What's Your Next Move: User Activity Prediction in Location-based Social Networks

Jihang Ye* Zhe Zhu* Hong Cheng†

Abstract

Location-based social networks have been gaining increasing popularity in recent years. To increase users' engagement with location-based services, it is important to provide attractive features, one of which is geo-targeted ads and coupons. To make ads and coupon delivery more effective, it is essential to predict the location that is most likely to be visited by a user at the next step. However, an inherent challenge in location prediction is a huge prediction space, with millions of distinct check-in locations as prediction target. In this paper we exploit the check-in category information to model the underlying user movement pattern. We propose a framework which uses a mixed hidden Markov model to predict the category of user activity at the next step and then predict the most likely location given the estimated category distribution. The advantages of modeling the category level include a significantly reduced prediction space and a precise expression of the semantic meaning of user activities. Extensive experimental results show that, with the predicted category distribution, the number of location candidates for prediction is 5.45 times smaller, while the prediction accuracy is 13.21% higher.

1 Introduction

Location-based social networks (LBSNs) [21] have been increasingly popular recently, in which millions of users are sharing their locations or geo-tagged information with friends through *check-ins*. LBSNs have largely influenced people's life style – they can share their experiences in a timely fashion; meanwhile, they also keep informed of the most up-to-date trends. To offer better service, it is very important for LBSN providers to predict the most likely location to be visited by a user. By knowing the next move of a user, LBSN providers can make geo-targeted ads and coupon delivery more effective. Hence, accurate location prediction can help improve the user experience and increase users' engagement with LBSN consequently.

At first glance, existing location prediction methods on GPS trajectories [22, 16] can be directly applied on the LBSN data, since the two data types exhibit certain similarity. But after analyzing the check-in records collected from a real online location-based social network, Gowalla, we found the LBSN check-in data exhibit some unique properties, which are different from the extensively studied GPS trajectories: (1) data sparseness. Only 10% users have more than 58 check-in records in the whole 12-month period, which is the average check-in times per user, showing a low check-in frequency. In addition, the spatial gap between any two consecutive check-ins is typically in the scale of kilometer, while the spatial gap between consecutively logged GPS points is typically $5-10$ meters [22]; (2) the semantic meaning. Each check-in record is labeled with the name and category of the location in LBSN, while a GPS point consists of only latitude, longitude and time stamp. The check-in category information reflects the user preference and the heterogeneity of user behavior. With these differences, the existing techniques designed on GPS trajectories are not suitable to be applied directly to the LBSN data.

An inherent challenge in location prediction is that the location prediction space is very huge – there may be millions of distinct check-in locations in an LBSN. As a result it is very hard to build a model by incorporating user movement pattern, preference and temporal spatial information to predict locations directly and achieve satisfactory performance. Considering this challenge, we propose to decompose the original problem into two sub-problems: (1) predicting the category of user activity at the next step; and then (2) predicting a location given the estimated category distribution. For example, an LBSN may predict a user's next activity to be *entertainment*, then it will likely predict the location to be a *cinema* in the user's vicinity. An obvious advantage of this approach is a significantly reduced prediction space, as there are only a small number of categories such as *food*, *shopping*, *entertainment*, etc. More importantly, by focusing on the category level we can model the underlying movement pattern of a user and capture the semantic meaning of the user's activities. Therefore, the problem we study in this work

*Authors contributed equally to this work.

†The Chinese University of Hong Kong. Email: {yjh010, zzhu}@alumni.ie.cuhk.edu.hk, hcheng@se.cuhk.edu.hk

is: *how do we predict a user's activity category at the next step and predict the most likely location, given a sequence of his/her check-in records as observation*?

In this work, we propose to use the hidden Markov model (HMM) to model the user movement pattern and the dependency between a user's check-in activities at the category level. The abstract states in HMM can well model the sparse LBSN data where users only check-in at points of interest (POIs) or certain places they deem important or interesting, as the hidden states capture the essential behavioral patterns of LBSN users, rather than focusing on the finer granularity and consecutive points as in GPS trajectories. As users' activities exhibit a strong temporal and spatial pattern, we train a mixed HMM by incorporating the temporal and spatial information, to further improve the model accuracy. After predicting the category of a user for the next step, we use some simple schemes to predict a location given the category distribution. To the best of our knowledge, our work is the first to model the user activity category for location prediction.

Extensive experimental evaluation shows that, (1) our mixed HMM with user clustering achieves 44.35% category prediction accuracy, outperforming the baseline methods by up to 14.85%; (2) when given the estimated category distribution, the number of location candidates we need to consider is 5.45 times smaller than without the predicted category, while the location prediction accuracy is 13.21% higher, as many irrelevant candidates are filtered out given the category; and (3) our method outperforms PMM [5], a human mobility model for location prediction by 31.89%.

The rest of this paper is organized as follows. Section 2 describes data collection procedure and data properties. Section 3 describes the HMM model for category prediction. Section 4 introduces several schemes for location prediction given the category distribution. We present extensive experimental results in Section 5 and discuss related work in Section 6. Finally Section 7 concludes our paper.

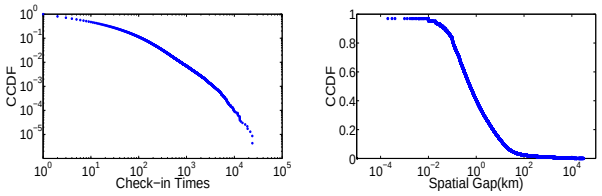
2 Data Analysis

In this section we describe our data collection procedure and analyze the data properties.

