Graph Structure Learning on User Mobility Data for Social Relationship Inference

Guangming Qin[1], Lexue Song[2], Yanwei Yu[1*], Chao Huang[3], Wenzhe Jia[1], Yuan Cao[1], Junyu Dong[1]
[1]College of Computer Science and Technology, Ocean University of China
[2]Department of Data Science, Duke Kunshan University
[3]Department of Computer Science, University of Hong Kong
qinguangming@stu.ouc.edu.cn, ls440@duke.edu, yuyanwei@ouc.edu.cn,
chaohuang75@gmail.com, jiawenzhe@stu.ouc.edu.cn, {cy8661,dongjunyu}@ouc.edu.cn

Abstract

With the prevalence of smart mobile devices and location-based services, uncovering social relationships from human mobility data is of great value in real-world spatio-temporal applications ranging from friend recommendation, advertisement targeting to transportation scheduling. While a handful of sophisticated graph embedding techniques are developed for social relationship inference, they are significantly limited to the sparse and noisy nature of user mobility data, as they all ignore the essential problem of the existence of a large amount of noisy data unrelated to social activities in such mobility data. In this work, we present **S**ocial **R**elationship **I**nference **Net**work (SRINet), a novel Graph Neural Network (GNN) framework, to improve inference performance by learning to remove noisy data. Specifically, we first construct a multiplex user meeting graph to model the spatial-temporal interactions among users in different semantic contexts. Our proposed SRINet tactfully combines the representation learning ability of Graph Convolutional Networks (GCNs) with the power of removing noisy edges of graph structure learning, which can learn effective user embeddings on the multiplex user meeting graph in a semi-supervised manner. Extensive experiments on three real-world datasets demonstrate the superiority of SRINet against state-of-the-art techniques in inferring social relationships from user mobility data. The source code of our method is available at https://github.com/qinguangming1999/SRINet.

Introduction

The advance in positioning technologies and the prevalence of location-based online services such as Uber, Location-Based Social Networks (LBSNs) such as Foursquare, and geo-tagged social media such as Twitter have generated massive human mobility data (Noulas et al. 2011). These mobility data provide us with the opportunity to deeply analyze human behaviors in different scenarios. Understanding the underlying human mobility patterns has been shown to benefit various spatio-temporal applications, *e.g.*, POI recommendation, urban planning, business site selection, traffic scheduling, and many more (Dai et al. 2021b, 2022, 2021a; Chen et al. 2022). To be specific, uncovering social relationships among users is crucial for many real-world problems, such as friend recommendation (Yang et al. 2022), advertisement targeting (Wang et al. 2011), and ride-sharing (Cici et al. 2014). Intuitively, there are certain correlations between social relationships and human mobility data (Cho, Myers, and Leskovec 2011). Using the correlations, several works (Eagle, Pentland, and Lazer 2009; Wang, Li, and Lee 2014; Pham, Shahabi, and Liu 2013, 2016) have been conducted to try to infer friendships between users from human mobility trajectory data.

Despite the effectiveness of existing social relationship inference methods (Wang, Li, and Lee 2014; Pham, Shahabi, and Liu 2013, 2016), most of them are based on meeting frequency and only focus on pairwise relationship inference individually. They work poorly for inactive users since they fail to solve the data sparsity challenge of inactive users. Recently, graph embedding (Perozzi, Al-Rfou, and Skiena 2014; Grover and Leskovec 2016; Liu et al. 2020) and graph convolutional networks (GCNs) (Kipf and Welling 2017; Veličković et al. 2018; Hamilton, Ying, and Leskovec 2017; Liu et al. 2021; Yu et al. 2022) have been developed for learning node representations in graph-structured data. A line of works based on network embedding and GCNs have been proposed, which no longer require hand-crafted features, but treat users as nodes and model the meeting events between users as a homogeneous graph (Yu, Wang, and Li 2018) or a heterogeneous graph (Backes et al. 2017; Wu et al. 2019). They leverage network embedding or GCNs to learn representations of nodes (users), which take into account the relation propagation among users and even consider the rich semantic information, and thus improving prediction performance. More recently, LBSN2Vec++ (Yang et al. 2022) and MSC-LBSN (Huynh et al. 2022) are proposed for friendship prediction based on an LBSN heterogeneous hypergraph consisting of four different data domains, *i.e.*, spatial, temporal, semantic and social domains. LBSN2Vec++ learns node embeddings from both friendship homogeneous edges and check-in heterogeneous hyperedges sampled from the LBSN heterogeneous hypergraph by a random-walk-with-stay scheme, while MSC-LBSN learns from friendship edges, check-in hyperedges, and node personas at the same time, and uses these to devise multiple representations for each user that respect their multiple roles in a social context.

*Corresponding author.

Although state-of-the-art works learn correlations between users in various contexts by designing complex graph embedding approaches to improve performance, they all ignore the essential problem: *does all human mobility data reflect social activities among users?* The answer is obviously not. For example, if a user goes shopping individually in supermarket, this check-in record does not reflect any social relationship. But in such a large public place, many users may check in, resulting in a mass of records being mistaken for social activities. On the other hand, the large amount of check-in records that are not related to social activities seriously hurt model inference performance (Yu, Wang, and Li 2018). Therefore, for social relationship inference, it is not necessary to use all human mobility data. *However, how to weed out these irrelevant check-ins (noisy data) from all mobility data remains a significant challenge.*

To tackle the aforementioned challenge, we develop a GNN model for social relationship inference, named SRINet, which is an iterative graph structure learning framework that can continuously remove noisy edges produced by social-irrelevant check-in records from the constructed multiplex user meeting graph. Specifically, we first construct a multiplex user meeting graph to model the spatio-temporal interactions among mobile users. To learn effective user representations, our SRINet effectively integrates GCNs and graph structure learning, which introduces graph sparsification constraint to learn topology filters to weed out noisy edges from the multiplex user meeting graph. Experimental results on three real-life mobility datasets show that our model significantly outperforms several strong baselines (**8.32**% ROCAUC and **10.19**% PRAUC gains on average) in social relationship inference task.

