PRED: Periodic Region Detection for Mobility Modeling of Social Media Users

Quan Yuan[†], Wei Zhang[‡], Chao Zhang[†], Xinhe Geng[†], Gao Cong[§], Jiawei Han[†]
[†]Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801
[‡]School of Computer Science and Software Engineering, East China Normal University, Shanghai, China, 200062
[§]School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798
[†]{qyuan, czhang82, xgeng2, hanj}@illinois.edu [‡]zhangwei.thu2011@gmail.com [§]gaocong@ntu.edu.sg

ABSTRACT

The availability of massive geo-annotated social media data sheds light on studying human mobility patterns. Among them, periodic pattern, *i.e.,* an individual visiting a geographical region with some specific time interval, has been recognized as one of the most important. Mining periodic patterns has a variety of applications, such as location prediction, anomaly detection, and location- and time-aware recommendation. However, it is a challenging task: the regions of a person and the periods of each region are both unknown. The interdependency between them makes the task even harder. Hence, existing methods are far from satisfactory for detecting periodic patterns from the low-sampling and noisy social media data.

We propose a Bayesian non-parametric model, named **P**eriodic **RE**gion **D**etection (PRED), to discover periodic mobility patterns by jointly modeling the geographical and temporal information. Our method differs from previous studies in that it is non-parametric and thus does not require priori knowledge about an individual's mobility (*e.g.,* number of regions, period length, region size). Meanwhile, it models the time gap between two consecutive records rather than the exact visit time, making it less sensitive to data noise. Extensive experimental results on both synthetic and real-world datasets show that PRED outperforms the state-of-the-art methods significantly in four tasks: periodic region discovery, outlier movement finding, period detection, and location prediction.

1. INTRODUCTION

The wide availability of geo-annotated social media data, such as tweets, Foursquare check-ins and Instagram photos, enables us to discover various mobility patterns [4, 27, 51, 54, 55, 58]. Among them, periodic mobility pattern has long been considered as one of the most important. Periodic mobility pattern, loosely defined as the repeating activities at certain locations with some time interval [27], can be observed virtually on each person. For example, a person may have breakfast in a region with several coffee houses every morning. This person may also shop in and then have dinner around a supermarket every Friday evening (Figure 1). In such cases, the coffee house region and the supermarket region are the two regions where the user exhibits periodic visiting behavior, with

WSDM 2017, February 06-10, 2017, Cambridge, United Kingdom

DOI: http://dx.doi.org/10.1145/3018661.3018680

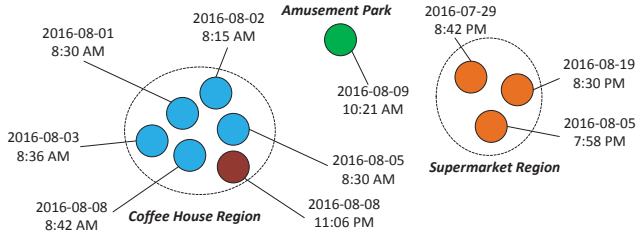


Figure 1: The historical geo-annotated records of a person

periods of one day and one week, respectively. Discovering periodic behaviors can help us better understand users' mobility as well as enhance an assortment of applications, such as anomaly detection, location prediction, location- and time-aware recommendation [50]. For example, based on the periodic pattern, we can detect the person's show-up in the coffee house region at 11:06 PM as an unusual movement; we can also anticipate that the person is highly likely going to visit the supermarket region on Friday evenings, and recommend a nearby restaurant to her.

However, it is challenging to discover periodic mobility regions of a user from her social media records, each of which contains geo-coordinates and time information. The reasons are three-fold. First, we have only GPS records with time information, but neither regions nor periods are known. It is difficult to discover which set of records from a user's social media should comprise a periodic region. Second, different users may have different numbers of regions, *e.g.,* a student may have one region at her campus, and a businessman may have more regions including home, office, shopping regions. A user may also whimsically visit locations out of regular regions (geographical irregularity, *e.g.,* the amusement park in Figure 1). Detecting irregular visits and discovering a proper number of regions to model an individual's mobility are non-trivial. Third, the irregular, low-sampling and short-span natures of social media data make it difficult to discover periods of regions: i) users usually do not strictly follow periods, *i.e.,* she might skip one visit or visit the region at a different time (temporal irregularity, *e.g.,* 11:06 PM at the coffee house region in Figure 1); ii) a user is unlikely to post about each activity on social media at every visited location (low-sampling rate *e.g.,* visits on August 4, 6, and 7 are missing even if she visits the coffeehouse every morning); iii) the spans of most users' records are usually very short, *e.g.,* less than one year.

In this paper, we study the problem of discovering periodic regions of social media users, *i.e.,* the geographical regions which a user visits periodically. Our goal is to automatically discover a proper number of regions for each individual as well as associated visiting periods. Existing studies, however, cannot address this task. Some studies [4, 41, 51] assume both the number of a user's regions and the period of each region are known in advance,

e.g., each user has two regions with 1 week as the period. Others [44, 51] rely on records at venues, *i.e.,* points of interests, but venue information may not be available in many social media services, *e.g.,* Twitter. Li *et al.* [27] propose to discover regions by kernel density estimation (KDE), then to estimate the period for each region by Fast Fourier Transform. However, the separation of region detection and period estimation hinders this study's effectiveness, because this method may cluster records with different periods into one region, making it difficult to determine periods. In addition, setting the bandwidth for KDE is still an open question.

To address the challenges, we propose a Bayesian non-parametric model **P**eriodic **RE**gion **D**etection (PRED) for users' periodic mobility modeling, which clusters records with proximate locations and same periods into a region. Our key observation is, *if two records follow a periodic pattern with a specific period, the gap time between which should approximately be a multiple of the period.* For example, the gap time between the person's visits at the coffee house on August 3 and 5 is 47.9 hours (Figure 1), approximately doubling the 24-hour period. Thus, instead of modeling the exact visiting time, we consider the gap time between visits. Therefore, the low-sampling and short spanning problem is alleviated. For example, even though we have only 3 observations in the supermarket region (Figure 1), we can infer that the period is 1 week based on the gap time (around 1 and 2 weeks). To exploit temporal information, PRED models the gap time by a Gaussian distributions, where the mean is the estimated periods (*e.g.,* 24 hours), and the variance allows the fluctuating visiting times. The temporally irregular visits (*e.g.,* the visit at 11:06 PM) will be detected and excluded based on its gap time (14.4 hours) between other visits. Thus, the temporal irregularity problem is mitigated. To exploit geographical information, PRED models a region by a bivariant Gaussian distribution over coordinates. As a result, close coordinates are likely to be clustered. The geographically irregular visits form isolated regions, and are thus detected. In a nutshell, PRED jointly exploits geographical and temporal information under the framework of Dirichlet Process, so it is able to automatically determine the number of regions.

Nevertheless, such a model introduces great difficulty in parameter estimation. From the location perspective, records are independent from each other; from the time perspective, they are not. This is because we model the gap time between records instead of absolute timestamps. The sequential nature of time makes parameter estimation even more difficult. As a result, conventional parameter estimation methods such as Markov Chain Monte Carlo (MCMC) cannot be applied directly. To overcome this difficulty, we propose a novel estimation method that takes into account the position of a record in a region's record sequence.

In summary, our major contributions are outlined as follows:

1. We formulate the problem of discovering periodic regions to model social media users' mobility.
2. We propose a novel Bayesian non-parametric model PRED to describe the periodic behaviors in different regions. PRED jointly models the geographical and temporal information. It can also well handle the problems of low-sampling, irregularity and short spanning. In addition, we develop a novel sampling-based parameter estimation method for model inference.
3. We conduct experiments on both real and synthetic datasets, and find that PRED outperforms state-of-the-art methods significantly on various tasks.

The organization of the paper is as follows. We first review existing studies in section 2, and then introduce our PRED model and its inference method in section 3. The experimental results are presented in Section 4. In the end, section 5 concludes this paper.

2. RELATED WORK

In this section, we review existing studies on mobility modeling, location prediction, and periodicity detection.