2.1 Data Collection Launched in 2009 as an online LBSN, Gowalla (http://gowalla.com) allows users to share their locations or geo-tagged information such as comments and photos with friends through check-ins. We have crawled 13 million check-in records of over 230,000 users from Gowalla for 12 months, from September 2009 to August 2010. For each check-in record we obtain the following information: user id, check-in time, latitude, longitude, name and category of the check-in location. The average check-in times of a user during the 12-month period is 58, thus we choose the users with check-in times no less than 58 as active users. After the data selection we get 23,040 users with a total of 6,634,176 check-in records, accounting for half of all the collected records. We treat the check-in records by a user within one day as a check-in sequence and thus obtain 1,054,689 sequences. Table 1 shows an example dataset with user check-in sequences.

Table 1: Example of User Check-in Sequences

User	Date	Check-in Records
John	1/13	⟨08:17, 41.89°, −87.65°, *Starbucks*, Food⟩, ⟨09:30, 41.88°, −87.63°, *City Hall*, Community⟩, ⟨12:35, 41.88°, −87.62°, *Subway*, Food⟩, ⟨17:22, 41.99°, −87.73°, *Macy's*, Shopping⟩
Andy	1/27	⟨11:37, 37.42°, −122.17°, *McDonald's*, Food⟩, ⟨18:01, 37.53°, −122.07°, *Park*, Outdoors⟩, ⟨19:30, 37.54°, −122.06°, *Italian*, Food⟩



(a) CCDF of Check-in Times Per User (b) CCDF of Spatial Gap Between Consecutive Check-ins

Figure 1: Data Statistics

2.2 Dataset Properties Figure 1(a) shows the complementary cumulative distribution function (CCDF) of the check-in times per user. The check-in data is very sparse; among all 230,000 users, only 23,040 (10%) users have more than 58 check-in records during the whole 12-month period, and only 1% have more than 760 check-in records during this period.

Figure 1(b) shows the CCDF of the spatial gap between any two consecutive check-ins within a day. 40% of all consecutive check-ins have a gap larger than 1 kilometer, much longer than the spatial gap between consecutive points in GPS trajectories, which is typically $5 - 10$ meters. This shows the sparseness of the check-in data from a different perspective. Thus the techniques for mining GPS trajectories [22] are not suitable to be applied to the LBSN data.

Figure 2 shows the number of check-ins by category and hour. In Gowalla there are 9 categories: *Community, Entertainment, Food, Night life, Outdoors, Shopping, Travel, Events, and None.* We can observe the following phenomena: (1) most categories exhibit a strong temporal pattern – the check-in times start to decrease since midnight and reach the minimum at 9 am. The check-in activities start to rise at 10 am and peak in the

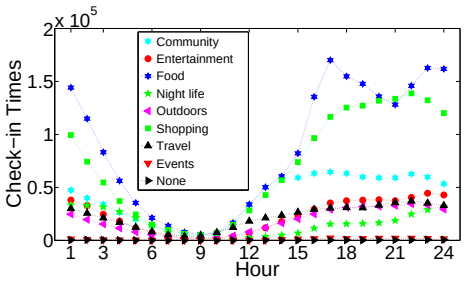


Figure 2: Check-ins by Category and Hour

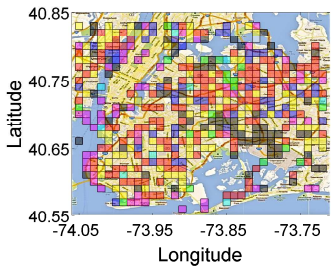


Figure 3: Category by Region in New York City. Community (blue), Entertainment (green), Food (red), Night life (cyan), Outdoors (magenta), Shopping (yellow), Travel (black), Events (white).

afternoon or evening; and (2) the check-in distribution by category is quite imbalanced – *Food* and *Shopping* are the two dominant categories, while *Events* and *None* have very few check-in records. Thus if we simply predict based on popularity, *Food* will always be predicted.

Figure 3 shows the spatial pattern of the check-in categories in New York City. We divide the whole area into 0.01 longitude by 0.01 latitude square regions and color a region with the dominant check-in category within that region, if the check-in percentage of the dominant category is 50% or above. We can observe that: (1) many regions are dominated by a certain category, exhibiting a strong correlation between the location and the type of activities; and (2) user activity categories are quite diverse across the spatial area.

3 User Activity Prediction

In this section we study how to predict the user activity at the category level.

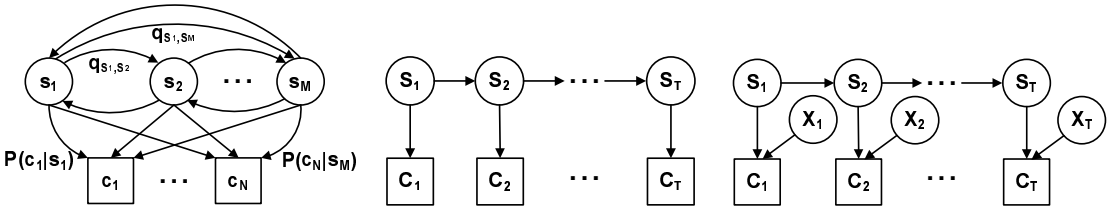
3.1 Definitions Consider a set of user check-in sequences $\mathcal{L} = \{l_1, l_2, \ldots, l_n\}$, and each sequence $l_i = r_1 r_2 \ldots r_{m_i}$ consists of a series of $m_i \geq 1$ check-in records. Each record r is a tuple in the form of $r = \langle uid, time, latitude, longitude, location, category \rangle$, where *uid* is the user id, *time* is the check-in time, *latitude* and *longitude* specify the geographic position of the check-in record. *location* is the name of the check-in location, e.g., Starbucks, Walmart, etc., and *category* is a label describing the type of location. In the collected dataset, there are totally 817,683 distinct locations and 9 categories.

