Modeling Check-in Preferences with Multidimensional Knowledge: A Minimax Entropy Approach

Jingjing Wang, Min Li, Jiawei Han and Xiaolong Wang
Department of Computer Science
University of Illinois at Urbana - Champaign
Urbana, IL 61801
{jwang112, minli3, hanj, xwang95}@illinois.edu

ABSTRACT

We propose a single unified minimax entropy approach for user preference modeling with multidimensional knowledge. Our approach provides a discriminative learning protocol which is able to simultaneously a) leverage explicit human knowledge, which are encoded as explicit features, and b) model the more ambiguous hidden intent, which are encoded as latent features. A latent feature can be carved by any parametric form, which allows it to accommodate arbitrary underlying assumptions. We present our approach in the scenario of check-in preference learning and demonstrate it is capable of modeling user preference in an optimized manner.

Check-in preference is a fundamental component of Point-of-Interest (POI) prediction and recommendation. A user's check-in can be affected at multiple dimensions, such as the particular time, popularity of the place, his/her category and geographic preference, etc. With the geographic preferences modeled as latent features and the rest as explicit features, our approach provides an in-depth understanding of users' time-varying preferences over different POIs, as well as a reasonable representation of the hidden geographic clusters in a joint manner. Experimental results based on the task of POI prediction/recommendation with two real-world check-in datasets demonstrate that our approach can accurately model the check-in preferences and significantly outperforms the state-of-art models.

1. INTRODUCTION

As the check-in feature becomes increasingly popular in major social network services (SNS) such as Foursquare, Facebook, etc., numerous research efforts have been aimed at mining users' check-in behaviors. In this paper, we consider the problem of modeling users' time-aware check-in preferences. Formally, our goal is to learn a time-aware distribution over POIs for each user: $p(l|u,t)$, where u denotes a user, t denotes a time point, l denotes a POI and $p(l|u,t)$ denotes the conditional probability that l is checked in given that the user is u and the time point is t. This distribution allows us to predict what are the top places a user would like to check in at a given time, which can be of great interest to both business owners and advertisement providers.

To approach this problem, we propose a discriminative learning framework which allows a subset of the features to be latent under the *minimax entropy* principle. In contrast to the standard discriminative learning protocol (e.g. SVM, logistic regression) where features are readily available before training, we introduce the concept of *latent features*. The value of a latent feature is not known before training, but is specified by a parametric form with unknown parameters. The parametric form can capture arbitrary underlying assumptions to describe the feature. For example, if a set of latent features are cluster indicators, the parameters can specify the underlying clustering structure. During the training process, the latent parameters are jointly learned with the classification task. We illustrate in the following paragraphs why this is the desired strategy.

Why maximum entropy?

A naive way to estimate $p(l|u,t)$ is simple counting. For each user u at time t, we can get the histogram of POIs (l's) and view it as the objective distribution. While this distribution perfectly fits the seen data, it is not generalizable, *i.e.*, it can never predict unvisited POIs for users and will fail to generate outputs for unseen time points.

We prefer a model which *explains the seen data well and meanwhile has good generalizability.* To this end, instead of exactly matching $p(l|u,t)$ to the empirical distribution, it is natural to extract features from the `user-time-POI` $\langle utl \rangle$ tuples and impose the constraints that $p(l|u,t)$ match the empirical statistics in the feature space. Among these qualified distributions, we select the distribution with the maximum entropy as the optimal distribution, as it assumes least bias on the model beyond the constraints we specify [11].

Why minimax entropy? (Why latent features? Why should they be jointly learned?)

User preferences over POIs can be affected by explicit features such as the category of a POI, the day of a week, etc., meanwhile it can also be affected by the more ambiguous features such as the geographic region, which is less clear how to encode as features effectively. For example, it is not straightforward to draw the boundary for "downtown Manhattan" or to classify if a POI belongs to it. Therefore, we introduce latent features to model this kind of ambiguity. Taking the geographic feature as an example, we can assume there exist geographic clusters, each of which is specified by latent parameters: a center (coordinates of latitude and longitude) and a radius (a positive real number). Given a POI, we define a weight vector over different clusters as a

WSDM'16, February 22–25, 2016, San Francisco, CA, USA.

ISBN 978-1-4503-3716-8/16/02...$15.00

DOI: http://dx.doi.org/10.1145/2835776.2835839

latent feature vector, where the weight on each cluster is determined by a parametric function which takes the latitude-longitude of the POI as input. With both explicit and latent features, we propose a minimax entropy approach to jointly learn the latent parameters together with the check-in preferences ($p(l|u,t)$). The joint learning approach is motivated by the fact that *the clustering structure is not only determined by geographic proximity, but also affected by how well it explains user check-ins.* For example, even if two POIs are very close to each other geographically, if they have never been visited by the same user, it may not be appropriate to put them into the same cluster. In sum, the jointly learned geographic clusters are specially tailored to boost the learning task's performance rather than just provide a standalone clustering results.

Contributions

- We propose a single unified minimax entropy approach which elegantly leverages explicit features and latent features for user preference modeling. It boosts the flexibility and expressiveness of the standard discriminative learning models significantly.
- Flexible as the way latent features are defined by parametric forms, the parameters governing the latent features are recovered jointly with the learning task in a principled way, which serve to explain the inherent structure of the learning task.
- We demonstrate the effectiveness of our approach in the context of check-in preference learning with its rich types of information. It opens up a promising direction for preference learning with multidimensional heterogeneous knowledge.

The rest of the paper is organized as follows. Section 2 first details the modeling of users, POIs and the way we specify the geographic clusters; and then formally defines the problem. We introduce our framework for check-in preference modeling in Section 3 and discuss related work in Section 4. We analyze the connections of our approach to other standard approaches in Section 5, report our experimental results on real-world data in Section 6 and conclude our study in Section 7.

2. PROBLEM FORMULATION

In this section, we define the POI profiles and user profiles with both explicit knowledge and the latent geographic clustering structure governed by latent parameters. Then we give the formal definition of check-in preference modeling. The notations used in this paper are summarized in Table 1.

Let U, T, L, C be the user set, time set, POI set and category set respectively. Our data contains the histories of user check-ins.

DEFINITION 1 (CHECK-IN). *A check-in is denoted by a user-time-POI tuple $\langle utl \rangle$, where $u \in U, t \in T$ and $l \in L$. Each POI l is associated with its category, latitude and longitude. The time is represented by the day of week and hour of day*[1].

[1]There are 7x24 unique values in T under this setting. However, one can index time with finer or coarser granularity as well. Overlapped time intervals are also allowed.

