# Trip purpose – Survey

## I. *Overview*:

This short survey summaries some existing work on trip purpose inference. As the data used in most of these papers is unlabeled, such as taxi GPS trajectories and cellular data, the main approach is to exploit surrounding POIs' information, mainly categories, to infer trip purposes. As a result, there are two ways to explain purposes of trips: based on both origin and destination points, or on only destination points. In the former methods, categories or category distribution of surrounding regions of starting and ending points are extracted and used to interpret the trip's purpose. Other factors, such as time, are also utilized for better prediction. For example, the trip from resident area to office area at 9AM is likely a trip to work. Example papers from this category are [1], [2], [8]. The second approaches only use the information of destination points to define trip purpose. Note that, many papers on activity prediction/recommendation are also included in this survey since they define categories of POIs as activities, and hence belong to the second approach.

## II. *Survey*:

### 1. Human Mobility Synchronization and Trip Purpose Detection with Mixture of Hawkes Processes (KDD'17)

a) *Problem*: Detect the trip purpose of trajectories.

b) *Method*: In this paper, a trajectory consists of 3 elements: the origin region, the destination region and arrival time, which can help us to infer the purpose of the trajectory. For example, if the origin region contains mostly residents, the destination region contains mostly companies or office buildings, and the arrival time is from 7AM and 9AM, then the purpose of the trajectory is likely to work. In other words, the trip purpose is determined by the functions (or topics) of origin and destination regions, and the arrival time. As a result, a hierarchical topic model is proposed with two latent variables: trip purpose and POI topic. To generate a trip, first the trip purpose $m$ is drawn from a multinomial distribution $\pi$. Then, two POI topics $z_o$ and $z_d$ for origin and destination regions, respectively, are generated from $\Phi_m$. Finally, each POI $w^o$ in the origin region are generated from $\beta_{zo}$ (similar for the destination region). To exploit the time arrival factor, authors integrate **Mixture of Hawkes Process** into the model, as they assume that if two destination regions have similar time arrival pattern, trajectories to them are likely to have the same purpose.

c) *Data*: 70M taxi trajectories from NYC. From the experiments, we first can infer the meaning of each POI topic (such as nightlife-related, dining-related, office topics) by considering its top POIs. Then by checking topic distributions of origin and destination regions, we can infer a trajectory purpose *m*.

| TOPIC 1 | prob. | TOPIC 2 | prob. | TOPIC 3 | prob. | TOPIC 4 | prob. | TOPIC 5 | prob. |
|---|---|---|---|---|---|---|---|---|---|
| Bar | 0.1884 | Chinese Rest. | 0.1286 | Bar | 0.0933 | Office | 0.3331 | Clothing Store | 0.0995 |
| Home | 0.0953 | Italian Rest. | 0.0913 | Italian Rest. | 0.0565 | General Entertain | 0.1035 | Cafe | 0.0693 |
| Nightclub | 0.0571 | Asian Rest. | 0.0541 | American Rest. | 0.0442 | Hotel | 0.1023 | Office | 0.0574 |
| Event Space | 0.0495 | Tea Room | 0.0481 | Wine Bar | 0.0373 | Building | 0.0869 | Coffee Shop | 0.0535 |
| Cocktail Bar | 0.0495 | Bar | 0.0472 | Sushi Rest. | 0.0319 | Event Space | 0.0593 | Cosmetics Shop | 0.0419 |
| Lounge | 0.0495 | Spa or Massage Parlor | 0.0416 | Mexican Rest. | 0.0306 | Sandwich Place | 0.0376 | General Entertain | 0.0408 |
| Speakeasy | 0.0471 | Salon or Barbershop | 0.0403 | Lounge | 0.0297 | Hotel Bar | 0.0342 | French Rest. | 0.0406 |
| Breakfast Spot | 0.0382 | Vietnamese Rest. | 0.039 | Pizza Place | 0.0278 | Lounge | 0.0342 | High Tech Outlet | 0.0388 |
| French Rest. | 0.0334 | Art Gallery | 0.0342 | Coffee Shop | 0.0256 | Other Outdoors | 0.0298 | Salon or Barbershop | 0.0368 |
| Boat or Ferry | 0.0316 | Cocktail Bar | 0.0316 | Salon or Barbershop | 0.0256 | Performing Arts Venue | 0.0289 | Miscellaneous Shop | 0.0331 |
| **TOPIC 6** | **prob.** | **TOPIC 7** | **prob.** | **TOPIC 8** | **prob.** | **TOPIC 9** | **prob.** | **TOPIC 10** | **prob.** |
| College Acad. | 0.0808 | Park | 0.1343 | Art Gallery | 0.2773 | American Rest. | 0.1023 | Home | 0.2005 |
| Food Truck | 0.0756 | Other Outdoors | 0.1 | Park | 0.1021 | Deli or Bodega | 0.0619 | Building | 0.0591 |
| University | 0.0653 | Scenic Lookout | 0.0767 | Other Outdoors | 0.0892 | Office | 0.0569 | Deli or Bodega | 0.0471 |
| College Library | 0.0639 | General Travel | 0.0753 | Cafe | 0.0555 | Pizza Place | 0.0464 | Pizza Place | 0.0442 |
| General College/University | 0.0573 | Building | 0.074 | Playground | 0.049 | Bar | 0.0448 | Laundromat or Dry Cleaner | 0.0342 |
| College Dorm | 0.0565 | Airport | 0.074 | Automotive Shop | 0.0386 | Food Truck | 0.0434 | Coffee Shop | 0.0317 |
| Cafe | 0.0499 | Harbor or Marina | 0.0616 | Event Space | 0.033 | Sandwich Place | 0.0392 | Drugstore or Pharmacy | 0.0291 |
| Plaza | 0.0485 | Taxi | 0.0534 | Strip Club | 0.0265 | Coffee Shop | 0.0346 | Chinese Rest. | 0.0256 |
| Park | 0.0382 | Government Building | 0.048 | Sculpture Garden | 0.0241 | Burger Joint | 0.0326 | Mexican Rest. | 0.0236 |
| College Classroom | 0.0374 | Seafood Rest. | 0.0343 | Plaza | 0.0233 | Cafe | 0.0307 | Apartment Building | 0.0206 |