Our goal is to learn a prediction model from $\mathcal{L}$. Given a test sequence $l_{test} = r_1 r_2 \ldots r_t$, we want to predict the location of the next check-in record r_{t+1}.

3.2 Category Prediction based on HMM As our dataset contains 817,683 distinct locations, this forms a huge prediction space for location prediction. In addition, as both the temporal and spatial gaps between two check-in locations are quite large, compared with the gaps between two logged points in GPS trajectory data, it is more difficult to model the dependency between two check-in locations. Consequently it is very challenging to predict the locations directly. But when we abstract to the more generalized category level, the dependency between categories reflects the dependency between different user status, e.g., *work*, *entertainment*, etc., with more statistical significance. Thus we decompose the prediction problem into two sub-problems: (1) predicting the category of user activity; and then (2) predicting the location of user activity given the predicted category distribution. In the following we will focus on the task of category prediction first.

We propose to use a hidden Markov model (HMM) to model the dependency between categories. In our category prediction problem, we define a set of hidden states $\mathcal{S}=\{s_1, s_2, \ldots, s_M\}$ which correspond to the underlying status of a user, and a set of observations $\mathcal{C}=\{c_1, c_2, \ldots, c_N\}$ which correspond to the nine categories in our problem, i.e., $N = 9$. The HMM has the following three key parameter components, which can fully describe the HMM: (1) initial state probability π_{s_i} for each hidden state $s_i \in \mathcal{S}$; (2) state transition probability q_{s_i,s_j} from the hidden state s_i to s_j where $s_i, s_j \in \mathcal{S}$; and (3) state-dependent output probability $\mathrm{P}(c_j|s_i)$, which determines the probability of the check-in category $c_j \in \mathcal{C}$ given the hidden state $s_i \in \mathcal{S}$. The graphical representation of the HMM with probabilistic parameters is shown in Figure 4(a).

We abstract a user's check-in activities within one day as a sequence of categories of length T, i.e., $l = C_1 C_2 \ldots C_T$ (abbreviated as $l = C_{1:T}$), and use such category observation sequences to train the HMM. $C_t \in \mathcal{C}$ is a random variable representing the observed category at time t, $1 \leq t \leq T$. Each C_t is uniquely associated with a random variable $S_t \in \mathcal{S}$, representing the unknown hidden state at time t. Figure 4(b) shows an instantiated HMM structure for the observation sequence $l = C_{1:T}$, illustrating the conditional dependencies between hidden states and observed categories. In the



(a) HMM with Probabilistic Parameters (b) Conditional Dependence Graph of HMM (c) Mixed HMM with Covariates

Figure 4: HMM Model for LBSN Prediction

following, we will discuss the parameter estimation in HMM, as well as category prediction based on the learnt HMM model.

3.2.1 Parameter Estimation for HMM Existing methods on parameter estimation for HMM, such as the Baum-Welch algorithm [4], use an EM approach which aims at maximizing the likelihood of the observation sequence. The overall likelihood of any observation sequence should be summed over all possible routes through the underlying hidden states and is expressed in Equation 3.1. Since we assume the HMM is time-homogeneous, the state transition probabilities and the state-dependent output probabilities do not change with time t.

(3.1)
$$P(C_{1:T}) = \sum_{S_1=s_1}^{s_M} \cdots \sum_{S_T=s_1}^{s_M} \pi_{S_1} \prod_{t=2}^{T} q_{S_{t-1},S_t} \prod_{t=1}^{T} P(C_t|S_t)$$

As Equation 3.1 computes the summation over M^T components, it is computationally intractable. According to [23], we can reformulate Equation 3.1 as Equation 3.2, which is expressed by matrix multiplications and can reduce the computational cost. Here π is a $1 \times M$ initial state distribution vector, $\mathcal{Q}$ is an $M \times M$ hidden state transition matrix where $\mathcal{Q}_{ij} = q_{s_i,s_j}$, and $\mathcal{P}_{C_t}$ is an $M \times M$ diagonal matrix with $P(C_t|s_i)$ $(1 \le i \le M)$ on the diagonal and other entries as 0. Then we can use the Baum-Welch algorithm to estimate the hidden state transition probabilities and the state-dependent output probabilities.

$$L_T = P(C_{1:T}) = \pi \mathcal{P}_{C_1} \mathcal{Q} \mathcal{P}_{C_2} \cdots \mathcal{Q} \mathcal{P}_{C_T} \mathbf{1}^\top \tag{3.2}$$

Since we consider a time-homogeneous transition matrix $\mathcal{Q}$, the initial state distribution π for a hidden Markov chain is defined as the stationary distribution of the transition matrix $\mathcal{Q}$, which satisfies $\pi = \pi\mathcal{Q}$. Therefore, π can be estimated when $\mathcal{Q}$ is available.

To decide the right number of hidden states M in learning an HMM, we use *Bayesian Information Criterion* (BIC) [15] to evaluate the HMM with different state numbers, and a smaller BIC value always leads to better model fitness.

