Context-aware Location Annotation on Mobility Records through User Grouping

Yong Zhang[1], Hua Wei[2], Xuelian Lin[1], Fei Wu[2], Zhenhui Li[2], Kaiheng Chen[1], Yuandong Wang[1], and Jie Xu[3]

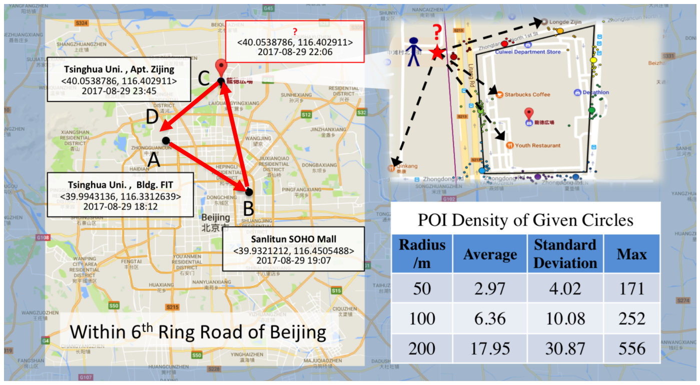
[1] School of Computer Science and Engineering, Beihang University, Beijing, China
{zhangy11, linxl, chenkh, wangyd}@act.buaa.edu.cn
[2] College of Information Science and Technology, The Pennsylvania State University, USA
{hzw77, fxw133, jessieli}@ist.psu.edu
[3] School of Computing, University of Leeds, Leeds, UK
j.xu@leeds.ac.uk

Abstract. Due to the increasing popularity of location-based services, a massive volume of human mobility records have been generated. At the same time, the growing spatial context data provides us rich semantic information. Associating the mobility records with relevant surrounding contexts, known as the location annotation, enables us to understand the semantics of the mobility records and helps further tasks like advertising. However, the location annotation problem is challenging due to the ambiguity of surrounding contexts and the sparsity of personal data. To solve this problem, we propose a Context-Aware location annotation method through User Grouping (CAUG) to annotate locations with venues. This method leverages user grouping and venue categories to alleviate the data sparsity issue and annotates locations according to multi-view information (spatial, temporal and contextual) of multiple granularities. Through extensive experiments on a real-world dataset, we demonstrate that our method significantly outperforms other baseline methods.

1 Introduction

In recent years, location-based services have been widely used in our daily lives and generated a massive volume of human mobility records (e.g., transportation records) and online spatial context data (e.g., venue database). The combination of mobility records with *relevant* contexts helps reveal the semantic of user movement and is known as the *semantic annotation of mobility records* [1]. In this paper, we use venue dataset as the context and consider the problem of mapping a user's location to a venue he might actually visit. The work can have important applications, such as user profiling, recommendation, and advertisement targeting. For example, as shown in Fig. 1, if we know a person often moves from a university to entertainment venues at night, and go back very late, we could infer this person is a sparky college student, and recommend some recreational activities to him. Besides, smart city applications can benefit from such semantic understandings of the raw transportation data.

However, it is hard to associate right venues with mobility records. The challenges are mainly two folds: (1) Both recorded locations and surrounding contexts are *ambiguous*. For a given mobility record of a user, the observed location could have noises, and



Radius /m	Average	Standard Deviation	Max
50	2.97	4.02	171
100	6.36	10.08	252
200	17.95	30.87	556

Fig. 1. An example of location annotation problem

there may be many possible venues around. As Fig. 1 shows, the number of POIs in some areas of Beijing can reach 500 (according to the data from AutoNavi[4]). (2) The user data maybe *sparse*. Though the total number of mobility records is large, each user may only have a limited number of personal data. According to the data from UCAR[5], less than 10% users have their trips recorded over 3 times within one month. In addition, most POIs only have a few visit records except some popular ones.

For the annotation problem, some previous work mainly considers the distance between the context location and the location of a user [2–4], while others further consider personal preference [5,6]. However, modeling personal preference straightforwardly requires adequate records for each individual user, which conflicts with the data sparsity. Furthermore, these methods do not consider the influence of contextual mobility records (e.g., former and later records of a given record). For example, a person who has just visit a restaurant is less likely to visit a restaurant again in a short time.

