Exploiting Spatio-Temporal User Behaviors for User Linkage

Wei Chen
School of Computer Science and Technology, Soochow University, China
wchzhg@gmail.com

Hongzhi Yin*
School of ITEE, The University of Queensland, Brisbane, Australia
db.hongzhi@gmail.com

Weiqing Wang
School of ITEE, The University of Queensland, Brisbane, Australia
weiqingwang@uq.edu.au

Lei Zhao
School of Computer Science and Technology, Soochow University, China
zhaol@suda.edu.cn

Wen Hua
School of ITEE, The University of Queensland, Brisbane, Australia
w.hua@uq.edu.au

Xiaofang Zhou
School of ITEE, The University of Queensland, Brisbane, Australia
zxf@itee.uq.edu.au

ABSTRACT

Cross-device and cross-domain user linkage have been attracting a lot of attention recently. An important branch of the study is to achieve user linkage with spatio-temporal data generated by the ubiquitous GPS-enabled devices. The main task in this problem is twofold, i.e., how to extract the representative features of a user; how to measure the similarities between users with the extracted features. To tackle the problem, we propose a novel model STUL (Spatio-Temporal User Linkage) that consists of the following two components. 1) Extract users' spatial features with a density based clustering method, and extract the users' temporal features with the Gaussian Mixture Model. To link user pairs more precisely, we assign different weights to the extracted features, by lightening the common features and highlighting the discriminative features. 2) Propose novel approaches to measure the similarities between users based on the extracted features, and return the pair-wise users with similarity scores higher than a predefined threshold. We have conducted extensive experiments on three real-world datasets, and the results demonstrate the superiority of our proposed STUL over the state-of-the-art methods.

KEYWORDS

Cross-domain; User linkage; Spatio-temporal behaviors

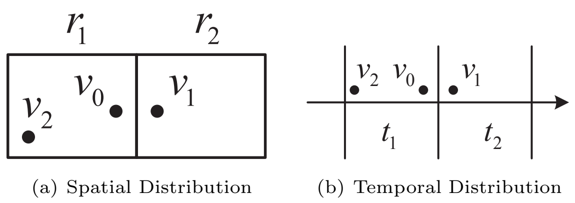
1 INTRODUCTION

The proliferation of GPS-enabled devices and mobile techniques has led to the emergence of large amount of spatio-temporal information. For example, the vehicles equipped with GPS can generate lots of trajectories, which consist of a sequence of points that are sampled in a short time period, to keep track of moving objects. Meanwhile, the widespread of location based social networks, such as Facebook, Twitter, and Foursquare have generated massive discrete check-in data [20], as many users share their status associated with locations and timestamps. The availability of spatio-temporal information offers a good opportunity to model users' spatio-temporal behaviors [23][18]. On the other hand, user linkage, which aims at connecting the same users across different platforms, has attracted much attention. User linkage benefits widespread real applications, such as prediction [13][21], data fusion [28], recommendation [19][22], etc. This paper focuses on leveraging the increasingly available spatio-temporal information in user linkage.

However, to the best of our knowledge, there is only one work utilizing the users' spatial and temporal features simultaneously to achieve user linkage [14]. In that work, locations and times are divided into bins, and each spatio-temporal record is associated with a bin (r, t) where r is a region and t represents a time interval. The similarities between users are inferred based on users' co-occurrences in each bin. Nonetheless, time and space are intrinsically continuous. Discretization of time and space inevitably leads to information loss, especially for the points near the boundaries. Assume that u_0 is a user on platform A while u_1 and u_2 are two users on platform B. To simplify the problem, we assume that there is only one activity record v_0, v_1 and v_2 for each user u_0, u_1 and u_2 respectively. The distributions of these activity records in terms of space and time are given in Figure 1(a) and 1(b) respectively. Based on [14], u_0 and u_2 have a larger probability to be linked together, as they co-occur in both the spatial bin r_1 and the temporal bin t_1. However, compared with u_2, u_1 is more similar to u_0 in terms of both spatial distribution in Figure 1(a) and temporal distribution in Figure 1(b). Thus, the discretization based method cannot capture the similarity between features that are divided into different bins. Besides, discretization of time and space always begs the question of selecting the region or time interval size, and the size is invariably too small for some regions and too large for others.

*This author is the corresponding author.

CIKM'17, November 6–10, 2017, Singapore.
© 2017 ACM. ISBN 978-1-4503-4918-5/17/11...$15.00
DOI: http://dx.doi.org/10.1145/3132847.3132898



(a) Spatial Distribution (b) Temporal Distribution

Figure 1: Activity Distribution in Terms of Bins

In addition to the aforementioned problems, another major issue that needs to be addressed in user linkage is to prioritize the extracted spatio-temporal features. Riederer et al. treat each feature equally in [14]. However, we argue that these spatio-temporal features are not equally important in user linkage. In the task of user linkage, we seek to identify what are unique about a user, in order to distinguish him/her from others. In this process, the "uncommon", which we refer to as *discriminative*, spatio-temporal features actually play a more important role. For example, some locations are very popular and frequently visited by many people, such as the downtown in the city and a popular restaurant, whereas other locations are more specific only to a few people, such as a private house. In a popular public place, it is more likely for two different users to co-locate by coincidence. Thus, such spatial features are less helpful to identify the same users.

To tackle the aforementioned problems, we propose a novel spatio-temporal user linkage model STUL to exploit the continuous spatio-temporal features associated with user activities, and compute their weights based on their discriminability. In this model, time and space are treated as continuous variables. A novel method extended from density-based clustering (DP) [15] is proposed to extract users' spatial features. Compared with the bin-based method in [14], this approach is more tolerable to anomalous points and more likely to capture the real spatial behaviors of a user, as it treats the space in a continuous way and thus is able to extract regions with arbitrary shapes. To model users' temporal behaviors, an approach based on Gaussian Mixture Model (GMM) is proposed in STUL. This approach aims at modeling users' periodical behaviors centered on multiple time points.

Inspired by TF-IDF (Term Frequency Inverse Document Frequency), STUL develops a novel weight scheme to compute weights for the extracted spatio-temporal features, by lightening the common features with smaller weights and highlighting the discriminative features with larger weights. Based on the extracted features and associated weights, STUL develops three novel similarity measures and combines them in a unified way to compute the final similarities between users. The pair-wise users with similarity scores higher than a given threshold are returned as the linked user pairs.

To sum up, we make the following major contributions in this study.

文章要点

- To the best of our knowledge, we are the first to exploit the spatio-temporal behaviors in a continuous way to achieve user linkage.
- We propose a novel model STUL, where a DP-based method is developed to model the spatial behaviors, and an approach based on GMM is used to model the users' temporal behaviors. STUL introduces a novel weight scheme to distinguish the discriminative features from the common features. Based on the extracted features and associated weights, STUL develops a novel similarity measure which combines three different similarity measures to compute the similarities between users.
- We conduct extensive experiments on real datasets, and the experiment results demonstrate the high performance of STUL.

The rest of paper is organized as follows. The related work is presented in Section 2. We formulate the problem in Section 3. The spatio-temporal features are extracted in Section 4. We measure the similarities between users in Section 5 and conduct experiments in Section 6, which is followed by the conclusion in Section 7.

