Mining Friendships Through Spatial-temporal Features in Mobile Social Networks

Jianwei Niu*, Danning Wang*, Jie Lu*
*State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering, Beihang University, Beijing 100191, China
Email: niujianwei@buaa.edu.cn; wangdanning@buaa.edu.cn; lu_jiee@163.com

***Abstract*—With the rapid popularization of smartphones and tablets, there are thousands of applications based on mobile social networks. The big data from these networks provide a huge potential to shed light on the mobility patterns of users. These big data enable a deeper understanding of users' preferences and behaviors and will help us mine users' friendship in both physical and digital worlds. In this paper, we firstly divide user mobility patterns into different categories to portray the characteristics of user encounter more precisely. Then, with combining proximity data from bluetooth devices and location data from cellular towers, we introduce a set of spatial-temporal features, including the encounter entropy, which measures the probability of encounters between different mobile users. Using these spatial-temporal features, we provide a novel model to infer user friendship by analyzing the social context of users and their encounters. To address the class imbalance problem in the dataset and improve the prediction accuracy of friendship, we employ the sampling method and evaluate our model with three different classifiers. The experimental results show that our encounter entropy feature has a striking effect to infer user friendship, and our model based on these spatial-temporal features can achieve pretty good accuracy in predicting friendship over real human mobility traces without privacy-sensitive information disclosure.**

***Index Terms*—mobile social networks; friendship inferring; proximity and location data; spatial-temporal features**

I. Introduction

Over the last couple of years, Location-Based Social Networks (LBSNs) and Mobile Social Networks (MSNs) have gained a great attention from both the research community and the industry [1]. More and more mobile applications and systems are developed based on these networks, and attract millions of users. Compared with traditional social networks, LBSNs and MSNs achieve further growth in providing the location-based information. These location features, integrated with the temporal features and social connections revealed through social networks, provide an unparalleled opportunity to study human social behaviors and promote a wide variety of applications and services such as tour planning [2], interest or product recommendation [3], friend sensing [4], location-based advertising [5], and traffic forecasting [6]. The prevalence of these applications and services in turn calls for systematic research on new computing techniques for discovering knowledge from user trajectory data [7].

However, there are many challenges in this space. One of the challenges is to infer properties of human social behaviors from a variety of data [8]. Eagle et al. introduced measures of user similarity based on user mobility and employed these measures to infer network structures by using mobile phone data [9]. Results showed that phone communication between users by far contributed most to predict friendship. However, it is difficult to perform direct measurements, say keeping track of users call logs or text messages data, due to the concerns of compromising user privacy. Cranshaw et al. introduced a set of location-based features based on Facebook's social network for inferring social network ties from co-location features and user mobility data [10]. Unfortunately, the performance of their approach is not encouraging (the best accuracy and recall ratios are 68% and 37%, respectively) according to their experimental results.

In this paper, we firstly divide user mobility patterns into different categories to portray the encounter features between users more precisely compared with [11]. Then we introduce a novel set of spatial-temporal features which combine proximity data from bluetooth devices and location data from the cell tower together, including encounter entropy features, for analyzing the social context of users and their encounters. Statistical analysis in our feature dataset reveals the problem of class imbalance which can mislead us since the class distribution is heavily biased towards encounters between strangers. To address this problem and improve the accuracy of friendship prediction, we employ different sampling methods [12] and evaluate our model with three different classifiers. By applying these features and sampling methods, our model can achieve pretty good accuracy and F-score in predicting friendship without using privacy-sensitive information like call logs or text messages from smartphones based on real human mobility traces.

This work makes the following primary research contributions:

1. We divide user mobility patterns into different categories based on our observations and analyses of users' social context in order to portray the encounter characteristics between users more precisely.
2. We introduce a novel set of spatial-temporal features, including encounter entropy, which measure the diversity of encounters between users. Our results also show that our proposed encounter entropy is very effective to infer friendships.

978-1-4673-8590-9/15/$31.00 ©2015 IEEE

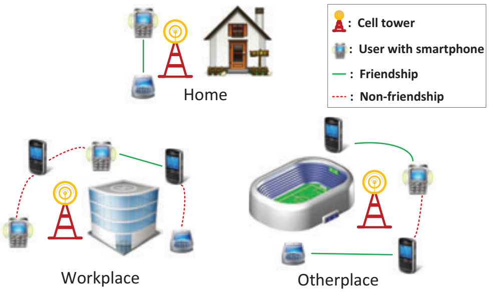


Fig. 1. Mining friendships with proximity and location data. Users with their smartphones encounter with each users at a certain time and location. Note that when the two users are friends, their connections are marked with a green line; otherwise, their connections are marked with a red line.

3. We employ different sampling methods on the imbalanced feature dataset and make a comparison between these sampling methods to address the issue of class imbalance. We validate the performance of our proposed model with three different classifiers over real human traces.

The remaining sections of this paper are organized as follows. Section 2 surveys existing literature in this area. Section 3 describes our proposed model. In Section 4, we describe the spatial-temporal features. In Section 5, our experimental results are presented. Lastly, conclusions are given in Section 6.