To tackle these problems, we propose a Context-Aware location annotation method through User Grouping, named CAUG. In this method, the correlation between mobility contextual records is captured by contextual features. And the data sparsity issue is mainly compensated by considering information of user groups, which is based on our intuition that users who share similar mobility patterns are likely to visit similar venues under the same condition. To summarize, we make the following contributions:

1. We propose an iterative grouping method to group users based on the similarity of their mobility patterns, which are captured by a Hidden Markov Model (HMM) [7]. The user-grouping method alleviates the data sparsity to a great extent.
2. We apply a ranking model to annotate locations, with a strategy that integrates multi-view (spatial, temporal and contextual) information extracted from users and POIs' historical information at different granularities. The comprehensive consideration guarantees the effectiveness of annotation.
3. We evaluate our method with a 14-month real-world dataset from a car-hailing company. Experimental results show our method can produce effective user groups and contextual features. Meanwhile, this method outperforms other baseline methods.

[4] A map service provider. https://en.wikipedia.org/wiki/AutoNavi.
[5] A chauffeured car service provider in China. https://www.crunchbase.com/organization/ucar.

Table 1. Summary of notations

Notation	Terms	Description
p	Point-of-Interest (POI)	$p\langle id, name, l, c\rangle$, where $p.l$ is a location defined by longitude and latitude, and $p.c$ represents categories of POI
c	Categories of POI	$c\langle c_1, c_2, c_3\rangle$, where $c.c_1$, $c.c_2$ and $c.c_3$ are p's 1^{st} class, 2^{nd} class and 3^{rd} class categories, respectively
g	Grid	$g\langle row, col\rangle$, an indexed grid (780 × 780 meters) in a city
tp	Period of a Day (POD)	A pre-clustered time period of a day, including morning, noon, afternoon, evening and late night
s	Spatio-temporal Area	$s\langle g, tp\rangle$, denoting a grid g at time period tp
z	User Activity	$z\langle v.c, tp\rangle$, where $v.c$ is a venue category and tp is a time period
x	Stop-Point	$x\langle u, l, t, p\rangle$, denoting a geographic location $x.l$ where a user $x.u$ *actually* picked up or droped off at time $x.t$ around a POI $x.p$
r	Travel Record	$r\langle x^s, x^e\rangle$, where $r.x^s$ and $r.x^e$ are the start stop-point and the end stop-point of a user, respectively
T'	Tajectory	A sequence of stop-points of a user

2 Preliminary

This section describes some basic terms used in this paper. The notations are summarized in Table 1.

We use transportation data from UCAR for our study, including a mass of travel records. When using UCAR's online car-hailing service to book a trip, a user can search based on address or based on the name of the POI and then selects a POI for pick-up and a POI for drop-off. After the trip finished, it will be recorded as a *travel record* which consists of a start *stop-point* and an end *stop-point*. It should be noted a POI selected by a user can be either a *venue* or an *indistinct place*. A *venue* refers to a place like a restaurant or a cinema where people conduct specific activities, while an *indistinct place* is a place like a crossroad or a public parking lot which is hardly the final intention of a user. In order to make users' trips more semantic, those *stop-points* whose POIs are indistinct places are to be annotated with venues users might actually visit.

A *trajectory* represents a sequence of stop-points with contextual relations. Therefore, we actually concatenate a set of *travel records* back to a *trajectory* if the end *stop-point* of the previous trip and the start *stop-point* of the next trip has similar time or location. The notion of *trajectory* enables our model to consider contextual correlation not only between a start *stop-point* and an end *stop-point*, but also between *travel records*. Specifically, given a sequence of *stop-points* $T = x_1^s x_1^e x_2^s x_2^e ... x_n^s x_n^e$ of a user u, a time gap threshold $\Delta_t > 0$ and a distance threshold $\Delta_d > 0$, a subsequence $T' = x_i^s x_i^e x_{i+1}^s x_{i+1}^e ... x_{i+k}^s x_{i+k}^e$ is a *trajectory* of T if T' satisfies: (a) $\forall 1 < j \leq k$, $x_j^s.t - x_{j-1}^e.t \leq \Delta_t$ or $d(x_{j-1}^e.l, x_j^s.l) \leq \Delta_d$, where $d(l_a, l_b)$ stands for the distance between location l_a and l_b; and (b) there are no longer sub-sequences in T that contains T' and satisfies condition (a). Also, for a given *stop-point* x in T', its former and later points are denoted as $\overleftarrow{x}$ and $\overrightarrow{x}$, respectively.

