Mining Semantic Mobility Patterns from Sparse Semantic Trajectories

ABSTRACT

1. INTRODUCTION

2. THE BACKGROUND MODEL.

2.1 Model Description.

The background model is an extension of the Gaussian mixture model such that each component generates a Gaussian as well as a number of keywords from the vacabulory. One may concern that a Gaussian may not well model the spatial shape for a latent topic. However, with a large enough number of components, the spatial distribution of a non-Gaussian topic is simply split into several states.

We consider the input $\mathcal{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_N\}$, which are N places and each place $\mathbf{o}_n$ is described by a location $\mathbf{x}_n$ and a word vector $\mathbf{y}_n$. We use π to represent the prior distributions over the latent components. In addition, we assume each component generates the textual observations from a multinomial parameterized by $\theta_{kv}(1 \leq k \leq K, 1 \leq v \leq V)$ such that θ_{kv} is the probability of generating word v from component k; the geographical observation is from a Gaussian, parameterized by μ_k and Σ_k.

2.2 Parameter Inference.

We use EM to obtain the parameters for the background model. For each observation $\mathbf{o}_n$, we use $\mathbf{z}_n$ to represent its membership over the K latent components. Note that $\mathbf{z}_n$ has the 1-of-K representation. Namely, if the latent component is k, then $z_{nk} = 1$ and all the other positions in the vector $\mathbf{z}_n$ are 0.

2.2.1 *E-Step.*

For ease of representation, we define

$$\gamma(z_{nk}) = p(z_{nk} = 1 | \mathbf{o}_n, \Theta^0).$$

This can be easily computed using the Bayes rule, namely

$$\gamma(z_{nk}) = \frac{\pi_k p(\mathbf{o}_n|\phi_k)}{\sum_{k=1}^{K} \pi_k p(\mathbf{o}_n|\phi_k)} = \frac{\pi_k p(\mathbf{x}_n, \mathbf{y}_n|\phi_k)}{\sum_{k=1}^{K} \pi_k p(\mathbf{x}_n, \mathbf{y}_n|\phi_k)}$$

2.2.2 *M-Step.*

We have

$$\ln p(\mathbf{o}_n, \mathbf{z}_n) = \sum_{k=1}^{K} z_{nk} [\ln \pi_k + \ln p(\mathbf{o}_n|\phi_k)].$$

The Q-function is thus

$$\begin{aligned} Q &= \sum_{n=1}^{N} \sum_{\mathbf{z}_n} \ln p(\mathbf{o}_n, \mathbf{z}_n|\Theta) p(\mathbf{z}_n|\mathbf{o}_n, \Theta^0) \\ &= \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) [\ln \pi_k + \ln p(\mathbf{o}_n|\phi_k)] \end{aligned}$$

The parameters are estimated as follows:

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} \gamma(z_{nk})$$

$$\theta_{kv} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) y_{nv}}{\sum_{n=1}^{N} \sum_{v=1}^{V} \gamma(z_{nk}) y_{nv}}$$

$$\mu_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

$$\Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

3. HIDDEN MARKOVE MODEL

Consider a sequence $\{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \ldots, (\mathbf{x}_N, \mathbf{y}_N)\}$. We use $\mathbf{o}_n = (\mathbf{x}_n, \mathbf{y}_n)$ to denote the observations at timestamp n, where $\mathbf{x}_n$ is a two-dimensional vector that represents the latitude and longitude; and $\mathbf{y}_n$ is a V-dimensional vector that represents the keywords using the bag-of-words model (V is the size of vocabulory). Each observation $(\mathbf{x}_n, \mathbf{y}_n)(1 \leq n \leq N)$ corresponds to a latent state $\mathbf{z}_n \in \mathbf{Z} = \{1, 2, \ldots, K\}$. Each state $\mathbf{z}_n$ generates observations along both the textual and geographical dimensions.

Transition. There is a prior distribution π for the initial latent state $\mathbf{z}_1$, *i.e.*, $\pi_k = p(\mathbf{z}_1 = k)$. In addition, the transitions between the latent states are characterized by a $K \times K$ matrix $\mathbf{A}$. Suppose the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...$10.00.

latent sate at timestamp $n-1$ is j, the probability for the latent state at timestamp n to be k is $\mathbf{A}_{jk} = p(\mathbf{z}_n = k|\mathbf{z}_{n-1} = j)$. Note that the next state only depends on the current state and is time-homogeneous (the transition matrix $\mathbf{A}$ does not change with time).

Observation.