3.2.2 Check-in Category Prediction When given a length-t test category sequence $l_{test} = C_{1:t}$, we can predict the category C_{t+1} at time $t+1$ with the learnt HMM. This prediction is divided into two steps:

$$S_{t+1} = \arg\max_{s_i \in \mathcal{S}} P(S_{t+1} = s_i | C_{1:t})$$
$$C_{t+1} = \arg\max_{c_j \in \mathcal{C}} P(c_j | S_{t+1})$$

where we compute the most likely hidden state S_{t+1} first, and then predict the most likely category C_{t+1} given S_{t+1}.

Prediction of the next state S_{t+1} given the observation $C_{1:t}$ can be computed from $q_{S_t,S_{t+1}}$ and $P(S_t|C_{1:t})$, according to the law of total probability:

$$P(S_{t+1} = s_j | C_{1:t}) = \sum_{i=1}^{M} q_{s_i,s_j} P(S_t = s_i | C_{1:t}) \tag{3.3}$$

where $P(S_t = s_i|C_{1:t})$ can be computed using the "filtering" approach. Specifically, we compute $P(S_t = s_i|C_{1:t})$ from Equation 3.4 where $\mathcal{Q}_i$ is the i^{th} column of transition matrix $\mathcal{Q}$, and L_t is the likelihood of $C_{1:t}$ computed from Equation 3.2.

(3.4)
$$P(S_t = s_i|C_{1:t}) = (\pi \mathcal{P}_{C_1} \mathcal{Q} \mathcal{P}_{C_2} \mathcal{Q} \cdots \mathcal{P}_{C_{t-1}} \mathcal{Q}_i P(C_t|S_t))/L_t$$

After getting the most likely hidden state S_{t+1}, we obtain the category distribution $P(c_j|S_{t+1})$, $1 \le j \le N$, and predict the most likely category as $C_{t+1} = \arg\max_{c_j \in \mathcal{C}} P(c_j|S_{t+1})$.

3.3 Mixed HMM with Temporal and Spatial Covariates The HMM described above only models the dependencies (or transitions) between the check-in categories. Given the same observation sequence $C_{1:t}$, the HMM always generates the same prediction regardless of the specific temporal and spatial information of the observed check-ins. However, in reality, the LBSN users' check-in behaviors are influenced by their surroundings. For example, the HMM may predict the most likely category as *Outdoors* after observing *Food*. However, if given the current time as midnight, the most

likely next category will be *Night life* instead; or if given the current user location as a shopping mall, then the most likely next category will be *Shopping* instead. Figures 2 and 3 also show how the category strength is affected by temporal and spatial factors. Therefore, we propose to build a more general prediction model, known as mixed HMM [1], by incorporating both temporal and spatial covariates to improve the model accuracy. Such a mixed HMM is "context aware", i.e., exploits the knowledge of user time and geographic position. Figure 4(c) shows the graphical representation of the mixed HMM, where the check-in category C_t is determined by both the hidden states and the temporal spatial covariates. We use $\overrightarrow{X_t}$ to represent the temporal spatial covariates at time t.

To formulate the state-dependent probability when incorporating the temporal spatial covariates $\overrightarrow{X_t}$, we follow the multivariate logit model, which has been studied extensively to model parametric probability function [7].

(3.5)
$$P(C_t = c_j | S_t = s_i, \overrightarrow{X_t}) = \frac{\exp(\alpha_{s_i}^{c_j} + \overrightarrow{\beta_{s_i}^{c_j}} \cdot \overrightarrow{X_t})}{\sum_{k=1}^{N} \exp(\alpha_{s_i}^{c_k} + \overrightarrow{\beta_{s_i}^{c_k}} \cdot \overrightarrow{X_t})}$$

where $\alpha_{s_i}^{c_k}$ is a state s_i-specific coefficient for the observed check-in category c_k (also known as "intercept"), $1 \le k \le N$, and $\overrightarrow{\beta_{s_i}^{c_k}}$ is a vector of state s_i-specific response coefficients for category c_k under the temporal spatial covariates $\overrightarrow{X_t}$.

We use a numerical vector to represent the check-in time information in terms of days of the week, i.e., Sunday to Saturday, and hours of a day. To represent the spatial information, a straightforward way is to use the latitude and longitude of the check-in records. However, the latitude and longitude do not reflect the property of a location related to category prediction. Thus we use the following representation of spatial information: we collect all check-in records for a location and compute a probability distribution of the check-in categories. The category distribution vector of a location is used as the spatial covariate. The temporal and spatial information of a check-in record at time t jointly forms the covariates $\overrightarrow{X_t}$.

3.3.1 Parameter Estimation for Mixed HMM

In the mixed HMM, for each hidden state and observed check-in category pair (s_i, c_j), where $s_i \in \mathcal{S}$ and $c_j \in \mathcal{C}$, we need to estimate the intercept parameter $\alpha_{s_i}^{c_j}$ and the response parameter vector $\overrightarrow{\beta_{s_i}^{c_j}}$. As there are totally $M \times N$ state-category pairs, the number of estimated parameters in the mixed HMM is much larger than in the basic HMM. EM algorithm is very slow when handling a large number of parameters. Moreover, it is very complicated to maximize the state-dependent probability in Equation 3.5 through derivation. To overcome these problems, we consider another way in HMM parameter estimation, Markov chain Monte Carlo (MCMC) Bayes Estimation, an iterative sampling approach.