To summarize, we make the following contributions:

- We propose a novel method SRINet to improve social relationship inference performance by combining GCNs with graph structure learning to eliminate noisy data in user mobility data.
- We design a multiplex graph structure learning framework by applying graph sparsification constraint on the multiplex user meeting graph for better user representation learning.
- We conduct extensive evaluations on three real-life mobility datasets. Experimental results show that our model significantly outperforms state-of-the-art baselines by 5.35%$\sim$12.07% and 7.93%$\sim$16.34% improvements in terms of ROCAUC and PRAUC.

Related Work

In literature, there are a number of studies on how to infer social relationships using human mobility trajectories. Eagle, Pentland, and Lazer (2009) attempt to discover the correlation between meeting events and social relationships using mobile phone data. Li et al. (2011) find that the meeting time could indicate different types of relationships, *e.g.*, colleagues meeting during the daytime vs. friends meeting at night. Pham, Shahabi, and Liu (2013) propose an entropy-based model (EBM) that considers the diversity of meeting events and location entropy in friendship inference, which is further extended in (Pham, Shahabi, and Liu 2016) by incorporating location semantics and stay duration. Wang, Li, and Lee (2014) develop a unified framework (PGT) that takes personal background, global background, and temporal factor into consideration. Hsieh, Yan, and Li (2015); Hsieh and Li (2019) use graph features (*e.g.*, Jaccard and Katz) computed from a co-location graph to measure relationships for user pairs. *However, all methods above make pairwise relationship inference independently.*

Following the recent advances in graph embedding techniques (Grover and Leskovec 2016; Perozzi, Al-Rfou, and Skiena 2014), several embedding frameworks (Backes et al. 2017; Yu, Wang, and Li 2018; Zhou et al. 2018) have been proposed for mobility relationship inference. Backes et al. (2017) first apply graph embedding into social relationship inference. Specifically, they employ random walk based embedding on a user-location bipartite graph to infer social links. Yu, Wang, and Li (2018) propose a graph embedding method based on hierarchical walk sampling on a user meeting graph for mobility relationship inference. Wu et al. (2019) first apply GCNs (Kipf and Welling 2017) to learn user embeddings on a user mobility heterogeneous graph that incorporates user-user, user-location, and location-location relations. These methods leverage graph embedding techniques to enable the propagation of relationships, thus improving model inference performance.

In addition, a few recent works are proposed to study the impact of mobility and social relationships on each other using LBSN data. Yang et al. (2019) propose a hypergraph embedding approach (LBSN2Vec) for automatic feature learning from the LBSN heterogeneous hypergraph, which is extended to LBSN2Vec++ (Yang et al. 2022) that further considers the heterogeneous nature of the LBSN hypergraph. Zhang, Lai, and Wang (2020) devise a multi-view matching network to learn three view-specific representations (*i.e.*, social, spatial, and temporal factors) and fuse them for final link inference. Recently, Huynh et al. (2022) develop an embedding technique that utilises multiple representations to capture all of the high-order, dynamic and multi-role contexts in the LBSN hypergraph data for both friend suggestion and POI recommendation. *These approaches enhance hypergraph embedding techniques, but they all ignore the fact that there are a large number of noisy edges that are not related to social relationships in the constructed graph, which severely limits the performance of existing models.*

Our model goes beyond the state of the art by developing a graph structure learning framework that can remove noisy edges generated by social-irrelevant check-ins from the user meeting graph to effectively learn node representations that are beneficial for revealing social relationships.

Problem Definition

Let $U = \{u_1, u_2, \ldots, u_m\}$ denote the collection of all users.

Definition 1 (Check-in Record). *A check-in record is a triple $\langle u, t, p\rangle$ that represents user u visiting POI p at time t. Here, p is a uniquely identified venue in the form of $\langle p_{id}, category, \ell\rangle$, where p_{id} is the POI identifier, $category$ denotes its category, and ℓ is the geographical coordinates*

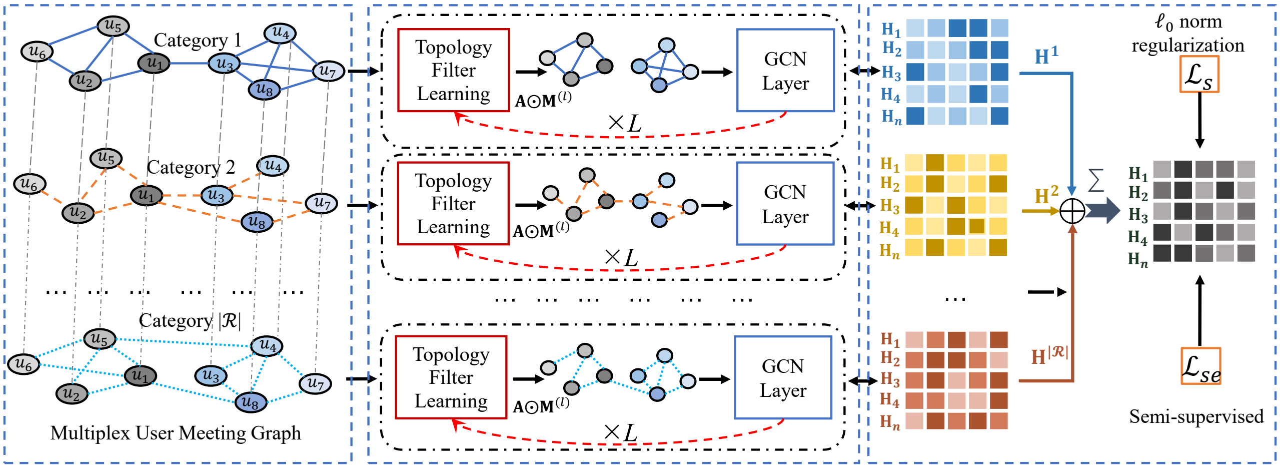


Figure 1: The overview of the proposed framework

of the POI (i.e., longitude and latitude).