- For the textual part, we assume the words $\mathbf{y}_n$ are generated from a multinomial distribution, namely

$$p(\mathbf{y}_n|\mathbf{z}_n = k) = \prod_{v=1}^{V} \theta_{kv}^{y_{nv}}.$$

where θ_{kv} is the probability for state k to generate word v.

- For the geographical part, we assume the location $\mathbf{x}_n$ is generated from a Gaussian mixture that has M components, namely

$$p(\mathbf{x}_n|\mathbf{z}_n = k) = \sum_{m=1}^{M} c_{km} b_{km}(\mathbf{x}_n),$$

where $b_{km}(\mathbf{x})$ is the normal distribution with parameters μ_{km} and Σ_{km}, *i.e.*, $b_{km}(x) = \mathcal{N}(\mu_{km}, \Sigma_{km})$; and c_{km} is the probability for choosing Guassian component m in state k.

Also, we assume the textual observations and the geographical observations are generated independently, hence

$$p(\mathbf{x}_n, \mathbf{y}_n|\mathbf{z}_n = k) = p(\mathbf{x}_n|\mathbf{z}_n = k) \cdot p(\mathbf{y}_n|\mathbf{z}_n = k).$$

4. THE MIXTURE MODEL

4.1 Model Description.

In the mixture model, we have a HMM model parameterized by Θ_H and a background model parameterized by Θ_B. The prior probability of choosing the HMM model is λ $(0 < \lambda < 1)$. Given a length-2 sequence $\mathbf{X}$, we associate it with a binary latent variable $\mathbf{h}$, $\mathbf{h} = 1$ indicates $\mathbf{X}$ is generated from HMM, and $\mathbf{h} = 0$ means $\mathbf{X}$ is generated from the background model. We have multiple sequences $\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^R$.

4.2 Parameter Inference.

We can again use the EM algorithm to infer the parameters.

4.2.1 E-Step.

In the E-step, the latent variables $\mathbf{h}^r, \mathbf{z}^r, \mathbf{g}^r$ are computed, assuming we are given a set of old parameters Θ^0. For each sequence $\mathbf{X}^r$, we define the following quatities.

Latent variable $\mathbf{h}^r$. The probability distribution of the underlying model (HMM or Background) for the r-th data point $\mathbf{X}^r$:

$$\kappa(\mathbf{h}^r) = p(\mathbf{h}^r|\mathbf{X}^r, \Theta^0).$$

$\kappa(\mathbf{h}_1^r)$ is the probability that $\mathbf{X}^r$ is generated from HMM. The quantity can be computed as follows:

$$p(\mathbf{h}^r = 1|\mathbf{X}^r, \Theta^0) = \frac{\lambda^0 p(\mathbf{X}^r|\Theta_H^0)}{\lambda^0 p(\mathbf{X}^r|\Theta_H^0) + (1-\lambda^0) p(\mathbf{X}^r|\Theta_B^0)}$$

Latent variable $\mathbf{z}_n^r$. The probability distribution of the latent state at position n for $\mathbf{X}^r$:

$$\gamma(\mathbf{z}_n^r) = p(\mathbf{z}_n^r|\mathbf{X}^r, \Theta^0).$$

$\gamma(z_{nk}^r)$ is the probability that the latent state at position n is k for the r-th sequence. The computation of $\gamma(\mathbf{z}_n^r)$ can be achieved using the forward-backward algorithm.

$$\gamma(\mathbf{z}_n^r) = \frac{\alpha(\mathbf{z}_n^r)\beta(\mathbf{z}_n^r)}{p(\mathbf{X}^r)},$$

where

$$\alpha(\mathbf{z}_n^r) = p(\mathbf{o}_1^r, \mathbf{o}_2^r, \ldots, \mathbf{o}_n^r, \mathbf{z}_n^r)$$
$$\beta(\mathbf{z}_n^r) = p(\mathbf{o}_{n+1}^r, \ldots, \mathbf{o}_n^r|\mathbf{z}_n^r)$$

Both quantities can be evaluated iteratively, namely

$$\alpha(\mathbf{z}_n^r) = p(\mathbf{o}_n^r|\mathbf{z}_n^r) \sum_{\mathbf{z}_{n-1}^r} \alpha(\mathbf{z}_{n-1}^r) p(\mathbf{z}_n^r|\mathbf{z}_{n-1}^r)$$
$$\beta(\mathbf{z}_n^r) = \sum_{\mathbf{z}_{n+1}^r} \beta(\mathbf{z}_{n+1}^r) p(\mathbf{o}_{n+1}^r|\mathbf{z}_{n+1}^r) p(\mathbf{z}_{n+1}^r|\mathbf{z}_n^r).$$