Table 1: Summary of Notations

Symbol	Description
u, U	a user, user set
t, T	a time index, time index set; $day(t)$ and $hour(t)$ denote the day index and hour index of t, respectively
l, L	a POI, POI set; $cat(l)$ denotes the category of l
C	category set
$\mathbf{o} = (o_1, o_2, ..., o_R)$	the centers of the geographic clusters
$\mathbf{r} = (r_1, r_2, ..., r_R)$	the radiuses of the geographic clusters
$\mathbf{c}^u = (c_1^u, c_2^u, ..., c_C^u)$	u's category preference
$\mathbf{g}^u = (g_1^u, g_2^u, ..., g_R^u)$	u's geographic preference
$\mathbf{c}^l$	l's one-hot encoding of its category
$\mathbf{g}^l = (g_1^l, g_2^l, ..., g_R^l)$	l's weights on different regions
p^l	l's global popularity
$\mathbf{d}^l = (d_1^l, d_2^l, ..., d_7^l)$	l's daily popularity profile
$\mathbf{h}^l = (h_1^l, h_2^l, ..., d_{24}^l)$	l's hourly popularity profile
$\pi_{utl} = p(l\|u,t)$	the conditional probability of checking in at POI l given a user u and time t
$\tilde{\pi}_{ut} = \tilde{p}(u,t), \tilde{\pi}_{utl} = \tilde{p}(l\|u,t)$	the empirical distributions estimated from data
Π	the true check-in preference distribution

DEFINITION 2 (REGION). *A region is a geographic cluster defined by the latitude and longitude of the center $o = (o_{lat}, o_{lon})$ and a radius $r > 0$. The (o, r)'s are the latent parameters.*

DEFINITION 3 (POI PROFILE). *A POI l is represented by a profile*[2] $\rho(l) = [\mathbf{c}^l, \mathbf{g}^l(\mathbf{o}, \mathbf{r}), \mathbf{d}^l, \mathbf{h}^l, p^l]$.

- $\mathbf{c}^l$ (a one-hot encoding of l's category): $\mathbf{c}^l$ has $c_i^l = 1$ if the i-th category in C is the category of l and 0 otherwise.
- $\mathbf{g}^l$ (the geographic profile of l): The geographic profile of a POI is modeled by a weight vector over different regions. The weight is determined by the POI's distance to the center of a region and the radius of the region:

$$g_i^l = \exp(-\frac{dist(l, o_i)}{r_i}) \quad (1)$$

  where $dist(\cdot,\cdot)$ is the Euclidean distance[3].

  When $dist(l, o) = 0$, the weight reaches its maximum 1; as $dist(l, o)$ becomes larger, the weight decreases towards 0. The radius r controls the decreasing speed w.r.t $dist(l, o)$. A smaller r indicates a more concentrated cluster, *i.e.*, the weight decreases drastically as the distance increases. Note that the weight function does not necessarily have to be defined in this way. A function that can satisfy the desired properties suffices.
- p^l (global popularity of l): The global popularity of a POI is defined as the total number of check-ins at this POI.
- $\mathbf{d}^l, \mathbf{h}^l$ (the daily popularity profile and hourly popularity profile of l): POIs have time varying popularity as

[2]We use bold letters to denote column vectors. The comma between column vectors indicates a vertical stack of the vectors.

[3]Other distance measures apply as well.

well. For example, a nightclub has its rush hours at night but is either closed or rarely visited before sunset. We compute the time varying popularity based on the aggregate statistics from all users. d_i^l is the proportion of check-ins at l that happen on the ith day of a week and h_i^l is the proportion of check-ins at l that happen on the ith hour of a day.

DEFINITION 4 (USER PROFILE). *A user u is represented by a profile $\rho(u) = [\mathbf{c}^u, \mathbf{g}^u(\mathbf{o}, \mathbf{r})]$.*

- $\mathbf{c}^u$ (user u's preference over categories): We define user u's preference of category i (*i.e.*, c_i^u) to be the proportion of his/her check-ins that fall into category i.
- $\mathbf{g}^u$ (user u's preference over regions): In addition to the category preference, users are also characterized by their geographic preferences over different regions. We define user u's geographic preference of a region i (*i.e.*, g_i^u) to be the aggregate weights at region i of all his/her check-ins .

We are now able to formulate the check-in preferences modeling problem as follows.

PROBLEM 1 (CHECK-IN PREFERENCES MODELING). *Given a training set of user check-in tuples, where each tuple $\langle utl \rangle$ is associated with a user profile $\rho(u)$ and a POI profile $\rho(l)$, jointly learn the conditional probability of checking in at POI l given a user u and time t, denoted by $\pi_{utl} = p(l|u,t), \forall u,t,l$; and the geographic clustering structure governed by latent parameters $\mathbf{o}$ and $\mathbf{r}$.*

3. A MINIMAX ENTROPY APPROACH FOR MODELING CHECK-IN PREFERENCES

In this section, we first assume the latent parameters are given, *i.e.*, all the features are explicit, and present the maximum entropy (MaxEnt) model for learning the check-in preferences. Then we present the proposed minimax entropy model which estimates the latent parameters jointly with the preference learning.

3.1 A Maximum Entropy Model

The most aggressive way to model the check-in preferences is just to let π_{utl} equal the empirical distribution[4] $\tilde{\pi}_{utl} = \frac{\#\langle utl \rangle}{\sum_l \#\langle utl \rangle}$. However, this will overfit the data and is not generalizable. We want to construct a model which explains the seen data well, and meanwhile has good generalizability. To this end, we adopt the maximum entropy principle to specify $\{\pi_{utl}\}$, *i.e.*, we choose the most "uniform" distribution with carefully chosen constraints instantiated by features. These constraints should guarantee that our model accords with the data statistics we feel essential in modeling the check-in preferences.

[4]In this paper, we use $\#\langle utl \rangle$ to denote the number of appearances of the check-in tuple $\langle utl \rangle$ in the data, and $\#$ to denote the total number of check-ins. We use " $\tilde{}$ " to denote the empirical distribution. Later we will also see $\tilde{p}(u,t) = \tilde{\pi}_{ut} = \frac{\sum_l \#\langle utl \rangle}{\#}$

3.1.1 Features Based on Multidimensional Preferences

We consider the following factors to model check-in preferences: temporal preference, category preference, geographic preference and the popularity of the POI. Consider the following scenario: on a Friday evening, Alice just finished yet another week of hard work; she would like to have a great dinner at a seafood restaurant and then she figures a popular Boiling Crab branch is just nearby. Then it is very likely she checks in at this place. We design the following features to instantiate the constraints which will be used to specify our model $\{\pi_{utl}\}$.

- **Category Preference**. The extent to which a POI l matches a user u's category preference is estimated by $f_c(\langle utl \rangle) = \mathbf{c}^{u^T} \mathbf{c}^l$.
- **Geographic Preference**. The extent to which a POI l matches a user u's geographic preference is estimated by $f_g(\langle utl \rangle) = \mathbf{g}^{u^T} \mathbf{g}^l$.
- **Temporal Preference**. If we represent each time index t with two one-hot encodings: $\mathbf{d}^t$, $\mathbf{h}^t$ for the day and hour respectively, the extent to which a POI l's daily popularity matches a time t is estimated by $f_d(\langle utl \rangle) = \mathbf{d}^{l^T} \mathbf{d}^t$, and hourly popularity by $f_h(\langle utl \rangle) = \mathbf{h}^{l^T} \mathbf{h}^t$.
- **Popularity Preference**. As more popular POIs usually would expect more check-ins, we assign a popularity preference for each POI without distinguishing users. $f_p(\langle utl \rangle) = p^l$.