2.1 Mobility Modeling & Location Prediction

A number of studies have been proposed to model user mobility behaviors. Brockmann *et al.* [2] find human mobility behavior can be approximated by a continuous-time random-walk model with long-tail distributions. Gonzalez *et al.* [14] find that users periodically return to a few previously visited locations, and the mobility of each user can be modeled by a stochastic process centered at a fixed point. Cho *et al.* [4] observe that the mobility of each user is centered at several regions, and the probability that a user stays at a region is influenced by time. They propose a generative model, Periodic Mobility Model (PMM), which predicts a user's location by estimating the regions in which a target user most likely stays at a target time. Tarasov *et al.* [41] follow this paper and model a region by radiation model [40]. Isaacman *et al.* [19] examine spatiotemporal distributions of people's call records data to study people's mobility at a metropolitan scale. Deb *et al.* [5] and Zhang *et al.* [55] employ the Hidden Markov Model to extract latent semantic locations. Wang *et al.* [44] propose a hybrid mobility model that combines regularity and conformity of human mobility. Jiang *et al.* model human dynamic behaviors with tensor method [20] and Minimum Description Length principle [21]. None of these studies are capable of detecting true periods. In contrast, we focus on detecting unknown periods and unknown regions in which the periodic behaviors occur.

Location prediction that aims to predict the geo-location of an individual at a specific time has attracted much research attention due to its wide applications in recommendation, targeting advertising [57], *etc.* Most of existing studies are proposed to predict the next location of a user given her current location, under the framework of Hidden Markov Model (HMM) [3, 32, 48], trajectory pattern mining [34], Hierarchical Pitman-Yor language model [12], regression model [36], Factorized Markov Chain [10], Recurrent Neural Network [30]. However, the periodic mobility behavior enables us to predict the location of a user at any time. Several methods on this topic have been proposed by employing HMM [38], decision tree [33], topic model [51, 53], radiation model [41], matrix factorization [29], gravity model [44], and discrete choice model [24], but they require either social links [38, 41] or venues [24, 29, 33, 38, 44, 51] as the input. The requirements also exist in the studies on next location prediction [3, 10, 12, 30, 36, 48]. However, social links of individuals are not always available, and locations are often represented by geo-coordinates instead of venues (*e.g.,* in Twitter). PMM [4] and its variation [41] are the most generic approaches to location prediction, but they assume that users' visiting periods are known in advance.

2.2 Periodicity Detection

There has been extensive research work [15, 16, 43, 45, 46] on mining periodic patterns from sequence data. Han *et al.* [15, 16] propose to find the periodic patterns that appear frequently in a given itemset sequence. Their introduced partial periodic patterns are later extended in different ways [43, 45, 46]. Despite the inspiring results, these pioneering studies focus on efficiency.

Researchers have also proposed methods for detecting periods by employing autocorrelation (ACF) [1, 7–9], chisquared test [31], max subpattern tree [39], suffix tree [37], pattern combination [35], projection [47], Lomb-Scargle periodogram [13], and the combination of ACF and FFT [42]. Nevertheless, methods above are designed for the sequence data with relatively high sampling rate,

Table 1: Symbols

Symbol	Description
D_u, $\|D_u\|$	record collection of user u, the size of D_u
R_u	the set of regions of user u
θ	multinomial distribution of regions
μ_r, Σ_r	mean and covariance matrix of Gaussian distribution over geo-coordinates of region r
ν_r, σ_r^2	mean and variance of Gaussian distribution over time specific to region r
T_r	period of region r
d_i	record $d_i = \{l_i, t_i\}$
l	geographical coordinates
$t, t_{r,i}$	time point, the i^{th} time point assigned to r
$tg(t_i, t_j)$	gap time between records d_i and d_j
$e_{i,j}$	time remainder of the gap time $tg(t_i, t_j)$
$c_{i,j}$	period count of the gap time $tg(t_i, t_j)$
S	Observation sequence
α	hyper parameter for Chinese Restaurant Process
υ_0, κ_0, ρ_0, Ψ_0	Normal-Wishart prior for μ_r and Σ_r
ϵ_0, λ_0, τ_0, ψ_0	Normal-Gamma prior for ν_r and σ_r
β	spatial noise rate
γ	temporal noise rate
δ	sampling rate
n	number of period segments in the observations

instead of low-sampling social media data. Li *et al.* [28] study to find periodicity from sequences that have incomplete observations. The key idea is to segment the time series into small chunks and overlay them based on each candidate period. However, to generate sufficient statistics it requires a long-time series that may not hold in social media data.

To the best of our knowledge, there is only one study that tries to discover periodic regions along with the periodic behaviors [27]. They first perform KDE to detect a number of regions. Then for each region, they combine FFT and ACF to detect the underlying periodic behaviors. As time is not considered when detecting regions, many noise records or even records with different periods are clustered, making it difficult to detect periods. In contrast, our method jointly models the spatial and temporal observations, and takes into account the gap time to address data sparsity problem.

3. PROPOSED MODEL

We first formulate the problem of periodic regions discovery, and then describe PRED model and its parameter estimation method. Finally we introduce its application on user location prediction. Table 1 lists all the notations used in this paper.

3.1 Problem Statement

Let D_u be the collection of records of user u, and each record $d_i \in D_u$ is a 2-tuple $d_i = \{l_i, t_i\}$, where l_i and t_i represent the geographical coordinates and posting time of d_i, respectively. A user u has a periodic visiting behavior at a region r with period T if u is likely to visit r every T hours. Here regions are a set of geographical clusters within which most records in D_u are observed.

Given D_u, our goal is to find (1) a set of geographical regions R_u at which user u has periodic visiting behavior, and (2) the period T_r associated with each region $r \in R_u$.

3.2 Periodic Region Detection Model

We build our Periodic Region Detection Model based on the following intuitions:

Table 2: Time Example

ID	time	exact time	gap	rmdr.	ct.
d_1	D1 8:30 AM	8.50	–	–	–
d_2	D2 8:15 AM	32.25	23.75	23.75	1
d_3	D3 8:36 AM	56.60	24.35	24.35	1
d_4	D5 8:30 AM	104.50	47.90	23.90	2
d_5	D8 8:42 AM	176.70	72.20	24.20	3

1. A user's mobility centers at several personal geographical regions, *e.g.,* home region, shopping region, working region, *etc.*, and the user tends to visit places within these regions.
2. A region may have a visiting period, *e.g.,* 1 week for shopping regions, and 1 day for dining regions.
3. If a region has a periodic visiting pattern, the gap time between its consecutive visiting records should approximate to a multiple of its period.

Note that we can always get a visiting period for a region even if there is no periodic visiting behavior on it, but the detected periods will be very large, and the supports will be small. Thus, such regions are not supposed to have periodic visiting patterns. Following the settings of previous studies [4, 27], we assume each region has only one visiting period (if there are multiple periods, the overall period will be their least common multiple). This suggests that the geographical and temporal information can cooperate for region discovery. In fact, we believe it is important to *exploit spatial and temporal information jointly*. On the one hand, if we discover regions first based on the geo-coordinates only, it is difficult to estimate the periods if two regions are close in distance but have different periods; on the other hand, if we only consider time in the first step, it is hard to detect all possible periods because all periods are interleaving with each other, mixed with noise.

How to determine the number of personal regions is a crucial problem. Most existing topic model based studies assume all users share the number of regions, and the region number is known [4,17, 49, 51]. In real world, however, different users may have different numbers of regions, and it is difficult to tell the numbers in advance. For example, a student may have only one region (campus), but a white collar may have more regions for working, dining, shopping, *etc.* Although kernel based method [27] can detect regions from data, how to set kernel bandwidths and how to determine threshold to filter noise remain unsolved.