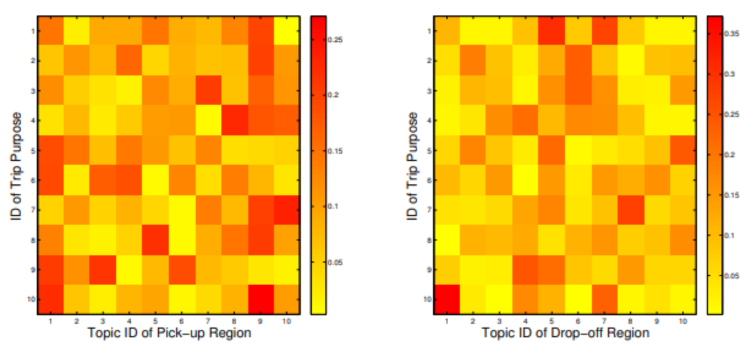
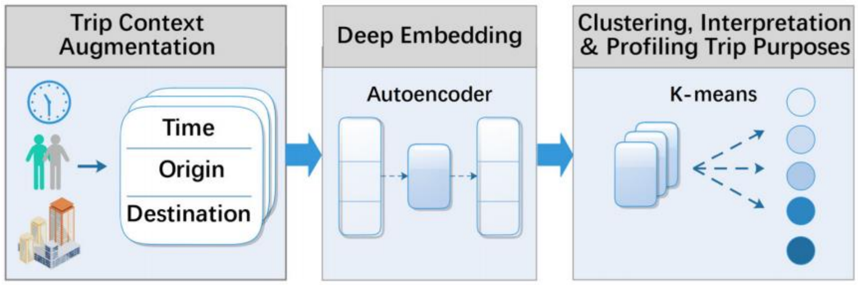



**Figure 4: POI Topic distribution over latent trip purposes for origin and destination**

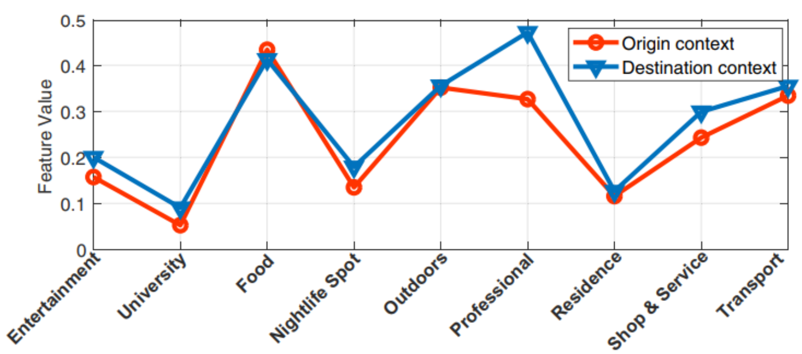
## 2. Trip2Vec: a deep embedding approach for clustering and profiling taxi trip purposes (Personal and Ubiquitous Computing' 18)

a) *Problem*: Infer the trip purpose of taxi trips.

b) *Method*: the proposed method contains two steps: clustering and profiling taxi trips. The first task aims to group taxi trips with similar travel purposes, and the second task is to interpret the trip purposes. The model consists of 3 modules:




The first component, trip context augmentation, extracts different kinds of context information, such as time, origin and destination, that are related to the trip as the trip features. Subsequently, trip features are passed through the second component, namely deep embedding, to learn their latent representation. Finally, trips are clustering based on their latent vectors using K-mean algorithm. To interpret the trip purpose of each cluster, authors compare features between origin and destination context of trips in the cluster, together with the average hour time in the temporal context. For example, if the average arrival time is 1PM, and the difference between contexts as shown in below figure, trips in this cluster is related to "Working" after lunch.




c) *Data*: 13M taxi trips and 220K check-ins in NYC. However, the evaluation is solely based on human evaluation on visualization.