II. RELATED WORK

Several important studies have been done to reveal huge potential of using data collected from smartphones to study human social behaviors. In [13], Li et al. mined the similarity geographically between users based on their location histories. They proposed a framework called hierarchical-graph-based similarity measurement for effectively modeling individual's location history and measuring the similarity among users in geographic information systems. In [14], Hristova et al. applied a new model that allows them to distinguish between social ties with varying strength. They built multiplex interaction graph by combining different communication layers to capture different types of interactions and relationships between the same two nodes. They found that strong social ties are characterized by maximal use of communication channels, while weak ties by minimal use.

Eagle et al. tried to infer friendship network structures by exploiting user mobility and interaction data collected from smartphones [9]. They introduced a set of features such as the proximity of the users at work on workday or weekend, the proximity of the users who were off-campus on weekday or weekend, whether there was mobile phone communication between them. A regression analysis was conducted using self-report survey data for the actual user relationships to study which factors contribute most to inferring friendship. The results revealed that phone communication was the most important predictor for friendship inferring, followed by the proximity on weekend.

In [4], Quercia et al. used short-range technologies (e.g., Bluetooth) on users' mobile phones. Users could keep track of other phones in their proximity and 'sense' their friends. They proposed a framework called FriendSensing that automatically recommended friends by logging and analysing co-location data based on social network theories of geographical proximity. They validated that the strategy of keeping track of how much time people spend co-located (duration) outperforms the strategy of simply keeping track of how may times people meet each other (frequency). In [10], Cranshaw et al. exploited the location histories of 489 users of a location sharing social network for relationships inferring by analyzing the user mobility patterns and structural properties of their social network. They defined location entropy, which based on location features, to analyze the social context of a geographic region. Using these features, they then proposed a framework for inferring social network ties from co-location data and inferring the number of friends from user mobility data. They found that the entropy of a location was a valuable tool for analyzing social mobility data. By their definitions, locations with high entropy were precisely the places where chance encounters were more likely to happen, thus locations with high entropy were thus much more likely to be random occurrences than locations with low entropy.

III. Model Description

In this paper, we use the dataset from Reality Mining study [15] to infer friendship between users. The Reality Mining study consists of 97 subjects (students and faculty at MIT) with Nokia 6600 smartphones. The dataset [16] includes mobile call logs, bluetooth devices in proximity, cell tower IDs and other information from the context application from each individual over the course of the academic year. In addition, self-report survey data were conducted online which include dyadic questions regarding the average reported proximity and friendship with the other subjects, as well as questions concerning the individual's general satisfaction with his or her work group. Note that we label friendship between users according to the users' self-report survey data. The whole dataset represented approximately 450,000 hours of information about users' location, interaction, communication and device usage behavior.

In this paper, we try to infer users' friendship through continuous tracking of their proximity data from bluetooth interfaces and location data from cell towers. The ground truth about which node pairs are friends in real life comes from the self-report survey data. We will give a detail description of modeling the dataset in this section.

A. The description of the original dataset

We primarily put emphasis on proximity and location data collected from users' mobile phones. The dataset contains 97 subjects (users) which can freely move to different districts and interact with others through their mobile phones. We will give the following mathematical symbols to help us portray users' mobile social network in the dataset.

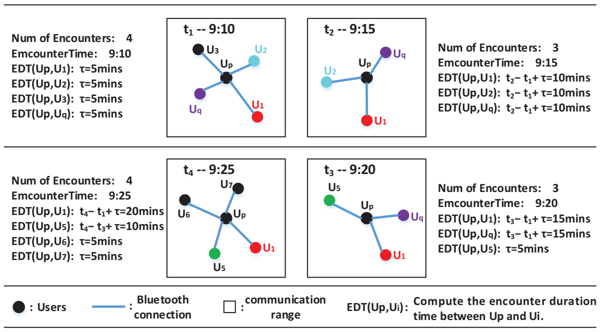


Fig. 2. The bluetooth encounter network for a specific user u_p. At time t_1, user u_p discovers u_q, u_1, u_2 and u_3. We can compute the Encounter Duration Time (EDT) between u_p and u_i according to Eq. 1.

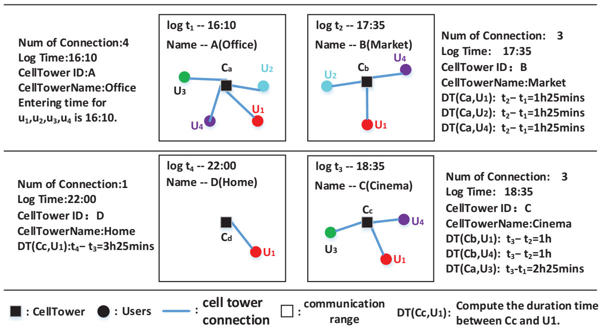


Fig. 3. Cell tower connection network. Users u_1, u_2, u_3 and u_4 are in the cell tower c_1's communication range at t_1. Places such as 'Office' and 'Home' are the names that users named the locations. We can calculate the connection duration time between c_i and u_p simply by subtracting two enter time when u_p transits from c_i to c_j.