The estimation of q_{s_i,s_j} in Section 3.2.1 is subject to the constraint $\sum_{j=1}^{M} q_{s_i,s_j} = 1$, which rarely happens during sampling. In order to satisfy this constraint and facilitate MCMC Bayes Estimation, we specify the state transition probability q_{s_i,s_j} from the hidden state s_i to s_j as follows:

$$\begin{aligned} q_{s_i,s_1} &= \frac{\exp(\mu_{s_i,s_1})}{1+\exp(\mu_{s_i,s_1})} \\ q_{s_i,s_j} &= \frac{\exp(\mu_{s_i,s_j})}{1+\exp(\mu_{s_i,s_j})} - \frac{\exp(\mu_{s_i,s_{j-1}})}{1+\exp(\mu_{s_i,s_{j-1}})} \\ q_{s_i,s_M} &= 1 - \frac{\exp(\mu_{s_i,s_{M-1}})}{1+\exp(\mu_{s_i,s_{M-1}})} \\ & i \in \{1,2,\ldots,M\}, j \in \{2,\ldots,M-1\}, \\ & \mu_{s_i,s_1} \le \mu_{s_i,s_2} \le \cdots \le \mu_{s_i,s_{M-1}} \end{aligned} \tag{3.6}$$

The above formulation converts the original problem of estimating q_{s_i,s_j} to estimating a newly introduced variable μ_{s_i,s_j}. In Equation 3.6 we impose a constraint on $\mu_{i,j}$ to be non-decreasing with j when given a specific i, to make sure the estimated $q_{s_i,s_j} \ge 0$.

From Equations 3.5 and 3.6, we need to estimate the parameter vector $\overrightarrow{\psi} = \{\mu_{i,j}, \alpha_k^m, \overrightarrow{\beta_k^m}\}$ $(i,j,k \in \mathcal{S}, m \in \mathcal{C})$ in the mixed HMM with the temporal spatial covariates. MCMC Bayes Estimation aims at sampling properly to simulate the original parameter according to its posterior distribution. Since normal distribution is always introduced to simulate parameter prior, we employ multivariate normal distribution to describe the prior distribution of $\overrightarrow{\psi}$. According to Bayes' Theorem, the conditional posterior distribution of $\overrightarrow{\psi}$ can be defined with multivariate normal priors as

$$\begin{aligned} & \{\overrightarrow{\psi} | \overrightarrow{X_1}, \overrightarrow{X_2}, \ldots, \overrightarrow{X_T}, C_1, C_2, \ldots, C_T\} \\ & \propto L_{\overrightarrow{\psi}}(\mathbf{X}, \mathcal{L}) N(\overrightarrow{\psi})_{\overrightarrow{\psi_0}, \mathbf{V}_{\psi_0}} \\ & \propto L_{\overrightarrow{\psi}} |\mathbf{V}_{\psi_0}|^{-1/2} \exp[-\frac{1}{2}(\overrightarrow{\psi} - \overrightarrow{\psi_0})' |\mathbf{V}_{\psi_0}|^{-1} (\overrightarrow{\psi} - \overrightarrow{\psi_0})] \end{aligned} \tag{3.7}$$

where $\overrightarrow{\psi_0}$ and $\mathbf{V}_{\psi_0}$ are properly assumed as diffuse priors of estimated parameters $\overrightarrow{\psi}$, $L_{\overrightarrow{\psi}}$ is computed according to Equation 3.2 when given parameters $\overrightarrow{\psi}$, $\mathcal{L}$ is the set of observed sequences and $\mathbf{X}$ is the corresponding temporal spatial covariates. Since Equation 3.7 does

Algorithm 1 Metroplis-Hastings Bayes Sampling

Input: K(iterations), $\mathcal{L}$(sequences), $\mathbf{X}$(covariates)
Output: $\overrightarrow{\psi_1}, \overrightarrow{\psi_2}, \ldots, \overrightarrow{\psi_K}$(sampled parameters)

1: $\overrightarrow{\psi_0} \leftarrow$ initial assignment of $\overrightarrow{\psi}$;
2: **for** $j = 1 : K$ **do**
3: generate $\overrightarrow{\Delta\theta} \sim N(\mathbf{0}, \sigma^2\Theta)$, $\overrightarrow{\theta} \leftarrow \overrightarrow{\psi_{j-1}} + \overrightarrow{\Delta\theta}$;
4: generate $\gamma \sim U(0, 1)$;
5: **if** $\gamma \leq \min\{\frac{L_{\overrightarrow{\theta}}(\mathbf{X},\mathcal{L})\cdot N(\overrightarrow{\theta})_{\overrightarrow{\psi_0},\mathbf{V}_{\psi_0}}}{L_{\overrightarrow{\psi_{j-1}}}(\mathbf{X},\mathcal{L})\cdot N(\overrightarrow{\psi_{j-1}})_{\overrightarrow{\psi_0},\mathbf{V}_{\psi_0}}}, 1\}$ **then**
6: $\overrightarrow{\psi_j} \leftarrow \overrightarrow{\theta}$;
7: **else**
8: $\overrightarrow{\psi_j} \leftarrow \overrightarrow{\psi_{j-1}}$;

not have a closed form, the Metropolis-Hastings (M-H) algorithm is used to draw from the conditional distribution of $\overrightarrow{\psi}$. The complete estimation procedure is shown in Algorithm 1. $\overrightarrow{\Delta\theta}$ is also a random draw from multivariate normal distribution $N(\mathbf{0}, \sigma^2\Theta)$; to reduce the autocorrelation among the MCMC draws, σ and Θ are chosen adaptively according to [2]. After collecting all the $\{\overrightarrow{\psi_1}, \ldots, \overrightarrow{\psi_K}\}$ from M-H sampling, it is trivial to estimate values of each respective parameter in $\overrightarrow{\psi}$.