**3. TripImputor: Real-Time Imputing Taxi Trip Purpose Leveraging Multi-Sourced Urban Data (IEEE Transactions on Intelligent Transportation Systems'18)**

a) *Problem*: predict the POI category (i.e., activity) that a user is likely to take when he/she gets off a taxi.

b) *Method*: Authors propose a two-phase framework, namely TripImputor. In the first stage, POIs are clustered based on spatial proximity. Each cluster is called a candidate activity area (CAA). In the second step, they propose a Bayesian model to predict the probabilities of a user taking one of 9 activities (i.e., POI categories) for the drop-off point. First, given the location of the drop-off point, we select top-k nearest CAAs within walkable distance. Then, for each chosen CAA, the probability to visit it $P(CAA_i|(x,y))$ is computed based on its distance to the drop-off point. Next, the probability of taking an activity $a_j$ given the drop-off point, time and $CAA_i$ is computed: $P(a_j \mid (x,y), t, CAA_i)$. In this probability, the main component is $P((x,y) \mid a_j, t, CAA_i)$ which is computed based on fraction of $a_j$ in $CAA_i$ and the attractiveness of $CAA_i$ at time $t$.

$$\begin{aligned} &P((x, y)|a_j, t, CAA_i) \\ &\quad \propto \frac{numberof\,POIs(a_j, CAA_i)}{numberof\,POIs(CAA_i)} \times A_i(t) \\ &s.t. \sum\nolimits_{j=1}^{n} P((x, y)|a_j, t, CAA_i) = 1 \end{aligned}$$

Finally, for a taxi trip $(x,y,t)$, the probability of passengers taking a given activity $a_j$ after getting off the taxi can be approximated by the following equation

$$\begin{aligned} &P(a_j|(x, y), t) \\ &\quad \propto P(CAA_i|(x, y)) \times P(a_j|(x, y), t, CAA_i) \\ &s.t. \sum\nolimits_{j=1}^{n} P(a_j|(x, y), t) = 1 \end{aligned}$$

The second phase of the framework is to speed up the prediction step for the real-time response.

c) Data: road network, Foursquare check-ins, and taxi GPS trajectory in NYC. To evaluate the framework, they use the travel purpose survey data at the same city as groundtruth. They first map the Foursquare categories with ones in the survey data, and then can evaluate the performance of the proposed method.

**4. Fine-Grained Urban Event Detection and Characterization Based on Tensor Cofactorization (IEEE Transactions on Human-Machine Systems'17)**

a) *Problem*: Detect urban event by identifying unusual irregularities of human movement.

b) *Method*: Big events in cities, such as public concerts, festival parades and sport games, usually cause unusual irregularities of human movement. The paper proposes a tensor cofactorization-based framework on mobility data for event detection. Specifically, two tensors are constructed from the data. The *crowd mobility tensor A* is built from bike trip records, where *A(r,t)* stores the number of bikes arriving at region *r*, during time span *t*. The social activity tensor *B* is built from check-ins data, where *B(r,t,c)* stores the number of check-ins in region *r*, during time span *t* of category *c*. Then, two tensors are decomposed simultaneously such that the region factors and time factors are shared by both tensors. As a result, we can approximate tensor *A* by the number of basic patters: $= \sum_{k=1}^{K} \hat{A}^k$ . To detect mobility outliers, we arrange these basic pattern components into a vector: $\boldsymbol{\theta}(r,t) = (\theta_1, \theta_2, \ldots, \theta_K)$, where $\theta_k = \hat{A}^k(r,t)$. Based on those vectors, we can detect mobility outliers, corresponding to outlier vectors, by some clustering algorithms. For example:

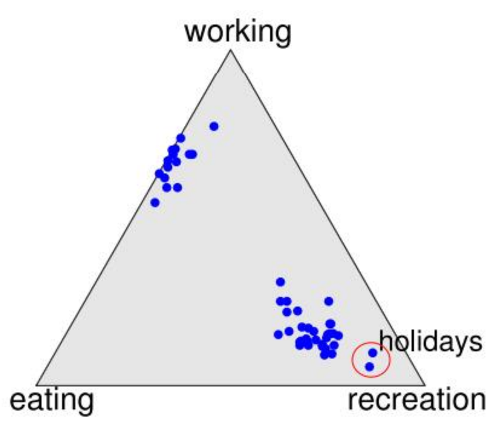



Fig. 8. Ternary plot of the three example basic pattern vectors in the National Mall region from 9:00 to 10:00 over six consecutive weeks (05/28/2012–07/08/2012).

## 5. Travel Purpose Inference with GPS Trajectories, POIs, and Geo-tagged Social Media Data (BIGDATA'17)

a) *Problem*: Trip purpose inference

b) *Method*: The paper proposes a HMM-based model to predict user activity (purpose) on each of his/her stop in a trajectory. Particularly, the model is formulated as following equation and figure:

$$\begin{aligned} P(a,c,l) &= P(a_0)P(c_0|a_0)P(l_0|c_0) \\ &\cdot \left(\prod_{i=1}^{N} P(a_i|a_{i-1})P(c_i|a_i)P(l_i|c_i)\right) \end{aligned}$$