1) Let $U = \{u_1, u_2, \ldots, u_m\}$ denote the user set where m represents the total number of the users. In the user set, we can identify every user u_i by the ID number.
2) Let $C = \{c_1, c_2, \ldots, c_n\}$ denote the cell tower set where n represents the total number of cell towers. Each cell tower has been assigned an area ID that can be logged by the mobile phones.
3) Let $T = \{t_1, t_2, \ldots, t_k\}$ denote the timestamp set. In the dataset, an entry time will be recorded when a user transits to a new location (cellular tower). In addition, the user will discover other bluetooth devices (users) at t_i on each scan intermittently. In this paper, we consider all these timestamps t_i as a set T.

B. Bluetooth Encounter and Cell Tower Location Networks

When a user u_j is discovered by a periodic bluetooth scan performed by another user u_i, we call these two users u_i and u_j encounter through bluetooth. In order to better describe the bluetooth encounter between users, we take users and their interactions as a network (*The Bluetooth Encounter Network*). Similarly, we consider logs which record users' transitions between cellular towers as another network (*The Cell Tower Location Network*). We describe the two networks as follows:

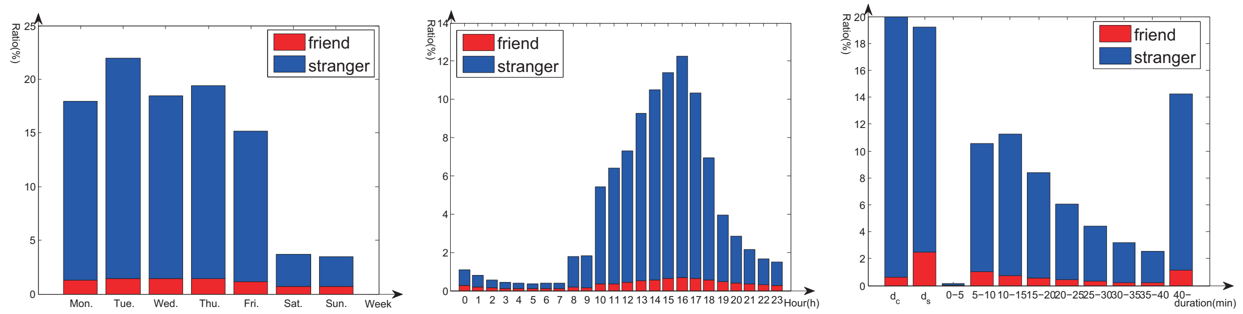
1) ***The Bluetooth Encounter Network****:* In *the Bluetooth Encounter Network*, a node represents a user, and an edge (u_p, u_q) between users u_p and u_q represents that u_p encounters u_q via bluetooth (u_p is discovered by the bluetooth scan performed by u_q) at a certain time. In the dataset, more than one nodes may be discovered on each encounter. Let $EU_p = \{(u_p, U_p, t_k) | u_p \in U, U_p \subset U, t_k \in T\}$ denote as the set of users discovered by user u_p. The elements of the triple (u_p, U_p, t_k) represent that user u_p encounters other users (defined as the set of U_p) at a certain time t_k. In fact, we refer to this network as an undirected graph. The number of vertices (users) in this network is 97. We analyze this network and find that there are 758,904 edges in the bluetooth encounter network, and only 64,065 edges represent encounters between friends. In other words, above 90% of encounters occurred are chance (irrelevant) encounters in the dataset. Fig. 2 shows the the bluetooth encounter network for a specific user u_p.

2) ***The Cell Tower Location Network****:* We create an undirected graph based on user's cellular tower location logs. An edge exists between u_p (user) and c_r (cell tower) if user u_p is within the communication range of the cell tower c_r at time t_k. Similar to the bluetooth encounter network, $EC_p = \{(u_p, c_r, t_k) | c_r \in C, t_k \in T\}$ is denoted as the set of time-stamped transitions among different cell towers of user u_p. The elements of the triple (u_p, c_r, t_k) represent that the mobile phone user u_p transits to cellular tower c_r at time t_k. Using the cellular towers IDs and the respective transition timings (timestamps when users hand off between different cellular towers), a user's position can be localized to within 100-200m in the dataset. In the cell tower location network, there are 97 users and 2,873,251 edges. Fig. 3 shows an example of the cell tower connection networks.

C. Mobility patterns

In our proposed model, we firstly divide user mobility patterns (the period during the encounter, the encounter duration time and the location the encounter takes place) into different categories to portray the encounter of users precisely.

1) ***Time Periods****:* Social relationships between users have huge impact on the period during the encounter. For example, people prefer to spend more time with their friends than colleagues on holiday. Therefore, we analyse the dataset and Fig. 4(a) shows the distribution of encounters between friends and strangers from Monday to Sunday. We observe that changes in the ratio of the encounters between strangers are more significant than that between friends, which illustrates that links (encounters) between friends are more stable than strangers. Specifically, changes in the ratio of encounters between friends are from about 0.66% to 1.18% (0.66% $\sim$ 0.68% for weekends and 1.14% $\sim$ 1.18% for weekdays) while changes in the ratio of encounters between strangers are from 2.80% to 20.54% (2.80% $\sim$ 2.97% for weekends and 13.99% $\sim$ 20.54% for weekdays). As shown in Fig. 4(b), the changes in the ratio of encounters between strangers (0.68% $\sim$ 11.58%) are more significant than that between friends (0.09% $\sim$ 0.26%). The encounters between people occurred after 18:00pm should be interpreted differently with that during the working hours (8:00am $\sim$ 18:00pm). That is to say, encounters



(a)

(b)

(c)

Fig. 4. (a) The distribution of encounters between friends and strangers occurred from Monday to Sunday. (b) The distribution of encounters between friends and strangers occurred in 24 hours a day. (c) The distribution of the encounter duration for friends and strangers. Note that d_s and d_c represent the first and the second situation mentioned in Section III (C:Encounter Duration).