Problem 1 (Location Annotation) *Given a trajectory T' of a user u, for each stop-point x in T' whose POI $x.p$ is an indistinct place or unknown, a location annotation method provides a list of venues u might visits.*

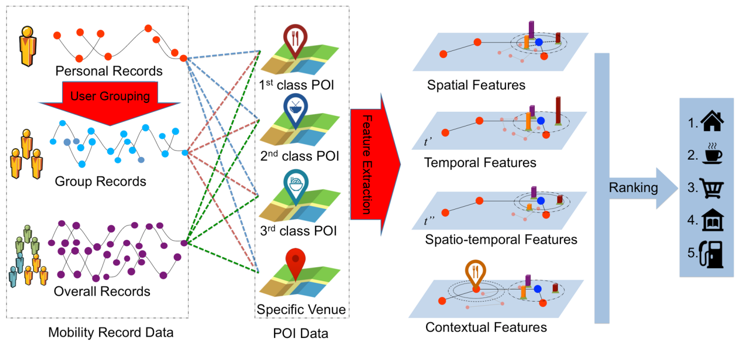


Fig. 2. An overview of CAUG

3 Method

In this section, we first introduce the overview of our Context-Aware location annotation method through User Grouping (CAUG). Then we present a user-grouping method based on HMM, followed by a feature extraction method and venue ranking models.

3.1 Overview of CAUG

The overall framework of CAUG is shown in Fig. 2, mainly including *user grouping*, *feature extracting* and *venue ranking*. When extracting features from historical records, we organize records into multiple granularities (represented by colored broken lines).

First, from the perspective of users, we observe the personal records are usually too sparse to do the personalized annotation and the overall records are too inconsistent as different users usually have totally different mobility patterns. Thus, we are motivated to find a middle-level granularity called user group to compensate, based on our intuition that users who share similar mobility patterns may have similar visit tendencies under the same condition (e.g., a group of colleagues may live in the same housing area and often go to a specific bar near their homes after work). In this way, we organize user travel records into three levels, i.e., personal level, group level and overall level.

Second, from the perspective of venues, we observe though most venues are seldom visited, similar venues have similar visited tendencies. Therefore, we consider not only the visit history of venues themselves but also that of the venue categories. We adopt a classification method of venue categories defined by AutoNavi and organize a specific venue into four classes. For example, a specific Starbucks coffee shop belongs to the shop itself (4^{th} class), Starbucks (3^{rd} class), cafe (2^{nd} class) and catering services (1^{st} class). Note that the venue category can vary with different classification methods based on different POI datasets. Combining the above two kinds of granularities, i.e., three levels of user records and four classes of venues, we finally get $3 \times 4 = 12$ granularities.

For a stop-point in a trajectory of a user, CAUG first selects all venues within a distance as candidates. Then, for each candidate venue, CAUG extracts a series of spatial, temporal, spatio-temporal and contextual features for each of the twelve granularities. Finally, CAUG returns a ranked list of Top-k venues through a ranking model.

3.2 User Grouping

When people move from one place to another place, their activities are in nature sequential and have mobility patterns. The mobility patterns of users could be captured by HMM, which is a general class of graphical model, describing the process of generating an unobservable state sequence from a hidden Markov chain and generating the observed sequence from the state sequence. The HMM method is widely used to model the mobility of users [4, 8]. Hence, we also use it for learning representations of users' mobility patterns and further use it in an iterative grouping.

HMM Formulating. We assume there are K activities $Z = \{z_1, z_2, ..., z_K\}$ (*i.e., hidden states*) and M spatio-temporal areas $S = \{s_1, s_2, ..., s_M\}$ (*i.e., observations*). As a user movement process is shown in Fig. 3, each observation s_n in the observed sequence $s_1 s_2 ... s_N$ corresponds to a state $z_n \in Z$, and the state seqeunce $z_1 z_2 ... z_N$ follows a transition regulation. Therefore, we consider three factors when formulating the HMM: (1) the probabilities of activities users begin, (2) the probabilities of transition between activities, and (3) the probabilities of users appearing at one area given the activity they are doing. More formally, the HMM is parameterized by $\lambda = (\pi, A, B)$, where $\pi = (\pi_i)$ is a K-dimensional vector which defines the initial distribution over the K activities, $A = [a_{ij}]_{K \times K}$ is a matrix that defines the transition probabilities among the K activities, and $B = [b_{ij}]_{K \times M}$ is a matrix which defines the emission probabilities of M spatio-temporal areas over the K activities.