Let $\mathbf{f} = [f_c, f_g, f_d, f_h, f_p]^T$. It[5] measures how a POI matches a user's preference at a particular time. We employ constraints that require our model to accord with the data at each dimension of the preferences, *i.e.*, the model distribution matches the empirical distribution at the feature space:

$$\mathbb{E}_\pi(\mathbf{f}) = \mathbb{E}_{\tilde{\pi}}(\mathbf{f})$$

$$i.e., \quad \sum_{u,t,l} \tilde{p}(u,t) p(l|u,t) \mathbf{f} = \sum_{u,t,l} \tilde{p}(u,t) \tilde{p}(l|u,t) \mathbf{f}$$

$$i.e., \quad \sum_{u,t,l} \tilde{\pi}_{ut} \pi_{utl} \mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut} \tilde{\pi}_{utl} \mathbf{f}$$

where $\mathbb{E}$ denotes expectation. Note that we do not model the joint distribution of u and t (*i.e.*, $p(u,t)$) since the goal is to predict l given u and t. We let $p(u,t) = \tilde{p}(u,t) = \tilde{\pi}_{u,t}$. The model parameters[6] here contain $\{\pi_{utl}, \forall u,t,l\}$ only. This also classifies our problem as a discriminative learning task (as opposed to generative learning).

3.1.2 A Maximum Entropy Model with Fixed Latent Parameters

With the constraints defined above, we formulate our MaxEnt model in this section.

[5]A complete notation should be $\mathbf{f}(\langle utl \rangle) = (f_c(\langle utl \rangle), f_g(\langle utl \rangle), f_d(\langle utl \rangle), f_h(\langle utl \rangle), f_p(\langle utl \rangle))^T$, in the following of the paper, we omit $(\langle utl \rangle)$ for brevity and readability.

[6]We slightly abuse the terminology *parameter*. Model parameters refer to π_{utl} and latent parameters refer to $(\mathbf{o}, \mathbf{r})$.

The conditional entropy of π_{utl} is given by

$$H(\pi) = -\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} = -\mathbb{E}_{\pi}(\ln \pi_{utl})$$

As discussed in the previous section, we constrain the distribution π to a set $\mathcal{C}$ of allowed probability distributions:

$$\mathcal{C} = \{\pi | \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f}\}$$

By the MaxEnt principle, we should select a model from $\mathcal{C}$ with maximum $H(\pi)$:

$$\pi^* = \arg\max_{\pi \in \mathcal{C}} H(\pi)$$

Therefore we have the following MaxEnt model for π:

$$\max_{\pi} -\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} \tag{2}$$

$$s.t. \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f} \tag{3}$$

$$\sum_{l} \pi_{utl} = 1 \quad \forall u,t \tag{4}$$

$$\pi_{utl} > 0 \quad \forall u,t,l \tag{5}$$

Note that equation (3) is a vector form of $|\mathbf{f}| = 5$ constraints, corresponding to the 5 dimensional preferences.

We solve the constrained optimization problem in the dual space (see Appendix A), which gives the following form of π_{utl}:

$$\pi_{utl} = \frac{\exp(\mathbf{w}^T\mathbf{f}(\langle utl \rangle))}{\sum_l \exp(\mathbf{w}^T\mathbf{f}(\langle utl \rangle))} \quad \forall u,t,l \tag{6}$$

where $\mathbf{w}$ is the Lagrange coefficients. Solving the primal problem turns out to be maximizing the log likelihood of the data with π_{utl} specified by Equation (6). Therefore we obtain the optimal $\mathbf{w}^*$ from the maximum likelihood estimation:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} LL \tag{7}$$

$$LL = \sum_{utl} \tilde{\pi}_{ut}\tilde{\pi}_{utl} \ln \pi_{utl} \tag{8}$$

Finally, the solution for the primal problem is given by:

$$\pi^*_{utl} = \frac{\exp(\mathbf{w}^{*T}\mathbf{f})}{Z_{ut}}, \quad Z_{ut} = \sum_l \exp(\mathbf{w}^{*T}\mathbf{f}) \quad \forall u,t,l$$

where $\mathbf{w}^*$ is the optimal Lagrange coefficients, each element of which corresponds to a constraint in Equation (3).

3.2 Recovering Latent Parameters via Minimax Entropy

In the previous section, we have completed the discussion for the case where we assume the latent parameters are given so that all the features are explicit. Now let us bring the latent features back. We have f_g as a *latent feature* which is parameterized by $(\mathbf{o}, \mathbf{r})$. Therefore $\mathbf{w}^*$ is also parameterized by $(\mathbf{o}, \mathbf{r})$. The optimal solution is thus $\pi^*(\mathbf{o}, \mathbf{r})$:

$$\pi^*_{utl}(\mathbf{o}, \mathbf{r}) = \frac{\exp(\mathbf{w}^*(\mathbf{o}, \mathbf{r})^T\mathbf{f}(\langle utl \rangle)(\mathbf{o}, \mathbf{r}))}{\sum_l \exp(\mathbf{w}^*(\mathbf{o}, \mathbf{r})^T\mathbf{f}(\langle utl \rangle)(\mathbf{o}, \mathbf{r}))} \quad \forall u,t,l$$

We propose that ***the optimal*** $(\mathbf{o}, \mathbf{r})$ ***should be chosen such that*** $\pi^*(\mathbf{o}, \mathbf{r})$ ***is minimized*** and justify this statement in this section.

To measure the quality of the check-in preference distribution, we use the standard Kullback-Leibler (KL) divergence [12] from $\pi^*(\mathbf{o}, \mathbf{r})$ to the true user check-in preference Π. Π is the true conditional distribution: $\Pi_{utl} = p_{true}(l|u,t)$[7]. The optimal $(\mathbf{o}, \mathbf{r})$ should give the smallest KL divergence:

$$(\mathbf{o}^*, \mathbf{r}^*) = \arg\min_{\mathbf{r}>0, \mathbf{o}} KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r}))$$

where

$$\begin{aligned} KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r})) &= \mathbb{E}_{\Pi}(\ln \Pi_{utl}) - \mathbb{E}_{\Pi}(\ln \pi^*_{utl}) \\ &= -\mathbb{E}_{\Pi}(\ln \pi^*_{utl}) - H(\Pi) \end{aligned}$$

The difficulty here is that the true distribution Π is unknown, thus we cannot directly evaluate the first term. However, under the assumption that our sample size is reasonably large, which means the expected feature statistics $\mathbb{E}_{\Pi}(\mathbf{f})$ can be approximated exactly by neglecting the estimation errors in the observed statistics $\mathbb{E}_{\tilde{\pi}}(\mathbf{f})$, we obtain the following theorem.

