pnas.0900282106 arXiv:<https://www.pnas.org/doi/pdf/10.1073/pnas.0900282106>
- [11] Ji Gao, Xiao Huang, and Jundong Li. 2021. Unsupervised Graph Alignment with Wasserstein Distance Discriminator. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (Virtual Event, Singapore) (KDD '21)*. Association for Computing Machinery, New York, NY, USA, 426–435. doi:10.1145/3447548.3467332

- [12] Renhe Jiang, Xuan Song, Zipei Fan, Tianqi Xia, Zhaonan Wang, Quanjun Chen, Zekun Cai, and Ryosuke Shibasaki. 2021. Transfer Urban Human Mobility via POI Embedding over Multiple Cities. *ACM/IMS Trans. Data Sci.* 2, 1, Article 4 (Jan. 2021), 26 pages. doi:10.1145/3416914
- [13] Alexander Kalinowski and Yuan An. 2020. A Survey of Embedding Space Alignment Methods for Language and Knowledge Graphs. arXiv:2010.13688 [cs.CL]. <https://arxiv.org/abs/2010.13688>
- [14] Jie Li, Fanzi Zeng, Zhu Xiao, Hongbo Jiang, Zhirun Zheng, Wenping Liu, and Ju Ren. 2020. Drive2friends: Inferring Social Relationships From Individual Vehicle Mobility Data. *IEEE Internet of Things Journal* 7, 6 (2020), 5116–5127. doi:10.1109/JIOT.2020.2974669
- [15] Lihuan Li, Hao Xue, Yang Song, and Flora Salim. 2024. T-JEPA: A Joint-Embedding Predictive Architecture for Trajectory Similarity Computation. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems* (Atlanta, GA, USA) (SIGSPATIAL '24). Association for Computing Machinery, New York, NY, USA, 569–572. doi:10.1145/3678717.3691271
- [16] Yongkang Li, Zipei Fan, and Xuan Song. 2025. Heterogeneous Hyperbolic Hypergraph Neural Network for Friend Recommendation in Location-based Social Networks. *ACM Trans. Knowl. Discov. Data* 19, 3, Article 57 (Feb. 2025), 29 pages. doi:10.1145/3708999
- [17] Yongkang Li, Zipei Fan, Du Yin, Renhe Jiang, Jinliang Deng, and Xuan Song. 2023. HMGCL: Heterogeneous multigraph contrastive learning for LBSN friend recommendation. *World Wide Web* 26, 4 (2023), 1625–1648.
- [18] Yongkang Li, Zipei Fan, Jixiao Zhang, Dengheng Shi, Tianqi Xu, Du Yin, Jinliang Deng, and Xuan Song. 2022. Heterogeneous Hypergraph Neural Network for Friend Recommendation with Human Mobility. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) (CIKM '22). Association for Computing Machinery, New York, NY, USA, 4209–4213. doi:10.1145/3511808.3557609
- [19] Zhehan Liang, Yu Rong, Chenxin Li, Yunlong Zhang, Yue Huang, Tingyang Xu, Xinghao Ding, and Junzhou Huang. 2021. Unsupervised Large-Scale Social Network Alignment via Cross Network Embedding. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management (Virtual Event, Queensland, Australia)* (CIKM '21). Association for Computing Machinery, New York, NY, USA, 1008–1017. doi:10.1145/3459637.3482310
- [20] Yan Liu, Bin Guo, Daqing Zhang, Djamel Zeghlache, Jingmin Chen, Sizhe Zhang, Dan Zhou, Xinlei Shi, and Zhiwen Yu. 2021. MetaStore: A Task-adaptive Meta-learning Model for Optimal Store Placement with Multi-city Knowledge Transfer. *ACM Trans. Intell. Syst. Technol.* 12, 3, Article 28 (April 2021), 23 pages. doi:10.1145/3447271
- [21] Jingkai Peng, Fei Xiong, Shirui Pan, Liang Wang, and Xi Xiong. 2023. Robust Network Alignment with the Combination of Structure and Attribute Embeddings. In *2023 IEEE International Conference on Data Mining (ICDM)*. 498–507. doi:10.1109/ICDM58522.2023.00059
- [22] Huy Pham and Cyrus Shahabi. 2016. Spatial influence - measuring followship in the real world. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. 529–540. doi:10.1109/ICDE.2016.7498268
- [23] Huy Pham, Cyrus Shahabi, and Yan Liu. 2013. EBM: an entropy-based model to infer social strength from spatiotemporal data. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data* (New York, New York, USA) (SIGMOD '13). Association for Computing Machinery, New York, NY, USA, 265–276. doi:10.1145/2463676.2465301
- [24] Huy Pham, Cyrus Shahabi, and Yan Liu. 2016. Inferring Social Strength from Spatiotemporal Data. *ACM Trans. Database Syst.* 41, 1, Article 7 (March 2016), 47 pages. doi:10.1145/2877200
- [25] Guangming Qin, Jianpeng Qi, Bin Wang, Guiyuan Jiang, Yanwei Yu, and Junyu Dong. 2024. Multi-relational graph attention network for social relationship inference from human mobility data. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24, Kate Larson (Ed.). International Joint Conferences on Artificial Intelligence Organization*. 2315–2323.