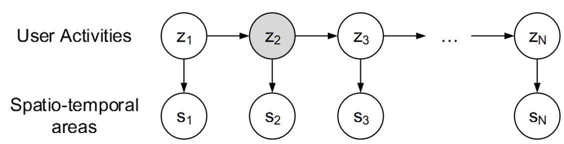


Fig. 3. The illustration of the HMM

Parameter Inference. Given R trajectories, we first generate a *observed* sequence and a corresponding *state* sequence for each trajectory by the following ways: (1) map stop-points in the trajectory to spatio-temporal areas, and consider the sequence of spatio-temporal areas as the observed sequence, (2) map stop-points in the trajectory to activities, and consider the activity sequence as the state sequence. However, because a proportion of stop-points' POIs may be indistinct places, their corresponding activities are unclear (represented by grey circles in Fig. 3). In this case, the parameters of the model are estimated in the following way:

$$\pi_i = \frac{R_i + \alpha}{R_\pi + K\alpha}, i = 1, ..., K \tag{1}$$

$$a_{ij} = \frac{A_{ij} + \alpha}{\sum_{k=1}^{K} A_{ik} + K\alpha}, i = 1, ..., K; j = 1, ..., K \tag{2}$$

$$b_{ij} = \begin{cases} \dfrac{B_{ij} + \alpha}{\sum_{m=1}^{M} B_{im} + M^s\alpha} & , z_i.tp = s_j.tp \\ 0 & , z_i.tp \neq s_j.tp \end{cases}, i = 1, ..., K; j = 1, ..., M \tag{3}$$

where R_i is the number of state sequences that begin with activity z_i and R_π is the number of state sequences whose first state is explicit. A_{ij} is the frequency of transferring from state i at time t to state j at time $t+1$, which is counted according to state sequences, skipping indistinct activities among them. B_{ij} is the frequency of appearing in spatio-temporal area s_j when doing activity z_i. M^s is the number of grids. The α is the smoothing parameter of the additive smoothing[6].

Iterative Grouping. Given a set of users $U = \{u_1, u_2, ..., u_D\}$ and their trajectories, we first initialize their groups. Then, we employ an iterative refinement framework to further group users based on their mobility patterns. During each iteration, we generate a better representation of each group's mobility pattern and then assign every user to a more appropriate group. The major steps are described as follows:

Step 1: Initialization. In this step, instead of assigning every user to a group randomly, we vectorize every user and use clustering algorithms like k-means to preliminarily cluster users according to their mobility patterns, so as to reduce the time cost in the subsequent iteration. Specifically, we first train an HMM $H_u = (\pi_u, A_u, B_u)$ for each user u by the aforementioned learning method, whose parameters reflect the mobility pattern. Then, we simply reshape (π_u, A_u, B_u) to an E-dimensional vector, where $E = K \times (1 + K + M)$. After vectorizing all users, we stack all vectors to be a matrix $I_{D\times E}$. Since E is a very large integer, we leverage PCA to reduce the dimensionality and then apply k-means to get the initial user groups $\varphi = \{g_1, g_2, ..., g_G\}$ where G is a user-defined group number. Next, for each $g \in \varphi$, we train an HMM H_g which represents the mobility pattern of group g. Finally, we get an initial HMM ensemble $\Phi^{(0)} = \{H_g^{(0)} \mid g \in \varphi\}$.

Step 2: Grouping. For each user u, let us denote the set of u's trajectories as J_u, in which the j-th trajectory is T_u^j. Based on the latest HMM ensemble $\Phi^{(t)}$, we assign u to a new group g^{t+1} by a way of voting, considering all trajectories in J_u. Users belong to g make up a new set $S_g^{t+1} = \{u \mid \forall j, v_u(g) \geq v_u(g_j), 1 \leq j \leq G\}$.

The voting value that u gives to g is:

$$v_u(g) = \sum_{j=1}^{|J_u|} \frac{1}{Z_u^j} p(T_u^j \mid g; \Phi^{(t)}) \tag{4}$$

where $p(T_u^j \mid g; \Phi^{(t)})$ is the probability of observing T_u^j given group g's group-level HMM $H_g^{(t)}$ and can be computed by the Forward Scoring algorithm of HMM, and $Z_u^j = \sum_{i=1}^{G} p(T_u^j \mid g_i; \Phi^{(t)})$ is the normalization term.

Step 3: Updating. For each g in φ, we utilize the trajectories belong to group g to train an HMM $H_g^{(t+1)}$ by the aforementioned learning method. Thus, we generate a new ensemble of HMMs $\Phi^{(t+1)} = \{H_g^{(t+1)} \mid g \in \varphi\}$.

Step 4: Iteration. After updating the ensemble of HMMs, we go back to step 2 for further iterations. The algorithm will stop when the number of reassigned users is lower than a preset value (e.g., 1% of total user number D).

At last, users with the similar mobility are grouped, upon which we get group-level features along with personal-level and overall-level features.

[6] https://en.wikipedia.org/wiki/Additive_smoothing

3.3 Feature Extraction

In this section, we introduce features from the point of views of multiple granularities and multiple views. Given a stop-point x belongs to a trajectory T' of a user u. We searched out venues $V = \{v_1, v_2, ..., v_n\}$ that are within distance d from location $x.l$ as candidates. For each candidate venue v_i, we extract a series of features, mainly based on travel history of different granularities.