Definition 2 (Mobility Trajectory)**.** *The mobility trajectory of a user u is a $(\langle u, t_1, p_1\rangle, \langle u, t_2, p_2\rangle, \ldots, \langle u, t_n, p_n\rangle)$ of check-in records generated by user u in chronological order, denoted by Tr_u.*

Given the above definitions, we formally define our studied problem in this work as follows:

Problem (Mobility Relationship Inference)**.** *Given a set of users U and their mobility trajectories, our goal is to learn an inference model $\mathcal{F}(u_i, u_j) \rightarrow \hat{y}$ for each pair of users, where $\hat{y} \in \{0, 1\}$, and $\hat{y} = 1$ indicates that they are friends, otherwise, they are not.*

In general, mobility relationships among mobile users are commonly learned from their spatial-temporal interactions. In literature, the interaction behavior, *i.e.*, a *meeting event*, is usually defined as (Yu, Wang, and Li 2018):

Definition 3 (Meeting event)**.** *Given a time threshold τ, user u_i and user u_j are considered to have a meeting event if they checked in at the same place within τ, i.e., $\exists\langle u_i, t, p\rangle \in Tr_{u_i}, \langle u_j, t', p'\rangle \in Tr_{u_j}$ such that $p = p'$ and $|t - t'| \leq \tau$.*

Definition 4 (Meeting Frequency)**.** *The meeting frequency between users u_i and u_j is the number of all meeting events between them. We denote it as $m_{i,j}$.*

Methodology

In this section, we present the details of our graph neural network model SRINet (as shown in Figure 1). It consists of three key components: (1) *multiplex user meeting graph modeling*, (2) *graph structure learning network*, and (3) *model learning*. *First*, we construct a multiplex user meeting graph to model meeting events among all users and incorporate meeting location semantics. *Second*, we present the graph structure learning network to learn topology filters with graph sparsification constraint. *Finally*, we introduce the joint learning objective function combining semi-supervised learning and regularization constraints to guide model learning.

Multiplex User Meeting Graph Modeling

To infer social relationships, many works model the users' mobility data into a complex heterogeneous graph, including user-user, user-location, and location-location, and even construct a heterogeneous hypergraph in state-of-the-art works (Yang et al. 2022; Huynh et al. 2022), incorporating time and activity categories. Nevertheless, the essence of discovering potential friend relationships from user mobility data lies in the interactions between users, therefore, inspired by (Yu, Wang, and Li 2018), we adopt the user meeting graph, denoted by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the collection of all mobile users, and $\mathcal{E}$ is the collection of edges corresponding to meeting events between users.

To capture the influence of meeting location semantics on friend relationship reference, we further construct the semantics aware multiplex user meeting graph $\mathcal{G} = \{\mathcal{V}, \{\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_{|\mathcal{R}|}\}\}$. $\mathcal{R}$ is the collection of semantic categories. As illustrated in Figure 1, multiplex user meeting graph is a heterogeneous graph that contains only one type of user node and multiple types of edges, and each type of edge corresponds to a meeting location semantics, *i.e.*, POI category. Specifically, for each pair of u_i and u_j, each edge $e^r_{i,j} \in \mathcal{E}_r$ represents the mobility relationship between them with respect to category r, and its weight is defined by their meeting frequency at POIs with category r. Therefore, the number of types of edges in the user meeting graph is determined by the number of POI categories, and each layer represents the interaction of all users at POIs of that category. For convenience, hereinafter, the user graph refers to the user meeting graph.

Graph Structure Learning Network

As analyzed above, performing representation learning directly on the constructed user graph cannot achieve the desired effect, because the user graph contains a large number of noisy edges, which seriously disturb the propagation of relationships. Therefore, if noisy edges (*e.g.*, occasional meeting events) can be removed from the user graph, the graph will be able to more accurately represent the social interactions between users, and thus the learned user representations will be able to more effectively predict friendships among users.

To this end, inspired by recent works on graph structure learning (Luo et al. 2021; Jin et al. 2020; Zhao et al. 2021), we propose the SRINet framework to learn a clean user graph on human mobility data for social relationship reference. SRINet framework is a GNN model that integrates topology structure learning and representation learning, which is compatible with existing GCNs, such as GCN (Kipf and Welling 2017), GAT (Veličković et al. 2018), etc. Additionally, the proposed SRINet is an end-to-end learning network that can be jointly optimized with GCNs in a semi-supervised manner.

Topology Filter Learning. The goal of the topology learning network is to learn a filter weeding out irrelevant edges for GCN layer. We first present topology filter learning on a single-layer user graph.

We introduce a binary mask matrix, *i.e.*, topology filter, $\mathbf{M}^{(l)} \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{V}|}$ for the l-th GCN layer, where $\mathbf{M}_{i,j}^{(l)}$ indicates whether there should be an edge between users u_i and u_j (0 means should not, *i.e.*, noisy edge). Therefore, the filtered adjacency matrix is $\mathbf{A}^{(l)} = \mathbf{A} \odot \mathbf{M}^{(l)}$, where $\mathbf{A}$ is the adjacency matrix of the user graph, and $\odot$ is the element-wise product. The topology filters for all layers are learned with graph sparsificaton constraint introduced below.

Graph Sparsification Constraint. One potential way to remove noisy edges from adjacency matrix $\mathbf{A}$ is to regularize matrix $\mathbf{M}^{(l)}$ with ℓ_0 norm, *i.e.*, penalize the number of non-zero entities in $\mathbf{M}^{(l)}$ for each layer, to enforce the filter matrix with the property of sparsity:

$$\sum_{l=1}^{L} \|\mathbf{M}^{(l)}\|_0 = \sum_{l=1}^{L} \sum_{e_{i,j}\in\mathcal{E}} |\mathbf{M}_{i,j}^{(l)}|, \tag{1}$$

where L is the number of layers of GCNs.