In this paper, we utilize the Dirichlet Process to generate the regions. A well-known metaphor is the Chinese Restaurant Process (CRP), a stochastic process in which customers select seats at a restaurant with an infinite number of tables. The first customer randomly selects a table to sit, while the other customers can either sit at a new table or select an occupied table with probability proportional to prior α and the number of customers at an occupied table, respectively. We employ CRP to cluster records into regions, which can automatically estimate the number of regions. From the geographical perspective, a region r is modeled by a bivariant Gaussian over the latitude and longitude coordinates, parameterized by the mean vector μ_r and covariance matrix Σ_r. The probability that region r generates location l is:

$$P(l|r) = \mathcal{N}(l|\mu_r, \Sigma_r) \tag{1}$$

It is more complicated to model time. Most existing studies [4, 51] assume the visiting period of a place is known, *e.g.,* 24 hours, and use Gaussian distribution to model the *visiting time* within each periodic segment. However, such methods are not able to estimate the actual periods. Thus, if the defined period is not true (*e.g.,*

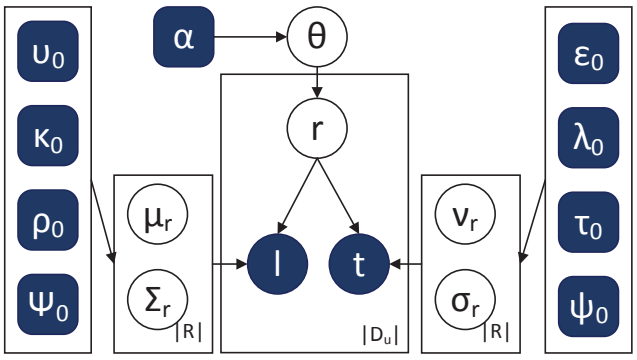


Figure 2: Graphical Model

1 week for shopping region), these methods will make mistakes (predict that the user will go shopping every day).

In this paper, instead of modeling the exact visiting time, we model the gap time between consecutive records. Our key insight is, if there is a periodic visiting pattern, the gap time between every pair of consecutive records should approximate to a multiple of its period. To see this, recall the person who always visits a coffeehouse region at 8:30 AM everyday (records are shown in Table 2). Obviously, the period is 24 hours. If the user rigorously goes to the coffeehouse every morning and reports her visits (*e.g.,* d_1, d_2 and d_3 for day 1 to day 3), and the gap time (gap) are around 24 hours. However, if some records are missing (day 4, 6 and 7), the gap time (47.90 and 72.20) does not center around the period any more. Nevertheless, if we divide the gap time by the period (24 hours), the remainders (rmdr.) fall around the period again. This suggests us to exploit the gap time to model the periodic time pattern.

Specifically, we define the remainder $e_{i,j}$ of the gap time $tg(t_i, t_j)$ between records d_i and d_j divided by the period T as follows:

$$e_{i,j} = \begin{cases} mod(tg(t_i,t_j),T) & mod(tg(t_i,t_j),T) > T/2, \\ mod(tg(t_i,t_j),T)+T & \text{otherwise.} \end{cases} \quad (2)$$

Then suppose a user u visited region r at time $t_1, t_2, ..., t_h$ with period T, the probability that u will visit r at time t_i is as follows:

$$P(t_i|r) = \mathcal{N}(e_{h,i}|\nu_r, \sigma_r^2), \quad (3)$$

where ν_r and σ_r^2 are the mean and variance of the univariate Gaussian distribution for time.

In summary, the generative process is illustrated in Algorithm 1, and the graphical model is shown in Figure 2.

3.3 Parameter Estimation

The total likelihood of full observation of D_u is:

$$\begin{aligned} & P(\boldsymbol{l}, \boldsymbol{t}, \boldsymbol{r}|\alpha, \upsilon_0, \kappa_0, \rho_0, \Psi_0, \epsilon_0, \lambda_0, \tau_0, \psi_0) \\ = & \int P(\boldsymbol{r}|\boldsymbol{\theta}) P(\boldsymbol{\theta}|\alpha) d\boldsymbol{\theta} \cdot \\ & \int P(\boldsymbol{l}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) P(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\upsilon_0, \kappa_0, \rho_0, \Psi_0) d\boldsymbol{\mu} d\boldsymbol{\Sigma} \cdot \\ & \int P(\boldsymbol{t}|\boldsymbol{\nu}, \boldsymbol{\sigma}) P(\boldsymbol{\nu}, \boldsymbol{\sigma}|\epsilon_0, \lambda_0, \tau_0, \psi_0) d\boldsymbol{\nu} d\boldsymbol{\sigma} \end{aligned} \quad (4)$$

We employ collapsed Gibbs sampling to obtain samples of the hidden variable assignments $\boldsymbol{r} = \{r_i\}_{i=1}^{|D_u|}$, and to estimate the unknown parameters $\{\boldsymbol{\theta}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\nu}, \boldsymbol{\sigma}\}$.

Based on Equation 4, we derive the updating equation for region r_i for record d_i as follows:

Algorithm 1: Generative Process

for *the record* $d_i \in D_u$ **do**
  Draw a region r_i based on $CRP(r|\alpha)$;
  if $r_i \notin R_u$ **then**
    Draw geographical distribution $\mathcal{N}(\mu_{ri}, \Sigma_{ri}) \sim Normal\text{-}Wishart(\boldsymbol{\upsilon}_0, \kappa_0, \boldsymbol{\rho}_0, \zeta_0)$;
    Draw periodic pattern $\mathcal{N}(\nu_{ri}, \sigma_{ri}) \sim Normal\text{-}Gamma(\epsilon_0, \lambda_0, \tau_0, \zeta_0)$;
    Add r_i into R_u;
  Draw a location $l_i \sim \mathcal{N}(l|\boldsymbol{\mu}_{ri}, \boldsymbol{\Sigma}_{ri})$;
  Draw a time $t_i \sim \mathcal{N}(mod(t, t', \nu_r)|\nu_r, \sigma_r^2)$;

$$\begin{aligned} P(r_i = r|\boldsymbol{r}_{\neg i}, .) \propto \quad & P(r|\boldsymbol{\theta}_{\neg i}) \cdot P(l_i|\mu_{r,\neg i}, \Sigma_{r,\neg i}) \cdot \\ & P(t_i|\nu_{r,\neg i}, \sigma_{r,\neg i}) \end{aligned} \quad (5)$$

$\boldsymbol{\theta}_{\neg i}$, $\mu_{r,\neg i}$, $\Sigma_{r,\neg i}$, $\nu_{r,\neg i}$, $\sigma_{r,\neg i}$ are parameters after excluding d_i.

The first part of the right hand side of Equation 5 follows the Chinese restaurant process:

$$P(r|\boldsymbol{\theta}_{\neg i}) = \begin{cases} \frac{\alpha}{\sum_{r' \in R_u} n_{r',\neg i}} & \text{if } r \notin R_u \\ \frac{n_{r,\neg i}}{\sum_{r' \in R_u} n_{r',\neg i}} & \text{if } r \in R_u \end{cases} \quad (6)$$

$n_{r,\neg i}$ is the number of records assigned to r after excluding d_i.

The second part is the posterior predictive probability of l_i being generated by region r after excluding d_i. Since the location distribution is modeled by bivariant Gaussian, the posterior predictive follows the bivariant student t-distribution:

$$t_{\rho-1}(\mu_{r,\neg i}, \frac{\Psi_{r,\neg i}(\kappa+1)}{\kappa(\rho-1)}), \quad (7)$$

where $\mu_{r,\neg i} = \frac{\kappa\upsilon + n_{r,\neg i}\bar{l}_{r,\neg i}}{\kappa + n_{r,\neg i}}$, $\kappa = \kappa_0 + n_{r,\neg i}$, $\rho = \rho_0 + n_{r,\neg i}$

$$\begin{aligned} \Psi_{r,\neg i} = \quad & \Psi_0 + \sum_{l \in \boldsymbol{l}_{r,\neg i}} (l - \bar{l}_{r,\neg i})(l - \bar{l}_{r,\neg i})^T + \\ & \frac{\kappa n_{r,\neg i}(\upsilon_0 - \bar{l}_{r,\neg i})(\upsilon_0 - \bar{l}_{r,\neg i})^T}{\kappa + n_{r,\neg i}} \end{aligned}$$

Here $\bar{l}_{r,\neg i}$ and $\boldsymbol{l}_{r,\neg i}$ are the mean coordinates and the set of all coordinates of records assigned to region r after excluding d_i, respectively. The derivations are omitted due to limited space. More details can be found in [22].

The third part of Equation 5 is the posterior predictive probability of t_i being generated by region r after excluding d_i. However, compared with the second part, it is much more difficult to get the closed form of the third part, because d_i is not independent from other records given that the temporal Gaussian distribution is defined in terms of gap time instead of time points.