THEOREM 1. *The KL divergence from* $\pi^*(\mathbf{o}, \mathbf{r})$[8] *to the true distribution* Π *is given by* $KL(\Pi, \pi^*) = H(\pi^*) - H(\Pi)$

PROOF. We need to prove $\mathbb{E}_{\Pi}(\ln \pi^*_{utl}) = -H(\pi^*)$. As shown before, π^* has the following form:

$$\pi^*_{utl} = \frac{\exp(\mathbf{w}^{*T}\mathbf{f})}{Z_{ut}}, Z_{ut} = \sum_l \exp(\mathbf{w}^{*T}\mathbf{f}) \quad \forall u,t,l$$

where $\mathbf{w}^*$ is the optimal Lagrange coefficients. Hence we have

$$\begin{aligned} \mathbb{E}_{\Pi}(\ln \pi^*_{utl}) &= \mathbb{E}_{\Pi}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\Pi}(\ln Z_{ut}) \\ &= \mathbb{E}_{\tilde{\pi}}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\tilde{\pi}}(\ln Z_{ut}) \\ &\qquad \text{by } \mathbb{E}_{\tilde{\pi}}(\mathbf{f}) = \mathbb{E}_{\Pi}(\mathbf{f}) \\ &= \mathbb{E}_{\pi^*}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\pi^*}(\ln Z_{ut}) \\ &\qquad \text{by Equation (3)} \\ &= \mathbb{E}_{\pi^*}(\ln \pi^*_{utl}) = -H(\pi^*) \end{aligned}$$

and the result follows. □

As the entropy of Π is fixed, and the entropy of π^* is parameterized by $(\mathbf{o}, \mathbf{r})$, in order to minimize $KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r}))$, we conclude that the latent variables should be estimated by minimizing the maximized entropy:

$$(\mathbf{o}^*, \mathbf{r}^*) = \arg\min_{\mathbf{o},\mathbf{r}} \sum_{u,t,l} -\tilde{\pi}_{ut}\pi^*_{utl}(\mathbf{o}, \mathbf{r}) \ln \pi^*_{utl}(\mathbf{o}, \mathbf{r}) \tag{9}$$

Therefore, we obtain our entire minimax entropy framework

[7] As before, we do not model $\Pi_{ut} = p_{true}(u,t)$ and let $\Pi_{ut} = \tilde{\pi}_{ut}$.

[8] For brevity, we use π^* short for $\pi^*(\mathbf{o}, \mathbf{r})$ in this proof

as summarized in the following program:

$$\min_{\mathbf{o},\mathbf{r}} \max_{\pi} -\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} \tag{10}$$

$$s.t. \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f} \tag{11}$$

$$\sum_{l} \pi_{utl} = 1 \quad \forall u,t \tag{12}$$

$$\pi_{utl} > 0 \quad \forall u,t,l \tag{13}$$

$$r_i > 0 \quad \forall i \tag{14}$$

3.3 The Learning Algorithm

Algorithm 1: The learning Algorithm for the Minimax Entropy Approach of Check-in Preferences Modeling

Input: A user check-in database $\{\langle utl \rangle\}$
Output: Check-in preference $\{\pi_{utl}\}, \forall u,t,l$; geographic clustering parameters $(\mathbf{o},\mathbf{r})$

1 Do a K-means clustering on the latitude-longitude coordinates of the POIs. Initialize $\mathbf{o}^*$ and $\mathbf{r}^*$ to be centers and average distances to the centers.
2 **for** *iter = 1:Maxiter* **do**
3 **MaxEnt step.** With $(\mathbf{o},\mathbf{r})$ fixed to $(\mathbf{o}^*,\mathbf{r}^*)$, solve the MaxEnt problem to obtain $\mathbf{w}^*$.

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} LL$$

$$\text{where } LL(\mathbf{w},\mathbf{o}^*,\mathbf{r}^*) = \sum_{utl} \tilde{\pi}_{ut}\tilde{\pi}_{utl} \ln \pi_{utl},$$

$$\pi_{utl} = \frac{\exp(\mathbf{w}^T\mathbf{f}(\mathbf{o}^*,\mathbf{r}^*))}{\sum_l \exp(\mathbf{w}^T\mathbf{f}(\mathbf{o}^*,\mathbf{r}^*))}$$

MinEnt step. With $\mathbf{w}$ fixed to $\mathbf{w}^*$, estimate the latent parameters $(\mathbf{o},\mathbf{r})$.

$$(\mathbf{o}^*,\mathbf{r}^*) = \arg\min_{\mathbf{r}>0,\mathbf{o}} -LL$$

$$\text{where } LL(\mathbf{w}^*,\mathbf{o},\mathbf{r}) = \sum_{utl} \tilde{\pi}_{ut}\tilde{\pi}_{utl} \ln \pi_{utl},$$

$$\pi_{utl} = \frac{\exp(\mathbf{w}^{*T}\mathbf{f}(\mathbf{o},\mathbf{r}))}{\sum_l \exp(\mathbf{w}^{*T}\mathbf{f}(\mathbf{o},\mathbf{r}))}$$

4 **end**

While it is hard to obtain a close form solution, we propose a neat coordinate descent learning procedure to solve the optimization problem.

We convert the inherent MaxEnt part to the dual space (see details in Appendix A) which reduces the problem to the following form:

$$\min_{\mathbf{r}>0,\mathbf{o}} \min_{\mathbf{w}} -LL \quad i.e., \quad \min_{\mathbf{r}>0,\mathbf{o},\mathbf{w}} -LL \tag{15}$$

where LL is given by Equation (8).

The objective now is to find the set of $(\mathbf{w},\mathbf{o},\mathbf{r})$ which minimizes the minus log likelihood LL of the data. This is divided to solving a MaxEnt problem (finding $\mathbf{w}^*$ with $(\mathbf{o},\mathbf{r})$ fixed) and a MinEnt problem (finding $(\mathbf{o}^*,\mathbf{r}^*)$ with $\mathbf{w}$ fixed).

Algorithm 1 sketches the learning algorithm. First, the geographic centers are initialized by a K-means clustering; the radius for each cluster is initialized by the average distance to the center. After initialization, we solve the MaxEnt and MinEnt problems alternately to get the optimal $(\mathbf{w},\mathbf{o},\mathbf{r})$. Both sides of optimization are solved by the L-BFGS [17] algorithm. See Appendix B for the optimization details.

4. RELATED WORK

Modeling the time aware check-in preference of users is the fundamental component of location[9] prediction and location recommendation. We review previous study on location prediction/recommendation tasks. Then we review the background of related discriminative models.