Since the nondifferentiability and combinatorial nature of $2^{|\mathcal{E}|}$ possible states of $\mathbf{M}^{(l)}$, optimizing this penalty (*i.e.*, ℓ_0 regularization) is computationally intractable (Louizos, Welling, and Kingma 2017). To optimize the ℓ_0 regularization, each binary number $\mathbf{M}_{i,j}^{(l)}$ can be considered to be drawn from a Bernoulli distribution parameterized by $\pi_{i,j}^{(l)}$, *i.e.*, $\mathbf{M}_{i,j}^{(l)} \sim Bern(\pi_{i,j}^{(l)})$. Therefore, optimizing ℓ_0 norm for $\mathbf{M}^{(l)}$ can be reformulated as penalizing $\sum_{e_{i,j}\in\mathcal{E}} \pi_{i,j}^{(l)}$ (Louizos, Welling, and Kingma 2017; Luo et al. 2021).

The value of $\pi_{i,j}^{(l)}$ can be regarded as a measure of the quality of the corresponding edge in the user graph. A small value of $\pi_{i,j}^{(l)}$ means that the edge $e_{i,j}$ is likely to be noise and then should be removed in the following GCN layer. Now the regularization of the reformulated form is continuous, and various gradient estimators such as the REINFORCE (Williams 1992) could be employed, but they suffer from high variance or biased gradients (Mnih and Rezende 2016; Bengio, Léonard, and Courville 2013; Gal, Hron, and Kendall 2017). To efficiently optimize the ℓ_0 norm with gradient methods, we employ the continuous cumulative distribution and the reparameterization trick (Rezende, Mohamed, and Wierstra 2014; Kingma, Salimans, and Welling 2015), and thus we can reformulate $\sum_{e_{i,j}\in\mathcal{E}} \pi_{i,j}^{(l)}$ as an expectation over a parameter-free noise distribution $p(\epsilon)$ and a deterministic and differentiable transformation $f(\cdot,\cdot)$ parameterized by $a_{i,j}^{(l)}$ and $\epsilon^{(l)}$:

$$\mathbf{M}_{i,j}^{(l)} = f(a_{i,j}^{(l)}, \epsilon^{(l)}), \epsilon^{(l)} \sim p(\epsilon). \tag{2}$$

To learn the noisy edges, we adopt a parameterized deep neural network $f_{\theta_l}^{(l)}$ to compute $a_{i,j}^{(l)}$ from node hidden representations: $a_{i,j}^{(l)} = f_{\theta_l}^{(l)}(h_i^{(l)}, h_j^{(l)})$. To further obtain $\mathbf{M}$, we apply a hard-sigmoid on the concrete distribution (Jang, Gu, and Poole 2016; Maddison, Mnih, and Teh 2016; Louizos, Welling, and Kingma 2017). Specifically, we fist sample $s_{i,j}^{(l)}$ from a binary concrete distribution with the location $a_{i,j}^{(l)}$:

$$s_{i,j}^{(l)} = \sigma((\log\epsilon - \log(1-\epsilon) + a_{i,j}^{(l)})/T), \epsilon \sim \mathcal{U}(0,1), \tag{3}$$

where T is the temperature, $\sigma(\cdot)$ is the Sigmoid function, and $\mathcal{U}(\cdot,\cdot)$ is the Uniform distribution.

The binary concrete is a smooth approximation to Bernoulli random variables, which allows for gradient-based optimization through the reparameterization trick. The temperature T controls the degree of approximation. With $T = 0$ we can recover the original Bernoulli distribution, whereas with $0 < T < 1$ we can obtain a probability density that concentrates its mass near the endpoints (*i.e.*, 0 and 1). To make the probability of a noise edge to be exactly zero, we can stretch this distribution to the range (γ, η) with $\gamma < 0$ and $\eta > 1$ and then apply a hard-sigmoid. We use the following estimator for $\mathbf{M}_{i,j}^{(l)}$:

$$\mathbf{M}_{i,j}^{(l)} = \min(1, \max(0, \bar{s}_{i,j}^{(l)})), \bar{s}_{i,j}^{(l)} = s_{i,j}^{(l)}(\eta-\gamma)+\gamma. \tag{4}$$

Eventually, the ℓ_0 regularization in Eq. (1) can be expressed as (Louizos, Welling, and Kingma 2017):

$$\begin{aligned}\mathcal{L}_s &= \sum_{l=1}^{L} \sum_{e_{i,j}\in\mathcal{E}} (1 - P_{\bar{s}_{i,j}^{(l)}}(0|\theta_l)) \\ &= \sum_{l=1}^{L} \sum_{e_{i,j}\in\mathcal{E}} (a_{i,j}^{(l)} - \tau \log\frac{-\gamma}{\eta}),\end{aligned} \tag{5}$$

$P_{\bar{s}_{i,j}^{(l)}}(0|\theta_l)$ is the cumulative distribution function of $\bar{s}_{i,j}^{(l)}$.

Notice that low-rank constraint (*i.e.*, nuclear norm regularization) is another potential implementation of structural topology learning. But low-rank constraint is more efficient for matrices that contain a lot of redundant and correlated information, such as image data. Since most users have few check-ins in collected mobility data, that is, the number of check-ins follows a long-tail distribution (Yu, Wang, and Li 2018), the user meeting graph is a relatively sparse matrix where social interactions are partially visible and contain a lot of noise. We experimentally verified that low-rank constraint has no obvious effect on removing noise edges in the scenario of this work but hurts mode performance, hence we do not use low-rank constraint in our model.