We need to consider the change of gap time of region r_{d_i} when we remove d_i from it, where r_{d_i} is the region assigned to d_i before sampling. As time is a sequential variable, we first find the time points $t_{i'}$ and $t_{i''}$ that are precedent and consecutive to t_i in r_{d_i} (Figure 3). If t_i is the first time point in r_{d_i}, excluding d_i will result in removing the gap time $tg(t_i, t_{i''})$ (Figure 3 (a)). Similarly, if t_i is the last time point in r_{d_i}, excluding d_i will lead to the removal of gap time $tg(t_{i'}, t_i)$ (Figure 3 (b)). If t_i is in the middle of the time sequence, after removing d_i, both $tg(t_{i'}, t_i)$ and $tg(t_i, t_{i''})$ will be deleted. In addition, as $t_{i'}$ and $t_{i''}$ become consecutive, a new gap time $tg(t_{i'}, t_{i''})$ will be added (Figure 3 (c)). Now, we re-estimate the parameters for the temporal Gaussian model for the sampling based on the remaining gap time.

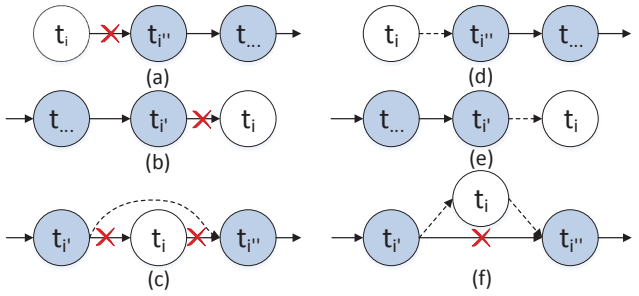


Figure 3: Inserting and deleting t_i. Dashed lines (gap time) are added, and lines with crosses are removed

When calculating the posterior predictive probability for t_i, we are actually calculating how likely t_i can be inserted into region r. Similarly, when deleting d_i, if t_i is at the boundary of r's time sequence, we just add one gap time, *i.e.,* $tg(t_{i'}, t_i)$ or $tg(t_i, t_{i''})$ (Figure 3(e) and (f)). If t_i is in the middle of the time sequence of r, two new gap time $tg(t_{i'}, t_i)$ and $tg(t_i, t_{i''})$ will be added, and the gap time $tg(t_{i'}, t_{i''})$ will be removed (Figure 3 (f)).

Since temporal pattern is defined as Gaussian distribution over gap time, given a region r, the posterior predictive of adding or deleting a gap time with remainder e follows the student t-distribution:

$$t_{2\tau}(e|\nu_r, \frac{\psi_r(\lambda+1)}{\tau\lambda}), \tag{8}$$

where $\nu_r = \frac{\lambda\epsilon_0 + (n_r-1)\bar{e}_r}{\lambda + n_r - 1}$, $\lambda = \lambda_0 + n_r - 1$, $\tau = \tau_0 + \frac{n_r-1}{2}$

$$\psi_r = \psi_0 + \frac{1}{2}\sum_{e' \in \boldsymbol{e}_r}(e' - \bar{e}_r)^2 + \frac{\lambda_0(n_r-1)(\bar{e}_r - \epsilon_0)^2}{2(\lambda_0 + n_r - 1)}$$

Based on the above analysis, we derive the third part of Equation 5:

$$\begin{cases} t_{2\tau}(e_{i,i''}|\nu_{r,\neg i}, \frac{\psi_{r,\neg i}(\lambda+1)}{\tau\lambda}) & \text{if } t_i < t_{r,0} \\ t_{2\tau}(e_{i',i}|\nu_{r,\neg i}, \frac{\psi_{r,\neg i}(\lambda+1)}{\tau\lambda}) & \text{if } t_i > t_{r,n_{r,\neg i}} \\ t_{2\tau}(e_{i',i}|\nu_{r,\neg i}, \frac{\psi_{r,\neg i}(\lambda+1)}{\tau\lambda}) \cdot t_{2\tau}(e_{i,i''}|\nu_{r,\neg i}, \frac{\psi_{r,\neg i}(\lambda+1)}{\tau\lambda}) & \\ / t_{2\tau}(e_{i',i''}|\nu_{r,\neg i}, \frac{\psi_{r,\neg i}(\lambda+1)}{\tau\lambda}) & \text{otherwise} \end{cases}$$

Here $\bar{e}_{r,\neg i}$ and $\boldsymbol{e}_{r,\neg i}$ are the average time remainders and the set of all time remainders assigned to region r after excluding d_i, respectively. $t_{r,k}$ is the k^{th} time point in region r.

However, after each sampling iteration, ν_r is not the period of region r, because it is estimated based on time reminders, but the time remainders are calculated based on the ν_r in last sampling steps. To address this problem, we need to turn to the period count (ct.) $c_{i,j}$ of the gap time $tg(t_i, t_j)$ between records d_i and d_j divided by ν:

$$c_{i,j} = \frac{tg(t_i, t_j) - e_{i,j} + \nu}{\nu} \tag{9}$$

Different count values c correspond to different potential periods $c\nu$. Recall the example in Table 2. Suppose in current iteration, ν_r' is 8. Then the count 3 appears twice (corresponds to the gap time 23.75 and 24.35), and the counts 6 and 9 appears 6 and 9 times, respectively (corresponds to gap time 47.90 and 72.20). The three count values correspond to three ν_r values, namely, 24 ($3\nu_r'$), 48 ($6\nu_r'$), and 72 ($9\nu_r'$) hours. Intuitively, the best ν_r should have smaller variance value in terms of its remainders, and its corresponding count should appear more times. Thus, we select $\nu_r = c\nu_r'$ with the smallest score $\frac{\sum_{e\in\boldsymbol{e}_r}\frac{(e-\nu_r)^2}{|\boldsymbol{e}_r|-1}}{\#c}$, where e is calculated with $T = \nu_r$, and $\#c$ is the number of times count c appears. For example, the scores for $\nu_r = 24$, 48, and 96 are 0.0391, 563.278, and 579.278, respectively. Thus, 24 is the optimal value for ν_r.

Based on the sampling results, we can calculate the unknown parameters as follows:

$$\widehat{\theta}_r = \frac{n_r + \alpha/R_u}{\sum_{r'} n_{r'} + \alpha}, \quad T_r = \widehat{\nu}_r = \nu_r$$
$$\widehat{\sigma}_r^2 = \frac{\sigma_r^2}{\tau}, \quad \widehat{\boldsymbol{\mu}}_r = \boldsymbol{\mu}_r, \quad \widehat{\boldsymbol{\Sigma}}_r = \frac{\boldsymbol{\Sigma}_r}{\rho} \tag{10}$$

3.4 Location Prediction

The detection of a user's periodic regions enables us to predict her location in the future. Specifically, given a target user u, her records D_u, and a target time t, we can calculate the probability p_r of each region r based on Equation 5 without excluding any record. Then, we use the mean $\widehat{\boldsymbol{\mu}}_r$ of the region r that has the greatest probability as the predicted location. Another way is to build a Gaussian Mixture Model for each target time based on the estimated $\{p_r\}_{r=1}^{|R_u|}$, $\{\widehat{\boldsymbol{\mu}}_r\}_{r=1}^{|R_u|}$, and $\{\widehat{\boldsymbol{\Sigma}}_r\}_{r=1}^{|R_u|}$, and sample a coordinate pair as the prediction results. We tried both methods, and found their performance is similar.

Remarks: Our model is robust to noise. For the spatial noise, *i.e.,* the user's visits at locations out of her regularly visited regions, our model can simply detect and exclude them as new (noise) regions because they are less likely to be generated by the geographical Gaussian distributions of existing regions. It is more complicated to deal with temporal noise, *i.e.,* the user's visits within regularly visited regions but the visiting time is different. Recall the example in Table 2. Suppose the person visited the coffee house again at 11:06 PM on day 8 (denoted by record d_6). Then the time gap between d_5 and d_6 is 14.4 hour. At the beginning, our model does not know this is an irregular visit, and thus takes this visit into consideration when estimating ν. Suppose the newly estimated ν is 20 hour. Given $\nu = 20$, the remainders of the gaps between records d_1 to d_5 are around 24. Thus, after several iterations, ν becomes closer to 24, and the probability that the corresponding Gaussian distribution generates the gap between d_5 and d_6 becomes smaller. In this end, d_6 will be excluded from the region as noise.