4.1 Location Prediction/Recommendation

There has been a substantial amount of research on location prediction/recommendation ever since the GPS devices became widely available. The prediction and recommendation tasks are closely related since they both predict a list of locations which are evaluated by the prediction accuracy. There are several subtle differences though. Location prediction usually focuses more on the places which have been already visited by a user and largely depends on the time point. Therefore, spatio-temporal regularity usually plays an important role in the task. On the contrary, location recommendation task focuses more on the unvisited locations based on collaborative filtering. The recommendation may or may not be time aware as well. Unlike movie recommendations where one may not want to watch a movie he/she has already watched before, a location can be checked in repeatedly by a user. Therefore it is desirable to include the places which have been visited before in the recommendation. In this paper, we do not distinguish between visited locations and new locations but output a distribution over all locations, where the most probable ones can be used for both prediction and recommendation.

One line of research [24, 25, 26, 27] focus on the study of GPS trajectories collected from human movements. Location prediction/recommendation on the trajectory data is a simpler task compared to the check-in data since trajectories contain consecutive movements of users which are very dense. The Nokia Research Center collected GPS data from 200 smartphone volunteers in the course of 1 year and launched a next place prediction challenge [14] in 2012. The best entries achieved prediction accuracies of above 50%.

However, location prediction/recommendation with the check-in data from LBSN is much more challenging due to the sparseness. Cheng *et al.* [4] propose a mixed hidden Markov model to predict the category of a user's next move and then predict the location given the category. However, while human movements may be Markovian, people usually do not check in at every POI they visit. Gao *et al.* [9, 10] explore the Hierarchical Pitman-Yor process [19] and view the check-in sequences as a language model to encode the historical effects. This method works much better for GPS trajectories [10] than check-in data [9] because the model also assumes dependencies between consecutive check-ins. Cho *et al.* [5], Yuan *et al.* [23] and Gao *et al.* [8, 7] highlight the daily periodicity of check-ins and show that temporal effects have significant influence on capturing users' check-in behaviors. Ye *et al.* [9] incorporate geographic influence to a collaborative filtering model by assuming a power-law distribution of the pairwise check-in distances. Cheng *et*

[9]In this paper, we use "location" and "POI" interchangeably as long as there is no ambiguity.

al. [3] extend this work to multi-center geographic distributions and combine it with a matrix factorization model. Yin *et al.* [21], Kurashima *et al.* [13] and Liu *et al.* [15, 16] propose generative models which introduce the concept of user/location profiles. Our approach is able to incorporate the various factors from the previous work and model them in a unified way.

4.2 Related Discriminative Models

The maximum entropy principle was first proposed by E.T. Jaynes [11] in 1957. It provides a very general rationale why we should select the model with the maximum entropy. It has seen widescale applications to real world problems recently especially within the natural language processing field [2]. Introducing latent variables to discriminative models is not new. The minimax entropy principle is first proposed in the computer vision community by Zhu *et al.* [30] for feature selection within the MaxEnt framework, which has motivated various methods of statistical estimation and pattern recognition. Lately Zhou *et al.* [28] adopted this methodology to study a crowdsourcing problem and demonstrated substantial performance improvement over existing methods. Previous work [20, 29, 22, 18] have introduced hidden variables to other discriminative models such as SVMs and CRFs in various forms as well. However, allowing features to be governed by a parametric form is innovative, which directly corresponds to the generative assumptions of the features. This makes our approach especially suited to user preference modeling where multidimensional heterogeneous information need to be modeled. The optimization of the parameters is naturally guided by the MinEnt principle, which not only makes it theoretically elegant but also leads to a handy coordinate descent solution.

5. DISCUSSIONS

In this section, we analyze the connections from our approach to several standard approaches. We also explain how various additional information can be incorporated and how cold start issue is naturally handled by our approach.

5.1 Connection to Maximum Likelihood Estimation

Our model has the intuitive interpretation of a discriminative maximum likelihood estimation (MLE). We have already seen that the final objective is to seek a maximum likelihood estimator $(\mathbf{w}^*, \mathbf{o}^*, \mathbf{r}^*)$ for the objective function LL, with the conditional probability defined as $p(l|u,t) = \pi_{utl} = \frac{\exp(\mathbf{w}^T\mathbf{f})}{\sum_l \exp(\mathbf{w}^T\mathbf{f})}$.

5.2 Connection to Matrix Factorization based Collaborative Filtering

Our model is a linear model in the sense that the prediction score is determined by $\mathbf{w}^T\mathbf{f}(\langle utl\rangle)$, and $\mathbf{w}$ is determined not only by the user check-in data, but also on the features $\mathbf{f}(\langle utl\rangle)$. In the standard matrix factorization (MF) model for recommendation where the access to meaningful information such as category, latitude and longitude is limited, it is still possible to perform prediction via purely utilizing the factorization of the user-item rating matrix R as approximated by the product of two low-rank matrices. Specifically, by carefully selecting a reasonable dimension parameter K which is much smaller than the number of users M and items N, MF approximates $R \approx U^T V$ where $U^{M\times K}$ is a user matrix and $V^{K\times N}$ is an item matrix. An interesting analogy is that the columns of U (or V) can be viewed as profiles of users (or items). However, unlike in our model where parameter estimation ($\mathbf{w}$) is performed and latent geographic clusters are learned jointly, this approach computes the prediction score in a rather simplified manner as $u^T v$ where u is the corresponding column in U for a user and v is the column in V for an item.

5.3 Incorporate Various Information

To make our model concrete, we defined every detail of how the features are generated. Nevertheless, the features do not necessarily have to be defined as we did in the previous sections. In this paper, we follow a natural thought that the category, geographic, temporal and popularity preferences are influential factors for a check-in. However, we can model other types of information into our learning framework as well in the forms of both explicit and latent features. For example, if the description and reviews of POIs are available, we can incorporate text features as explicit features. If social network information is available, we can incorporate friend clusters as latent features. With both explicit and latent features, our approach models ambiguous knowledge together with explicit knowledge in a unified manner to find the best possible way to utilize them.

5.4 Cold Start

As in most recommendation problems, cold start is an important issue in preference learning. If we have little historical data for a user, predicting her preference typically falls back to an appropriate way of utilizing "independent" features that do not reply on histories, such as gender, age, hometown, etc. Our model can elegantly handle such cases by just taking care of these information as additional features. They can be both explicit or latent. In this study, we do not have those demographic information available thus we do not define them in the profiles. However, these features can be utilized exactly the same way as the defined ones. We don't even need to worry about how to distinguish cold start users from the heavy users since the automatically learned weights help us to do the trade-off. In the extreme case where a user has no history at all, the prediction will fall back to a regression on those "independent" features. This treatment of cold start scenarios is of the similar style as in [1], with better expressiveness, reduced model complexity and simpler optimization procedure.

6. EXPERIMENTS

We introduce our datasets and report our experimental results in this section. We evaluate our proposed method on the location prediction/recommendation task.