The initial quantities are given as follows:

$$\alpha(\mathbf{z}_1^r = k) = p(\mathbf{o}_1^r, \mathbf{z}_1^r = k) = \pi_k p(\mathbf{o}_1^r|\phi_k).$$

and $\beta(\mathbf{z}_N^r) = 1$ for all settings of $\mathbf{z}_N^r$. Moreover, we have $p(\mathbf{X}^r)$ computed as $p(\mathbf{X}^r) = \sum_{\mathbf{z}_N^r} \alpha(\mathbf{z}_N^r)$.

Latent variable $\mathbf{g}_{nk}^r$. The probability distribution of the Gaussian component for the latent variable at position n and state k:

$$\rho(\mathbf{g}_{nk}^r) = p(\mathbf{g}_{nk}^r|\mathbf{X}^r, \Theta^0).$$

$\rho(g_{nkm}^r)$ is the probability that the latent state at position n is k, and the Gaussian component is m. Once $\gamma(z_{nk}^r)$ is computed, the quantity $\rho(g_{nkm}^r)$ can be easily computed:

$$\rho(g_{nkm}^r) = \gamma(z_{nk}^r) \frac{c_{km} b_{km}(\mathbf{x}_n^r)}{\sum_{m=1}^{M} c_{km} b_{km}(\mathbf{x}_n^r)}.$$

Latent variable ξ^r. The probability of transiting from a state at position $n-1$ to a state at position n:

$$\xi(\mathbf{z}_{n-1}^r, \mathbf{z}_n^r) = p(\mathbf{z}_{n-1}^r, \mathbf{z}_n^r|\mathbf{X}^r, \Theta^0).$$

$\xi(z_{n-1,j}^r, z_{nk}^r)$ is the probability that the latent state is j at position $n-1$, and state k at position n. ξ can be computed as

$$\xi(\mathbf{z}_{n-1}^r, \mathbf{z}_n^r) = \frac{\alpha(\mathbf{z}_{n-1}^r) p(\mathbf{o}_n^r|\mathbf{z}_n^r) p(\mathbf{z}_n^r|\mathbf{z}_{n-1}^r) \beta(\mathbf{z}_n^r)}{p(\mathbf{X}^r)}$$

4.2.2 M-Step.

The Q-function is:

$$
\begin{aligned}
&L(\Theta)\\
=&\sum_{r=1}^{R}\sum_{\mathbf{h}^r}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln p(\mathbf{X}^r,\mathbf{h}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta)\\
=&\sum_{r=1}^{R}\sum_{\mathbf{h}^r}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\mathbf{h}^r\{\ln[\lambda p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_H)]\\
&+(1-\mathbf{h}^r)\ln[(1-\lambda)p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_B)]\}\\
=&\sum_{r=1}^{R}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r=1,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln\lambda\\
&+\sum_{r=1}^{R}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r=1,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_H)\\
&+\sum_{r=1}^{R}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r=0,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln(1-\lambda)\\
&+\sum_{r=1}^{R}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r=0,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln p(\mathbf{X}^r|\Theta_B)\\
=&\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\ln\lambda+\sum_{r=1}^{R}\kappa(\mathbf{h}_0^r)\ln(1-\lambda)\\
&+\sum_{r=1}^{R}\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{h}^r=1,\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\ln p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_H)\\
&+\sum_{r=1}^{R}\kappa(\mathbf{h}_0^r)\ln p(\mathbf{X}^r|\Theta_B)
\end{aligned}
$$

Note that $p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_B)=p(\mathbf{X}^r|\Theta_B)$ because for the background model, there is no latente variables. For the last term in the above equation, we can actually ignore it because Θ_B is already known and there are no variables for estimation.