We assume there is only one visiting peak in each period. However, for some regions there might be more than one peak. For example, the visiting records at home may center at the peaks 7:00 AM and 9:00 PM. In such a case, two regions will be detected, which share similar geographical coverage and the same period, but the visiting time are different: One corresponds to 7:00 AM; the other corresponds to 9:00 PM. It is desirable to merge such regions into one. To achieve this, we can employ many metrics to measure the proximity between the geographical Gaussian densities of two regions with the same period, such as Kullback-Leibler Distance, Earth Movers Distance [26], Normalized L2 Distance, and normalized cross-likelihood ratio (NCLR) [25]. We will leave this for future work, even though our current simplified model can achieve good results on real-world datasets (Section 4).

4. EXPERIMENTS

In this section, we first evaluate PRED on synthetic datasets under different scenarios, and then examine its effectiveness in location prediction. In the end, we provide two case studies to illustrate the results of PRED model[1].

[1]The data and code are available at http://www.quan-yuan.com/datacode.html

4.1 Periodicity Detection on Time Series Data

4.1.1 Experimental Settings

Data Generation: We generate synthetic datasets by a set of parameters, and use the datasets with known periods to test the effectiveness of the proposed method under different scenarios. The datasets are generated as follows [28]: (1) fix a period T (*e.g.,* 24) and the center of visiting time t (*e.g.,* 12). The visiting time within each period t_i around t is modeled by a decay function $P(t_i) = exp(-tg(t, t_i))$ [6, 52]. (2) set $N = nT$ as the length of the time series. (3) sample the time sequence with sampling rate δ, *i.e.,* the record in each period segment can be observed with probability δ. (4) with temporal noise rate γ, the record within a period segment is generated randomly (*i.e.,* uniformly sampled). The default values are: $T = 50, t = 0, n = 300, \delta = 0.1, \gamma = 0.1$.

Methods for Comparison:

- **Fast Fourier Transform (FFT)**: We employ discrete Fourier transform to find the spectral with the highest power, and use its corresponding period as the result.
- **Autocorrelation and Fourier Transform (Auto)**: We first calculate the auto-correlation of the time series, and then use FFT to select the period with the highest power as the result.
- **Auto-correlation and Fourier Transform (Periodogram)** [27]. We first apply Fourier Transform to the time series and identify the range of candidate periods corresponding to the greatest power. Then for each period range given by the periodogram, we test whether there is a peak within it. If there is a peak, we return the location of the peak as the result.
- **Discrepancy-based method (Discrepancy)** [28]. We calculate the discrepancy score for each candidate period, which measures to which extant the records concentrate on a set of time points if we segment the records based on the period and overlay them together. The period with the largest discrepancy score is output as the result.
- **Period Detection (PD)**. Our proposed method without considering spatial information.

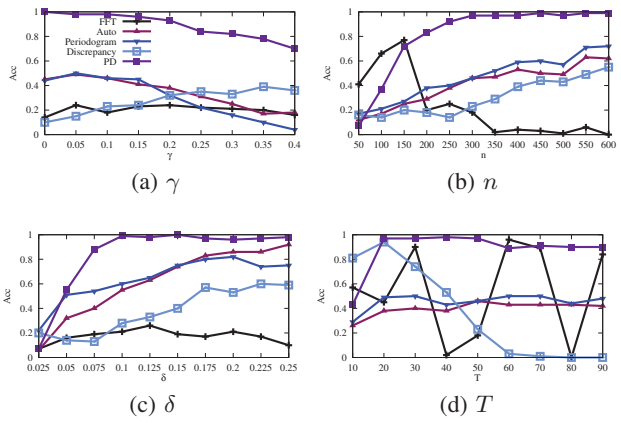
Since all these baselines are designed for discrete time series, and the ground truth periods T are integers, in order to make a fair comparison, we round the outputs of our method to integers. The default values of our hyper-parameters are set as follows: $\alpha = 0.1$, $\epsilon_0 = 24$, $\lambda_0 = 0.01$, $\tau_0 = 0.5$, $\psi_0 = 0.1$. The parameters of baselines are set to be their suggested values.

Evaluation Metric: Given a sequence of time records, we are interested in how well a method can discover its period. For each parameter setting, we repeat the experiment for 100 times, and report accuracy, *i.e.,* the percentage of correct period detections over 100 trials. For our method, we report the period with the maximum number of records as the prediction.

4.1.2 Performance Study

We are interested in the performance of methods on the datasets generated by different combinations of parameters. Specifically, for each time, we vary the value of a parameter, and fix the values of the others at their default values. The accuracies of methods are plotted in Figure 4.

Figure 4 shows, with the decrease of noise rate γ and the increase of repetition number n and sampling rate δ, all methods generally achieve better results for periodicity detection. Among the baselines, the performance of some baselines, *e.g.,* FFT, is poor. The performance of other baselines, *e.g.,* Auto, and Discrepancy can achieve good results only when the data sequence is of high quality,



(a) γ (b) n (c) δ (d) T

Figure 4: Comparison Results with Various Data Generation Parameter Settings

e.g., large number of repetition n and high sampling rate δ. The accuracy curves of FFT fluctuate severely when varying n or T, probably due to the spectral leakage problem. In contrast, the curves of Auto is more stable because it generates a smoothed version of data for Fourier transform. Discrepancy is sensitive to period length T: with the increase of T, the performance of Discrepancy drops dramatically. Potential reason is the score of the randomized sequences penalizes large periods (see [28] for details).

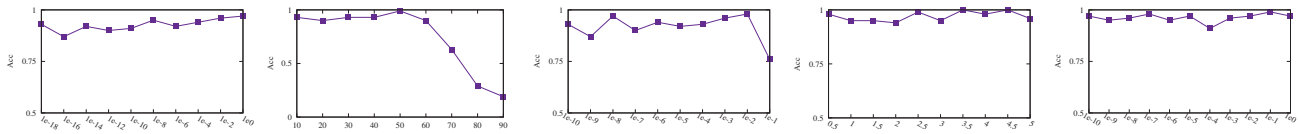
Among all methods compared, our method PD outperforms baselines under various settings of parameters even when the noise rate γ is large, the number of repetition n is small, and low sampling rate δ is small and the period length T is large. This is because we model the gap time between records rather than the exact time, resulting in better robustness under various sampling rates, period lengths, and repetition numbers. The irregular time form isolated Gaussian components and are thus detected, so our model is less sensitive to high noise rate.

We are also interested in whether PD is sensitive to the hyper-parameters. As shown in Figure 5, the performance is relatively stable. The only exception is ϵ_0: when ϵ_0 is greater than the period, PD will work badly. This is because ϵ_0 controls the initial period of PD. When the period is small, we can increase it by multiplying by the count $\#c$, but when the period is too large, it would be difficult to reduce it. In practice, we can use a small value, *e.g.,* 20, as ϵ_0.

4.2 Periodic Region Detection on Spatiotemporal Data

4.2.1 Experimental Settings

Data Generation: We generate synthetic datasets by a set of parameters as follows [27,28]: (1) fix the number of region R, and for each region $r \in R$ we generate its distribution over geo-coordinates. In this paper, we use Cauchy distribution, a well-known Levy Flight distribution [23], as the geographical distribution. For each Cauchy distribution, we sample its mean coordinates by uniform distribution over a 10*10 map, and randomly generate a value smaller than 1 as its scale parameter. (2) For each region, randomly generate a period between 30 and 100. (3) Set $N = nT_{max}$ as the maximum time in the time series, where T_{max} is the largest period of all regions. (4) For each period segment of each region r, generate a spatiotemporal record d with the probability δ. (5) With the probability of $(1-\beta)$, the coordinates of d are generated according to their geographical distribution, otherwise a random coordinates are selected based on the uniform distribution over the map (spa-



(a) α (b) ϵ_0 (c) λ_0 (d) τ_0 (e) ψ_0

Figure 5: Parameter Influence on Period Identification

tial noise). (6) If d is not a spatial noise, with the probability of $(1-\gamma)$, generate a time for it based on r's temporal distribution (See Section 4.1), otherwise a random time is selected. The time for all records with spatial noise is also randomly selected. The default values of R and β are 3 and 0.1, respectively. Other parameter values remain the same as in Section 4.1.