Multi-granularity Features. As mentioned in Sect. 3.1, we leverage multi-granularity features to alleviate the data sparsity issue. The kinds of feature granularities can be divided into multi-user and multi-venue granularities. For multi-user granularities, we generate features for a given user u from the travel history of all users, u himself and the group g he belongs to. And for multi-venue granularities, we generate features for a given venue v from visit history of v itself (X_v) and v's category ($X_{v.c}$) which can be further sub-divided into $X_{v.c.c_1}$, $X_{v.c.c_2}$ and $X_{v.c.c_3}$. The symbol X_v stands for the set of stop-points where each stop-point x's POI $x.p$ is the same as the given venue v, and similarly $X_{v.c.c_i}$ stands for the set of stop-points where each stop-point x's POI category $x.p.c.c_i$ is the same as the given venue category $v.c.c_i$.

By combining these two kinds of granularities, we can extract history-related features from the user u's visit history to v and $v.c$, his group's visit history to v and $v.c$ and all users' visit history to v and $v.c$, respectively.

Multi-view Features. Given a stop-point x and a candidate venue v, we mainly consider four types of features: the spatial relationship - $F_s(x, v)$, the temporal relationship - $F_t(x, v)$, the spatio-temporal relationship - $F_{st}(x, v)$ and the contextual relationship - $F_c(x, T', v)$, where T' is the trajectory x belongs to. In the following, we only introduce features generated from X_v due to the limitation of space. Note that the feature generated from $X_{v.c}$ is similar.

(1) Spatial features of $F_s(x, v)$ reflect visit preference related to geographic factors, including 2 parts:

- Revised distance $dist_{rv}(x, v)$. We observe distance $d(v.l, x.l)$ sometimes mislead annotation. For example, if a user gets off a vehicle at the roadside ($x.l$) and get into a large supermarket v represented geographically by only one point $v.l$, $d(v.l, x.l)$ may be bigger than distances from $x.l$ to many other venues. Thus, we extract typical stop-points (e.g., colored circles in Fig. 1) of venues by applying Affinity Propagation Clustering to historical stop-points of venues. Then we consider the distance between $x.l$ and the closest typical stop-point as the distance feature.
- Spatial conditional frequency $freq_d(x, X_v) = |\{x_h \in X_v | d(x.l, x_h.l) < \Delta'_d\}|$: the number of historical stop-points in X_v around $x.l$. Spatial adjacent points may have similar visit preference.

(2) Temporal features of $F_t(x, v)$ consist of 6 temporal conditional frequencies, which reflect different visit preference under different temporal conditions. We first define a temporal condition set $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_6\}$. Given time t' of a historical stop-point and t of an unannotated stop-point, Λ contains: 1) $POD(t') = POD(t)$, where

$POD(t)$ maps time t to a time period tp; 2) t' is in the weekend if t is in the weekend, otherwise t' is in the weekday; 3) $DOW(t') = DOW(t)$, where $DOW(t)$ maps time t to day of week; 4) t' and t are on the same day, which reflects the situation (e.g., a sales promotion) on that day of venue v; 5) t' is within 30 days before and after t, which reflects recent situations of venue v; and 6) t' is within 90 days before and after t, which reflects long-term situations of venue v. Then, for each condition $\lambda(t', t) \in \Lambda$, we get a conditional frequency $freq(X_v, x, \lambda) = |\{x_h \in X_v | \lambda(x_h.t, x.t)\}|$ as a feature, which stands for the number of historical stop-points in X_v satisfying temporal condition λ.

(3) Spatio-temporal features of $F_{st}(x, v)$.

- Spatio-temporal conditional frequency $freq_{st}(x, X_v)$=$|\{x_h \in X_v \mid d(x.l, x_h.l) < \Delta'_d \wedge I(x.t, x_h.t) < \Delta'_t\}|$: the number of historical stop-points in X_v which satisfy both spatial and temporal constraints, where $I(t, t')$ computes the time interval between t and t' in the span of 24 hours.