First we evaluate the effectiveness of our method by accuracy of prediction under various settings. Then we zoom in to see the benefits from optimizing the latent parameters. At the end we conduct an efficiency study and analyze the scalability of our method.

6.1 Data

We conduct our experiments on two public real world datasets [6] obtained from Foursquare[10]. The first dataset (**CA**) contains 483,813 check-in records of 4,163 users in

[10]https://foursquare.com

California USA ranging from December 2009 to June 2013. The second dataset (**World**) contains 2,290,996 check-in records of 11,326 users around the world ranging from January 2011 to December 2011. We preprocess the datasets by removing the users and POIs with check-ins fewer than 20. Each check-in record consists of a user ID, a POI ID, a check-in timestamp, and the latitude and longitude of the POI. The first dataset has the category information of each POI while the second one does not.

We sort each user's check-ins chronologically and assign the first 80% of the check-ins to the training set and the remaining 20% to the test set.

6.2 Implementation

6.2.1 Smoothing the Category and Temporal Preference

Smoothing is a common practice to avoid overfitting and mitigate the effect from noise when estimating categorical distributions. It assigns a tiny probability to the categories that are not seen in the data. In our model, we do a simple add-one smoothing to the category preference and temporal preference of users.

6.2.2 Parameter Regularization

We incorporate a standard L_2 regularization on $\mathbf{w}$ in the MaxEnt step to avoid overfitting and numerical problems. The objective function becomes $-LL + \frac{1}{2}\beta||\mathbf{w}||_2^2$ where β is the regularization parameter. From our experiments, we found that our model is insensitive to parameter regularization. We set β to 0.2.

6.2.3 Number of Geographic Clusters

The number of clusters affect the granularity of the geographic regions. It can be empirically set by cross validation or specified by human knowledge of how fine grained regions we want to achieve. In this paper, we set the cluster number to 30 for the CA dataset and 200 for the World dataset.

6.2.4 Number of Iterations

We set the global iteration number $Maxiter$ to 20 and run 10 iterations within each L-BFGS step based on the empirical study of the convergence rate.

6.3 Effectiveness Study

6.3.1 Methods for Comparison

Existing models on location prediction/recommendation are usually specifically designed emphasizing a particular set of factors. Unlike our model, most of them cannot be generalized to take arbitrary features. In this study, we consider category, time, popularity and geographic coordinates. Thus we compare our method (the basic **MaxEnt** model with K-means initialization, and the full **Minimax** model) with the following three state-of-art models which can accept the same set of features.

- **PMM**. A spatial-temporal location *prediction* model proposed in [5], which studies the spatial-temporal regularity of user mobilities and builds a generative model for check-ins.
- **HMM**. A mixed hidden Markov location *prediction* model proposed in [4], which first predicts the category of user activity at the next step and then predict the most likely location given the estimated category distribution. This model is compared to only for the CA dataset because the category information is not available for the World dataset.
- **TGM**. A time-aware location *recommendation* model proposed in [23], which employs a user-based collaborative filtering framework with geographic influence incorporated by a linear combination. For the CA dataset, we enhance this model by a further linear combination with the category distribution at the prediction time for fair comparison.

6.3.2 Evaluation on Accuracy

Evaluation Metrics. We compute the accuracy of both location and category prediction on the test set for the CA dataset and the accuracy of location prediction for the World dataset. For each $\langle utl \rangle$ in the test data, we return the top-k locations predicted by each model for (u, t). As long as the true location l lies in the top-k set, we consider it as a correct prediction. For categories, we obtain the category list associated with the top-k predictions and evaluate the accuracy in the same way.

Performance. As shown in Figure 1, our method significantly outperforms the three baselines w.r.t both POI and category prediction at all position k's. TGM is not working well because 1) it takes a binary user-location matrix as the input for collaborative filtering which completely ignores the preference over different visited POIs; 2) it involves geographic and temporal influences in an ad-hoc manner which is difficult to coordinate in the optimal way; 3) the way it encodes the geographic knowledge is to do a power-law fitting of consecutive check-in distances, which is sensitive to outliers and cannot capture the clustering effect of check-ins. HMM relies a lot on the Markovian assumption of user activity. If a user's check-ins are not so dense (which is usually the case since people do not check-in at every POI they visit), the dependency between consecutive check-ins are weakened. Once the Markovian assumption does not hold, good performance would not be guaranteed. PMM gives the worst performance. The generative assumption that movements are governed by Gaussian spatial-temporal clusters is too strict and limits the model's expressiveness and generalizability. Another interesting phenomenon we can observe is that despite the lack of category information for the World dataset, the location prediction accuracy is higher than the CA dataset for all the models. In fact, the World dataset has comparative number of POIs with the CA dataset but has substantially large number of checkin records. This makes the learning task easier for all the models. The performance difference is more significant on the CA dataset, which concludes that when we have limited number of observations for training, our MiniMax model generalizes better than the baseline models.

6.3.3 The Influence of Latent Features

Performance. We plot the accuracy@top-5 of our model as the latent parameters are optimized with 20 iterations in Figure 2. The accuracy continues to improve as the latent parameters are optimized. It is also worth noting that the convergence is very fast.

A Visualization of the Geographic Clusters. To illustrate the intuition behind optimizing the latent parameters, we show a snapshot of the San Francisco Bay Area