With the estimated parameters in the mixed HMM, prediction of hidden state S_{t+1} and check-in category C_{t+1} can be done similarly as in Section 3.2.

3.4 User Preference Modeling Besides temporal and spatial information, the LBSN users' check-in behaviors will also be influenced by their preferences or interests. Intuitively, if users have similar preferences, they will likely have similar check-in patterns; similarly, if users have quite different preferences, they will likely have quite different check-in patterns. It is less accurate to use a single HMM to model all users' behaviors who may have diverse interests. Thus we consider modeling user preference to further improve the model accuracy.

In this work, we propose to model user preference with his/her check-in activities. Specifically, let c_j^i be the total number of check-in records of user i on category j. The preference of user i is defined as the distribution of his/her check-in categories, $\overrightarrow{p^i} = (p_1^i, p_2^i, \ldots, p_N^i)$, where $p_j^i = \frac{c_j^i}{\sum_{k=1}^{N} c_k^i}$. With this preference feature vector, we can group users with similar preferences into clusters using k-means. For each cluster, we will train an HMM using the check-in records belonging to that cluster.

When given a test sequence $l_{test} = C_{1:t}$ by a user u, if u is a returning user, we can find the cluster which u belongs to, and then use the corresponding HMM to make the prediction C_{t+1}. But if u is a new user with no past check-in activities, we need to determine which cluster u should belong to. A possible solution is to compute the preference vector $\overrightarrow{p^u}$ from the only available observation sequence $C_{1:t}$ of u and then find the closest cluster to $\overrightarrow{p^u}$. But if the observation $C_{1:t}$ is too sparse to estimate the user preference, we can resort to the original non-clustered HMM.

4 Location Prediction

Given the category distribution generated by HMM, we predict the most likely location to be visited. In this paper, we simply use a ranking approach on the training check-in records to predict a user's location. The ranking score of a location is treated as an indicator of the predicted probability of that location. Given a user's current physical position, i.e., latitude and longitude, we draw a $d \times d$ square region centered at the user's position, where d is the region range parameter. We consider all locations falling into the region as candidates for prediction. Given the most likely category, we rank the candidate locations belonging to this category and return top-1, or more generally, top-k ($k \geq 1$) locations as our prediction. Note that in our scenario it is reasonable to predict more than one location (i.e., $k > 1$), as in reality an LBSN provider always delivers coupons of multiple retailers or brands. If there is no location belonging to the first category in the square region, we will rank locations belonging to the second most likely category and so on.

We propose four location ranking schemes:

- *Check-in count*: rank a location based on the total check-in times on the location.
- *User count*: rank a location based on the total number of users who have ever checked in on the location.
- *User count* × *Check-in count*: rank a location based on the product of User count and Check-in count.
- *Max check-in count by user*: rank a location based on the maximum check-in times by a user on the location.

5 Experimental Evaluation

We present extensive experimental evaluation of our methods for category and location prediction.

5.1 Data Preparation As introduced in Section 2.1, we have collected 1,054,689 check-in sequences. We randomly select 900,000 check-in sequences as the training set to learn the HMM and use the remaining sequences as the test set. In the category prediction

phase, for each test sequence $l_{test} = r_1 r_2 \dots r_t$, we randomly choose a time stamp $1 < t' \leq t$, and use all check-in categories $C_1 C_2 \dots C_{t'-1}$ prior to t' as well as the check-in temporal and spatial information till t' as the observation to predict $C_{t'}$. Then we compare the predicted $C_{t'}$ with the ground truth to calculate the category prediction accuracy. In the second stage, we predict the most likely location at time t' given the predicted category distribution, then compare it with the ground truth and report the prediction accuracy.

5.2 Category Prediction First, we learn the HMM from the training set for category prediction. According to BIC [15], we define 8 states in basic HMM (BHMM) and 5 states in mixed HMM (MHMM). We compare the category prediction accuracy of our HMM based approaches with the following baseline schemes.

- Global Frequency (**Freq**): Use the most frequent check-in category within a cluster as prediction.
- Order-n Markov: Predict the next category from its preceding length-n sequence of categories. Here we consider $n = 1$ and $n = 2$. It has been utilized widely in GPS trajectories or WiFi network [16].
- HITS Ordering (**HITS**): Predict the most likely category $C_{t'}$ using HITS algorithm when given category $C_{t'-1}$. This method was designed for location recommendation in GPS trajectories [22].

We clustered the training sequences into k clusters based on user preference and tested $k = 3, 5, 7, 9$. For comparison, we also tested the category prediction accuracy without clustering, i.e., $k = 1$. The category prediction accuracy is shown in Figure 5.

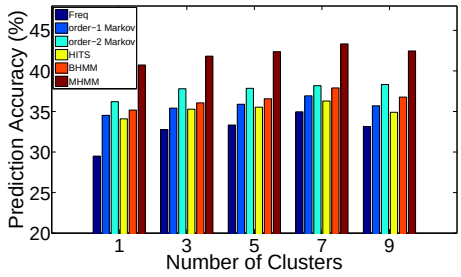


Figure 5: Category Prediction Accuracy

From Figure 5, we can observe that, (1) mixed HMM achieves the highest accuracy for all k values, outperforming all other methods by a large margin. The prediction accuracy of mixed HMM is 5.56% – 6.44% higher than that of basic HMM, by incorporating the temporal spatial covariates; and (2) when we model user preference through clustering according to their historical check-ins, and make category prediction using cluster-specific HMM, the prediction accuracy increases with the increase of k. The accuracy is the highest when $k = 7$, e.g., 44.35% for mixed HMM. This shows the effectiveness of user preference modeling and confirms that user preference indeed affects his/her check-in behaviors. The accuracy decreases when we further increase k to 9. In the following location prediction experiment, we use mixed HMM with 7 clusters.