2 RELATED WORK

2.1 User Linkage

The user linkage across different platforms was firstly proposed in [24], where the disconnected websites were connected with the proposed approaches that leverage social medias collective patterns. Following that study, a novel method to identify users based on web profile matching has been proposed to aggregate online friends into a single integrated environment [17], which also extended its effectiveness by incorporating the user's friend network. Next, the user's profile that contains more abundant information was characterized to accomplish user linkage [2][11][25]. To merge user's contacts from different social services or compose more complete social graph for many social-powered applications, a Conditional Random Fields based approach was designed for user profile matching [2]. Especially, the work is extremely suitable for cases when profile data is poor, incomplete or hidden due to privacy settings. Human behaviors with regard to the usages of online usernames was firstly proposed by [11] to tackle the problem of linking users across multiple online communities, where an alias-disambiguation step was proposed to differentiate users with the same usernames. Furthermore, Zafarani et al. [25] have introduced a behavior model to find a mapping among identities of individuals across social media sites with more features, such as usernames, language, and writing styles. Unlike the aforementioned work, [12] investigated the problem of large-scale social identity linkage across different social media platforms. The developed framework HYDRA consists of three components, i.e., user's behaviors modeling, structural consistency graph construction, and mapping function learning. In the following studies, a novel energy-based model COSNET has been proposed to address the problem of potential inconsistency of matchings between multiple networks by considering both local and global consistency among multiple networks [27], where a subgradient algorithm was developed to accelerate the model training. In recent advance, Riederer et al. [14] investigated the problem of linking users with location data, where locations and times were divided into bins, and the scores of user pairs were measured based on these bins. Connecting users across multiple platforms brings opportunities for many applications and insights on users' behaviors, yet it raises privacy concerns. To tackle the problem, [5] studied the problem of reliable profile matching by exploiting public attributes of users. To avoid cross-site identity linking attacks, novel methods were developed in [1].

2.2 Mining of Discriminative Features

Mining and highlighting the discriminative features have received great attention, which underly a wide range of applications [10][30][8][6]. The discriminative color features method, which applies an effective color model, a novel similarity measure, and effective color feature extraction methods, was proposed to improve face recognition performance [10], where the discriminative color features were extracted from a compact color image representation. Zhu et al. [30] proposed a diversified discriminative feature selection method for graph classification, where the discriminative score was used to select frequent subgraph features, and a new diversified discriminative score was proposed to select features that have a higher diversity. To overcome the problem of complicated data process in social network area, an unsupervised feature selection algorithm was proposed due to the unlabeled nature of datasets in social network [6]. The idea of [4] is similar to our study, i.e., highlighting the discriminative features with higher score, and lightening the common features with smaller score. However, since their methods were designed to characterize objects in heterogeneous information network, and each object was represented by a feature tree, these methods cannot be applied to our study.

In spite of the great contributions made by the aforementioned studies, none of them consider the discrimination and continuity of the spatio-temporal features while linking across domains users with these features. To tackle the problem, we propose the model STUL in this study.

3 PROBLEM FORMULATION

In spatio-temporal databases, there are two different and important types of data , i.e., the check-in data and the trajectory data.

Definition 1. ***User Activity****. A user activity on both types of data is defined as $d = (u, lat, lng, t)$, where u is the user id, lat and lng denote the latitude and longitude of the location where this activity is taken and t represents the time.*

The check-in data is an important type of spatio-temporal data with low sample rate, where the time span between two locations is usually large. The check-in places associated with user activities in Twitter, Facebook, and Foursquare are meaningful (i.e., restaurants, hotel and so on). These meaningful check-ins are informative in modeling users' spatio-temporal behaviors. Thus, following many existing work [9][23], check-in data are applied directly without any preprocess in STUL.

无预处理

Trajectory data is another important type of spatio-temporal data. Different from the check-in data, trajectory data are densely sampled, which means that two adjacent points in a trajectory are sampled in a short time period. The dense sampling of trajectory data brings redundancy, where many points are meaningless, for example, a point in a high way. We argue that a meaningless point in trajectory data is not helpful in identifying users' spatial-temporal behavior patterns. To remove the meaningless points, we only focus on the regions where a user visits repeatedly. To find these regions, we introduce *stay point*, which is an important notion in existing research work [29].

Definition 2. ***Stay Point*** *[29]. Given a trajectory $\tau = (p_1, p_2, \cdots, p_n)$, where p_i is in the form of (lat, lng, t), lat and lng denote the latitude and longitude respectively, and t is the corresponding timestamp, a stay point s stands for a region where a user stayed over a certain time interval. Given a time threshold δ_t and a distance threshold δ_d, if there exists a group of consecutive points $P = (p_i, p_{i+1}, \cdots, p_j)$ of τ such that $\forall i < k \leq j$, $Distance(p_i, p_k) \leq \delta_d$, and $|p_j.t - p_k.t| \geq \delta_t$, then we have a stay point s in the form of*

$$(s.lat, s.lng) = (\frac{\sum_{k=i}^{j} p_k.lat}{|P|}, \frac{\sum_{k=i}^{j} p_k.lng}{|P|})$$

平均

The notion *stay point* is proposed to mine regions where an individual stayed longer than a threshold time period. Based on this notion, we introduce the points used to mine the meaningful regions based on which we build our model.

Definition 3. ***Stay Region Candidate Points****. Given a trajectory $\tau = (p_1, p_2, \cdots, p_n)$, the start point p_1, the end point p_n, each point p_k of $P(P = (p_i, p_{i+1}, \cdots, p_j), \forall i < k \leq j, Distance(p_i, p_k) \leq \delta_d$ and $|p_j.t - p_k.t| \geq \delta_t)$ is defined as stay region candidate point, denoted as r_c.*

We call the meaningful regions used to mine users' spatio-temporal behaviors as "stay regions". To remove the redundancy in trajectory data, only the points that can generate a stay point s are considered in modeling users' spatio-temporal behaviors. These points are referred as "Stay Region Candidate Points". Additionally, it is worth noting that the start point and end point of the trajectory are also significant. For instance, given a set of historical trajectories, we may find a user's home region and work region directly from all start points and end points, as many users usually commute between their homes and companies. Aforementioned points are defined as stay region candidate points, since we can extract the stay regions of a user with them.

Problem 1. *Given two sets of users $U_1 = \{u_{11}, u_{12}, \cdots, u_{1n}\}$ and $U_2 = \{u_{21}, u_{22}, \cdots, u_{2m}\}$ on two platforms respectively, where each user is associated with a set of user activities defined in Definition 1, we aim at finding linked user pairs across the two platforms.*

Overview of our method. The overview of our method is presented in algorithm 1, which contains two components.