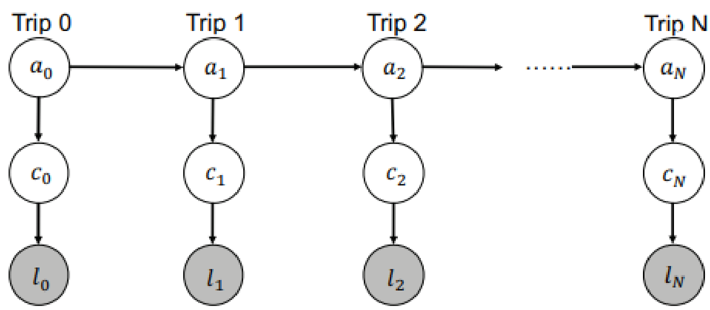



Figure 2: The Dynamic Bayesian Network

where, $P(a_i|a_{i-1})$ is the probability of the activity $a_i$ given previous activity $a_j$, $P(c_i|a_i)$ is the probability of the visited POI category given current activity $a_i$, and $P(l_i|c_i)$ is the probability of chosen location $l_i$ given currently chosen POI category $c_i$. Note that, list of activities $\{a_i\}$ and category $\{c_i\}$ are predefined, but unobserved in the data (as shown in the figure). The probability $P(l_i|c_i)$ is defined as follows:

$$\begin{aligned} P(l_i|c_i) &\propto \frac{P(c_i|l_i)}{P(c_i)} \\ &\propto \frac{P_{POI}(c_i|l_i)P_{tweet}(c_i|l_i)}{\sum_i P_{POI}(c_i|l_i)P_{POI}(c_i|l_i)} \times \frac{1}{P(c_i)}. \end{aligned}$$

where, $P(c_i|l_i)$ denotes the POI category distribution given a geo-location $l_i$, and is determined by two factors: the functionality distribution $P_{POI}(c_i|l_i)$, and the popularity distribution $P_{tweet}(c_i|l_i)$. The former one is obtained from the nearby POIs, and the latter measures the popularity of POI categories in the surrounding area of $l_i$, which obtained by extracting mentions of nearby POIs from tweets.

c) *Data*: GPS trajectories in California with activity labels.

## 6. Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs (IEEE Transactions on Systems, Man, and Cybernetics'15)

a) *Problem*: Given a user's current context (location and time), infer his/her activity (POI category).

b) *Method*: Based on users history, the proposed model models two types of user preference: spatial and temporal activity preferences. For the spatial activity preference, for each user, the model first finds his/her *personal functional regions* (PFRs), where the user usually performs certain specific activities. In other words, in a region, if the user visits POIs with the same category (such as restaurants) significantly more than other categories, the regions is a PFR of the user. Figure below is an example

Nightlife Region

| Category | Check-in count | Probability |
|---|---|---|
| Sports bars | 15 | 0.5 |
| Jazz bars | 9 | 0.3 |
| French Restaurant | 3 | 0.1 |
| Fast food restaurant | 2 | 0.0667 |
| Movie theater | 1 | 0.0333 |


Fig. 4. Example of a nightlife PFR of a user.

Then, given the user's current location *l*, the spatial activity preference $\Psi_{u,l}$ is calculated as: $\Psi_{u,l} = \sum_{r_u \in R_u} \psi_{u,r_u} \cdot w_{l,r_u}$, where $R_u$ is the set of PFRs of the user, $\psi_{u,r_u}$ is the activity preference distribution of user *u* in PFR $r_u$, and $w_{l,r_u}$ is the weight of PFR $r_u$ which is based on the distance between $r_u$ and *l*.

For the temporal activity preference, the model applies the tensor decomposition method, where tensor dimensions represent users, time and activities. The temporal activity preference is then is defined as: $\Psi_{u,t} = \{\hat{y}_{u,t,c} |\ c \in C\}$, where $\hat{Y}$ is the normalized recovered tensor.

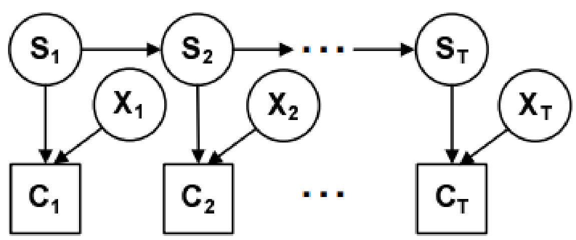
In the prediction step, one of two preferences is chosen to make the prediction, based on their performance given the current user's context (i.e., location and time).

c) *Data*: Foursquare and Gowalla. However, the data size is quite small: ~1K users, 10K-60K POIs.

**7. What's Your Next Move: User Activity Prediction in Location-based Social Networks (SDM'13)**

a) *Problem*: predict user's next POI's category and then next POI.

b) *Method*: Paper proposes an HMM-based model to model user check-ins sequences. In the model, besides hidden states and observations (categories), to exploiting temporal and spatial information, they introduce a feature vector X containing temporal and spatial information into the model. The model is shown as in below figure.




Here, the emission probability is redefined as:

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

The above model does not consider user preference. To model this factor, authors use the following approach: first, for each user, they construct his/her category distribution vector based on his/her history d. Then users are clustered based on their category vectors, and finally an HMM model is trained for each cluster.

After predicting new category $C_{t+1}$, we can predict then next POI by considering nearby POIs of this category.

c) *Data*: 1M Gowalla check-ins and 900K sequences.