TABLE I
DESCRIPTION OF THE TIME PERIOD DURING THE ENCOUNTER.

Notation	Time Period	Description
T_i^j	22:00 - 8:00	People usually have a sleep at night
	8:00 - 18:00	People usually work in the day
	18:00 - 22:00	People usually stay with friends or family

$i \in \{r, k, s\}$ represents the 3 different time periods and
$j \in \{h, w\}$ represents weekday or weekend when encounters occurred.

TABLE II
DESCRIPTION OF THE ENCOUNTER DURATION TIME (EDT).

Notation	Duration	Description
D_i	Chance	The ratio of the chance encounter is 19.97%
	0 - 5mins	The ratio of this duration is 19.34%
	5 - 30mins	The ratio of this duration is 60.68%
	30mins -	The ratio of this duration is 19.98%

$i \in \{c, s, m, l\}$ represents the 4 different EDT categories
when encounters occurred.

are more likely to be chance encounters if the encounters occurred during working hours.

Based on this consideration, we divide the timestamp set T into six subsets: $T = \{T_i^j, i \in \{r, k, s\}, j \in \{h, w\}\}$ where r, k, s represent 3 different time periods during the encounter (in Table I) and h, w represent weekday (Monday to Friday) or weekend (Saturday and Sunday) when the encounter occurred.

2) ***Encounter Duration:*** In the dataset, a bluetooth device (user) will scan for other nearby bluetooth devices intermittently. A sequence of bluetooth encounter events is shown in Fig. 2. We use t_i and t_j to represent the time that the first and the last encounter occurred between users u_p and u_q, respectively. Note that there exists a situation that the mobile user u_p encounters u_q only once ($t_j = t_i$). This situation can be divided into two categories: 1) the bluetooth device of user u_p scans only once during a period of time; 2) the bluetooth device of u_q is discovered only once during user u_p's periodically scans. We consider the encounter in the first situation is a normal encounter via bluetooth between users u_p and u_q and the encounter duration time for the first situation is $\tau/2$ where τ is the time interval of each scan. We consider the encounter in the second situation is a chance encounter. We give the following formula to calculate the Encounter Duration Time (EDT) between users u_p and u_q in Eq.(1) according to our above analysis.

$$EDT(u_p, u_q) = \begin{cases} \tau/2 & if\ j = i, \\ t_j - t_i + \tau & if\ j \neq i. \end{cases} \tag{1}$$

Therefore, we divide the encounter duration time into 4 categories based on the statistic results as shown in Fig. 4(c): $D = \{D_i, i \in \{c, s, m, l\}\}$ where D_i represents different EDT as described in the Table II.

3) ***Location:*** To better understand the context of each encounter observation, it's helpful to identify the type of location where the encounter takes place. For example, an encounter between two people occurred in a private residence should be viewed differently from that in a crowded shopping mall.

The location logs we extract from raw data are area and cell tower IDs. We can know the type of location around the cell tower from users' survey data. In this paper, we classify all the cellular tower IDs into 3 location groups: $L = \{L_i, i \in \{h, w, o\}\}$ where h, w and o represent 'Home', 'Work' and 'Others', respectively. We do not distinguish locations which are neither 'Home' nor 'Work' and group them into 'Others'. We believe that there is no significant difference between 'Shopping mall' and 'Theater', and the experimental results in Section V show that the classification of locations at this level of granularity is enough for our model to portray mobility patterns of users correctly.

IV. FEATURE EXTRACTION

In this section, we introduce our spatial-temporal features and classify them into two categories (Mobility and Encounter features). These features (outlined in Table III) we provided try to distinguish when an encounter between two users happens by chance, say two strangers shopping in a crowded shopping mall on weekday, and when an encounter is a social activity, say one inviting his/her friends for lunch on weekend.

A. Mobility features

The mobility features measure characteristics related to the size of the encountered user set and the encounter location set for each user. These features quantify the basic mobility properties of users' behaviors in their daily life.

For a user u_p, Bluetooth Encounter Frequency $BEF(u_p)$ is denoted as the number of encounter times for each user and $BEF_{T_i^j}(u_p)$ is denoted as the number of encounter times during different time periods for each user.

$$\begin{cases} BEF(u_p) = |\{(u_q, U_q, t_k) | u_q = u_p\}| \\ BEF_{T_i^j}(u_p) = |\{(u_q, U_q, t_k) | u_q = u_p, t_k \in T_i^j\}| \end{cases} \tag{2}$$

where $(u_q, U_q, t_k) \in EU_p$ and T_i^j is described in Table I. $|\cdot|$ represents the cardinality of a set. We also compute the total number of users discovered by each user's bluetooth devices. Similar to Bluetooth Encounter Frequency, Cell tower Location Frequency $CTL(u_p)$ is denoted as the number of transitions between different cell towers and $CTL_{T_i^j}(u_p)$ is denoted as the number of transitions during different time periods for each user.

$$\begin{cases} CTL(u_p) = |\{(u_q, c_r, t_k) | u_q = u_p\}| \\ CTL_{T_i^j}(u_p) = |\{(u_q, c_r, t_k) | u_q = u_p, t_k \in T_i^j\}| \end{cases} \tag{3}$$

where $(u_q, c_r, t_k) \in EC_p$.