First, a density based algorithm is used to extract the stay region distribution, and the Gaussian Mixture Model is used to extract the temporal features, which contain the global time distribution and local time distribution. Additionally, we assign different weights to these spatio-temporal features. The details of this component are discussed in Section 4. Second, we develop novel methods to calculate the similarity between users, and add the user pair $< u_{1i}, u_{2j} >$ into the result on condition that $S(u_{1i}, u_{2j}) \geq \theta$, where θ is a predefined threshold. The details of this component are discussed in Section 5.

```
Algorithm 1: Overall Algorithm
  Data: two sets of users $U_1 = \{u_{11}, u_{12}, \cdots, u_{1n}\}$ and
        $U_2 = \{u_{21}, u_{22}, \cdots, u_{2m}\}$, a threshold $\theta$
  Result: a set of user pairs
  for each user $u$ in $U_1$ and $U_2$ do
      extract stay regions of $u$ with a density based
      method, and assign different weights to these
      regions ;
      extract the global and local time distribution of $u$
      with GMM, and assign different weights to these
      time clusters;
  end
  for each user $u_{1i}$ in $U_1$ do
      for each user $u_{2j}$ in $U_2$ do
          calculate the similarity $S(u_{1i}, u_{2j})$ ;
          if $S(u_{1i}, u_{2j}) \geq \theta$, add the user pair $< u_{1i}, u_{2j} >$
          into the result ;
      end
  end
```

4 FEATURE EXTRACTION

To model users' spatial behaviors, we extract the stay regions. To model users' temporal behaviors, we extract the time distribution that includes global time distribution and time distribution in each stay region (i.e., local time distribution). Note that, we need to first extract all stay region candidate points if the input data consists of trajectories. We directly extract the spatio-temporal features if the input data consists of check-in records.

4.1 Stay Region Distribution

Stay regions are areas that contain more points than others, and a user will visit repeatedly. To extract these regions, we use an advanced density-based clustering (DP) method [15][3], which is based on the idea that cluster centers are surrounded by neighbors with lower local density and that they are at a relatively large distance from any points with a higher local density. This method has following advantages: 1) DP has been proved to be able to ignore anomalous points. 2) DP is able to extract regions with arbitrary shapes. 3) Compared with other clustering algorithms, DP requires less parameters, i.e., calculating the local density and the distance from points with higher local density for each stay region candidate point. Given a set of stay region candidate points $\{r_c^1, r_c^2, \cdots, r_c^n\}$, the local density $p_{r_c^i}$ and the distance from points with higher local density $\delta_{r_c^i}$ are defined as:

$$p_{r_c^i} = \sum_j \chi(d_{r_c^i, r_c^j} - d_c), \begin{cases} \chi(x) = 1, \ if \ x < 0 \\ \chi(x) = 0, \ otherwise \end{cases}$$

$$\delta_{r_c^i} = \begin{cases} \underset{p_{r_c^j} > p_{r_c^i}}{min} \ (d_{r_c^i, r_c^j}), \ if \ p_{r_c^j} > p_{r_c^i} \\ \underset{j}{max}(d_{r_c^i, r_c^j}), \ otherwise \end{cases}$$

where d_c is the cutoff distance given by users. $p_{r_c^i}$ denotes the number of points that are closer to the candidate point r_c^i than d_c. $\delta_{r_c^i}$ is the minimum distance between r_c^i and any other point with higher density, especially, $\delta_{r_c^i}$ is measured as the maximum distance $d_{r_c^i, r_c^j}$ if r_c^i has the highest density. Following the calculation of $p_{r_c^i}$ and $\delta_{r_c^i}$, the top-k centers with the highest value $\xi_{r_c^i} = p_{r_c^i} \times \delta_{r_c^i}$ are returned as cluster centers, and we assign different labels to these centers. Then, we assign each remaining point to the same cluster as its nearest neighbor of higher density. Obviously, these clusters are geographical regions that a user is more likely to visit and take activities than other places. Following the cluster extraction, we transform each cluster into a regular region, which is the foundation of similarity measure between two users. Given a cluster with points $\{r_c^i, \cdots, r_c^j\}$, the corresponding stay region is generated by connecting the marginal points of the cluster, denoted as R.

In real life, many people tend to visit popular areas, such as the downtown of a city, a large bus station, and a popular cinema. Then, many extracted stay regions of different users on a platform are overlapped. Such a phenomenon brings the great challenge to distinguish one user from others, and makes it hard to link the user to his/her account from another platform. To address the problem, we need to assign different weights to these stay regions. Given a set of users with corresponding regions in Table 1(a), the calculation of region weights presented in Table 1(b) consists of two steps. Next, we present the calculation of the weight $\omega(R_1^i)$.

Table 1: Region Weight Calculation

(a) User Region

User	Region
u_1	$(R_1^1, \cdots, R_1^l)$
u_2	$(R_2^1, \cdots, R_2^k)$
$\cdots$	$\cdots$
u_n	$(R_n^1, \cdots, R_n^m)$

(b) Region Weight

Weight
$\{\omega(R_1^1), \cdots, \omega(R_1^l))\}$
$\{(\omega(R_2^1), \cdots, \omega(R_2^k))\}$
$\cdots$
$\{\omega(R_n^1), \cdots, \omega(R_n^m))\}$

First, we compute the similarity between R_1^i and the regions of other users $\{u_2, u_3, \cdots, u_n\}$, i.e., the similarity between R_1^i and $\{R_2^1, \cdots, R_2^k, \cdots, R_n^1, \cdots, R_n^m\}$. The similarity is defined as $\sum_{R_o \in \mathcal{D}_R - \mathcal{D}_R^1} S(R_1^i, R_o)$, where $\mathcal{D}_R^1$ denotes the extracted regions of the user u_1, $\mathcal{D}_R$ denotes the extracted regions of all users, and $S(R_1^i, R_o)$ is defined as:

$$S(R_1^i, R_o) = \frac{|R_1^i \cap R_o|}{|R_1^i \cup R_o|} \tag{1}$$

where $|R_1^i \cap R_o|$ denotes the common areas of R_1^i and R_o, and $|R_1^i \cup R_o|$ represents the union areas of them. Then, we obtain the set $(\sum S(R_1^1, R_o), \sum S(R_1^2, R_o), \cdots, \sum S(R_1^l, R_o))$ for u_1, where $\sum\limits_{R_o \in \mathcal{D}_R - \mathcal{D}_R^1} S(R_1^i, R_o)$ is denoted as $\sum S(R_1^i, R_o)$ for the sake of convenience.

Second, we obtain $(\frac{N}{1+\sum S(R_1^1, R_o)}, \frac{N}{1+\sum S(R_1^2, R_o)}, \cdots, \frac{N}{1+\sum S(R_1^l, R_o)})$ by applying the function $f(x) = \frac{N}{1+x}$ following the computation method of IDF[1] to assign larger weights to regions with smaller $\sum S(R_1^i, R_o)$, where N is the possible maximum value of x. The smaller $\sum S(R_1^i, R_o)$ means it is more likely to distinguish u_1 from other users with the stay region R_1^i. Normalizing the vector, we obtain the weight of R_1^i:

$$\omega(R_1^i) = \frac{\frac{N}{1+\sum S(R_1^i, R_o)}}{\sum \frac{N}{1+\sum S(R_1^i, R_o)}} \tag{2}$$

4.2 Global Time Distribution

Time distribution is another important feature in our model. Unlike the study in [9], where the discretization method is adopted and timestamps are assigned to different time slices, we use the Gaussian Mixture Model (GMM) to model the time distribution of a user. This is because time is continuous, and a user usually takes activities around some time points [23]. For example, a user may get up around 7:00 am, have lunch around 12:00 am, and drive home around 18:00 pm. GMM is able to model these temporal behaviors, where timestamps are centered on some certain points.

In this section, we extract the temporal features from the global perspective, where the stay region factor is omitted. Given a set of stay region candidate points $(r_c^1, r_c^2, \cdots, r_c^n)$ of a user u, the Expectation Maximization (EM) algorithm is used to find optimal parameters with timestamp set $(r_c^1.t, r_c^2.t, \cdots, r_c^n.t)$.