**8. Spotting Trip Purposes from Taxi Trajectories: A General Probabilistic Model (Transactions on Intelligent Systems and Technology'17)**

a) *Problem*: infer trip purpose from taxi trajectory.

b) *Method*: Authors applied LDA to the proposed model to infer taxi trip purpose. Specifically, each POI can be considered as a document where its surrounding POIs are words. As a result, the origin and destination points are represented by POI Topics, and the purpose of a trip can be represented as linkage between the two POI Topics. For example, a trip from the "Home"-related POI Topic to "Work"-related POI Topic can be naturally inferred as for working purpose, while if the trip happens in the early morning, the inference for the purpose would be more accurate. As a result, they proposed LDA-based model for inferring trip purpose as below figure.

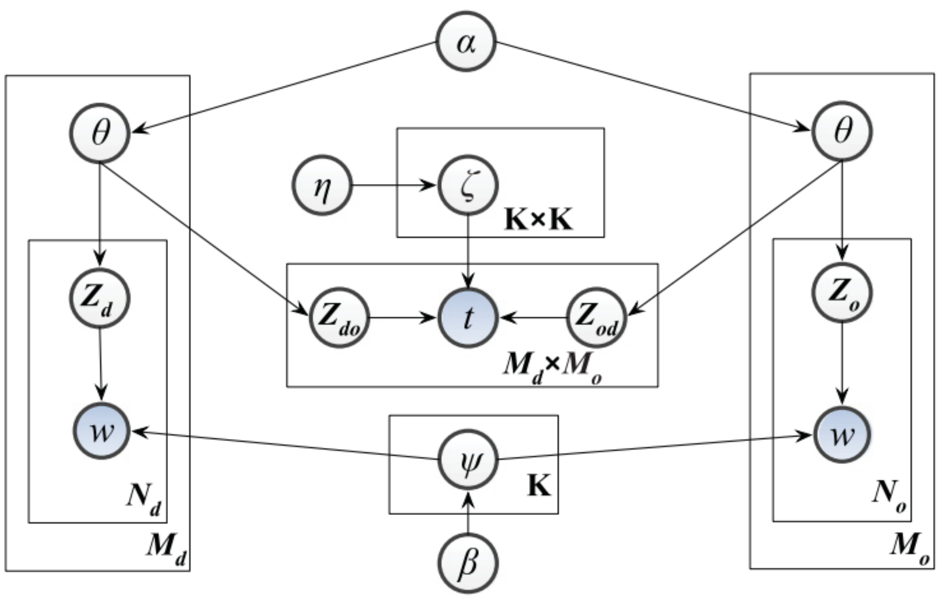



Fig. 1. POI Link Model.

Specifically, for each O-D pair, the POI topics $z_{od}$ and $z_{do}$ are generated from the multinomial distribution over topics θ, respectively, and then the pairwise topics together generate the time slots for the trip from the multinomial distribution over time specified by the topic pairs ξ, revealing how the POI distribution of the pickup and drop-off point interact with different time periods.

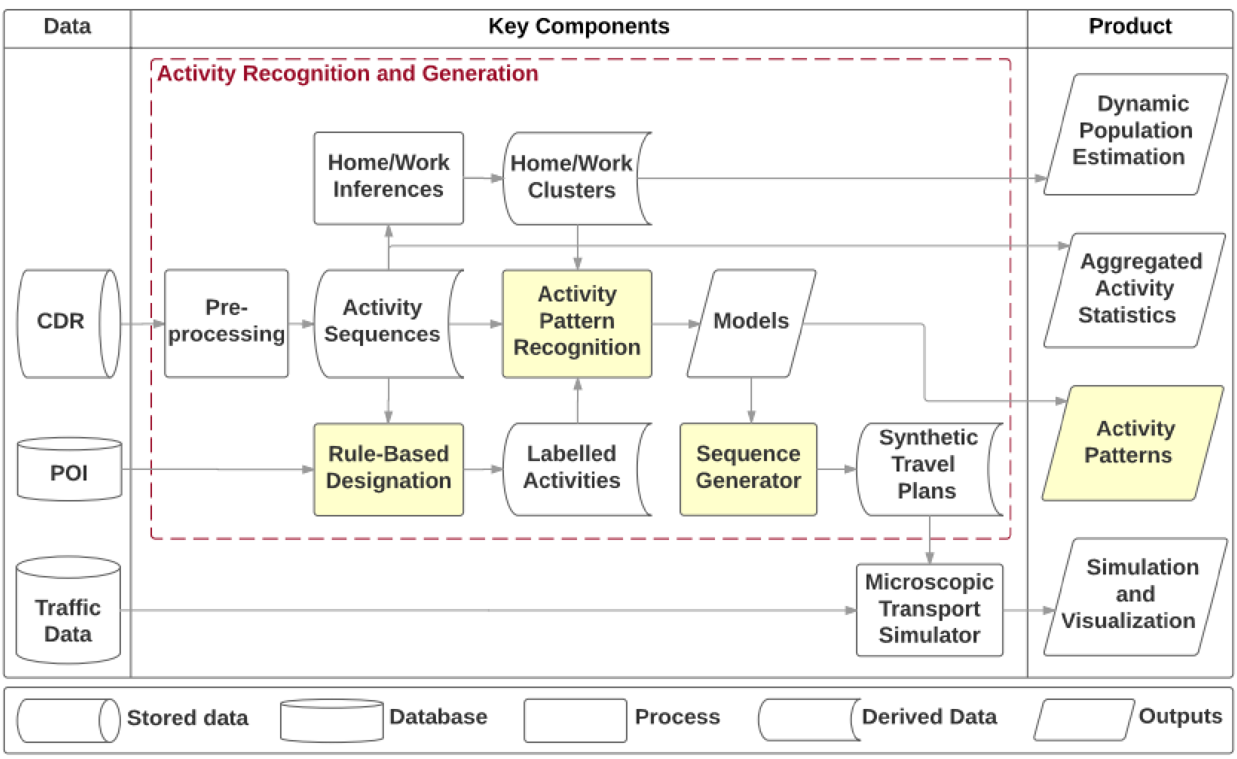
To infer trip purpose, for each origin *o* and destination *d*, we assign the POI Topic with the largest probability from its Topic distribution θ, that is $k_o^* = argmax_k \theta_o^k$, and $k_d^* = argmax_k \theta_d^k$. So, each trip is categorized into $K^2$ clusters (topics), and together with the time distribution of the topic pairs, we can further explain the purpose of each taxi trip.