(4) Contextual features of $F_c(x, T', v)$. The relevance between x and v is related to the trajectory T'. For example, if x's former point $\overleftarrow{x}$ and another stop-point $\overleftarrow{x_h}$ have similar characters (e.g., spatial adjacent or corresponding to the same POI category), x and x_h are likely to visit the same venue if they are spatial adjacent (e.g., a transition from housing areas to a specific company followed by a group of colleagues). Contextual features contains 2 parts:

- User activity inferred by group-level HMM. Instead of using Viterbi Algorithm directly, we consider the activities already known in trajectory T'.
- Contextual conditional frequencies. We first define a contextual condition set $\Omega = \{\omega_1, \omega_2, \omega_3\}$. Given two stop-points x and x', Ω contains: 1) $d(x.l, x'.l) < \Delta'_d$; 2) $x.p = x'.p$, which is the POI limit; and 3) $x.p.c = x'.p.c$, which is the POI category limit. Then, for each $\omega(x, x') \in \Omega$, we get a frequency feature about former points $freq(X_v, x, \omega) = |\{x_h \in X_v | \omega(\overleftarrow{x_h}, \overleftarrow{x})\}|$ and a frequency feature about later points $freq(X_v, x, \omega) = |\{x_h \in X_v | \omega(\overrightarrow{x_h}, \overrightarrow{x})\}|$, both of which stand for the number of historical stop-points in X_v satisfying contextual condition ω.

3.4 Venue Ranking

Our method has three variants to rank venues by relevance:

- *CAUG-LR*. For a stop-point, this method uses Logistic Regression [9] to do binary classification for every candidate venue. Then venues are ranked by probabilities.
- *CAUG-GBTree*. It uses XGBoost [10] to replace Logistic Regression, measuring the performance in tree based model.
- *CAUG-Rank*. It uses a learning-to-rank algorithm named LambdaMart to give ranked lists, which is a boosted tree version of LambdaRank based on RankNet [11].

4 Experimental Study

In this section, we introduce the datasets in experiments, methods for comparison, metrics for evaluation and performance of methods.

Table 2. Summary of datasets

Data Source	Domain	Entities	Description
Transporation	Orders Users	9,835,247 885,246	Sample over 600 thousand travel records whose stop points are labeled with venues from 6807 stable users (with at least 40 orders) and 3017 cold users (with no more than 40 orders) for following experiments
POI	Overall Venues	169,612 72,604	Include 18 first class categories, 245 second class categories and 819 third class categories

4.1 Setup

Datasets. In the work, we use a real operational transportation dataset collected by UCAR within 6^{th} Ring Road of Beijing during Jun. 1, 2015 to Aug. 31, 2016. Also, we make use of the POI dataset of AutoNavi for annotation and use the first-class category to make up the activity. The details are summarized in Table 2.

Algorithms. We implement three variants of our method, i.e. *CAUG-LR*, *CAUG-GBTree* and *CAUG-Rank*, which use the same features proposed in this paper. We also implement 2 straightforward methods (i.e., *Dist* and *DistR*) and 3 distinct methods (i.e., *MRF*, *DistHMM* and *LSRank*) as baselines:

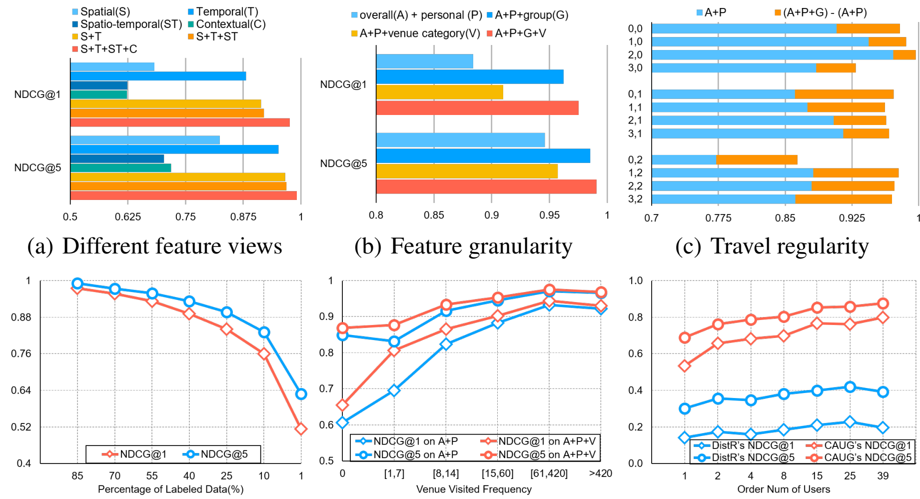
- *Dist*. It directly matches the closest venue to each stop-point.
- *DistR*. This method uses the revised distance $dist_{rv}$ to match the closest venue.
- *MRF*. A method based on Markov random field model [5], considering the distance factor, spatial and temporal regularity of human mobility.
- *DistHMM*. A method based on HMM [4], considering the distance and the historical consecutive transitions between POIs. Note that if there is no venue within 500m for a stop-point in the user's states, we will use the *Dist* model to annotate.
- *LSRank*. A learning-to-rank-based local search framework [6]. It uses features including the popularity of venues, distance between stop point and venues, temporal preference to venues and personal preference to venues.