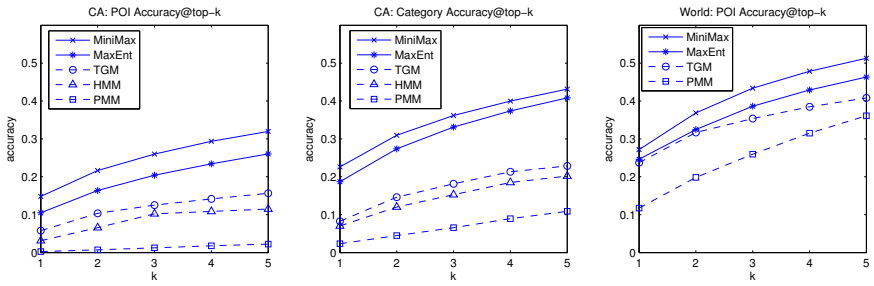


Figure 1: Prediction Accuracy @ top-k

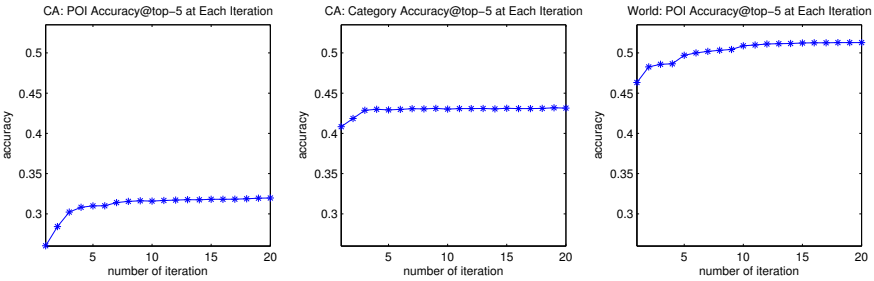
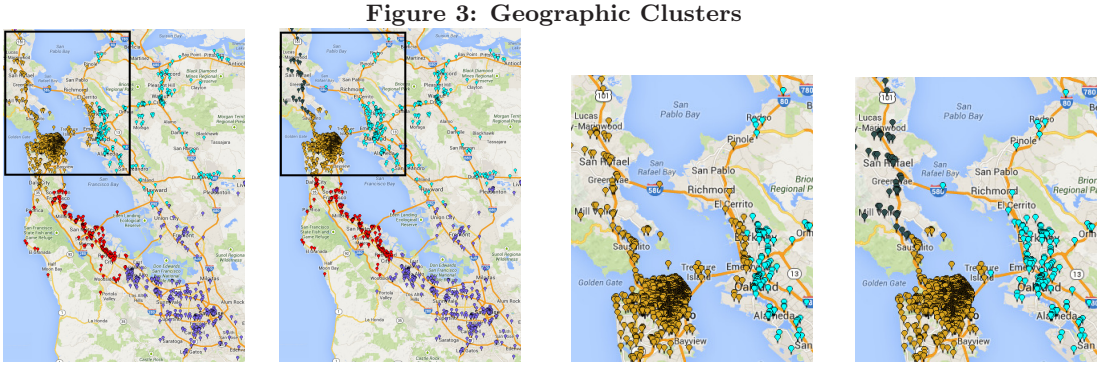


Figure 2: Prediction Accuracy @ top-5 as we optimize the latent parameters. The prediction becomes more accurate and the convergence is very fast.

Figure 3: Geographic Clusters



(a) SF k-means (b) SF (c) SF k-means (d) SF

geographic clusters obtained from our algorithm for the CA dataset in Figure 3(b). We assign each POI l to a cluster by selecting the largest weight of $\mathbf{g}^l$. Figure 3(a) shows the initial k-means clustering results.

The optimal clustering structure is refined from the K-means clustering via the interaction with the check-in preferences modeling. We can observe interesting refinements. As shown in Figure 3(d) and Figure 3(c), we zoom in to San Francisco (SF) city. As K-means clustering blindly clusters the POIs by geographic latitudes and longitudes, the cluster centered at SF (yellow) stretched to San Rafael, Oakland and Berkeley; while in the refined clusters, SF corresponds to a concentrated cluster. The SF cluster extends north right to the vicinity of the Golden Gate Bridge as tourists to SF would always like to explore the Golden Gate Bridge.

6.4 Efficiency Study

In this section, we first analyze the complexity of our algorithm and then present experimental results on the execution time.

6.4.1 *Complexity Analysis*

The coordinate descent algorithm contains a MaxEnt step and a MinEnt step. Within each step, the space and time consuming part lies in the evaluation of the function value and the gradient (see Appendix B), which determines the complexity of our algorithm. We show that both space and time complexity are linear w.r.t the number of users, time indices and POIs.

Space. At each iteration of both steps, we need to store π_{utl} for all (u,t) pairs that appear in the training set and any $l \in L$, which requires at most $O(|U||T||L|)$ space. Computation of the feature values are done at the beginning of each step and requires at most two components of (u,t,l), therefore does not affect the order of space complexity. The space required to store the current estimate of the solution in the MaxEnt step is the dimension of the features $\mathbf{f}$. In the

MinEnt step it is 3 times the number of geographic clusters, which is also not contributing to the order of complexity. Thus the overall space complexity is $O(|U||T||L|)$.

Time. At each iteration, to evaluate the function and gradient values, we need to compute π_{utl} for all (u,t) pairs that appear in the training set and any $l \in L$. Let the total iteration number be M and let the maximum function evaluation number be M_1 at one MaxEnt step and M_2 at one MinEnt step. The overall time complexity is $O(M(M_1+M_2)|U||T||L|)$.

6.4.2 *Execution Time Evaluation*

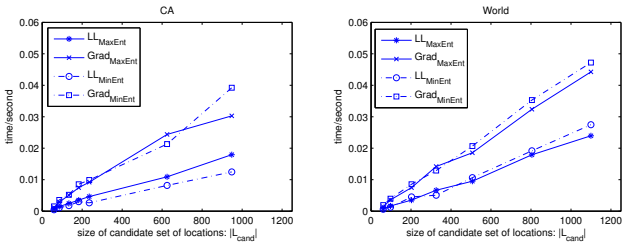


Figure 4: Average Execution Time of A Function/Gradient Evaluation

To examine the efficiency of our algorithm, we illustrate the execution time of a function/gradient evaluation for both the MaxEnt step and the MinEnt step. The time consuming computation of π_{utl} can be computed in parallel since $\{\pi_{utl}|l \in L\}$ can be computed for each (u,t) pair simultaneously. Therefore we examine the average execution time of a function/gradient evaluation over all (u,t) pairs. We vary the pruning threshold δ_α and obtain the *time - candidate set size* curves shown in Figure 4. They all exhibits a linear trend in $|L|$ while the gradient evaluation is more expensive than the function value evaluation. $|L_{cand}|$ is the average size of the candidate set over all (u,t) pairs. In the ideal case, the overall time complexity can be reduced to $O(M(M_1+M_2)|L_{cand}|)$.

7. CONCLUSIONS

In this paper, we develop a novel minimax approach for modeling time-aware check-in preferences. Specifically, our approach has the advantage of investigating the multidimensional knowledge of entities (users, locations) as well as jointly learning the latent geographic clustering. The proposed discriminative model can strike a good balance between explaining seen data and generalizing to unseen data by requiring the model to satisfy meaningful relaxed constraints. Going beyond check-in preference modeling, the proposed minimax entropy model also provides a general guidance to model ambiguous features with arbitrary parametric forms, which significantly boosts the flexibility and expressiveness of the standard discriminative learning models.

Acknowledgment

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-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov), and MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC.

8. REFERENCES

[1] D. Agarwal and B.-C. Chen. Regression-based latent factor models. KDD '09, New York, NY, USA. ACM.

[2] A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996.

[3] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.

[4] H. Cheng, J. Ye, and Z. Zhu. What's your next move: User activity prediction in location-based social networks. In *SDM*, pages 171–179, 2013.

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

[6] H. Gao and H. Liu. Location-based social network data repository, 2014.

[7] H. Gao, J. Tang, X. Hu, and H. Liu. Exploring temporal effects for location recommendation on location-based social networks. In *RecSys*, pages 93–100, 2013.

[8] H. Gao, J. Tang, X. Hu, and H. Liu. Modeling temporal effects of human mobile behavior on location-based social networks. In *CIKM*, pages 1673–1678, 2013.

[9] H. Gao, J. Tang, and H. Liu. Exploring social-historical ties on location-based social networks. In *ICWSM*, 2012.