c) *Data*: 188K GPS trajectories from NYC, 66K POIs from Foursquare.

**9. A Generative Model of Urban Activities from Cellular Data (IEEE Transactions on Intelligent Transportation Systems, 2018)**

1. *Problem*: Recognize the activities behind the cellular data.

2. *Method*: Yin et al. build a pipeline for recognize the activities (e.g., home, work, food) given a sequence of call detail records (CDRs). The pipeline includes a module to convert the raw data into stay points, and an input output hidden Markov model to learn the transitions between activities under different temporal contexts.




**Stay point detection** – Due to positioning errors and connection oscillations, the raw data are first preprocessed by spatial clustering to detect stay points as potential activities. Then, each raw CDR sequence is represented by a sequence of clusters. To label the activity of each cluster, the authors first take the location that a user stays for 50% of daytime (nighttime) as work place (home).

**Activity model** – an input output Markov model is proposed. The hidden states are treated as latent activities/purpose. The model takes the temporal context (weekday/weekend, hours in a day, number of hours spend at work) and output the observable information (e.g., distance from stay point to home/work, stay duration, previously visited or not).

3. *Data*: The model is trained on 20,000 users from a major mobile carrier in the US. To evaluate the quality of the model, the authors collect a ground truth data using a short range distributed antenna system which provides higher spatial resolution and the POI dataset from Google places. They apply several rules to label the activities.

| Activity | Duration (hours) | Start hour | Context | Location category |
|---|---|---|---|---|
| Lunch | 0.25 - 1 | 11-12 | | Food |
| Dinner | 0.25 - 2 | 17-18 | | Food |
| Shop | 0.25 - 1 | 7-9<br>14-15<br>20-21 | Home based or during evening commute | Shop |
| Transport | $< 0.25$ | | Commute | Transport |
| Recreation | 1-4 | 7-21 | Home based or during evening commute | Recreation |
| Personal | any | 7-21 | | Personal |
| Travel | any | any | | Out of the region |

**10. Location Prediction Through Activity Purpose: Integrating Temporal and Sequential Models (PAKDD 2017)**

1. *Problem*: Predict the next location based on predicted activity purpose.

2. *Method*: Liao et al. propose to predict the next location of a user via two steps. In the first step, they predict the next activity purpose (POI category) by combining a hidden Markov model $p_u(c_n|c_{n-1})$ and a temporal model $p_u(c_n|t_n)$ for each user $u$. The temporal model is obtained by tensor decomposition on a user-time-activity tensor built from the historical data. The authors choose one of the two models to use by building a binary classifier based on some handcrafted features extracted from the data. By applying the features extracted from the current check-in sequence, the output $y$ of the binary classifier specifies the model to use:

$$P_u(c_i|\tau^u_{\hat{t}}, \hat{t}) = \begin{cases} P_u(c_i|c_n), & \text{if y=1} \\ P_u(c_i|\hat{t}), & \text{if y} = -1 \end{cases}$$

In the second step, to predict the location given the category using kernel density estimation.