Multiplex Graph Learning

Let $\mathbf{H}^r$ be the output of the last GCN layer (*i.e.*, node representations) in graph structure learning network for the user graph consisting of edges of type r (*i.e.*, $\mathcal{E}_r$):

$$\mathbf{H}^r = GNN(\mathbf{A}_r, \mathbf{H}^{(0)}), \tag{6}$$

where $\mathbf{A}_r$ denotes the adjacency matrix of $\mathcal{E}_r$, $\mathbf{H}^{(0)}$ can be the user attribute matrix, or identity matrix, and GNN is the graph structure learning network described above.

Then, we fuse the node representations corresponding to all categories of user graphs to capture the influence of different meeting semantics:

$$\mathbf{H} = \frac{1}{|\mathcal{R}|}\sum_{i=1}^{|\mathcal{R}|}\mathbf{H}^i. \tag{7}$$

Model Learning

In this section, we present the objective function to train our model to learn user representations for friendship inference. Specifically, we train our SRINet in a semi-supervised learning manner.

We adopt the following binary cross-entropy loss function through negative sampling to optimize model parameters:

$$\begin{aligned}\mathcal{L}_{semi} = &- \sum_{(u,v)\in\Omega} \log\sigma(<\mathbf{H}_i^{\mathsf{T}}, \mathbf{H}_j>) \\ &- \sum_{(i',j')\in\Omega^-} \log\sigma(-<\mathbf{H}_{i'}^{\mathsf{T}}, \mathbf{H}_{j'}>),\end{aligned} \tag{8}$$

where $\mathbf{H}_i$ is the representation of user u_i. T denotes the vector transposition and $\sigma(\cdot)$ is the sigmoid function. $<,>$ can be any vector similarity measure function (*e.g.*, dot product used in this work). In addition, Ω is the set of positive user pair sampled from training set. Ω^- is the set of negative user pairs sampled from all pairs of users who have not met and are not included in the training set.

The final loss function to train our model is defined as:

$$\mathcal{L} = \mathcal{L}_{semi} + \omega\mathcal{L}_s, \tag{9}$$

where ω is used to control the contribution of graph sparsification constraint to the overall loss.

Experiments

In this section, we evaluate our proposed model on three real-world mobility datasets. The following research questions (RQs) are used to guide our experiments:

- **RQ1.** How does our proposed model perform in social relationship inference on real-world datasets compared to existing methods?
- **RQ2.** How does our SRINet perform on sparse data of inactive users in comparison to state-of-the-art baselines?
- **RQ3.** How does SRINet perform with different parameter settings (*e.g.*, training set ratio, hyperparameters ω, and dimension d)?

Dataset	Gowalla	Brightkite	Foursquare
City	Austin	SF and LA	NYC
#users	7,355	6,393	13,692
#check-ins	207,278	223,549	251,323
#locations	5,115	20,596	25,395
#friends	35,696	20,660	21,431

Table 1: Statistics of the datasets.

Datasets

We use three publicly available real-world mobility datasets, *i.e.*, Gowalla, Brightkite (Cho, Myers, and Leskovec 2011), and Foursquare (Yang et al. 2019), to evaluate the performance of models. Following previous works (Yu, Wang, and Li 2018; Wu et al. 2019; Yang et al. 2022; Huynh et al. 2022), we focus on four cities with most check-ins in our experiments: Austin in Gowalla dataset, San Francisco (SF) and Los Angeles (LA) in Brightkite dataset, and New York City (NYC) in Foursquare dataset. Since POI categories are not available in Gowalla and Brightkite, we collect the POI categories from public Foursquare API, and perform POI category matching as in (Yu, Wang, and Li 2018). Table 1 shows the basic statistics of three datasets.

It is worth noting that previous works (Wang, Li, and Lee 2014; Pham, Shahabi, and Liu 2016; Zhang, Lai, and Wang 2020; Huynh et al. 2022) all discard the sparse data of inactive users, while we do not remove any user mobility data in this work.

Baselines

We compare our SRINet against the following baselines:

- **DeepWalk** (Perozzi, Al-Rfou, and Skiena 2014) – performs random walks on network and uses skip-gram model to learn node representation.
- **node2vec** (Grover and Leskovec 2016) – extends DeepWalk by proposing a biased random walk sampling method with the balance between depth-first search and breadth-first search.
- **GCN** (Kipf and Welling 2017) – is the most representative one that performs convolutional operations over graph neighboring nodes for information aggregation.
- **emb-cat** (Yu, Wang, and Li 2018) – applies skip-gram based model on user graph with a hierarchical walk sampling scheme to learn node embeddings.
- **Heter-GCN** (Wu et al. 2019) – applies GCN on user mobility heterogeneous graph to learn node embeddings.
- **MVMN** (Zhang, Lai, and Wang 2020) – learns view-specific representations from social, spatial, and temporal views, and fuses them for final link inference.
- **LBSN2Vec++** (Yang et al. 2019, 2022) – learns node embeddings based on n-wise node proximity of sampled hyperedges from LBSN heterogeneous hypergraph.
- **MSC-LBSN** (Huynh et al. 2022) – is a multi-context embedding method to learn multiple representations for each user that respect their multiple roles in LBSN heterogeneous hypergraph.

Method	Gowalla		Brightkite		Foursquare	
	ROCAUC	PRAUC	ROCAUC	PRAUC	ROCAUC	PRAUC
DeepWalk	0.6035	0.6172	0.5963	0.6130	0.6137	0.6039
node2vec	0.6212	0.6159	0.6133	0.6014	0.6257	0.6211
GCN	0.6275	0.6318	0.6167	0.6312	0.6276	0.6153
emb-cat	0.6405	0.6337	0.6515	0.6426	0.6359	0.6496
Heter-GCN	0.7753	0.7867	0.7596	0.7481	0.7562	0.7633
MVMN	0.7612	0.7831	0.7456	0.7434	0.7573	0.7648
LBSN2Vec++	0.7521	0.7472	0.7286	0.7188	0.7352	0.7269
MSC-LBSN	<u>0.7892</u>	<u>0.8054</u>	<u>0.7705</u>	<u>0.7628</u>	<u>0.7736</u>	<u>0.7721</u>
SRINet-GS	0.7776	0.7815	0.7693	0.7763	0.7762	0.7613
SRINet-ML	0.8189	0.8432	0.7952	0.8216	0.8032	0.8156
SRINet (GCN)	**0.8296**	**0.8602**	**0.8091**	**0.8308**	**0.8163**	**0.8297**
SRINet (GAT)	**0.8315**	**0.8693**	**0.8166**	**0.8362**	**0.8290**	**0.8336**

Table 2: Performance comparison of all models on three real-world datasets.