[10] H. Gao, J. Tang, and H. Liu. Mobile Location Prediction in Spatio-Temporal Context. *the Procedings of Mobile Data Challenge by Nokia Workshop at the Tenth International Conference on Pervasive Computing*, June 2012.

[11] E. T. Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

[12] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, pages 79–86, 1951.

[13] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura. Geo topic model: Joint modeling of user's activity area and interests for location recommendation. WSDM '13, pages 375–384, New York, NY, USA, 2013. ACM.

[14] J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. M. T. Do, O. Dousse, J. Eberle, and M. Miettinen. From big smartphone data to worldwide research: The mobile data challenge. *Pervasive and Mobile Computing*, 9(6):752–771, 2013.

[15] B. Liu, Y. Fu, Z. Yao, and H. Xiong. Learning geographical preferences for point-of-interest recommendation. In *KDD*, pages 1043–1051, 2013.

[16] B. Liu and H. Xiong. Point-of-interest recommendation in location based social networks with topic and location awareness. In *SDM*, pages 396–404, 2013.

[17] D. C. Liu and J. Nocedal. On the limited memory bfgs

method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[18] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell. Hidden conditional random fields. *IEEE transactions on pattern analysis and machine intelligence*, 29(10):1848, 2007.

[19] Y. W. Teh. A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 985–992. Association for Computational Linguistics, 2006.

[20] S. Wang, D. Schuurmans, and Y. Zhao. The latent maximum entropy principle. *ACM Trans. Knowl. Discov. Data*, 6(2):8:1–8:42, July 2012.

[21] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen. Lcars: a location-content-aware recommender system. In *SIGKDD*, pages 221–229. ACM, 2013.

[22] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*. ACM, 2009.

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

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

[25] Y. Zheng and X. Xie. Learning location correlation from gps trajectories. In *Mobile Data Management*, pages 27–32, 2010.

[26] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma. Recommending friends and locations based on individual location history. *TWEB*, 5(1):5, 2011.

[27] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining correlation between locations using human location history. In *GIS*, pages 472–475, 2009.

[28] D. Zhou, J. C. Platt, S. Basu, and Y. Mao. Learning from the wisdom of crowds by minimax entropy. In *NIPS*, pages 2204–2212, 2012.

[29] J. Zhu, E. P. Xing, and B. Zhang. Partially observed maximum entropy discrimination markov networks. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *NIPS*. Curran Associates, Inc., 2009.

[30] S. C. Zhu, Y. N. Wu, and D. Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8):1627–1660, 1997.

APPENDIX

A. PRIMAL DUAL CONVERSION

The Lagrangian of the MaxEnt problem is

$$\begin{aligned}\mathcal{L} = &-\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}\ln\pi_{utl}\\ &+\sum_{\alpha}w_\alpha(\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}f_\alpha-\sum_{u,t,l}\tilde{\pi}_{ut}\tilde{\pi}_{utl}f_\alpha)\\ &+\sum_{u,t}\eta_{u,t}(\sum_{l}\pi_{utl}-1)\end{aligned}$$

where $\{w_\alpha\}$ and $\{\eta_{u,t}\}$ are the Lagrange multipliers.

Let $\frac{\partial\mathcal{L}}{\partial\pi_{utl}}=0$, we have

$$\begin{aligned}&-\tilde{\pi}_{ut}(1+\ln\pi_{utl})+\sum_\alpha w_\alpha(\tilde{\pi}_{ut}f_\alpha)+\eta_{u,t}=0\\ \Longleftrightarrow\ &\ln\pi_{utl}=\sum_\alpha w_\alpha f_\alpha+\frac{\eta_{u,t}}{\tilde{\pi}_{ut}}-1\end{aligned}$$

Apply the constraint $\sum_l\pi_{utl}=1\quad\forall u,t$, we can get

$$\pi_{utl}=\frac{\exp(\sum_\alpha w_\alpha f_\alpha)}{\sum_l\exp(\sum_\alpha w_\alpha f_\alpha)}\forall u,t,l \tag{16}$$

Plugging Equation (16) into $\mathcal{L}$ gives that $\mathcal{L}$ is the minus log likelihood of the data. Maximizing the primal problem becomes minimizing the dual problem, which turns out to be maximizing the log likelihood of the data with π_{utl} specified by Equation (16). Therefore $\mathbf{w}^*$ is the maximum likelihood estimation:

$$LL=\sum_{utl}\tilde{\pi}_{ut}\tilde{\pi}_{utl}\ln\pi_{utl},\quad \mathbf{w}^*=\arg\min_{\mathbf{w}}-LL$$

where π_{utl} is of the form given in Equation (16).

B. OPTIMIZATION DETAILS

We derive the gradients required by L-BFGS for both MaxEnt and MinEnt steps.

- The MaxEnt problem is an unconstrained optimization problem in the dual space. The gradient w.r.t $\mathbf{w}$ is given by

$$\frac{\partial LL}{\partial w_\alpha}=\sum_{u,t,l}\tilde{\pi}_{ut}\tilde{\pi}_{utl}f_\alpha-\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}f_\alpha$$

  where π_{utl} is given by Equation (16). This is the difference between the expectations of the feature f_α from the model and the empirical mean.

  The Hessian matrix is given by

$$\begin{aligned}\frac{\partial^2 LL}{\partial w_\alpha\partial w_\beta}=&\mathbb{E}_\pi[(\sum_{u,t,l}\tilde{\pi}_{ut}\tilde{\pi}_{utl}f_\alpha-\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}f_\alpha)\\ &(\sum_{u,t,l}\tilde{\pi}_{ut}\tilde{\pi}_{utl}f_\beta-\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}f_\beta)]\end{aligned}$$

  which is the covariance matrix of the features, and is thus positive definite[11]. This indicates that the MaxEnt problem is strictly convex and has a unique solution.

- The optimization over the latent parameters may or may not be convex, depending on the form of the chosen geographic weight function. In this paper, the problem is not convex and L-BFGS will converge to the local minimum. We take several trials of the iteration process to approach the global minimum.

  The gradient w.r.t $(\mathbf{o},\mathbf{r})$ is given by

$$\frac{\partial LL}{\partial z_i}=\sum_{u,t,l}\tilde{\pi}_{ut}\tilde{\pi}_{utl}w_g\frac{\partial f_g}{\partial z_i}-\sum_{u,t,l}\tilde{\pi}_{ut}\pi_{utl}w_g\frac{\partial f_g}{\partial z_i}$$

  where z_i can be o_{ilat}, o_{ilon} or r_i, w_g is the weight corresponding to the geographic feature f_g and

$$\frac{\partial f_g}{\partial z_i}=g_i^u\frac{\partial g_i^l}{\partial z_i}+\frac{\partial g_i^u}{\partial z_i}g_i^l$$

  with g_i^l given by Equation (1).

[11] only in rare cases it may be positive semi-definite