Metrics. We use Normalized Discounted Cumulative Gain (*NDCG*)[7] to measure whether the ground truth venue appears in the output ranked list weighted by the position. For each annotation, $NDCG_k = \sum_{i=1}^{k} \frac{rel_i}{log_2(i+1)}$, where $rel_i \in \{0, 1\}$, is the binary relevance of the result at position i. The higher the ground truth venue is ranked in our list, the higher the NDCG score will be, and a value of 0 indicates the ground truth is not in the Top-k ranked list. In this paper, $NDCG@k$ is the mean of the $NDCG_k$ for each annotation and $NDCG@1$ is the same as the Top-1 accuracy.

4.2 Results

In this section, we first use CAUG-Rank to evaluate the performance given different feature views, different granularities and different percentages of labeled data. Then we compare the effectiveness of our method with other baselines. Based on the actual situation of our dataset, the percentage of the labeled data is set to 55% for an experiment of the cold-start user and 85% for feature and comparison experiments.

[7] https://en.wikipedia.org/wiki/Discounted_cumulative_gain.



(a) Different feature views (b) Feature granularity (c) Travel regularity

(d) Percentages of labeled data (e) Venue visited frequency (f) Amount of personal order

Fig. 4. Experiment Results

The Impact of Feature Views. To evaluate the impact of different feature views on our methods, we test the effectiveness of our method on one view at a time and gradually combine them all together. The result in Fig. 4(a) shows temporal features are more effective than spatial features (0.88 versus 0.68 in NDCG@1). By combining temporal, spatial and spatio-temporal features, the NDCG@1 reaches over 0.92. Furthermore, the integration of contextual features further enhances the performance apparently, demonstrating it improves annotation in a different aspect with spatio-temporal factors.

The Impact of Feature Granularities. To evaluate the effectiveness of group and venue-category granularities, we start with features of both the overall and personal granularities without considering venue categories, then we add group and venue-category granularities. As Fig. 4(b) shows, when we only use overall and personal granularities, the NDCG@1 only reaches 0.88. By introducing the group and venue-category granularities, the NDCG@1 rises to about 0.97, which verifies the sparsity issue is alleviated. And we can find the group granularity is more effective than the venue-category granularities (0.96 versus 0.91). To further observe the impacts of group and venue-category granularities, the following two sub-experiments were conducted:

- *Impact on travel regularity.* We first define travel regularity by a 2-dimensional vector (L_{order}, L_{POI}), where L_{order} represents the level of order quantity and L_{POI} is the level of POI quantity. Intuitively, users who have visited various venues (L_{POI} is high) in his few number of travel records (L_{order} is low) are less regular in their mobilities. The result in Fig. 4(c) shows adding group granularity (G) to the model which just use overall and personal records (A+P) improves the performance, especially for those irregularly-traveling users.
- *Impact on visited frequency.* We test venues with different visited frequency. As shown in Fig. 4(e), popular venues have higher annotation accuracies. By adding venue-category granularity (V) to the model which just uses overall and personal records (A+P), the performance improves, especially for novel and cold venues.

Table 3. Experiment results of different methods

Methods	Dist	DistR	MRF	DistHMM	LSRank	CAUG-LR	CAUG-GBTree	CAUG-Rank
NDCG@1	0.105	0.381	0.438	0.428	0.820	0.960	0.973	0.975
NDCG@5	0.222	0.562	/	/	0.913	0.985	0.990	0.991

The Impact of Labeled Data Percentage. As shown in Fig. 4(d), the performance does not drop notably until the percentage of labeled data diminish to under 10%. This result provides an interesting message: for those who annotate locations manually, they could only annotate a small proportion of the records and leave the work to CAUG.

Comparison with other Methods. To evaluate the performances of different models, we compared 8 models. The result in Table 3 shows the performance of our method outperforms all others. NDCG@1 of Dist, DistR, MRF, DistHMM are under 0.50 because they oversimplify the factors influencing annotation. Since LSRank doesn't consider contextual information and multi-granularity, its NDCG@1 only reaches 0.80. CAUG-LR, CAUG-GBTree, and CAUG-Rank all annotate accurately (over 0.95), which reveals the effectiveness of our features. Besides, we compare DistR and our model for cold users without enough historical data. The result in Fig. 4(f) shows as the number of personal records grows, the overall NDCG gradually improves. Specifically, for a new user who only has one travel record, the NDCG@1 of DistR only reaches 0.10, while our model is above 0.50 owing to the contextual information and multi-granularity features enrich the information for modeling user preference.

5 Related Work

Researchers proposed numerous methods [2, 5, 8, 12, 13] for semantic annotation of mobility records according to their specific tasks or data.