Evaluation Metrics and Experiment Settings

In our evaluation, we use the Area Under the ROC Curve (ROCAUC) and the Area Under the Precision-Recall Curve (PRAUC) to quantify the performance of different methods. For social relationship inference, the pairwise cosine similarity of the learned user presentations in each method is used to compute a score that indicates the probability of two users being friends.

In our experiments, we use 25% friendships in social networks on each dataset as training set. Then we randomly sample another 5% friendship for validating, and use the remaining 70% for testing.

For baseline implementations, we use the released source code by their authors for evaluation, and the parameters of all baselines are tuned to be optimal. For our SRINet, we set user embedding dimension d to 512 unless stated otherwise, the number of convolution layers L to 2, tune learning rate from 0.0001 to 0.01, dropout to 0.01, and weight-decay to 0.0001, use early stopping mechanism, and set patience to 10 to avoid overfitting. The coefficient ω is set to 0.003 for three datasets. To capture more potential meeting events, we set τ to 2 hours.

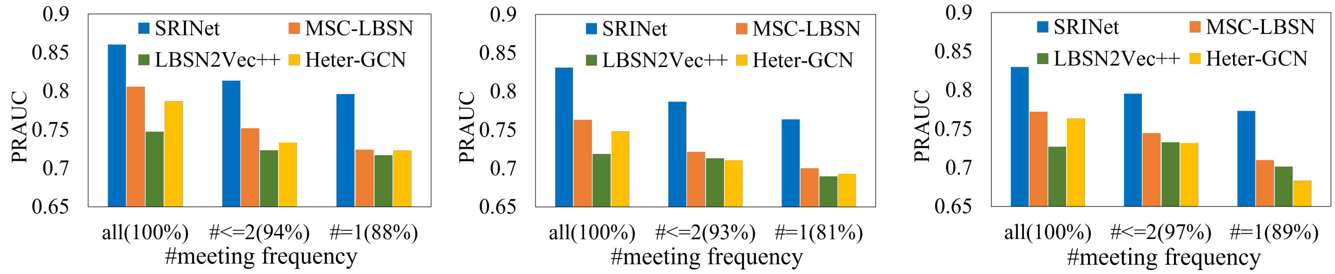
Experiment Results

Overall Performance (RQ1). The overall performance of all methods on three datasets is reported in Table 2, where the best two are shown in **bold** and the best among baselines is <u>underlined</u>. SRINet (GCN) and SRINet (GAT) are our model variants using GCN (Kipf and Welling 2017) and GAT (Veličković et al. 2018) as the GCN layer in SRINet respectively. SRINet-ML and SRINet-GS are two variants of our SRINet (GCN). Specifically, SRINet-ML does not consider POI categories, *i.e.*, only a single-layer user meeting graph as input. SRINet-GS removes graph sparsification constraint, *i.e.*, weight ω is set to 0 in $\mathcal{L}$.

From the results in Table 2, we can observe that SRINet achieves the best performance in terms of both ROCAUC and PRAUC on three mobility datasets. This is because our model is the first to consider the essential problem of poor performance of existing methods in discovering social relations from user mobility data – there are a large number of check-in records that are not related to social activities in human mobility data, resulting in a large number of noise edges in either user meeting graph or user mobility heterogeneous graph. Our designed graph structure learning framework effectively removes the noise edges in the constructed user meeting graph, thereby improving the model performance, which verifies that the essential problem we proposed does exist. Although state-of-the-art methods (*e.g.*, MSC-LBSN) improve the performance of friendship prediction by designing more sophisticated sampling methods and/or node representation learning methods, they all ignore this essential problem and avoid the impact of this essential problem on their models by discarding the sparse data of inactive users. Specifically, results in Table 2 indicate that our SRINet (GAT) achieves average gains of 8.32% ROCAUC and 10.19% PRAUC in comparison to the best-performed baseline across all datasets. Considering that the performance gain in social relationship inference by the recent work (*i.e.*, MSC-LBSN) is usually around 2-3% in both ROCAUC and PRAUC, this performance improvement achieved by our SRINet is significant.

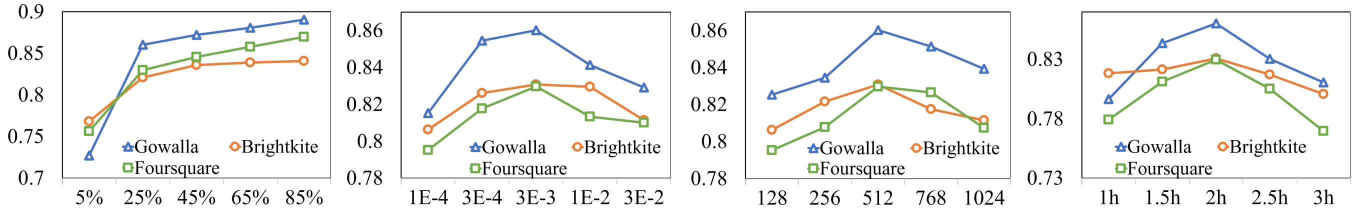
Additionally, our variant SRINet-ML still outperforms all baseline methods, which further illustrates the superiority of our proposed solution – it still works well on a simple user meeting graph using the classic GCN. Our SRINet further outperforms the variant SRINet-ML demonstrating the effect of our proposed multiplex user graph model for capturing the influence of meeting semantics on friendship inference. Furthermore, SRINet-GS performs much worse than SRINet (GCN) on all datasets, which reflects the crucial role of our designed graph sparsification constraint module in removing noisy edges. The comparison between SRINet (GCN) and SRINet (GAT) highlights that our model is compatible with different GCN models.