The Gaussian probability density function is:

$$N(r_c^i.t; \mu_k, \Sigma_k) = \frac{1}{\sqrt{2\pi \Sigma_k}} \exp\left[-\frac{1}{2}(r_c^i.t - \mu_k)^2 \Sigma_k^{-1}\right] \tag{3}$$

E-step: The probability of the sample $r_c^i.t$ generated by the k-th cluster (μ_k, Σ_k) is:

$$\gamma_{ik} = \frac{\alpha_k N(r_c^i.t|\mu_k, \Sigma_k)}{\sum\limits_{j=1}^{K} \alpha_k N(r_c^i.t|\mu_j, \Sigma_j)} \tag{4}$$

where K is the number of clusters.

M-step: The maximum likelihood method is used to update model parameters as follows:

$$\alpha_k = \frac{1}{n}\sum_{i=1}^{n} \gamma_{ik},\ \mu_k = \frac{\sum\limits_{i=1}^{n} \gamma_{ik} r_c^i.t}{\sum\limits_{i=1}^{n} \gamma_{ik}},\ \Sigma_k = \frac{\sum\limits_{i=1}^{n} \gamma_{ik}(r_c^i.t - \mu_k)^2}{\sum\limits_{i=1}^{n} \gamma_{ik}}$$

After updating parameters, we obtain a set of time clusters $\{T_1, T_2, \cdots, T_n\}$ for each user. Next, we introduce how to measure the similarity between two time clusters with the overlapping coefficient (OVL) method [7][16]. Given $T_1 = (\mu_1, \sigma_1)$ and $T_2 = (\mu_2, \sigma_2)$ with following density functions:

$$\begin{aligned} f_1(x) &= \frac{1}{\sqrt{2\pi}\sigma_1} \exp(-\frac{(x-u_1)^2}{2\sigma_1^2}) \\ f_2(x) &= \frac{1}{\sqrt{2\pi}\sigma_2} \exp(-\frac{(x-u_2)^2}{2\sigma_2^2}) \end{aligned} \tag{5}$$

First, we calculate the intersections of $f_1(x)$ and $f_2(x)$. Assume $\sigma_1 > \sigma_2$, we obtain the following intersections:

$$\begin{aligned} x_1 = (&\sqrt{2\sigma_1^2\sigma_2^2 \ln\frac{\sigma_1}{\sigma_2} - (\sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2) + \frac{(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)^2}{\sigma_1^2 - \sigma_2^2}} \\ &- \frac{\mu_1\sigma_2^2 - \mu_2\sigma_1^2}{\sqrt{\sigma_1^2 - \sigma_2^2}})/\sqrt{\sigma_1^2 - \sigma_2^2} \\ x_2 = (&-\sqrt{2\sigma_1^2\sigma_2^2 \ln\frac{\sigma_1}{\sigma_2} - (\sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2) + \frac{(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)^2}{\sigma_1^2 - \sigma_2^2}} \\ &- \frac{\mu_1\sigma_2^2 - \mu_2\sigma_1^2}{\sqrt{\sigma_1^2 - \sigma_2^2}})/\sqrt{\sigma_1^2 - \sigma_2^2} \end{aligned} \tag{6}$$

Then, the similarity $S(T_1, T_2)$ is given as:

$$S(T_1, T_2) = \int_{-\infty}^{x_1} f_2(x)dx + \int_{x_1}^{x_2} f_1(x)dx + \int_{x_2}^{\infty} f_2(x)dx \tag{7}$$

Given the user u_1 with time clusters $\{(\mu_1, \sigma_1), (\mu_2, \sigma_2), \cdots, (\mu_l, \sigma_l)\}$, the computation of $\omega(T_1^i = (\mu_i, \delta_i))$ consists of two steps. First, we compute the similarity between T_1^i and the time clusters of other users, i.e., $\sum\limits_{T_o \in \mathcal{D}_T - \mathcal{D}_T^1} S(T_1^i, T_o)$, where $\mathcal{D}_T^1$ denotes the extracted time clusters of u_1 and $\mathcal{D}_T$ denotes the extracted time clusters of all users. Then, we obtain a set $(\sum S(T_1^1, T_o), \sum S(T_1^2, T_o), \cdots, \sum S(T_1^l, T_o))$ for u_1, where $\sum\limits_{T_o \in \mathcal{D}_T - \mathcal{D}_T^1} S(T_1^i, T_o)$ is denoted as $\sum S(T_1^i, T_o)$.

Second, the function $f(x) = \frac{N}{1+x}$ is also used to assign larger weights to clusters with smaller $\sum S(T_1^i, T_o)$, and we obtain the vector $(\frac{N}{1+\sum S(T_1^1, T_o)}, \frac{N}{1+\sum S(T_1^2, T_o)}, \cdots, \frac{N}{1+\sum S(T_1^l, T_o)})$. Then, the weight $\omega(T_1^i)$ is defined as:

$$\omega(T_1^i) = \frac{\frac{N}{1+\sum S(T_1^i, T_o)}}{\sum \frac{N}{1+\sum S(T_1^i, T_o)}} \tag{8}$$

[1] https://en.wikipedia.org/wiki/Tf-idf

Table 2: An Example of Local Time Distribution

	Platform A	Platform B		
user	u_0	u_1	u_2	u_3
Global Time	$\{T_0^1, T_0^2\}$	$\{T_1^1, T_1^2\}$	$\{T_2^1, T_2^2\}$	$\{T_3^1, T_3^2\}$
Parameters	$\{(8:00, 2), (18:00, 1)\}$	$\{(8:00, 2), (18:00, 1)\}$	$\{(8:00, 2), (18:00, 1)\}$	$\{(12:00, 3), (22:00, 3)\}$
Stay Region	$\{R_0^1, R_0^2\}$	$\{R_1^1, R_1^2\}$	$\{R_2^1, R_2^2\}$	$\{R_3^1, R_3^2\}$
Location	$\{(3, 4; 8, 9), (10, 14; 15, 19)\}$	$\{(3, 4; 8, 9), (10, 14; 15, 19)\}$	$\{(10, 14; 15, 19), (3, 4; 8, 9)\}$	$\{(5, 6; 11, 12), (13, 14; 21, 20)\}$
Local Time	$T_0^1\|R_0^1, T_0^2\|R_0^2$	$T_1^1\|R_1^1, T_1^2\|R_1^2$	$T_2^1\|R_2^1, T_2^2\|R_2^2$	$T_3^1\|R_3^1, T_3^2\|R_3^2$

4.3 Local Time Distribution

Considering the time distribution from global perspective, we can obtain time clusters around some certain time points, such as $T_0 = (7:00, 2)$, $T_1 = (12:00, 1)$, and $T_2 = (18:00, 3)$. By further analyzing the user behaviors with spatial information, we may extract T_0 in home region, extract T_1 in school region, and extract T_2 in bus station, that is each stay region has its own time distribution. It is worth noting that this feature is of critical importance in modeling user behaviors. Considering the example in Table 2, where parameters corresponds to global time distribution, location corresponds to stay region, and local time denotes the time cluster extracted in current stay region. $T_0^1|R_0^1$ means we can extract the time cluster $T_0^1 = (8:00, 2)$ in stay region $R_0^1 = (3, 4; 8, 9)$, where $(3, 4)$ and $(8, 9)$ denote the lower-left and upper-right of current stay region R_0^1, respectively. We assume that $< u_0, u_1 >$ is an actual linked pair. Based on the global time distribution, we can remove the link $< u_0, u_3 >$ from the candidate set $\{< u_0, u_1 >, < u_0, u_2 >, < u_0, u_3 >\}$. However, we cannot confirm which pair in $\{< u_0, u_1 >, < u_0, u_2 >\}$ is the actual linked pair based on stay region distribution and global time distribution. Fortunately, by taking the time distribution in each stay region into account, we can prune the link $< u_0, u_2 >$, since u_0 and u_2 have opposite time distribution in stay regions.