Studies on traditional mobility data like GPS traces [2, 4] mainly consider the distance between the context location and the location of the user. Without considering the history of individual's movement, they cannot provide personalized annotation. Spinsanti et al. [14] add some manually defined semantic rules to calculate the possibility of a person visiting a POI. However, rules cannot be well-rounded. Yan et al. [4, 15] take transition relation of human movements into account and propose a method using HMM to annotate trajectories. Nevertheless, they ignored the temporal influences and only annotate locations with categories of POI other than specific POIs.

Due to the development of mobile Internet, massive geo-tagged social media (GeoSM) data combining texts with locations are generated. Wu et al. [3] and Zhang et al. [8] utilize noisy and sparse GeoSM data to discover proper activities or text tags of locations. Since sources of annotations are texts over the space, methods proposed by them cannot be applied to our problem. Some researchers utilize check-in data to study location annotation in [5, 6, 12, 16], which is similar to our work. However, because check-in records are usually not continual, both of them neglect mobility transitions.

Moreover, location annotation is similar to the problem of recommending a POI to a user at one location [17, 18]. Nevertheless, POI recommendation aims to rank those potentially interesting but previously unvisited venues higher, while location annotation does not follow this principle.

6 Conclusion

In this paper, we have proposed CAUG, an effective method to provide personalized location annotation through spatial, temporal and contextual factors, which can be generalized to many kinds of mobility data (e.g., locations collected by mobile apps). By constructing the sequence of locations, we take advantage of the transition relations among contextual mobility records to help annotate. We use HMM to model the users' mobility and group users based on their mobility patterns. With the help of user groups and venue categories, we effectively alleviate the issue of data sparsity. Experiments on a real-world dataset show that CAUG outperforms other 5 baseline models.

In the future, possible improvements can be reached through integrating more travel, contextual and user information. Besides, the grouping step is currently time-consuming and simply solved by parallel computing, which should be improved.

References

1. Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., et al.: Semantic trajectories modeling and analysis. ACM Computing Surveys (CSUR) **45**(4) (2013) 42
2. de Graaff, V., de By, R.A., van Keulen, M.: Automated semantic trajectory annotation with indoor point-of-interest visits in urban areas. In: SAC. (2016) 552–559
3. Wu, F., Li, Z., Lee, W.C., Wang, H., Huang, Z.: Semantic annotation of mobility data using social media. In: WWW. (2015) 1253–1263
4. Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K.: Semitri: a framework for semantic annotation of heterogeneous trajectories. In: EDBT. (2011) 259–270
5. Wu, F., Li, Z.: Where did you go: Personalized annotation of mobility records. In: CIKM. (2016) 589–598
6. Lian, D., Xie, X.: Learning location naming from user check-in histories. In: SIGSPATIAL. (2011) 112–121
7. Rabiner, L., Juang, B.: An introduction to hidden markov models. IEEE ASSP Magazine **3**(1) (1986) 4–16
8. Zhang, C., Zhang, K., Yuan, Q., Zhang, L., Hanratty, T., Han, J.: Gmove: Group-level mobility modeling using geo-tagged social media. In: KDD. (2016) 1305–1314
9. Berger, A.L., Pietra, V.J.D., Pietra, S.A.D.: A maximum entropy approach to natural language processing. Computational linguistics **22**(1) (1996) 39–71
10. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: KDD. (2016) 785–794
11. Burges, C.J.: From ranknet to lambdarank to lambdamart: An overview. Learning **11** (2010)
12. Lian, D., Xie, X.: Mining check-in history for personalized location naming. TIST **5**(2) (2014) 1–25
13. Nishida, K., Toda, H., Kurashima, T., Suhara, Y.: Probabilistic identification of visited point-of-interest for personalized automatic check-in. In: UbiCOMP. (2014) 631–642
14. Spinsanti, L., Celli, F., Renso, C.: Where you stop is who you are: understanding people's activities by places visited. In: BMI workshop. (2010)
15. Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K.: Semantic trajectories: Mobility data computation and annotation. TIST **4**(3) (2013) 49
16. Shaw, B., Shea, J., Sinha, S., Hogue, A.: Learning to rank for spatiotemporal search. In: WSDM. (2013) 717–726
17. Liu, Y., Pham, T.A.N., Cong, G., Yuan, Q.: An experimental evaluation of point-of-interest recommendation in location-based social networks. VLDB **10**(10) (2017) 1010–1021
18. Bao, J., Zheng, Y., Mokbel, M.F.: Location-based and preference-aware recommendation using sparse geo-social networking data. In: SIGSPATIAL. (2012) 199–208