B. Encounter features

We combine proximity data from the bluetooth devices and location data from the cell towers together to obtain the qualified encounter records between users. For users u_p and u_q, their encounter EUC_{pq} is denoted in Eq.(4).

$$EUC(u_p, u_q) = \{(t_k, d_i, l_j) | t_k \in T, d_i \in D, l_j \in L\} \tag{4}$$

where $u_p, u_q \in U$. T, D and L are described in the Section III. The process of obtaining EUC_{pq} is described as follows:

1) Firstly, we consider the leave time of a location $l_{t_i} (t_i \in T)$ is the entry time for the user transiting to the next location $l_{t_{i+1}}$. In other words, the entry time when a user enters into a location is also the leave time of the previous area. For a transition from cell tower c_m to c_n, we can get the entry time t_e and the leave time t_l of the location c_m.
2) Based on the above considerations, an encounter record EUC_{pq} will be generated when proximity and location information satisfy the condition:

$$\max(t_e, t_s) < \min(t_l, t_f) \tag{5}$$

where t_s ($t_s = t_k$) and t_f represent the start and end time of the encounter, which discussed in Section III(C).

The statistical results show that although there are about one million observed encounters (1,143,064), only roughly 10% (107,000) of these encounters occurred between friends. This illustrates that there exist large amount of chance encounters in our daily social interactions.

For a given encounter set $EUC(u_p, u_q)$, we compute the Encounter Frequency ($EFreq$) between users u_p and u_q as follows:

$$EFreq(u_p, u_q) = |EUC(u_p, u_q)| \tag{6}$$

Then, we give the different kind of encounter features as follows.

1) ***Single encounter patterns****:* In this section, we define the encounter frequency and probability between users u_p and u_q in different patterns according to the categories we divided in Section III-C. For example, we define $Freq_{T^w}(u_p, u_q) = |\{(t_k, d_i, l_j) | t_k \in T^w|$ and $D_s(u_p, u_q) = \frac{|\{(t_k, d_i, l_j) | t_k \in T^w\}|}{EFreq(u_p, u_q)}$ are the encounter frequency and probability between users u_p and u_q when the encounter occurred on weekdays. We can also define $Freq_{D_s}(u_p, u_q) = |\{(t_k, d_i, l_j) | d_i = D_s\}|$ and $P_{D_s}(u_p, u_q) = \frac{|\{(t_k, d_i, l_j) | d_i = D_s\}|}{EFreq(u_p, u_q)}$ are the encounter frequency and probability between users u_p and u_q when the encounter lasts less than 5 minutes, respectively. Similarity, we can easily define the frequency and probability in other patterns, such as $Freq_{T_s}(u_p, u_q)$ and $P_{T_s}(u_p, u_q)$, $Freq_{T_f}(u_p, u_q)$ and $P_{T_f}(u_p, u_q)$ and so on.

2) ***Hybrid encounter patterns****:* Obviously, the probability of encountering friends on weekend in a cinema is higher than that on workday at office, and encounters with short duration time in a shopping mall have a higher probability to be chance encounters than that with long encounter duration time at home. Therefore, single encounter patterns can be combined to form hybrid patterns which may bring 'more meaningful' information for distinguishing encounters between friends from chance encounters. $Freq_{T^h, L_h}(u_p, u_q) = |\{(t_k, d_i, l_j) | t_k \in T^h, l_j = L_h\}|$ and $P_{T^h, L_h}(u_p, u_q) = \frac{|\{(t_k, d_i, l_j) | t_k \in T^h, l_j = L_h\}|}{EFreq(u_p, u_q)}$ represent the frequency and probability of encounter occurred at home on weekends. Similarity, the Work Encounter Frequency ($Freq_{WE} = Freq_{L_w, T^w}(u_p, u_q)$) and Probability ($P_{WE} = P_{L_w, T^w}(u_p, u_q)$) describe the frequency and the probability of encounters between users u_p and u_q occurred at work places on weekdays. The Home Encounter Frequency ($Freq_{HE} = Freq_{L_h, T^h}(u_p, u_q)$) and Probability ($P_{HE} = P_{L_h, T^h}(u_p, u_q)$) describe the frequency and the probability of encounters between users u_p and u_q occurred at home on weekends.

3) ***Encounter Entropy features****:* We quantify people's predictable structures in their daily life using an information entropy metric. In information theory, entropy is the expected (average) value of the information contained in each message received. In our paper, the encounter entropy features characterize the diversity of encounters between users. In other words, high encounter entropy between users u_p and u_q represents that their encounters are harder to predict and likely to be chance encounters between strangers, while low encounter entropy represents that their encounters are characterized by strong patterns and likely to be intended encounters between friends.