Methods for Comparison:

- **Periodica** [27]: We first extract regions by kernel density estimation (KDE), and then estimate the period for each region using Periodogram, which has been evaluated in Section 4.1.
- **KernelDiscp**: As Periodogram is not effective in handling time sequence with large noise rate, to make a fair comparison we still use KDE to extract regions, but use the method Discrepancy to extract period.
- **Periodic Region Detection Separate (PREDsep)**. To test the effectiveness of jointly modeling spatial and temporal information, we introduce a new baseline which first extracts regions using CRP and then identifies the period using PD. Different from PRED, PREDsep does not exploit spatial and temporal information jointly.
- **Periodic Region Detection (PRED)**. Our proposed method.

Same as before, the parameters of baselines are set at their suggested values. We set the default values of our hyper-parameters as follows: $\kappa_0 = 0.01$, $\rho_0 = 2$, υ_0 and Ψ_0 are set using the mean and covariance matrix of the record coordinates of the user, respectively. For PRED, we first disregard time and run 100 iterations on geographic locations to discover initial regions, and then run another 500 iterations on both locations and time to discover periodic regions. We discard regions with less than 10 assigned records, or with period greater than the threshold 100 as noise.

Tasks & Evaluation Metric:

- **Periodic Region Detection**. We check whether a method can correctly identify the regions and their according periods. A detected region is correct if it shares the same period with its nearest true region, where the distance between regions are defined using L-2 distance. We use F-1 as the evaluation metric.
- **Outlier Visit Detection**. Recall that in our generated records, some are irregular (spatial or temporal). We are interested to test whether a method can correctly detect a visit is irregular. A record is regarded as irregular if it is associated with a noise region (PREDSep and PRED) or associated with no region (Periodica and KernelDiscp). We use F-1 as the evaluation metric.
- **Period Detection for Records**. When generating the datasets, we know the corresponding period of each record. A good method should be able to detect the period to which a record belongs to. We use accuracy as the evaluation metric.

4.2.2 Performance Study

We evaluate the performance of all methods under different settings of parameters for all three tasks, and plot the results in Figure 6. We observe that all methods will generate better results when the data is of better quality, *i.e.,* lower spatial noise rate β, lower temporal noise rate γ, greater sampling rate δ, larger number of period repetition n, and smaller number of regions R.

For region detection (Figures 6(a) to 6(e)), the performance of Periodica is always the worst, followed by KernelDiscp. The geographical regions extracted by the two methods are the same, but since the period detection algorithm of KernelDiscp is more effective in handling time series data with large noise rate, KernelDiscp can better find the periods for the extracted regions. Same with Periodica and KernelDiscp, PREDsep first detects regions and then estimates period for each of them. However, comparing with the two baselines, PREDsep achieves better performance. The reasons are two-fold: from the spatial perspective, PREDsep can automatically detect a proper number of regions, while the regions detected by Periodica and KernelDiscp rely on the bandwidth of Kernel density estimation greatly; from the temporal perspective, our periodicity detection method is less sensitive to small sampling rate and repetition number, as well as large noise ratio and period length. Comparing with PREDsep, PRED always generates superior results, because it exploits spatial and temporal information jointly when extracting periodic regions.

For outlier detection (Figures 6(f) to 6(j)), there are only three curves, because Periodica and KernelDiscp always generate the same results and thus their curves overlap with each other. Similarly as above, our method PRED performs much better than the baselines. It is interesting to observe that with the increase of noise rate β and γ, the performance of different methods doesn't drop obviously. This is because the number of noise records also grows. Thus, we only make comparison under the same noise rate.

For the task of period detection for records (Figures 6(k) to 6(o)), we still observe that PRED performs the best, followed by PREDsep. In contrast, Periodica and KernelDiscp are less effective in predicting the period for records.

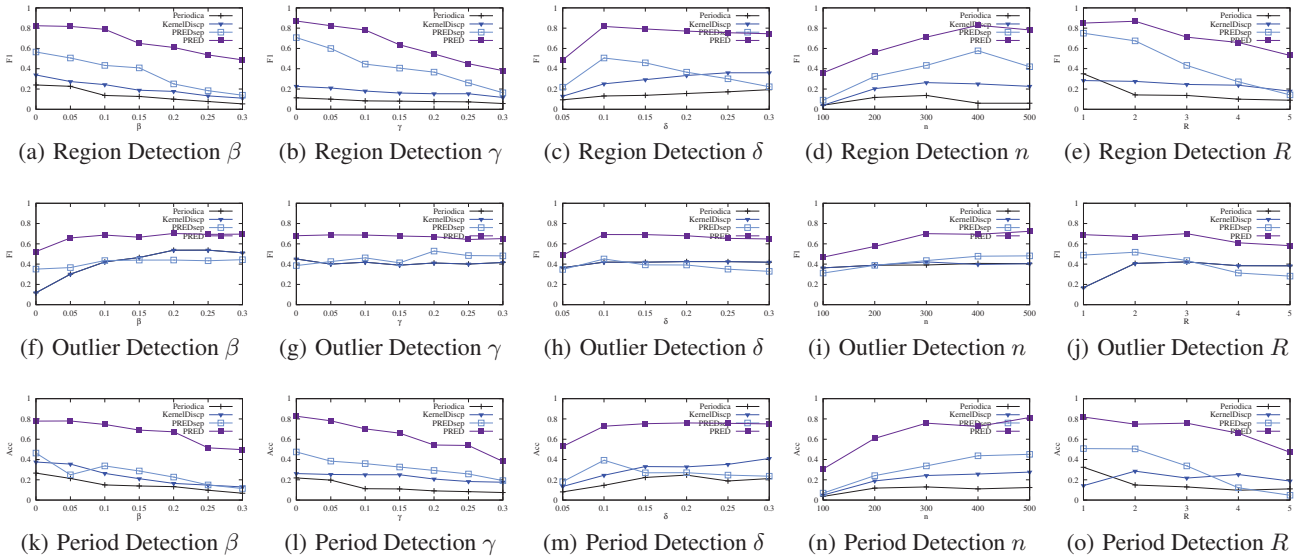
In summary, our proposed method PRED always generates the superior results. Potential reasons are two-fold: (1) the Dirichlet process is able to correctly identify the regions; and (2) jointly modeling location and time can help improve the effectiveness of periodic region detection.

Figure 7 shows the performance of PRED and PREDsep in region detection under different settings of hyper-parameters. The performance in other tasks share similar trends and thus are omitted due to space limit. We can find that the influence of hyper-parameters is limited, and PRED always outperforms PDsep.

4.3 Location Prediction on Social Media Data

4.3.1 Experimental Settings

Dataset and Evaluation Metric: We use Gowalla check-ins (**Gowalla**) and Tweets (**Twitter**) as two datasets for location prediction, *i.e.,* to predict the coordinates of a target user at a target

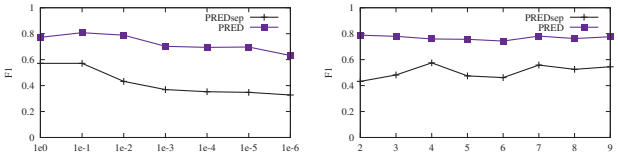


(a) Region Detection β (b) Region Detection γ (c) Region Detection δ (d) Region Detection n (e) Region Detection R

(f) Outlier Detection β (g) Outlier Detection γ (h) Outlier Detection δ (i) Outlier Detection n (j) Outlier Detection R

(k) Period Detection β (l) Period Detection γ (m) Period Detection δ (n) Period Detection n (o) Period Detection R

Figure 6: Comparison Results with Various Geo Parameter Settings



(a) κ_0 (b) ρ_0

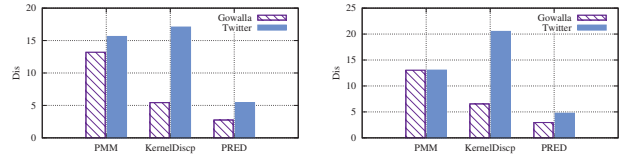
Figure 7: Parameter Influence on Region Detection

time. We extract Gowalla dataset from the data published by Cho *et al.* [4], and collect the geo-annotated tweets from the most recent 3,200 tweets of Twitter users. For each check-in and tweet, we map it to its corresponding city by reverse geo-coding, and retain the check-ins and tweets posted within the most frequently visited city of each user. Users who have at least 70 records are kept in datasets (each user must have more than 10 records on each day in [4], but most users cannot meet this requirement). For Gowalla, we sort users based on ID and keep the first 550 valid users, and for Twitter, we randomly selected 550 valid users. In the end, the average number of records of a user is 331.67 and 663.86, and the average sampling rate is 0.057 and 0.049 in Gowalla and Twitter, respectively. For each user, we use the first 90% of her records to build models, and use the remaining records to test the prediction performance. The effectiveness is measured by Macro (averaged by users) and Micro (averaged by test instances) Error distance (Dis) [51], which is the Euclidean distance between the true and predicted location of a testing record.