The example in Table 2 demonstrates the importance of modeling local time distribution in each stay region. To address the problem, EM algorithm is also used to evaluate parameters, and OVL is used to measure similarities. Namely, given a stay region of u, we extract the time clusters $\{(\mu_1, \sigma_1), (\mu_2, \sigma_2), \cdots, (\mu_m, \sigma_m)\}$ in this region, and assign weight to each cluster (μ_i, σ_i) based on the approaches proposed in Section 4.2.

5 SIMILARITY MEASURE

Based on the extracted spatio-temporal features and their weights, we develop novel approaches to measure the similarity between two users.

(1) **Stay Region Similarity**: Assume the stay regions of u_1 and u_2 are $\{(R_1^1, \omega(R_1^1)), (R_1^2, \omega(R_1^2)), \cdots, (R_1^m, \omega(R_1^m))\}$ and $\{(R_2^1, \omega(R_2^1)), (R_2^2, \omega(R_2^2)), \cdots, (R_2^n, \omega(R_2^n))\}$, the stay region similarity between u_1 and u_2 is defined as:

$$S(u_1, u_2)_r = \sum_{i=1}^{m}\sum_{j=1}^{n} S(R_1^i, R_2^j)\omega(R_1^i)\omega(R_2^j) \qquad (9)$$

where $S(R_1^i, R_2^j)$ is calculated according to Eq.(1)

(2) **Global Time Similarity**: Assume the global time distributions of u_1 and u_2 are $\{(T_1^1, \omega(T_1^1)), (T_1^2, \omega(T_1^2)), \cdots, (T_1^k, \omega(T_1^k))\}$ and $\{(T_2^1, \omega(T_2^1)), (T_2^2, \omega(T_2^2)), \cdots, (T_2^l, \omega(T_2^l))\}$, the global time similarity $S(u_1, u_2)_t$ is given as:

$$S(u_1, u_2)_t = \sum_{i=1}^{k}\sum_{j=1}^{l} S(T_1^i, T_2^j)\omega(T_1^i)\omega(T_2^j) \qquad (10)$$

where $S(T_1^i, T_2^j)$ is calculated based on Eq.(7).

(3) **Local Time Similarity**: Assume the time distribution in a stay region $(R_1^i, \omega(R_1^i))$ of u_1 is $\{(T_1^1, \omega(T_1^1)), (T_1^2, \omega(T_1^2)), \cdots, (T_1^k, \omega(T_1^k))\}$, and the time distribution in an extracted stay region $(R_2^j, \omega(R_2^j))$ of u_2 is $\{(T_2^1, \omega(T_2^1)), (T_2^2, \omega (T_2^2)), \cdots, (T_2^l, \omega(T_2^l))\}$, the local time similarity in these two regions is defined as $S(R_1^i, R_2^j)\omega(R_1^i)\omega(R_2^j) \times \sum_{i=1}^{k}\sum_{j=1}^{l} S(T_1^i, T_2^j) \omega(T_1^i)\omega(T_2^j)$. The similarity between u_1 and u_2 based on this feature is defined as:

$$\begin{aligned} S(u_1, u_2)_{rt} = &\sum_{i=1}^{m}\sum_{j=1}^{n}(S(R_1^i, R_2^j)\omega(R_1^i)\omega(R_2^j)\cdot \\ &\sum_{i=1}^{k}\sum_{j=1}^{l} S(T_1^i, T_2^j)\omega(T_1^i)\omega(T_2^j)) \end{aligned} \qquad (11)$$

Finally, the similarity between u_1 and u_2 is defined as:

$$S(u_1, u_2) = S(u_1, u_2)_r + S(u_1, u_2)_t + S(u_1, u_2)_{rt} \qquad (12)$$

6 EXPERIMENT

6.1 Experiment Settings

Three real-world datasets, which contain the trajectory data in Beijing and the check-in data in Foursquare, Twitter, and Instagram, are used to conduct experiments. The details of these datasets are presented in Table 3.

Beijing Walk Trajectories - Beijing Car Trajectories. The first dataset is collected by (Microsoft Research Asia) GeoLife project in Beijing [2] with different transportation modes, such as walk, car, bus, bike, etc. This dataset records a broad range of users' outdoor movements, such as shopping, sightseeing, driving, and cycling. To investigate the performance of the proposed model in linking the same users across different domains, we select the walk trajectory data and the car trajectory data, since they contain

[2] https://www.microsoft.com/en-us/research/people/yuzheng

Table 3: Properties of the Given Datasets

Dataset	Domain	Users	Trajectories	Locations/Check-ins
BJW-BJC	Walk	182	14337	2190957
	Car	182	5475	925380
FS-TW	Foursquare	89	-	3924
	Twitter	89	-	35384
IT-TW	Instagram	908	-	267029
	Twitter	908	-	357949

more locations than other transportation modes. The selected dataset contains 182 actual linked user pairs in real life. It contains 14337 walk trajectories with 2190957 locations, and 5475 car trajectories with 925380 locations. Each location is a tuple in the form of (user-id, trajectory-id, latitude, longitude, timestamp).

Foursquare-Twitter. Foursquare and Twitter are popular social networks, where users can post status associated with location information and timestamps. To evaluate the performance of our model in connecting the users across these two platforms with spatio-temporal features, we use the dataset provided by [26][14]. However, some users in the given dataset only contain one or two check-in records, which are insufficient to extract users' spatio-temporal features. Consequently, we preprocess the original dataset, and the users containing less than 30 check-ins are removed. Finally, we obtain a new dataset with 89 linked user pairs as the ground truth. The dataset contains 3924 check-ins in Foursquare and 35384 check-ins in Twitter, where each check-in is stored as a tuple (user-id, latitude, longitude, timestamp).

Instagram-Twitter. Instagram is another popular social network, where users can share photos associated with locations and timestamps. To link users across Instagram and Twitter, we use the dataset provided by [14]. Some users in the dataset also only contain one or two check-in records. As a consequence, we remove users who have less than 30 check-ins. Finally, the new dataset consists of 908 ground truth user pairs, and contains 267029 check-ins in Instagram and 357949 check-ins in Twitter, where each check-in is in the form of (user-id, latitude, longitude, timestamp).

In the rest of paper, we use BJW-BJC, FS-TW, and IT-TW to denote the first, second, and third dataset respectively for the sake of convenience.

Compared Algorithms. To investigate the performance of the proposed model, we compare the performances of the following algorithms.