- [26] Guangming Qin, Lexue Song, Yanwei Yu, Chao Huang, Wenzhe Jia, Yuan Cao, and Junyu Dong. 2023. Graph Structure Learning on User Mobility Data for Social Relationship Inference. *Proceedings of the AAAI Conference on Artificial Intelligence* 37, 4 (Jun. 2023), 4578–4586. doi:10.1609/aaai.v37i4.25580
- [27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763. <https://proceedings.mlr.press/v139/radford21a.html>
- [28] Yihong Tang, Ao Qu, Andy H.F. Chow, William H.K. Lam, S.C. Wong, and Wei Ma. 2022. Domain Adversarial Spatial-Temporal Network: A Transferable Framework for Short-term Traffic Forecasting across Cities. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) (CIKM '22). Association for Computing Machinery, New York, NY, USA, 1905–1915. doi:10.1145/3511808.3557294
- [29] Yi Wang, Tongya Zheng, Yuxuan Liang, Shunyu Liu, and Mingli Song. 2024. COLA: Cross-city Mobility Transformer for Human Trajectory Simulation. In *Proceedings of the ACM Web Conference 2024* (Singapore, Singapore) (WWW '24). Association for Computing Machinery, New York, NY, USA, 3509–3520. doi:10.1145/3589334.3645469
- [30] Yongji Wu, Defu Lian, Shuwei Jin, and Enhong Chen. 2019. Graph Convolutional Networks on User Mobility Heterogeneous Graphs for Social Relationship Inference.. In *IJCAI*. 3898–3904.
- [31] Takahiro Yabe, Kota Tsubouchi, Toru Shimizu, Yoshihide Sekimoto, and Satish V. Ukkusuri. 2020. Unsupervised Translation via Hierarchical Anchoring: Functional Mapping of Places across Cities. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (Virtual Event, CA, USA)* (KDD '20). Association for Computing Machinery, New York, NY, USA, 2841–2851. doi:10.1145/3394486.3403335
- [32] Yuchen Yan, Lihui Liu, Yikun Ban, Baoyu Jing, and Hanghang Tong. 2021. Dynamic Knowledge Graph Alignment. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 5 (May 2021), 4564–4572. doi:10.1609/aaai.v35i5.16585
- [33] Dingqi Yang, Daqing Zhang, and Bingqing Qu. 2016. Participatory Cultural Mapping Based on Collective Behavior Data in Location-Based Social Networks. *ACM Trans. Intell. Syst. Technol.* 7, 3, Article 30 (Jan. 2016), 23 pages. doi:10.1145/2814575
- [34] Dazhou Yu, Guangji Bai, Yun Li, and Liang Zhao. 2022. Deep Spatial Domain Generalization. In *2022 IEEE International Conference on Data Mining (ICDM)*. 1293–1298. doi:10.1109/ICDM54844.2022.00170
- [35] Zheng Zhang, Hossein Amiri, Dazhou Yu, Yuntong Hu, Liang Zhao, and Andreas Züfle. 2024. Transferable Unsupervised Outlier Detection Framework for Human Semantic Trajectories. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems* (Atlanta, GA, USA) (SIGSPATIAL '24). Association for Computing Machinery, New York, NY, USA, 350–360. doi:10.1145/3678717.3691324
- [36] Yuhao Zheng, Jinyang Wu, Zihao Cai, Senzhang Wang, and Jianxin Wang. 2024. AdaTM: Fine-grained Urban Flow Inference with Adaptive Knowledge Transfer across Multiple Cities. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (Boise, ID, USA) (CIKM '24). Association for Computing Machinery, New York, NY, USA, 3424–3432. doi:10.1145/3627673.3679856

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009