According to information theory, we define the encounter entropy between users u_p and u_q as follows:

$$H(P) = -\sum_{i=1}^{n} P_i \cdot \ln P_i \tag{7}$$

where P_i is the probability for each encounter pattern between users. For a example, we can get the workday encounter entropy for user pair u_p and u_q as $H_{T^w}(u_p, u_q) = -(\sum_{i=r,k,s} P_{T_i^w}(u_p, u_q) \cdot \ln P_{T_i^w}(u_p, u_q))$. Lastly, we list our main features in Table III.

V. The Results Of The Experiment

In this section, with extracted two category features (in Section IV), we will figure out the problem of predicting friendship based on mobility patterns of users. Firstly, we discuss the class imbalance problem in our feature dataset and apply different sampling methods (under-sampling and over-sampling) to address this problem and improve the friendship predicting accuracy. Secondly, we compare the performance of different sampling methods and then choose the over-sampling method to re-sample our feature dataset. Finally, we train three classifiers (SVM, Neural Network and Modest AdaBoost) to the re-sampled dataset and compare the performance among the three classifiers with the benchmark in Reality Mining Project [9]. In our paper, we use accuracy, precision, recall and F-score as our performance measures. Note that F-score is a measure of a test's accuracy which considers both the precision and the recall of the test.

A. Sampling methods

The dataset includes 66 pair of encounters between friends and 3386 pair of encounters between non-friends. Note that the feature dataset we extracted is imbalanced because the classes (the positive class represents encounters between friends and the negative class represents encounters between non-friends) are not approximately equally represented. Imbalance which on our dataset on the order of 51 ($\frac{3386}{66} \approx 51.3$) to 1 often leads to misunderstanding in predicting accuracy. Therefore, we re-sample the original dataset, either by over-sampling the minority class and/or under-sampling the majority class and then compare the performances with the SVM classifier.

*1) **Under-sampling**:* Under-sampling method [17] tries to improve minority class performance by dropping some of the majority samples at random. The disadvantage of the under-sampling method is that it cannot make full use of majority samples due to losing a great deal of important information in the majority samples.

*2) **Over-sampling**:* Over-sampling method [18] tries to improve minority class performance by increasing minority samples. We use the SMOTE [19] (Synthetic Minority Over-sampling Technique) algorithm to generate new synthetic minority class samples. The minority sample is over-sampled by introducing synthetic examples which are randomly chosen from its k nearest neighbors rather than coping minority samples. Specifically, for every sample in minority class x_i, the synthetic sample y_j generating according to its k nearest neighbors is computed as:

$$y_j = x_i + rand(0,1) * (x_i - x_{ij}) \tag{8}$$

where the function *rand(0,1)* represents a decimal number randomly generated between 0 and 1 and $x_{ij} (j = 1, 2, ..., k)$ are the k-th nearest neighbors of x_i.

TABLE IV
Sampling Methods[1]

Feature Dataset	*Under-Sampling*	*SMOTE*
Number of minority class samples	66[2]	3314
Number of majority class samples	74	3386[2]
Number of total samples	140	6700
k (parameter in SMOTE)[3]	None	3

[1] The SVM classifier uses the RBF (Radial Basis Function) as the kernel function and the grid search method as the parameter selection method.
[2] Before sampling, the number of minority samples and majority samples are 66 and 3386, respectively.
[3] Process of finding k nearest neighbours for each minority sample in SMOTE algorithm.

TABLE V
Performances of the different sampling methods with SVM.

Sampling method	**Accuracy**	**Precision**	**Recall**	**F-score**
Original dataset Mean(STD)*	0.9823 (0.0062)	0.8700 (0.2218)	0.1959 (0.1028)	0.3077 (0.1295)
Under-Samping Mean(STD)	0.7292 (0.0425)	0.7130 (0.0749)	0.7153 (0.0763)	0.7094 (0.0455)
SMOTE Mean(STD)	0.9470 (0.0309)	0.9176 (0.0132)	0.9838 (0.0215)	0.9496 (0.0437)

*3) **The comparison of different sampling methods**:* We use Support Vector Machine (SVM) as the machine learning algorithm to verify the performance of the different sampling methods (under-sampling and over-sampling).

We preformed 20 times of different sampling methods, and the predictions were conducted with a 10 fold cross validation. The detailed properties of the feature dataset and the SVM classifier are listed in Table IV. The performance of the SVM classifier is measured against the true values of whether the users are friends. Table V shows the average (Mean) and standard deviation (STD) of accuracy, precision, recall and F-score for the SVM classifier.

As shown in Table V, SVM achieves fairly high (98.23%) average accuracy but quite low average F-score (30.77%) in the imbalanced (original) dataset. It means that the model achieves high accuracy in inferring non-friendship ties (strangers) but very low accuracy in inferring friendship ties. It is unacceptable for applications or systems to recommend friends since sensing possible friends is more valuable than sensing strangers.

One noteworthy observation is that the SMOTE algorithm (over-sampling) outperforms the under-sampling method.

TABLE III
DESCRIPTION OF THE FEATURES.