Baseline 1 (GC): Existing work [9] has proposed a method to discretize the space into grid cells, and calculate the density $f(c)$ for each grid cell c with a kernel method:

$$f(c) = \frac{1}{n\gamma^2}\sum_{i=1}^{n}\frac{1}{2\pi}\exp(-\frac{|c-loc_i|^2}{2\gamma^2}$$
$$\gamma = \frac{1}{2}(\delta_x^2+\delta_y^2)^{\frac{1}{2}}n^{-\frac{1}{6}} \tag{13}$$

where $|c-loc_i|$ is the distance between c and location loc_i, δ_x and δ_y denote the standard deviations of the whole locations in its x and y-coordinates, respectively. Next, the algorithm returns the top-15% grid cells with the maximum density as dense regions that a user will visit repeatedly. Based on these regions, we calculate the similarity between users. For instance, given two users u_1 and u_2 with returned grid cells $C_1 = \{c_{11}, c_{12}, \cdots, c_{1m}\}$ and $C_2 = \{c_{21}, c_{22}, \cdots, c_{2n}\}$, respectively. The similarity $S(u_1, u_2)$ is defined as:

$$S(u_1,u_2) = \sum_{c_{1i}\in C_1}^{m}\sum_{c_{2j}\in C_2}^{n}\frac{|c_{1i}\cap c_{2j}|}{|c_{1i}\cup c_{2j}|} \tag{14}$$

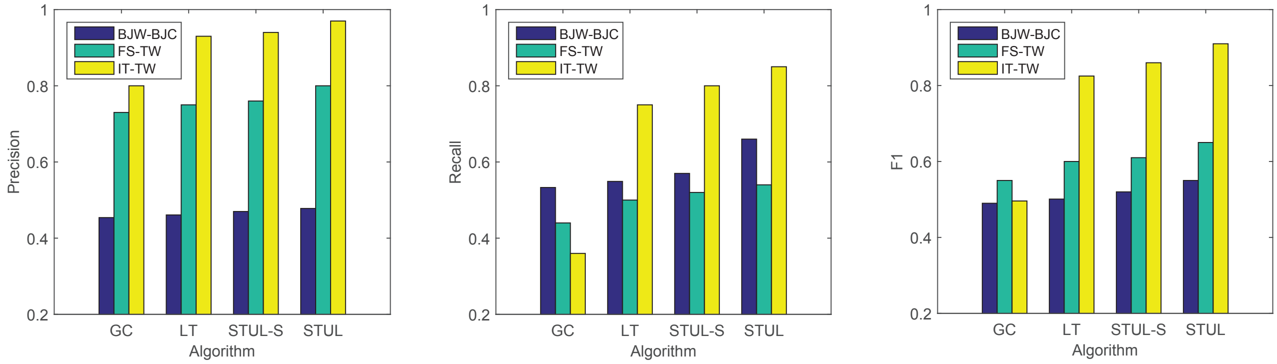
Baseline 2 (LT): Locations and times are divided into bins, where each record in region l during time interval t is associated with bin (l, t). Then, the user similarity is measured by the corresponding bins [14].

Baseline 3 (STUL-S): This is a simplified version of STUL, where the extracted stay region distribution and time distribution are directly used to measure the similarity between users without consideration of their weights. The similarity function is defined as:

$$\begin{aligned} S(u_1,u_2) = &\sum_{i=1}^{m}\sum_{j=1}^{n}S(R_{1i},R_{2j}) + \sum_{i=1}^{k}\sum_{j=1}^{l}S(T_{1i},T_{2j}) \\ &+ \sum_{i=1}^{m}\sum_{j=1}^{n}S(R_{1i},R_{2j})\sum_{i=1}^{k}\sum_{j=1}^{l}S(T_{1i},T_{2j}) \end{aligned} \tag{15}$$

STUL: All extracted features are taken into account to measure the similarity between users, i.e., the similarity $S(u_1, u_2) = S(u_1,u_2)_r + S(u_1,u_2)_t + S(u_1,u_2)_{rt}$ defined in Eq.(12). Unlike STUL-S, we assign different weights to the extracted features based on their uniqueness and discriminability.

Evaluation Methods. To evaluate the performances of GC, LT, STUL-S, and STUL, we use precision, recall, and F1. Given two sets of users $U_1 = \{u_{11}, u_{12}, \cdots, u_{1n}\}$ and $U_2 = \{u_{21}, u_{22}, \cdots, u_{2n}\}$, we assume $< u_{1i}, u_{2i} >$ $(1 \le i \le n)$ is an actual linked pair in real life. As presented in algorithm 1, the pair $< u_{1i}, u_{2j} >$ is added into the result if the similarity $S(u_{1i}, u_{2j})$ is larger than the user defined threshold θ. Note that, we can return the user pair $< u_{1i}, u_{2j} >$ with the maximum $S(u_{1i}, u_{2j})$, if there is no $S(u_{1i}, u_{2j}) \ge \theta$ and there does exist an actually linked user pair $< u_{1i}, u_{2j} >$. The precision is defined as the fraction of user pairs contained by the returned result that are correctly linked, and recall is defined



(a) Precision (b) Recall (c) F1

Figure 2: Performances of the proposed algorithms in different datasets

as the fraction of the actual linked user pairs contained by the returned result [12],

$$Recall = \frac{k}{n},\ Precision = \frac{k}{m}$$
$$F1 = \frac{2 * Recall * Precision}{Recall + Precision}$$

where $n(n = |U_1| = |U_2|)$ is the number of given user pairs, m is the number of returned user pairs, and k is the number of actual linked user pairs in the returned result.

6.2 Performance Evaluation

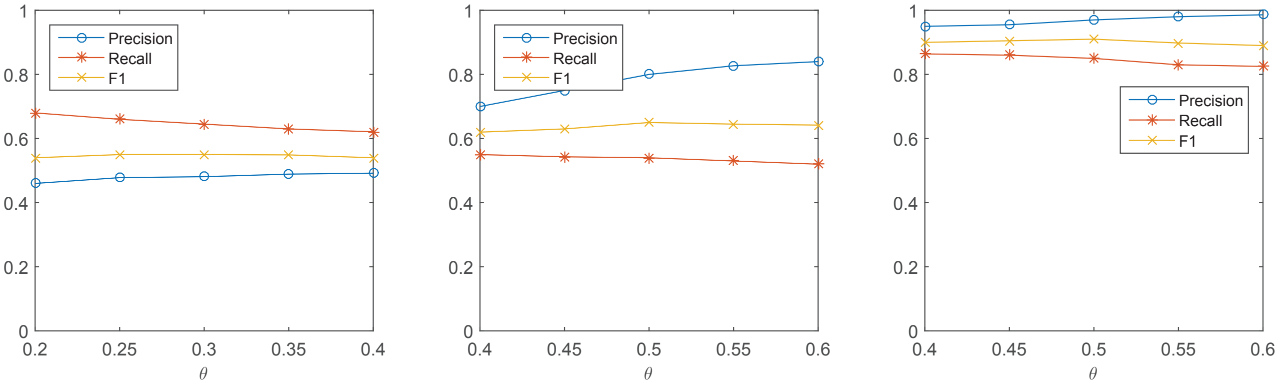
The precision, recall, and F1 of all algorithms are presented in Fig.2. As expected, LT has better performance than GC, since GC only considers grid cells with high density, whereas both spatial and temporal information are taken into account in LT. The simplified model STUL-S outperforms the discretization based methods GC and LT, and the reason is twofold. On one hand, STUL-S has considered the continuity of user regions. Compared with the rectangle grid cells, the stay regions extracted by STUL-S with arbitrary shapes are more likely to model the real regions of a user. On the other hand, STUL-S has considered the continuity of time, where the time clusters centered on some time points are extracted by the Gaussian Mixture Model. Unlike STUL-S, STUL assigns different weights to the extracted features, where the discriminative features are highlighted with larger weights and the common features are lightened with smaller weights. Then, the actual linked users are more likely to be connected with these features. As a result, STUL performs best in all datasets.