Performance *w.r.t.* Meeting Frequency (RQ2). To validate the effectiveness of our SRINet on sparse mobility data of inactive users, we report the performance of SRINet (GCN) and several stronger baselines on user pairs with low meeting frequency on three mobility datasets. Experimental



(a) Gowalla (b) Brightkite (c) Foursquare

Figure 2: Performance of different models on user pairs with low meeting frequency.



(a) PRAUC *w.r.t.* training set (b) PRAUC *w.r.t.* ω (c) PRAUC *w.r.t.* d (d) PRAUC *w.r.t.* τ

Figure 3: Hyperparameter impact of SRINet *w.r.t.* training set ratio, coefficient ω, dimension d, and time threshold τ.

results are shown in Figure 2, where the percentage on the horizontal axis indicates the proportion of the user pairs with that meeting frequency to all the user pairs who have met in each dataset. For example, 88% user pairs have a meeting frequency equal to 1 and 94% user pairs have a meeting frequency less than or equal to 2 on Gowalla. Notice that, in this experiment, we use PRAUC to evaluate model performance because precision-recall curves are better in evaluating performance with class imbalance, while ROC curves can be deceptive in this case (Davis and Goadrich 2006; Yang, Lichtenwalter, and Chawla 2015).

As shown in Figure 2, our SRINet significantly outperforms the strong baselines in all tested cases. Specifically, for user pairs who meet only once (*i.e.*, #frequency=1), SRINet improves state-of-the-art MSC-LBSN by 9.87%, 9.09%, and 8.94% on Gowalla, Brightkite, and Foursquare, respectively. It is very difficult to predict the relationships for user pairs with low meeting frequency, because firstly, the number of such user pairs is very large, secondly, most of them are participated by inactive users, and thirdly, most of the edges are generated by occasional meeting events (*e.g.*, checking in at a heavily visited public place). As verified in (Wu et al. 2019), the convolutional property of GCNs effectively models relationship propagation, thereby improving inference performance for such low-meeting frequency cases. However, the presence of a large number of noisy edges disturbs normal relation propagation, while our attempt to remove them significantly improves inference performance. This experiment suggests the rationality of our designed model, which effectively combines the relationship propagation ability of GNNs with the power of removing noisy edges of graph structure learning.

Parameter Sensitivity (RQ3). We also evaluate the sensitivity of SRINet *w.r.t.* different settings of training set ratio, hyperparameter ω, embedding dimension d, and time threshold τ. Figure 3 shows the experimental results of SRINet (GCN) on three datasets. We find that the performance of SRINet increases as the training set ratio increases, in particular, when the training set ratio grows from 5% to 25%, there is a large increase in performance, and as the training set continues to increase, the performance increases slowly. This is in line with expectations, and our model can achieve desired performance when only about 25% training set ratio is required. Figure 3(b) illustrates the performance of SRINet with respect to coefficient ω for regularizer $\mathcal{L}_s$. As expected, the performance of our model first increases and then drops as coefficient ω increases. This also verifies that graph sparsification constraint contributes to the model performance improvement. From Figure 3(c), we can see that the performance of SRINet first rises and then decreases slightly as dimension d increases, and achieves the best performance when $d = 512$. Finally, we investigate the impact of time threshold τ on our model. As shown in Figure 3(d), the model performance reaches its best at $t = 2$ hours, while the performance decreases for smaller or larger time intervals. The reason is that most social events generally last around two hours, and a smaller time interval may miss some meeting events, while a larger time interval brings more noise edges, both of which affect the performance.

Conclusion

In this paper, we present a novel graph structure learning framework, SRINet, for inferring social relationships from user mobility data. Our SRINet effectively learns user representations for friendship inference by integrating the representation learning ability of GNNs with the denoising power of graph structure learning. Experiments on three real-world mobility datasets show that our model significantly outperforms state-of-the-art baselines. In-depth analysis offers results of parameter sensitivity and sparse data evaluation.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China under grant Nos. 62176243, 61773331, 62202438, and 41927805, and the National Key Research and Development Program of China under grant Nos. 2018AAA0100602 and 2019YFC1509100.

References

Backes, M.; Humbert, M.; Pang, J.; and Zhang, Y. 2017. walk2friends: Inferring social links from mobility profiles. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 1943–1957.

Bengio, Y.; Léonard, N.; and Courville, A. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Chen, W.; Li, S.; Huang, C.; Yu, Y.; Jiang, Y.; and Dong, J. 2022. Mutual Distillation Learning Network for Trajectory-User Linking. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, 1973–1979.

Cho, E.; Myers, S. A.; and Leskovec, J. 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*, 1082–1090.

Cici, B.; Markopoulou, A.; Frias-Martinez, E.; and Laoutaris, N. 2014. Assessing the potential of ride-sharing using mobile and social data: a tale of four cities. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 201–211.

Dai, S.; Wang, J.; Huang, C.; Yu, Y.; and Dong, J. 2021a. Temporal Multi-view Graph Convolutional Networks for Citywide Traffic Volume Inference. In *2021 IEEE International Conference on Data Mining (ICDM)*, 1042–1047. IEEE.

Dai, S.; Wang, J.; Huang, C.; Yu, Y.; and Dong, J. 2022. Dynamic Multi-View Graph Neural Networks for Citywide Traffic Inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*.