Methods for Comparison We use **PMM** [4] as one baseline because it is the most relevant method that can predict location at any time on GPS coordinates data without requiring social information. In PMM, we build two GMMs for weekdays and weekends respectively instead of one GMM for each day, because the performance of the latter is worse. We use **KernelDiscp** as another baseline, which can find the active timeslot in a period, and has much better effectiveness than its original version Periodica [27]. Specifically, after training the model, we know the weight $P(r)$ and period T_r of each region r, and also know the weight $P_r(i)$ of each timeslot $i \in [0, T_r]$. Given a test instance d with time t, we select the region r that can maximize $P(r)P_r(t\%T_r)$, and return its center of mass as the result. There are many other methods on location prediction [24, 29, 33, 38, 44, 51]. We do not compare with them because they require venue information that is often not available in social networks like Twitter. Since PMM is one of the baselines already, we do not further compare with its variation [41]. We set the prior of period ϵ_0 in our model **PRED** to be 50 in this section for the purpose of fair comparison, while other parameters remain the same in Sections 4.1 and 4.2.

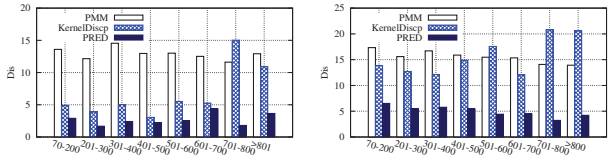
4.3.2 *Performance Study*

Figure 8 plots the macro and micro error distance of different methods on two datasets. We can observe that the error distance of PMM is large, probably because (1) two regions are not sufficient to model a user's mobility, and (2) a user may have mobility periods other than 1 week. KernelDiscp, which can discover various number of regions and their corresponding periods, generates better performance on Gowalla. However, its performance on Twitter is worse. This is because in Twitter data, each user has much more records. Since KernelDiscp does not consider time when discovering regions, it is more likely to group records with diverse periods into one region, making it difficult to estimate the correct region periods for location prediction. Our model PRED outper-



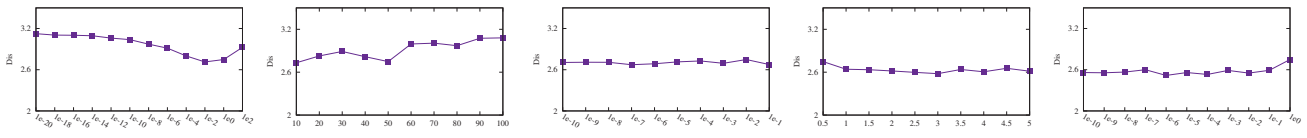
(a) Macro Error Distance (b) Micro Error Distance

Figure 8: Location Prediction Error Distance



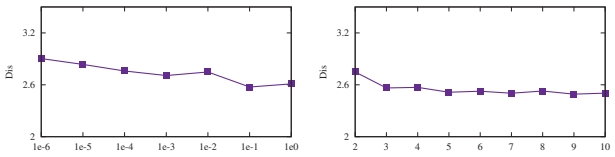
(a) Gowalla (b) Twitter

Figure 9: Error Distance on Different User Groups



(a) α (b) ϵ_0 (c) λ_0 (d) τ_0 (e) ψ_0

Figure 10: Error Distance under Different Temporal Hyper-parameters



(a) κ_0 (b) ρ_0

Figure 11: Error Distance under Different Spatial Hyper-parameters

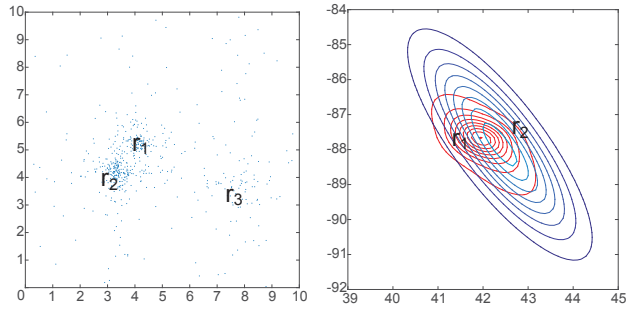
forms the baselines significantly on both datasets in both metrics (*e.g.,* the Micro Error Distance of PRED on Twitter is only 36.30% and 23.09% of that of PMM and KernelDiscp, respectively), owing to that we jointly model spatial and temporal information and detect regions and periods simultaneously.

We are also interested in whether one method can perform well even for users with few records. Thus, for each dataset, we divide users into 10 groups based on their record count N, and plot their respective error distance in Figure 9. We find that KernelDiscp performs worse for users with more records, which is in accordance with our previous observations on Twitter data. Comparing with the baselines, our model consistently achieves superior performance for users with different number of records.

We plot the error distance of PRED under different parameter settings in Figures 10 and 11. We find that PRED is not sensitive to the hyper-parameters, even for ϵ_0. Potential reason is the true periods in real-world data are often greater than 50.

4.4 Case Study

We use two cases to illustrate the results of the proposed PRED model. Figure 12(a) plots the synthetically generated dataset, in which there are three regions r_1, r_2 and r_3. Among them, r_1 and r_2 are close to each other, and the records of r_3 are scattered over a large area. The periods of the three regions are 30, 51, 82, respectively. On this dataset, the kernel density estimated based methods Periodica and KernelDiscp treat r_1 and r_2 as a single region. In addition, they cannot detect r_3 because of its low density. As a result, these two baselines cannot detect the correct periodic regions. In contrast, our proposed model PRED is able to extract the regions and detect their corresponding periods, because it jointly exploits spatial and temporal information.



(a) Synthetic Data (b) Twitter User Data

Figure 12: Case Study

We also train the PRED model on the records of a randomly selected Twitter user. PRED discovers 4 periodic regions with period 24 hours (1 day), and 1 region with period 168 hours (1 week). Since the distances among the 4 regions with period 24 hours are small, we combine them as a new region. We plot the contours of the two periodic regions in Figure 12(b), where region r_1 has period 24 hours, and region r_2 has period 168 hours. This case study shows that our proposed method PRED is effective in discovering periodic regions from geo-annotated social media records.

5. CONCLUSION

In this paper, we studied the novel problem of extracting periodic mobility patterns from the noisy and incomplete social media data, and proposed a Bayesian non-parametric method that jointly models geographical and temporal information. Different from existing work, our method does not need a-prior knowledge about user mobility and is robust to noise by modeling the time gap between records instead of exact visiting time. Our extensive experiments on both synthetic and real-world data show that our model outperforms the state-of-the-art methods significantly. In the future, it is interesting to use the extracted periodic patterns to improve other important location-based applications, such as location recommendation [18, 29, 57] and local event detection [11, 56].

6. ACKNOWLEDGEMENTS

Research was sponsored in part by the U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), National Science Foundation IIS-1320617 and IIS 16-18481, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov). Gao Cong is supported by a Singapore MOE Tier-2 Grant (MOE-2016-T2-1-137). The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies of the U.S. Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

7. REFERENCES

[1] C. Berberidis, W. G. Aref, M. Atallah, I. Vlahavas, A. K. Elmagarmid, et al. Multiple and partial periodicity mining in time series databases. In *ECAI*, volume 2, pages 370–374, 2002.

[2] D. Brockmann, L. Hufnagel, and T. Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–5, 2006.

[3] M. Chen, Y. Liu, and X. Yu. Nlpmm: a next location predictor with markov modeling. In *PAKDD*, pages 186–197. Springer, 2014.

[4] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD*, pages 1082–1090, 2011.

[5] B. Deb and P. Basu. Discovering latent semantic structure in human mobility traces. In *Wireless Sensor Networks*, pages 84–103. Springer, 2015.

[6] Y. Ding and X. Li. Time weight collaborative filtering. In *CIKM*, pages 485–492, 2005.

[7] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid. Periodicity detection in time series databases. *TKDE*, 17(7):875–887, 2005.