Category	Description
Mobility	The total number of encounter times for user u_p.
	The number of encounter times for user u_p on weekdays/weekends, the number of the start time of the encounter occurred in 22:00-8:00/8:00-18:00/18:00-22:00 time period.
	The number of transitions between different cell towers for user u_p.
	The number of transitions between different cell towers for user u_p on weekdays/weekends or at Home/Work/Others.
Encounter	The encounter frequency and probability between users u_p and u_q when the encounter occurred on weekdays/weekends or in 22:00-8:00/8:00-18:00/18:00-22:00 time period.
	The encounter frequency and probability between users u_p and u_q where the location the encounter takes place is Home/Work/Others.
	The encounter frequency and probability between users u_p and u_q when the encounter duration time is less than 5mins/5-30mins/more than 30mins.
	The encounter frequency and probability between users u_p and u_q that the encounter occurred on weekends and the encounter duration time is less than 5mins/5-30mins/more than 30mins.
	The encounter frequency and probability between users u_p and u_q that the encounter occurred on weekdays and the encounter duration time is less than 5mins/5-30mins/more than 30mins.
	The encounter frequency and probability between users u_p and u_q that the encounter occurred at the workplace and in 22:00-8:00/8:00-18:00/18:00-22:00 time period.
	The encounter entropy that measures the different locations the encounters take place.
	The encounter entropy that measures the different encounter durations between users.
	The encounter entropy that measures the different time periods when encounters occurred between users.

The reasons are as follows: 1) the under-sampling method would lose large important information by only taking a small fraction (only about 2.2% of the total number) of the majority samples (we sampled 74 majority class samples to keep balance); 2) the SMOTE method provides more related minority class samples to learn from, thus allowing a learner to carve broader decision regions, leading to more coverage of the minority class.

B. Comparison with the model in Reality Mining Project

We compare the three classifiers and Table VI shows the average (Mean) and standard deviation (STD) of accuracy, precision, recall and F-score for each classifier. We observe that the difference among the three classifiers are not significant. In particular, the Neural Network model seems to perform the best, having the best average accuracy (95.68%) and F-score (95.34%). It correctly identifies 3255/3314 (98.2%) friendship pairs and 3156/3386 (93.2%) non-friendship pairs. Unlike classifiers training without sampling, the overall accuracy of all these three classifiers based on over-sampled dataset are high since the class distribution is no more heavily biased towards encounters between non-friendships. There is no significant difference among all the three algorithms, which illustrates that our spatial-temporal features are valuable tools for analyzing human social mobility patterns and our model has strong predictive power for inferring friendships.

In reality mining project [9], the authors used the same dataset and trained a gaussian mixture model to detect patterns in proximity between users and correlate them with the type of relationship. They compared the ground truth with mobile phone communication, estimated location, proximity data and time of day. They accurately inferred 96% of non-friendships and 95% of friendships based on this observational data. We compare our novel model with the model in reality mining project in Fig. 5. The result shows that our model outperforms the model in reality mining project in inferring friendships between users. Note that we excluded users' communication logs to predict friendships due to concerns of compromising user privacy and still gained pretty good accuracy (95.68%) and F-score(95.34%).

TABLE VI
PERFORMANCES OF DIFFERENT CLASSIFIERS.

Classifier	Accuracy	Precision	Recall	F-score
SVM*	0.9470	0.9176	0.9838	0.9496
Mean(STD)	(0.0309)	(0.0132)	(0.0215)	(0.0437)
Neural Network*	0.9568	0.9300	0.9780	0.9534
Mean(STD)	(0.0159)	(0.0153)	(0.0282)	(0.0207)
Modest AdaBoost*	0.9509	0.9171	0.9886	0.9514
Mean(STD)	(0.0271)	(0.0333)	(0.0452)	(0.0532)

* The kernel using in SVM classifier was Radial Basis Function (RBF). In neural network classifier, the hidden layer size was 5 and the network training function that updated weight and bias values was Levenberg-Marquardt algorithm. The adaboost classifier was run 500 iterations using CART as the tree learner (with 2 splits).

C. Feature importance

We preform feature selection approach in random forest classifier to examine which features are working best and then we rank the importance of the features. The top 10 important features are listed in Table VII. It seems that the number of

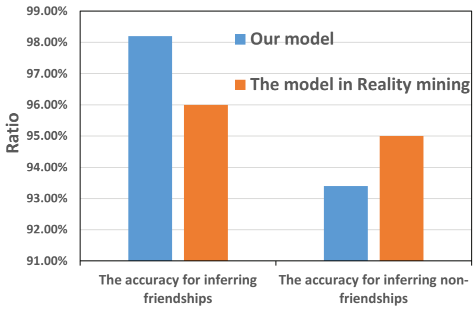


Fig. 5. The comparison between our model and model in Reality Mining Project [9].

TABLE VII
FEATURE IMPORTANCE.

Rank	RI*	Feature
1	100%	The number of encounters occurred at ‘Home’.
2	74.72%	The number of encounters occurred at 8:00-18:00.
3	50.30%	The probability of encounters occurred on weekends.
4	49.04%	The probability of encounters occurred at workplace and last more than 30 mins.
5	40.08%	The probability of encounters occurred at workplace and last between 5-30 mins.
6	33.52%	The encounter entropy that measures the different time periods.
7	33.44%	The encounter entropy that measures encounters at different location.
8	32.05%	The ratio of number of encounters between u_p and u_q to the total number of encounters of u_p.
9	30.35%	The encounter entropy that measures different encounter durations between users.
10	29.44%	The number of encounters occurred on weekdays.