5.3 Location Prediction In this experiment, we evaluate the location prediction accuracy under different ranking schemes, given the predicted category distribution from HMM (denoted as "*w/ categ.*"). For comparison we report the accuracy under different ranking schemes when category information is not given (denoted as "*w/o categ.*"). In this case, all locations falling into the square region are considered as candidates. We also test the location ranking schemes with user clustering versus without user clustering. Table 2 shows the location prediction accuracy when we return top-1, top-2 and top-3 locations as prediction. The accuracy is measured through comparing the predicted location with the true location in the check-in data. We treat a prediction as correct, as long as the true location is among the top-k ($k = 1, 2, 3$) returned locations. In this experiment, we set the region range $d = 400$ meters.

From Table 2 we observe that: (1) the location prediction accuracy when given the category is consistently higher than that without the category. The accuracy improvement is up to 13.21%. This shows that, location prediction is more accurate by modeling user movement pattern and check-in dependency at the category level. If without category, the ranking schemes *Check-in count* and *User count* are simply based on global popularity without modeling user behavior, thus they have a low accuracy of 26.57% and 27.07%. The average number of location candidates we consider for prediction is 9.64 when given the predicted category distribution, and is 52.55 when the category distribution is not given. Therefore, the candidate space is reduced by 5.45 times with our approach; (2) among the four location ranking schemes, *Max check-in count by user* consistently achieves the highest accuracy when combined with user clustering. It shows that if a location is repeatedly checked in many times by the same user, it could be an interesting location to be visited by users; (3) the accuracy with user clustering is consistently higher than that without user clustering. This demonstrates that considering user preference for prediction is more accurate, as users tend to visit the same location if they have similar preferences; and finally (4) the prediction accuracy increases as we return more locations, i.e., from top-1 to top-3. This is very intuitive. The accuracy difference between *w/ categ.* and *w/o categ.* gradually decreases

Table 2: Location Prediction Accuracy under Different Ranking Schemes

	Ranking Scheme	Top-1		Top-2		Top-3	
		w/o Categ.	w/ Categ.	w/o Categ.	w/ Categ.	w/o Categ.	w/ Categ.
without clustering	Check-in count	26.57	39.54	41.48	53.04	50.81	60.71
	User count	27.07	40.00	41.20	51.25	50.20	56.72
	User count × Check-in count	26.03	39.24	40.38	50.56	49.58	55.77
	Max check-in count by user	28.46	39.80	42.51	51.22	51.68	56.40
with clustering	Check-in count	33.59	43.99	48.68	57.55	57.79	61.87
	User count	36.05	45.23	50.29	58.04	58.79	61.99
	User count × Check-in count	32.33	43.22	47.11	56.93	56.30	61.37
	Max check-in count by user	**36.93**	**45.63**	**52.25**	**58.80**	**61.54**	**62.81**

when more locations are returned as prediction.

It is noteworthy that the accuracy of top-1 location prediction with category (e.g., 45.63% in Table 2) may be even higher than the accuracy of category prediction (44.35% in Figure 5). As described in Section 4, our location prediction is done based on the predicted category distribution. Within a bounded square region, we rank candidate locations belonging to the most likely category. But if there is no location belonging to the first category, we will consider locations belonging to the second most likely category and so on. Thus it is possible that in some cases the top-1 predicted category is wrong, but the top-1 location can still be correct, leading to a higher location prediction accuracy.

Comparison with State-of-the-art Methods. Cho et al. [5] developed a model, called PMM, of human mobility which can predict the locations and dynamics of future human movement. We compare our location prediction method (max check-in count by user) with PMM and report the prediction accuracy in Table 3. Our method outperforms PMM by 28.50% – 31.89%. A possible explanation is that PMM can indeed predict the mean geographic point of a specific state ("home" or "work") of a user effectively. However, there are many locations that are very close to the mean point in our dataset, causing the probabilities of these locations to be very close. Thus it can be very difficult to rank the right location on the top. In addition, as pointed out in their paper [5], PMM does not perform very well on the weekends when the periodic movement is less obvious. On the contrary, our method can model the dependency between the user's activities well and consider more features of locations, such as check-in count and user count, besides the geographic features. Therefore, it achieves a much higher prediction accuracy.

Table 3: Location Predication Accuracy Comparison

	Top-1	Top-2	Top-3
PMM [5] with two states	16.82	26.91	34.31
Max check-in by user	**45.63**	**58.80**	**62.81**

Order-n Markov model can also be applied to the check-in sequences directly on the location level, just like its application on WiFi mobility data [16]. However, we observe order-2 Markov model achieves a location prediction accuracy of 2.85% only in our dataset, due to the large number of distinct locations in LBSN.

6 Related work

Due to the increasing availability of GPS-enabled devices, research on mining GPS data [6, 9, 10, 19] has attracted a lot of attention during past several years. [22] by Zheng et al. aims to mine interesting locations and travel sequences from GPS trajectories. They model multiple individuals' location histories with a tree-based hierarchical graph and propose a HITS based inference model to infer the interest of a location. Some works study location recommendation based on GPS data. [20] aims to discover interesting locations and possible activities for recommendations. [17] proposes an item-based collaborative filtering algorithm to recommend shops based on users' past location history. Some studies [8, 3] consider user preference or friendship to enhance the recommendation or prediction accuracy. [19] studies mining individual life pattern from GPS data, which can be used for location prediction.