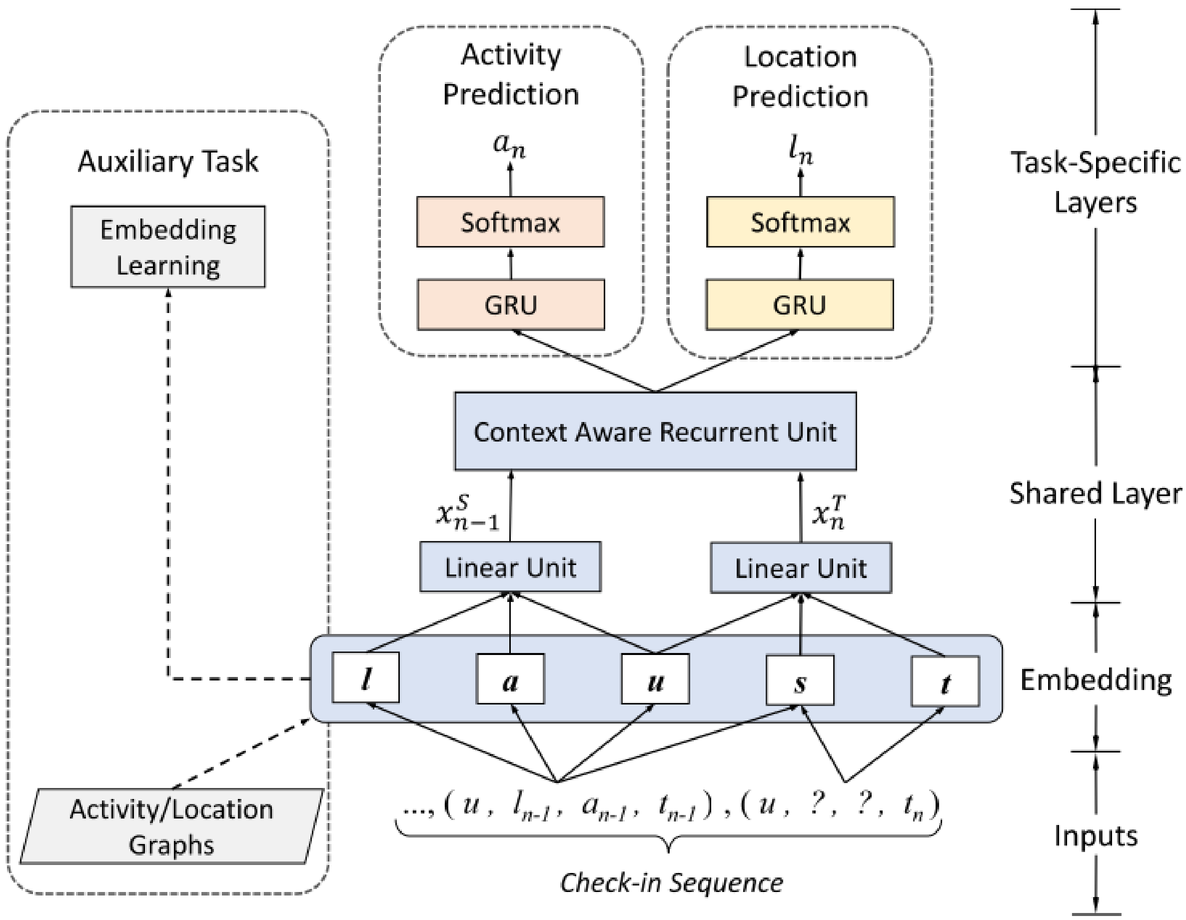
c) *Data*: Public check-in data from New York city and Tokyo. They 251 POI categories are used as activities.

**11. Predicting Activity and Location with Multi-task Context Aware Recurrent Neural Network (IJCAI 2018)**

a) *Problem*: Predict next location and activity given the historical check-in sequence.

b) *Method*: Liao et al. claim that the relationship between user's activities and locations is more complicated than that one decides the other. Then, they propose the concept of spatial-activity topic to model the correlation between activity and location. They develop a context aware recurrent unit to forget the previous sequential information if the time difference is large and propose a deep learning framework with recurrent neural network to learn the latent spatial-activity topics. They learn the embeddings of location and activities as auxiliary task to exploit the spatial distance among locations and the correlation between activities and locations.

To predict the next location and activity, they pass the historical sequence to the RNN to infer the next hidden state (the activity-location topic). At the last step, the model will output the activity and location for the next timespan given the current activity and location.




c) *Data*: Public check-in data from New York city and Tokyo. They extract 400 POI categories as activity keywords.

**12. Regions, Periods, Activities: Uncovering Urban Dynamics via Cross-Modal Representation Learning (WWW 2017)**

a) *Problem*: Recognize the activity given the time, and location.

b) *Method*: Zhang et al. propose a cross-modal embedding model to jointly learn the representation of location and temporal hotspot as well as words. They first detect spatial and temporal hotspots by performing mean shift procedure. Based on the hotspots, they define the spatial and temporal neighborhoods and propose two embedding algorithms to learn the embeddings of location, time and words.

ReconEmbed – This embedding algorithm follows the same motivation as word embedding, which aims to reconstruct as much records as possible.

GraphEmbed – This algorithm construct a heterogeneous graph with location, time and word as nodes and apply node embedding approach to learn the representation.