Estimating λ: Using the first line in the Q-function, we can get the estimation of λ as:

$$\lambda=\frac{\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)}{R}.$$

Estimating $\pi,\mathbf{A},\theta,c,\mu,\Sigma$: We proceed to discuss the estimations of the other parameters in the second line of the Q-function. To begin with, note that z_{nk} is a binary variable (1 indicates the latent state at n is k, and 0 indicates the latent sate is not k), and the same holds for $\mathbf{z}_{n-1}\cdot\mathbf{z}_n$. Hence, we have

$$\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\cdot z_{nk}^r=p(z_{nk}^r=1|\mathbf{X}^r,\Theta^0)=\gamma(z_{nk}^r),$$

$$\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\cdot z_{n-1,j}^r\cdot z_{nk}^r=\xi(z_{n-1,j}^r,z_{nk}^r).$$

Consider the second line in the Q-function, we have

$$
\begin{aligned}
&f(\Theta_H)\\
=&\sum_{r=1}^{R} p(\mathbf{h}^r=1|\mathbf{X}^r,\Theta^0)\\
&\cdot\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\mathbf{h}^r=1,\Theta^0)\ln p(\mathbf{X}^r,\mathbf{z}^r,\mathbf{g}^r|\Theta_H)\\
=&\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\ln p(\mathbf{z}_1^r|\pi)\\
&+\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\sum_{n=2}^{N}\ln p(\mathbf{z}_n^r|\mathbf{z}_{n-1}^r,\mathbf{A})\\
&+\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}^r\ln p(\mathbf{x}_n,\mathbf{g}_n^r|\phi_k)\\
&+\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r}\sum_{\mathbf{g}^r} p(\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)\sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}^r\ln p(\mathbf{y}_n|\phi_k)
\end{aligned}
$$

Note that, the first two terms are derived because the latent variable $\mathbf{g}^r$ is marginalized. Similarly, for the fourth term, since the quantity is irrelevant to the latent variable $\mathbf{g}^r$, marginalization of $p(\mathbf{z}^r,\mathbf{g}^r|\mathbf{X}^r,\Theta^0)$ over $\mathbf{g}^r$ can be done. Those four terms can be optimized separately.

Optimizing $\pi,\mathbf{A}$: Consider the first term, since the latent variable $\mathbf{z}_1^r$ can take a value from $\{1,2,\ldots,K\}$, we have

$$p(\mathbf{z}_1^r|\pi)=\prod_{k=1}^{K}\pi_k^{z_{1k}^r}.$$

Hence, the first term in the above equation becomes

$$
\begin{aligned}
f&=\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\sum_{k=1}^{K} z_{1k}^r\ln\pi_k\\
&=\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{k=1}^{K}\gamma(z_{1k}^r)\ln\pi_k.
\end{aligned}
$$

Using the Lagrange multiplier, we obtain the following optimal parameters:

$$\pi_k=\frac{\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\gamma(z_{1k}^r)}{\sum_{r=1}^{R}\sum_{j=1}^{K}\kappa(\mathbf{h}_1^r)\gamma(z_{1j}^r)}=\frac{\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\gamma(z_{1k}^r)}{\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)}$$

For the second term, we have

$$
\begin{aligned}
f&=\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{\mathbf{z}^r} p(\mathbf{z}^r|\mathbf{X}^r,\Theta^0)\sum_{n=2}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K} z_{n-1,j}^r z_{nk}^r\ln A_{jk}\\
&=\sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{n=2}^{N}\sum_{k=1}^{K}\sum_{j=1}^{K}\xi(z_{n-1,j}^r,z_{nk}^r)\ln A_{jk}
\end{aligned}
$$

Similarly, we can use the Lagrange multiplier to obtain the optimal parameter as

$$A_{jk} = \frac{\sum_{r=1}^{R}\sum_{n=2}^{N}\kappa(\mathbf{h}_1^r)\xi(z_{n-1,j}^r, z_{nk}^r)}{\sum_{r=1}^{R}\sum_{n=2}^{N}\sum_{i=1}^{K}\kappa(\mathbf{h}_1^r)\xi(z_{n-1,j}^r, z_{ni}^r)}$$

Optimizing θ:

Now we consider the fourth term:

$$f = \sum_{r=1}^{R}\kappa(\mathbf{h}_1^r)\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(z_{nk}^r)\ln p(\mathbf{y}_n^r|\phi_k)$$

Recall that, we model the generation of the words as the observations from multinomial distribution. For state k, we have

$$p(\mathbf{y}_n^r|\phi_k) = \prod_{v=1}^{V}\theta_{kv}^{y_{nv}^r}$$

then the objective function becomes

$$f = \sum_{r=1}^{R}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{v=1}^{V}\kappa(\mathbf{h}_1^r)\gamma(z_{nk}^r)y_{nv}^r\ln\theta_{kv}$$

Again, using the Lagrange multiplier, we obtain the parameter estimation as

$$\theta_{kv} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)y_{nv}^r\gamma(z_{nk}^r)}{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\gamma(z_{nk}^r)}.$$

Optimizing c, μ, Σ:

We proceed to consider the geographical part. Recall that we assume the geographical location $\mathbf{x}_n^r$ is generated from a Gaussian mixture ϕ_k that have M components, namely

$$p(\mathbf{x}_n^r, \mathbf{g}_n^r|\phi_k) = \sum_{m=1}^{M} g_{nm}^r c_{km} b_{km}(\mathbf{x}_n^r)$$

the objective function is now

$$f = \sum_{r=1}^{R}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{M}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)\ln(c_{km}b_{km}(\mathbf{x}_n^r))$$

The optimal parameters are computed as

$$c_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)}{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\gamma(z_{nk}^r)}$$

$$\mu_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)\mathbf{x}_n^r}{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)}$$

$$\Sigma_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)(\mathbf{x}_n^r-\mu_{km})(\mathbf{x}_n^r-\mu_{km})^T}{\sum_{r=1}^{R}\sum_{n=1}^{N}\kappa(\mathbf{h}_1^r)\rho(g_{nkm}^r)}$$

5. EXPERIMENTS

In this section, we evaluate the empirical performance of the proposed method. All the algorithms were implemented in JAVA and the experiments were conducted on a computer with Intel Core i7 2.4Ghz CPU and 8GB memory.

5.1 Experimental Setup

Data Sets. The real data set, referred to as Tweet, is collected from Twitter. This data set consists of the geo-tagged tweets of 14,909 users living in New York. There are totally 68,564 distinct places in the data set, distributed in a $0.5^\circ \times 0.5^\circ$ space. The average length of each trajectory, *i.e.*, number of check-ins, is 82.

5.2 Experimental Results

6. RELATED WORK

Generally, existing techniques related to our problem can be categorized as follows.

Mobility pattern mining.

Giannotti *et al.* [**?**] define the *T-pattern* in a collection of GPS trajectories. A T-pattern is a Region-of-Interest (RoI) sequence with temporal annotations, where each RoI as a rectangle whose density is larger than a threshold δ. However, their method still relies on rigid space partitioning. In addition, the threshold δ is hard to prespecify for our problem: a small δ will lead to very coarse regions while a large one may eliminate fine-grained patterns.

Another important line in trajectory data mining is to mine a set of objects that are frequently co-located. Efforts along this line include mining *flock* [**?**], *convoy* [**?**], *swarm* [**?**], and *gathering* [3] patterns. All these patterns differ from our work in two aspects: (1) they only model the spatio-temporal information without considering place semantics; and (2) they require the trajectories are aligned by the absolute timestamps to discover co-located objects, while we focus on the relative time interval in a trajectory.

Location prediction.

Monreale *et al.* [**?**] proposed a pattern-based location predictor.

Geo-textual data mining. Yin *et al.*

Semantic trajectory mining.

There are a few studies on mining sequential patterns in semantic trajectories. Alvares *et al.* [1] first identify the *stops* in GPS trajectories, then match these stops to semantic places using a background map. By viewing each place as an item, they extract the frequent place sequences as sequential patterns. Unfortunately, due to spatial continuity, such place-level sequential patterns can appear only when the support threshold is very low. Ying *et al.* [2] mine sequential patterns in semantic trajectories for location prediction. They define a sequential pattern as a sequence of semantic labels (*e.g.*, school $\rightarrow$ park). Such a definition ignores spatial and temporal information. In contrast, our fine-grained patterns consider the spatial, temporal and semantic dimensions simultaneously.

7. CONCLUSION

8. REFERENCES

[1] L. O. Alvares, V. Bogorny, B. Kuijpers, B. Moelans, J. A. Fern, E. D. Macedo, and A. T. Palma. Towards semantic trajectory knowledge discovery. *Data Mining and Knowledge Discovery*, 2007.

[2] J. J.-C. Ying, W.-C. Lee, T.-C. Weng, and V. S. Tseng. Semantic trajectory mining for location prediction. In *GIS*, pages 34–43, 2011.

Table 1: The word distribution of of the latent states.

State 0		State 5		State 8	
shop	0.1242106	travel	0.15717	hotel	0.0991
service	0.080308	airport	0.1264	residence	0.0577
apparel	0.0782	jfk	0.1052	hotel	0.03227
women	0.0632	terminal	0.0674	building	0.031497
store	0.051126	international	0.0545	inn	0.02978
food	0.03212	kennedy	0.05408	center	0.02891
mall	0.03123	station	0.0317	plaza	0.021429
shopping	0.021112	train	0.0221	home	0.02101

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