The soaring popularity of location-based services, e.g., Foursquare, Gowalla, etc., brings a new research area to us and only few studies have been done on LBSNs till now. LBSN data distinguishes from GPS data mainly in two aspects: (1) check-in sparseness, and (2) semantic tags with which we can model the underlying user movement pattern and infer the user preference. Scellato et al. [14] study the link prediction problem in LBSNs by exploiting place features, which are defined based on the properties of the places visited by users. But they do not consider the semantic tags of locations. There are some studies [12, 18] sharing a similar angle with us by exploiting the semantic tags in check-in data. [12] studies to cluster user and geographic region based on the check-in distribution of category within two cities and analyze the coherence and difference between them. The recent work by Ye et al. [18] develops a semantic annotation technique for LBSNs to automatically annotate places with category tags. [5] by Cho et al. studies user movement pattern

by exploiting the strong periodic behavior in LBSNs and develops a model of human mobility that combines periodic short range movements with travel due to the social network structure.

Research on hidden Markov model has developed rapidly during the last two decades. In the pioneering work [13], Rabiner provided a comprehensive view of employing HMM and demonstrated its effectiveness, especially in speech recognition. Recently, Altman [1] and Maruotti [11] have studied the extension from basic HMM to mixed HMM under a dynamic longitudinal data setting. The mixed HMM performs well under some standard data distributions, such as Poisson and Gaussian distributions; however, these models require choosing specific canonical link functions according to specific distribution requirements, which rarely happen in the LBSN domain. In our scenario, we consider temporal and spatial covariates in HMM to fully describe check-in behaviors with a more general form in expressing state-dependent observation probability.

7 Conclusion

In this paper we study location prediction in LBSNs through modeling user movement pattern and user preference at the category level. A mixed HMM is learnt through MCMC Bayes Estimation to predict the category of a user's next activity, and then predict a location given the category. To the best of our knowledge, our work is the first to model the category level for location prediction. Our approach can effectively reduce the location candidate number by 5.45 times, while improving the location prediction accuracy by 13.21%, according to our experiments on Gowalla data.

Acknowledgments

This work is supported by the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) Project No. CUHK 411211 and 411310.

References

[1] R. M. Altman, Mixed hidden markov models: An extension of the hidden markov model to the longitudinal data setting, *Journal of the American Statistical Association*, vol. 102, pp. 201–210, 2007.

[2] Y. F. Atchadé, An adaptive version for the metropolis adjusted langevin algorithm with a truncated drift, *Methodology and Computing in Applied Probability*, vol. 8, no. 2, pp. 235–254, 2006.

[3] L. Backstrom, E. Sun, and C. Marlow, Find me if you can: improving geographical prediction with social and spatial proximity, in *WWW*, 2010, pp. 61–70.

[4] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains, *The Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970.

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

[6] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi, Trajectory pattern mining, in *KDD*, 2007, pp. 330–339.

[7] W. H. Greene, *Econometric Analysis*, Prentice Hall, London, 5th edition, 2003.

[8] T. Horozov, N. Narasimhan, and V. Vasudevan, Using location for personalized poi recommendations in mobile environments, in *SAINT*, 2006, pp. 124–129.

[9] J. Krumm and E. Horvitz, Predestination: Inferring destinations from partial trajectories, in *Ubicomp*, 2006, pp. 243–260.

[10] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, Mining periodic behaviors for moving objects, in *KDD*, 2010.

[11] A. Maruotti, Mixed hidden markov models for longitudinal data: An overview, *International Statistical Review*, vol. 79, no. 3, pp. 427–454, 2011.

[12] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil, Exploiting semantic annotations for clustering geographic areas and users in location-based social networks, in *Proc. 3rd Workshop on Social Mobile Web*, 2011.

[13] L. R. Rabiner, A tutorial on hidden markov models and selected applications in speech recognition, *Proc. of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[14] S. Scellato, A. Noulas, and C. Mascolo, Exploiting place features in link prediction on location-based social networks, in *KDD*, 2011, pp. 1046–1054.

[15] G. Schwarz, Estimating the dimension of a model, *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[16] L. Song, D. Kotz, R. Jain, and X. He, Evaluating location predictors with extensive wi-fi mobility data, in *INFOCOM*, 2004, pp. 1414–1424.

[17] Y. Takeuchi and M. Sugimoto, CityVoyager: An outdoor recommendation system based on user location history, in *UIC*, 2006, pp. 625–636.

[18] M. Ye, D. Shou, W. C. Lee, P. Yin, and K. Janowicz, On the semantic annotation of places in location-based social networks, in *KDD*, 2011, pp. 520–528.

[19] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie, Mining individual life pattern based on location history, in *MDM*, 2009, pp. 1–10.

[20] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, Collaborative location and activity recommendations with gps history data, in *WWW*, 2010, pp. 1029–1038.

[21] Y. Zheng, Location-based social networks: Users, in *Computing with Spatial Trajectories*, Yu Zheng and Xiaofang Zhou eds. Springer 2011.

[22] Y. Zheng, L. Zhang, X. Xie, and W. Y. Ma, Mining interesting locations and travel sequences from gps trajectories, in *WWW*, 2009, pp. 791–800.

[23] W. Zucchini and I. L. MacDonald, *Hidden Markov models for time series: an introduction using R*. Chapman and Hall, London, 2009.