In addition, the results in Fig.2 demonstrate the universality of the proposed model STUL. This is because the trajectories in BJW-BJC are densely sampled, where the adjacent points in a trajectory are sampled in a short time period. The check-in records in Foursquare, Twitter, and Instagram are sampled with a low rate, where the time span between two check-ins is usually large.

Performance w.r.t. Varied θ. As discussed in algorithm 1, given two user datasets U_1 and U_2, the user pair $< u_{1i}, u_{2j} >$ $(u_{1i} \in U_1, u_{2j} \in U_2)$ is added into the result if $S(u_{1i}, u_{2j}) \geq \theta$. Naturally, the increase of θ will lead to the decrease of recall, as many candidate user pairs are filtered. After filtering user pairs with small similarities, the remaining user pairs with high similarities are more likely to be the actual linked user pairs, as which usually have larger similarities than unmatched user pairs in real life. Consequently, the increase of θ also leads to the increase of precision. To balance the precision and recall, we set $\theta = 0.25$ in BJW-BJC, $\theta = 0.5$ in FS-TW, and $\theta = 0.5$ in IT-TW.

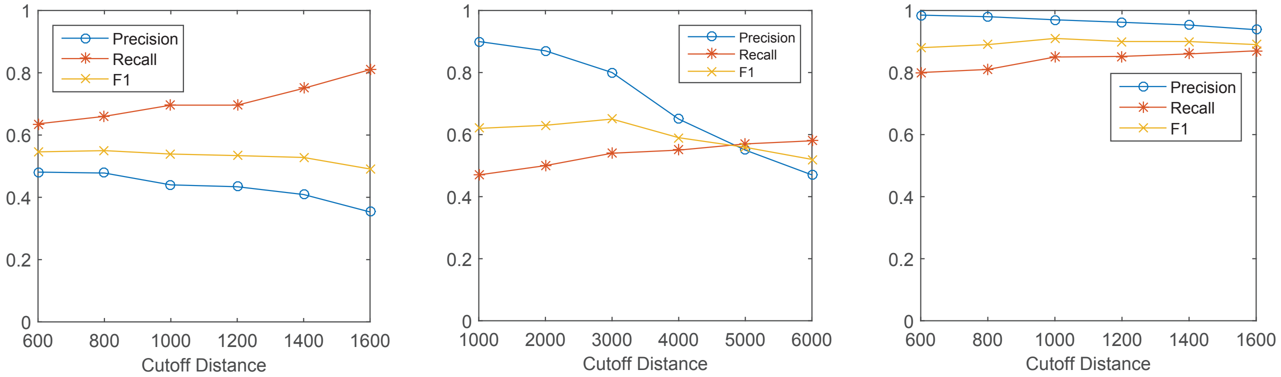
Performance w.r.t. Varied Cutoff Distance. The cutoff distance d_c is an important parameter of STUL, as p and δ are calculated based on d_c, where points within d_c are considered. From Fig.4, we observe that STUL achieves higher recall with the increase of d_c. This is because, given a larger d_c, the extracted stay regions will contain more points and have larger areas. Then, the across domain users are more likely to have common areas and larger user similarity, which leads to the increase of recall. However, for users on the same platform, a larger d_c will lead to the decrease of the distinction of their stay regions, since these regions tend to contain same points and have similar distribution with a larger d_c. Then, it is difficult to link one of them to the actual linked user from another platform. This difficulty leads to the decrease of of precision. Considering the diversity of the given datasets, we set different cutoff distances to balance the precision and recall, i.e., $d_c = 800\ m$ in the dataset BJW-BJC, $d_c = 3000\ m$ in FS-TW, and $d_c = 1000\ m$ in IT-TW.

Performance w.r.t. Varied ξ. Another import parameter of STUL is ξ, as the top-k centers with the maximum $\xi = p \times \delta$ are returned. In Fig.5, we set $p = (0, 1, 2, 3, 4, 5)$, $\delta = (0, 100, 200, 300, 400, 500)$, thus ξ is varied from 0 to 2500. This figure shows that the model STUL achieves the best performance in some certain points, although the change in performance is small when using different ξ. A too large or too small ξ is not appropriate. On one hand, given a small ξ, the users in a platform tend to have similar features as too many stay regions are returned. The overlap of these features leads to the bad performance of STUL. On the other



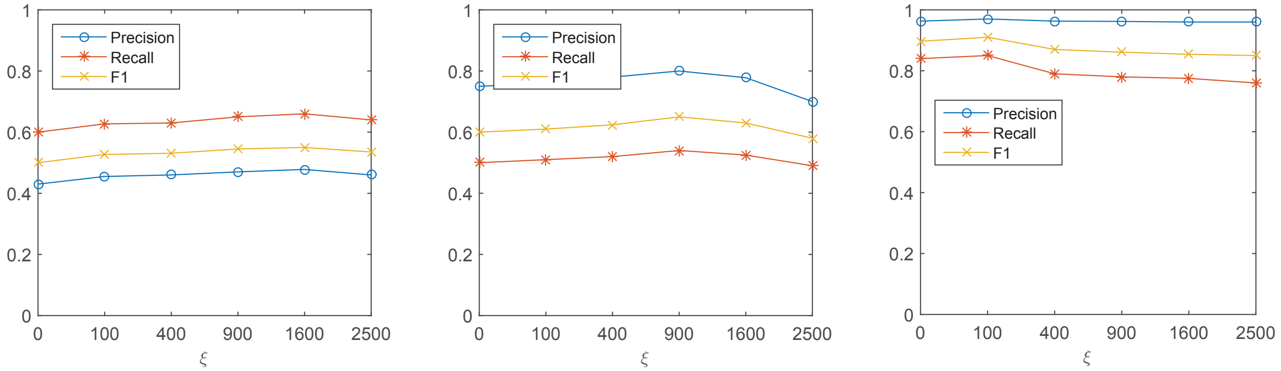
(a) BJW-BJC (b) FS-TW (c) IT-TW

Figure 3: Performance of STUL w.r.t. varied θ



(a) BJW-BJC (b) FS-TW (c) IT-TW

Figure 4: Performance of STUL w.r.t. varied cutoff distance



(a) BJW-BJC (b) FS-TW (c) IT-TW

Figure 5: Performance of STUL w.r.t. varied ξ

hand, given a large ξ, we can extract important and discriminative stay regions, as many centers with small ξ are filtered. However, the number of the returned regions is small, which also leads to the bad performance of STUL. To achieve the best performance, we set $\xi = 1600$ in BJW-BJC, $\xi = 900$ in FS-TW, and $\xi = 100$ in IT-TW.

Efficiency. Besides the high effectiveness, STUL is very efficient, where the average time it takes to find an actual linked user pair is 1.3s. Additionally, the stay regions extraction, time clustering, and user similarity computation can be parallelized with MapReduce or Spark. Thus, STUL is scalable to large scale datasets.