* RI represents relative importance which ranges from 0 to 100%. The higher the RI is, the more important the feature will be.

encounters occurred at ‘Home’ is the most effective feature for inferring friendships between users. The number of encounters occurred at 8:00 $\sim$ 18:00 is also an important feature. Note that our encounter entropy feature show relatively high importance among all the features.

VI. CONCLUSION

In this paper, we combine proximity data from bluetooth interfaces and location data from cellular towers to infer friendship by analyzing users’ temporal and spatial mobility patterns. We introduce a novel set of features, including encounter entropy to analyze the social context of users and their encounters after grouping user mobility patterns into different categories. Using these spatial-temporal features, we propose a novel model to infer friendships between different users. An over-sampling method is employed to address the class imbalance problem. We validate our spatial-temporal features with three different classifiers and our experimental results show that our proposed features and model perform well in friendship prediction without using privacy-sensitive information like call logs.

ACKNOWLEDGMENT

This work was supported by the 973 Program (2013CB035503), National Natural Science Foundation of China (61572060, 61170296 and 61190125), and the R&D Program (2013BAH35F01).

REFERENCES

[1] Y. Zheng, “Tutorial on location-based social networks,” in *Proc. 21th Int’l. Conf. World Wide Web*, vol. 12, 2012.

[2] C.-C. Yu and H.-P. Chang, *Personalized location-based recommendation services for tour planning in mobile tourism applications*. Springer, 2009.

[3] B. Liu and H. Xiong, “Point-of-interest recommendation in location based social networks with topic and location awareness.” in *SDM*, vol. 13, 2013, pp. 396–404.

[4] D. Quercia and L. Capra, “FriendSensing: recommending friends using mobile phones,” in *Conference on Recommender Systems*, 2009, pp. 273–276.

[5] Y. Chen and K. Schwan, “Opportunistic overlays: Efficient content delivery in mobile ad hoc networks,” in *Middleware 2005*. Springer, 2005, pp. 354–374.

[6] J. You and T. J. Kim, “Towards developing a travel time forecasting model for location-based services: A review,” in *Methods and Models in Transport and Telecommunications*. Springer, 2005, pp. 45–61.

[7] Y. Zheng, “Trajectory data mining: An overview,” *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 3, pp. 1 – 41, May 2015.

[8] L. Adamic and E. Adar, “How to search a social network,” *Social Networks*, vol. 27, no. 3, pp. 187 – 203, 2005.

[9] N. Eagle, A. S. Pentland, and D. Lazer, “Inferring friendship network structure by using mobile phone data,” *Proc. the National Academy of Sciences*, vol. 106, no. 36, pp. 15 274–15 278, 2009.

[10] J. Cranshaw, E. Toch, J. Hong, A. Kittur, and N. Sadeh, “Bridging the gap between physical location and online social networks,” in *Proc. 12th ACM Int’l. Conf. Ubiquitous computing*. ACM, 2010, pp. 119–128.

[11] E. Cho, S. A. Myers, and J. Leskovec, “Friendship and mobility: user movement in location-based social networks,” in *Proc. 17th ACM SIGKDD Int’l. Conf. Knowledge Discovery and Data Mining*. ACM, 2011, pp. 1082–1090.

[12] N. Chawla, “Data mining for imbalanced datasets: An overview,” in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. Springer US, 2005, pp. 853–867.

[13] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma, “Mining user similarity based on location history,” in *Proc. 16th ACM SIGSPATIAL Int’l. Conf. Advances in Geographic Information Systems*. ACM, 2008, p. 34.

[14] D. Hristova, M. Musolesi, and C. Mascolo, “Keep your friends close and your facebook friends closer: A multiplex network approach to the analysis of offline and online social ties,” *CoRR*, vol. abs/1403.8034, 2014. [Online]. Available: http://arxiv.org/abs/1403.8034

[15] N. Eagle and A. Pentland, “Reality mining: sensing complex social systems,” *Personal and ubiquitous computing*, vol. 10, no. 4, pp. 255–268, 2006.

[16] N. Eagle and A. S. Pentland, “CRAWDAD data set mit/reality (v.2005-07-01),” Downloaded from http://crawdad.org/mit/reality/, jul 2005.

[17] X.-Y. Liu, J. Wu, and Z.-H. Zhou, “Exploratory undersampling for class-imbalance learning,” *Trans. Sys. Man Cyber. Part B*, vol. 39, no. 2, pp. 539–550, apr 2009. [Online]. Available: http://dx.doi.org/10.1109/TSMCB.2008.2007853

[18] R. Barandela, R. Valdovinos, J. Snchez, and F. Ferri, “The imbalanced training sample problem: Under or over sampling?” in *Structural, Syntactic, and Statistical Pattern Recognition*, ser. Lecture Notes in Computer Science, A. Fred, T. Caelli, R. Duin, A. Campilho, and D. de Ridder, Eds. Springer Berlin Heidelberg, 2004, vol. 3138, pp. 806–814.

[19] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “S-mote: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, no. 1, pp. 321–357, 2002.