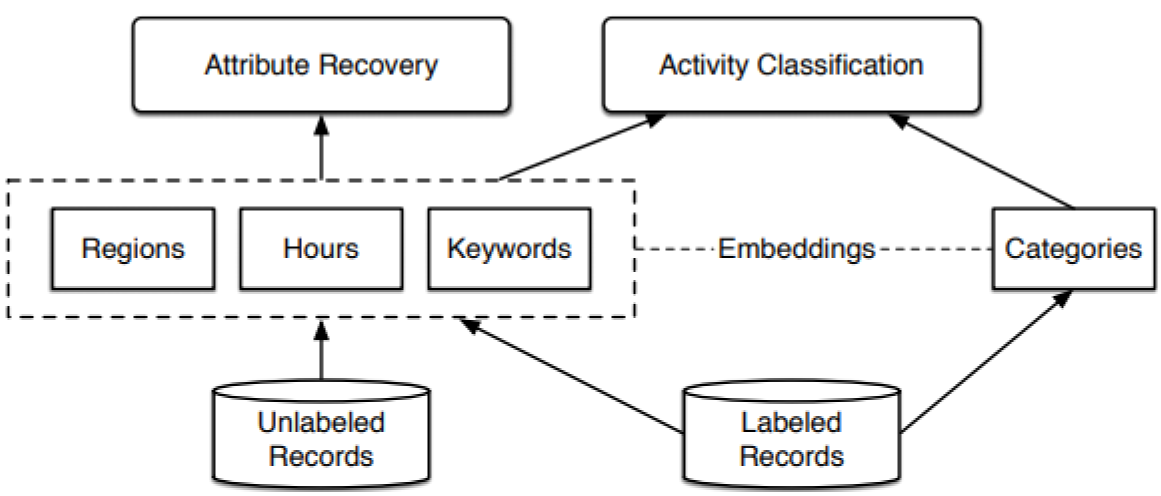
With the representation learned, give a location and time, they can infer the most possible activities each of which is a collection of words.

c) *Data*: 1.1 million geotagged tweets from LA. 0.6 million check-ins from 4SQ

**13. ReAct: Online Multimodal Embedding for Recency-Aware Spatiotemporal Activity Modeling (SIGIR 2017)**

a) *Problem*: Detect topical trajectory patterns.

b) *Method*: Zhang et al. propose an online learning approach to learn the multimodal embeddings of location, time and text by considering the recency of the records. To learn the embeddings, they consider two objectives for unlabeled data and labeled data respectively. For unlabeled data, they update the embeddings by maximizing the likelihood of recovering attributes of the records (time, location, words). For labeled data, they update the embeddings by maximizing the likelihood of predicting the correct category given time, location and words.



When new records come, model updates is conducted by a constraint learning process:

$$J_{\mathcal{R}_\Delta} = -\sum_{r\in\mathcal{R}_\Delta}\sum_{i\in r}\log p(i|r_{-i}) + \lambda \sum_{i\in L,T,W,C}\|\mathbf{v}_i - \mathbf{v}_i'\|^2$$

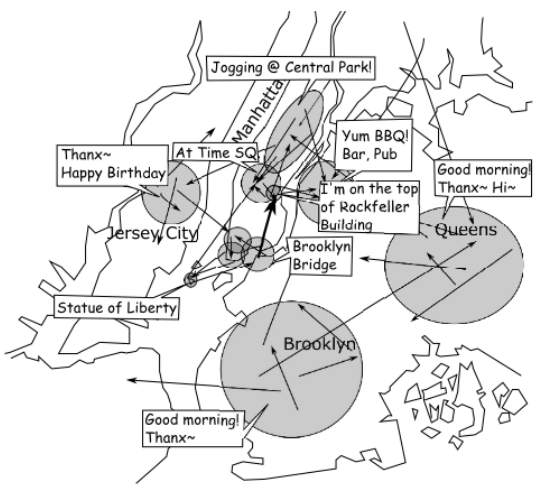
Where $\|v_i - v_i'\|$ penalize the updates that has very large variance to the previous model. In addition, the authors sample the historical data for the training with a decaying sampling rate to consider the freshness of the records.

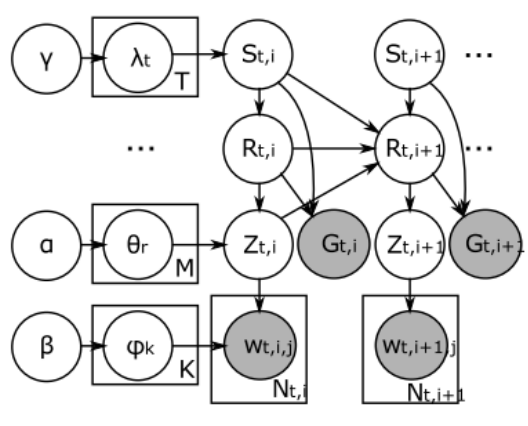
c) *Data*: 1.1 million geo-tagged tweets from LA, 1.2 million geo-tagged tweets from New York. Around 10% tweets in both datasets can be linked to POIs in Foursquare.

**14. TOPTRAC: Topical Trajectory Pattern Mining (KDD 2015)**

a) *Problem*: Detect topical trajectory patterns.

b) *Method*: Kim et al. propose a topic model to learn latent regions and topics from semantic-rich user trajectory data (e.g., geo-tagged tweets) and mine frequent topical transition patterns.




The proposed method first learn the topic $Z$ and latent region $R$ and a switch variable specifying whether the point is related to semantic regions $S$ for each check-in point by a probabilistic model. Then, the authors apply a frequent pattern mining approach on the sequence of learned $(Z, R, S)$. Given the words and location of a sequence, the model can infer the topics (can be considered as purpose) and regions for each check-in point.

c) *Data*: 9070 sequences of geo-tagged tweets in New York sCity and 809 trajectories of geo-tagged tweets in San Francisco.