7 CONCLUSION AND FUTURE WORK

Analyzing the users' spatio-temporal behaviors to achieve user linkage has received great attention. Unlike the existing studies that model the users' behaviors with discretization methods, such as grid cells [9] and bins [14], we develop novel approaches that consider the continuity of the spatio-temporal features. From spatial perspective, a density-based method is developed to extract the stay regions that a user will visit repeatedly. From temporal perspective, the Gaussian Mixture Model (GMM) is used to extract the global time distribution and local time distribution of a user. Additionally, we assign different weights to the extracted features with the goal of linking users more precisely. Next, the similarities between users are measured based on the extracted features, and the pair-wise users with similarity scores higher than the given threshold are returned. The extensive experiments on real-world datasets demonstrate the high performance and universality of the proposed model STUL. In the future work, to further improve the efficiency of cross-domain user linkage, we can construct index structure from spatial and temporal perspective to prune the unmatched user pairs.

Acknowledgments. This work is partially supported by ARC Discovery Project (DP170103954) and ARC Discovery Early Career Researcher Award (DE160100308). It is also partially supported by the National Natural Science Foundation of China under Grant Nos. 61572335, 61572336, 61303019, and 61402312, the Natural Science Foundation of Jiangsu Province of China under Grant No. BK20151223, the Natural Science Foundation of Jiangsu Provincial Department of Education of China under Grant No. 12KJB520017, and Collaborative Innovation Center of Novel Software Technology and Industrialization, Jiangsu, China.

REFERENCES

[1] Michael Backes, Pascal Berrang, Oana Goga, Krishna P Gummadi, and Praveen Manoharan. 2016. On Profile Linkability despite Anonymity in Social Media Systems. In *Proceedings of the Workshop on Privacy in the Electronic Society*. 25–35.

[2] Sergey Bartunov, Anton Korshunov, Seung-Taek Park, Wonho Ryu, and Hyungdong Lee. 2012. Joint link-attribute user identity resolution in online social networks. In *Proceedings of the Workshop on Social Network Mining and Analysis (SNA-KDD)*.

[3] Nurjahan Begum, Liudmila Ulanova, Jun Wang, and Eamonn Keogh. 2015. Accelerating dynamic time warping clustering with a novel admissible pruning strategy. In *KDD*. 49–58.

[4] Wei Chen, Feida Zhu, Lei Zhao, and Xiaofang Zhou. 2016. When Peculiarity Makes a Difference: Object Characterisation in Heterogeneous Information Networks. In *DASFAA*. 3–17.

[5] Oana Goga, Patrick Loiseau, Robin Sommer, Renata Teixeira, and Krishna P Gummadi. 2015. On the reliability of profile matching across large online social networks. In *KDD*. 1799–1808.

[6] Elham Hoseini and Eghbal G Mansoori. 2016. Selecting discriminative features in social media data: An unsupervised approach. *Neurocomputing* 205 (2016), 463–471.

[7] Henry F. Inman and Edwin L. Bradley. 1989. The overlapping coefficient as a measure of agreement between probability distributions and point estimation of the overlap of two normal densities. *Communications in Statistics: Theory and Methods* 18, 10 (1989), 3851–3874.

[8] Nikos Karampatziakis and Paul Mineiro. 2014. Discriminative Features via Generalized Eigenvectors. In *ICML*. 494–502.

[9] Zhenhui Li, Bolin Ding, Jiawei Han, Roland Kays, and Peter Nye. 2010. Mining Periodic Behaviors for Moving Objects. In *KDD*. 1099–1108.

[10] Chengjun Liu. 2011. Extracting discriminative color features for face recognition. *Pattern Recognition Letters* 32, 14 (2011), 1796–1804.

[11] Jing Liu, Fan Zhang, Xinying Song, Young-In Song, Chin-Yew Lin, and Hsiao-Wuen Hon. 2013. What's in a name?: an unsupervised approach to link users across communities. In *WSDM*. 495–504.

[12] Siyuan Liu, Shuhui Wang, Feida Zhu, Jinbo Zhang, and Ramayya Krishnan. 2014. Hydra: Large-scale social identity linkage via heterogeneous behavior modeling. In *SIGMOD*. 51–62.

[13] Guojun Qi, Charu C Aggarwal, and Thomas Huang. 2013. Link prediction across networks by biased cross-network sampling. In *ICDE*. 793–804.

[14] Christopher Riederer, Yunsung Kim, Augustin Chaintreau, Nitish Korula, and Silvio Lattanzi. 2016. Linking Users Across Domains with Location Data: Theory and Validation. In *WWW*. 707–719.

[15] Alex Rodriguez and Alessandro Laio. 2014. Clustering by fast search and find of density peaks. *Science* 344, 6191 (2014), 1492–1496.

[16] Friedrich Schmid and Axel Schmidt. 2006. Nonparametric estimation of the coefficient of overlapping?theory and empirical application. *Computational statistics and data analysis* 50, 6 (2006), 1583–1596.

[17] Jan Vosecky, Dan Hong, and Vincent Y Shen. 2009. User identification across multiple social networks. In *Networked Digital Technologies*. 360–365.

[18] Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Sadiq, and Xiaofang Zhou. 2017. ST-SAGE: A Spatial-Temporal Sparse Additive Generative Model for Spatial Item Recommendation. *TIST* 8, 3 (2017).

[19] Min Xie, Hongzhi Yin, Hao Wang, Fanjiang Xu, Weitong Chen, and Sen Wang. 2016. Learning graph-based poi embedding for location-based recommendation. In *CIKM*. 15–24.

[20] Hongzhi Yin, Bin Cui, Xiaofang Zhou, Weiqing Wang, Zi Huang, and Shazia Sadiq. 2016. Joint Modeling of User Check-in Behaviors for Real-time Point-of-Interest Recommendation. *TOIS* 35, 2 (2016).

[21] Hongzhi Yin, Zhiting Hu, Xiaofang Zhou, Hao Wang, Kai Zheng, Quoc Viet Hung Nguyen, and Shazia Sadiq. 2016. Discovering interpretable geo-social communities for user behavior prediction. In *ICDE*. 942–953.

[22] Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Quoc Viet Hung Nguyen. 2016. Adapting to user interest drift for poi recommendation. *TKDE* 28, 10 (2016), 2566–2581.

[23] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Magnenat-Thalmann Nadia. 2013. Who, where, when and what: discover spatio-temporal topics for twitter users. In *KDD*. 605–613.

[24] Reza Zafarani and Huan Liu. 2009. Connecting Corresponding Identities across Communities. In *ICWSM*. 354–357.

[25] Reza Zafarani and Huan Liu. 2013. Connecting users across social media sites: a behavioral-modeling approach. In *KDD*. 41–49.

[26] Jiawei Zhang, Xiangnan Kong, and Philip S. Yu. 2014. Transferring Heterogeneous Links Across Location-based Social Networks. In *WSDM*. 303–312.

[27] Yutao Zhang, Jie Tang, Zhilin Yang, Jian Pei, and Philip S Yu. 2015. Cosnet: connecting heterogeneous social networks with local and global consistency. In *KDD*. 1485–1494.

[28] Yu Zheng. 2015. Methodologies for cross-domain data fusion: An overview. *IEEE transactions on big data* 1, 1 (2015), 16–34.

[29] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. 2009. Mining interesting locations and travel sequences from GPS trajectories. In *WWW*. 791–800.

[30] Yuanyuan Zhu, Jeffrey Xu Yu, Hong Cheng, and Lu Qin. 2012. Graph classification: a diversified discriminative feature selection approach. In *CIKM*. 205–214.