[8] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid. Warp: time warping for periodicity detection. In *ICDM*, pages 138–145, 2005.

[9] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid. Stagger: Periodicity mining of data streams using expanding sliding windows. In *ICDM*, pages 188–199, 2006.

[10] S. Feng, X. Li, Y. Zeng, G. Cong, Y. M. Chee, and Q. Yuan. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI*, 2015.

[11] W. Feng, C. Zhang, W. Zhang, J. Han, J. Wang, C. Aggarwal, and J. Huang. STREAMCUBE: hierarchical spatio-temporal hashtag clustering for event exploration over the twitter stream. In *ICDE*, pages 1561–1572, 2015.

[12] H. Gao, J. Tang, and H. Liu. Mobile location prediction in spatio-temporal context. In *Nokia mobile data challenge workshop*, 2012.

[13] E. F. Glynn, J. Chen, and A. R. Mushegian. Detecting periodic patterns in unevenly spaced gene expression time series using lomb–scargle periodograms. *Bioinformatics*, 22(3):310–316, 2006.

[14] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[15] J. Han, G. Dong, and Y. Yin. Efficient mining of partial periodic patterns in time series database. In *ICDE*, pages 106–115, 1999.

[16] J. Han, W. Gong, and Y. Yin. Mining segment-wise periodic patterns in time-related databases. In *KDD*, pages 214–218, 1998.

[17] L. Hong, A. Ahmed, S. Gurumurthy, A. J. Smola, and K. Tsioutsiouliklis. Discovering geographical topics in the twitter stream. In *WWW*, pages 769–778, 2012.

[18] B. Hu and M. Ester. Spatial topic modeling in online social media for location recommendation. In *RecSys*, pages 25–32, 2013.

[19] S. Isaacman, R. Becker, R. Cáceres, M. Martonosi, J. Rowland, A. Varshavsky, and W. Willinger. Human mobility modeling at metropolitan scales. In *MobiSys*, pages 239–252, 2012.

[20] M. Jiang, P. Cui, F. Wang, X. Xu, W. Zhu, and S. Yang. Fema: flexible evolutionary multi-faceted analysis for dynamic behavioral pattern discovery. In *KDD*, pages 1186–1195. ACM, 2014.

[21] M. Jiang, C. Faloutsos, and J. Han. Catchtartan: Representing and summarizing dynamic multicontextual behaviors. In *KDD*. ACM, 2016.

[22] H. Kamper. Gibbs sampling for fitting finite and infinite gaussian mixture models. 2013.

[23] J. Klafter, M. F. Shlesinger, and G. Zumofen. Beyond brownian motion. *Physics today*, 49(2):33–39, 1996.

[24] R. Kumar, M. Mahdian, B. Pang, A. Tomkins, and S. Vassilvitskii. Driven by food: Modeling geographic choice. In *WSDM*, pages 213–222, 2015.

[25] V. B. Le, O. Mella, D. Fohr, et al. Speaker diarization using normalized cross likelihood ratio. In *INTERSPEECH*, volume 7, pages 1869–1872, 2007.

[26] E. Levina and P. Bickel. The earth mover's distance is the mallows distance: Some insights from statistics. In *ICCV*, volume 2, pages 251–256, 2001.

[27] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye. Mining periodic behaviors for moving objects. In *KDD*, pages 1099–1108, 2010.

[28] Z. Li, J. Wang, and J. Han. Mining event periodicity from incomplete observations. In *KDD*, pages 444–452, 2012.

[29] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui. Geomf: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *KDD*, pages 831–840, 2014.

[30] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In *AAAI*, 2016.

[31] S. Ma and J. L. Hellerstein. Mining partially periodic event patterns with unknown periods. In *ICDE*, pages 205–214, 2001.

[32] W. Mathew, R. Raposo, and B. Martins. Predicting future locations with hidden markov models. In *UbiComp*, pages 911–918, 2012.

[33] J. McGee, J. Caverlee, and Z. Cheng. Location prediction in social media based on tie strength. In *CIKM*, pages 459–468, 2013.

[34] A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. Wherenext: a location predictor on trajectory pattern mining. In *KDD*, pages 637–646, 2009.

[35] M. A. Nishi, C. F. Ahmed, M. Samiullah, and B.-S. Jeong. Effective periodic pattern mining in time series databases. *Expert Systems with Applications*, 40(8):3015–3027, 2013.

[36] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. Mining user mobility features for next place prediction in location-based services. In *ICDM*, pages 1038–1043, 2012.

[37] F. Rasheed, M. Alshalalfa, and R. Alhajj. Efficient periodicity mining in time series databases using suffix trees. *TKDE*, 23(1):79–94, 2011.

[38] A. Sadilek, H. Kautz, and J. P. Bigham. Finding your friends and following them to where you are. In *WSDM*, pages 723–732, 2012.

[39] C. Sheng, W. Hsu, and M. Li Lee. Mining dense periodic patterns in time series data. In *ICDE*, pages 115–115, 2006.

[40] F. Simini, M. C. González, A. Maritan, and A.-L. Barabási. A universal model for mobility and migration patterns. *Nature*, 484(7392):96–100, 2012.

[41] A. Tarasov, F. Kling, and A. Pozdnoukhov. Prediction of user location using the radiation model and social check-ins. In *KDD Workshop on Urban Computing*, page 8, 2013.

[42] M. Vlachos, S. Y. Philip, and V. Castelli. On periodicity detection and structural periodic similarity. In *SDM*, volume 5, pages 449–460. SIAM, 2005.

[43] W. Wang, J. Yang, and P. S. Yu. Meta-patterns: revealing hidden periodic patterns. In *ICDM*, pages 550–557, 2001.

[44] Y. Wang, N. J. Yuan, D. Lian, L. Xu, X. Xie, E. Chen, and Y. Rui. Regularity and conformity: Location prediction using heterogeneous mobility data. In *KDD*, pages 1275–1284, 2015.

[45] J. Yang, W. Wang, and P. S. Yu. Infominer: mining surprising periodic patterns. In *KDD*, pages 395–400, 2001.

[46] J. Yang, W. Wang, and P. S. Yu. Infominer+: mining partial periodic patterns with gap penalties. In *ICDM*, pages 725–728, 2002.

[47] K.-J. Yang, T.-P. Hong, Y.-M. Chen, and G.-C. Lan. Projection-based partial periodic pattern mining for event sequences. *Expert Systems with Applications*, 40(10):4232–4240, 2013.

[48] J. Ye, Z. Zhu, and H. Cheng. What's your next move: User activity prediction in location-based social networks. In *SDM*. SIAM, 2013.

[49] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Geographical topic discovery and comparison. In *WWW*, pages 247–256, 2011.

[50] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372. ACM, 2013.

[51] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Who, where, when and what: discover spatio-temporal topics for twitter users. In *KDD*, pages 605–613, 2013.

[52] Q. Yuan, G. Cong, and A. Sun. Graph-based point-of-interest recommendation with geographical and temporal influences. In *CIKM*, pages 659–668, 2014.

[53] Q. Yuan, G. Cong, K. Zhao, Z. Ma, and A. Sun. Who, where, when, and what: A nonparametric bayesian approach to context-aware recommendation and search for twitter users. *TOIS*, 33(1):2, 2015.

[54] C. Zhang, J. Han, L. Shou, J. Lu, and T. La Porta. Splitter: Mining fine-grained sequential patterns in semantic trajectories. *VLDB Endowment*, 7(9):769–780, 2014.

[55] C. Zhang, K. Zhang, Q. Yuan, L. Zhang, T. Hanratty, and J. Han. Gmove: Group-level mobility modeling using geo-tagged social media. In *KDD*, pages 1305–1314, 2016.

[56] C. Zhang, G. Zhou, Q. Yuan, H. Zhuang, Y. Zheng, L. Kaplan, S. Wang, and J. Han. Geoburst: Real-time local event detection in geo-tagged tweet streams. In *SIGIR*, pages 513–522, 2016.

[57] W. Zhang and J. Wang. Location and time aware social collaborative retrieval for new successive point-of-interest recommendation. In *CIKM*, pages 1221–1230, 2015.

[58] K. Zheng, Y. Zheng, N. J. Yuan, and S. Shang. On discovery of gathering patterns from trajectories. In *ICDE*, pages 242–253, 2013.