Dai, S.; Yu, Y.; Fan, H.; and Dong, J. 2021b. Personalized poi recommendation: spatio-temporal representation learning with social tie. In *International Conference on Database Systems for Advanced Applications*, 558–574. Springer.

Davis, J.; and Goadrich, M. 2006. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*, 233–240.

Eagle, N.; Pentland, A. S.; and Lazer, D. 2009. Inferring friendship network structure by using mobile phone data. *PNAS*, 106(36): 15274–15278.

Gal, Y.; Hron, J.; and Kendall, A. 2017. Concrete dropout. *Advances in neural information processing systems*, 30.

Grover, A.; and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 855–864. ACM.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 1025–1035.

Hsieh, H.-P.; and Li, C.-T. 2019. Inferring online social ties from offline geographical activities. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2): 1–21.

Hsieh, H.-P.; Yan, R.; and Li, C.-T. 2015. Where you go reveals who you know: analyzing social ties from millions of footprints. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 1839–1842. ACM.

Huynh, T. T.; Van Tong, V.; Nguyen, T. T.; Jo, J.; Yin, H.; and Nguyen, Q. V. H. 2022. Learning Holistic Interactions in LBSNs with High-order, Dynamic, and Multi-role Contexts. *IEEE Transactions on Knowledge and Data Engineering*.

Jang, E.; Gu, S.; and Poole, B. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Jin, W.; Ma, Y.; Liu, X.; Tang, X.; Wang, S.; and Tang, J. 2020. Graph structure learning for robust graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 66–74.

Kingma, D. P.; Salimans, T.; and Welling, M. 2015. Variational dropout and the local reparameterization trick. *Advances in neural information processing systems*, 28.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*.

Li, Z.; Lin, C. X.; Ding, B.; and Han, J. 2011. Mining significant time intervals for relationship detection. In *International Symposium on Spatial and Temporal Databases*, 386–403. Springer.

Liu, Z.; Huang, C.; Yu, Y.; and Dong, J. 2021. Motif-preserving dynamic attributed network embedding. In *Proceedings of the Web Conference 2021*, 1629–1638.

Liu, Z.; Huang, C.; Yu, Y.; Fan, B.; and Dong, J. 2020. Fast attributed multiplex heterogeneous network embedding. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 995–1004.

Louizos, C.; Welling, M.; and Kingma, D. P. 2017. Learning sparse neural networks through L_0 regularization. *arXiv preprint arXiv:1712.01312*.

Luo, D.; Cheng, W.; Yu, W.; Zong, B.; Ni, J.; Chen, H.; and Zhang, X. 2021. Learning to drop: Robust graph neural network via topological denoising. In *Proceedings of the 14th ACM international conference on web search and data mining*, 779–787.

Maddison, C. J.; Mnih, A.; and Teh, Y. W. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

Mnih, A.; and Rezende, D. 2016. Variational inference for monte carlo objectives. In *International Conference on Machine Learning*, 2188–2196. PMLR.

Noulas, A.; Scellato, S.; Mascolo, C.; and Pontil, M. 2011. An empirical study of geographic user activity patterns in foursquare. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 5, 570–573.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 701–710. ACM.

Pham, H.; Shahabi, C.; and Liu, Y. 2013. Ebm: an entropy-based model to infer social strength from spatiotemporal data. In *Proceedings of the 2013 ACM SIGMOD*, 265–276. ACM.

Pham, H.; Shahabi, C.; and Liu, Y. 2016. Inferring social strength from spatiotemporal data. *ACM Transactions on Database Systems (TODS)*, 41(1): 7.

Rezende, D. J.; Mohamed, S.; and Wierstra, D. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, 1278–1286. PMLR.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations (ICLR)*.

Wang, C.; Raina, R.; Fong, D.; Zhou, D.; Han, J.; and Badros, G. 2011. Learning relevance from heterogeneous social network and its application in online targeting. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 655–664.

Wang, H.; Li, Z.; and Lee, W.-C. 2014. PGT: Measuring mobility relationship using personal, global and temporal factors. In *Proceedings of 2014 IEEE International Conference on Data Mining (ICDM)*, 570–579. IEEE.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3): 229–256.

Wu, Y.; Lian, D.; Jin, S.; and Chen, E. 2019. Graph Convolutional Networks on User Mobility Heterogeneous Graphs for Social Relationship Inference. In *IJCAI*, 3898–3904.

Yang, D.; Qu, B.; Yang, J.; and Cudre-Mauroux, P. 2019. Revisiting user mobility and social relationships in lbsns: a hypergraph embedding approach. In *The world wide web conference*, 2147–2157.

Yang, D.; Qu, B.; Yang, J.; and Cudré-Mauroux, P. 2022. LBSN2Vec++: Heterogeneous Hypergraph Embedding for Location-Based Social Networks. *IEEE Transactions on Knowledge and Data Engineering*, 34(4): 1843–1855.

Yang, Y.; Lichtenwalter, R. N.; and Chawla, N. V. 2015. Evaluating link prediction methods. *Knowledge and Information Systems*, 45(3): 751–782.

Yu, P.; Fu, C.; Yu, Y.; Huang, C.; Zhao, Z.; and Dong, J. 2022. Multiplex heterogeneous graph convolutional network. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2377–2387.

Yu, Y.; Wang, H.; and Li, Z. 2018. Inferring mobility relationship via graph embedding. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(3): 1–21.

Zhang, W.; Lai, X.; and Wang, J. 2020. Social link inference via multiview matching network from spatiotemporal trajectories. *IEEE transactions on neural networks and learning systems*.

Zhao, J.; Wang, X.; Shi, C.; Hu, B.; Song, G.; and Ye, Y. 2021. Heterogeneous graph structure learning for graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 4697–4705.

Zhou, F.; Wu, B.; Yang, Y.; Trajcevski, G.; Zhang, K.; and Zhong, T. 2018. Vec2link: Unifying heterogeneous data for social link prediction. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 1843–1846.