LoCaTe: Influence Quantification for Location Promotion in Location-based Social Networks

Ankita Likhyani
Indraprastha Institute of Information Technology, Delhi, India
ankital@iiitd.ac.in

Srikanta Bedathur
IBM-Research Lab, Delhi, India
sbedathur@in.ibm.com

Deepak P.
Queen’s University Belfast
Northern Ireland, UK
deeaksp@acm.org

1 Introduction

With GPS1 becoming pervasive within smartphones, location-based social networks (LBSNs) such as Foursquare and Facebook places have gained popularity. These networks allow users to share their check-in information with their friends enabling LBSN users to be aware of their friend’s check-ins. This, combined with other factors, can influence the check-in preferences of users [Cho et al., 2011; Ye et al., 2011]. Quantifying the user-to-user influence—often modelled probabilistically [Goyal et al., 2010; Zhang et al., 2012]—is an essential building block in understanding the effects of information diffusion in LBSNs.

Influence maximization [Bouros et al., 2014; Li et al., 2014; Wu and Yeh, 2013] in social networks addresses the problem of finding a set of users who have a strong influence in the social network; these users are potentially good seed users to run promotion campaigns that target to maximize the reach of the campaign. Research interest in LBSNs has sparked interest in using geo-locations for influence maximization, leading to the location promotion problem [Zhu et al., 2015]. The location promotion problem is of significant importance for launching effective campaigns to help businesses gather more customers. This task instantiates the influence maximization task on a specified target location/venue (e.g., a particular restaurant), with the intent of finding a set of seed users who are well-positioned for the promotion of their business [Zhu et al., 2015]. Once a set of seed users is identified, it can be used to issue targeted special offers.

From the perspective of location promotion, the worth of a user - or the user’s “seedness” - is quantified as the number of users she can influence to visit a location, after her visit. The necessary step for the location promotion problem is thus a method to assess the likelihood that other users would visit the location following the candidate seed user’s visit. Influence quantification can factor in a variety of information that a LBSN offers: (i) geographic features: user’s mobility over different locations, (ii) semantic features: type/category of location (e.g., restaurant, cafe), (iii) social correlation: the relationship between users in the social network, and (iv) temporal correlation: the degree to which a user’s movement is correlated with another user’s. Previous work on influence quantification for location promotion has mostly focused on modelling geographic features and social correlation [Zhu et al., 2015]. Studies on semantic features such as category have been limited primarily since datasets containing such information have been scarce [Gao et al., 2012; Cho et al., 2011]; such deficiencies have been recently addressed by methods such as approximate spatial joins [Likhyani et al., 2015]. Thus, because of unavailability of category information intrinsically, the use of location semantics have not been explored in previous work. In this work, we join datasets from multiple sources to make location category usage possible.

The temporal correlation of users’ behavior has been modeled previously in online social networks, but not in LBSN as we do in this paper. The socially induced followship based on temporal correlation has been of interest in LBSN studies in other contexts [Pham and Shahabi, 2016].

In this paper, we address this research gap and make two-fold contribution: first, exploitation of the hitherto unexplored information, that of location categories, for influence quantification; second, we develop a novel model LoCaTe

---

1 https://en.wikipedia.org/wiki/Global_Positioning_System
that incorporates:
1. **Location affinity**: The mobility patterns of users that hold cues to whether they frequent the proximity of the target location.
2. **Category affinity**: The affinity of a user to the category or type of the location.
3. **Temporal correlation**: The temporal correlation of movements between the user and the candidate seedsset, thus modelling time-conditioned social followship.

As is the case of any influence quantification method, our model easily fits into any location promotion setting [Li et al., 2014; Zhu et al., 2015]. We empirically evaluate our approach over both publicly available real-world LBSN data as well as our own long-term crawl of Foursquare check-ins spanning more than one year. Overall, the proposed LoCaTe model can outperform state-of-the-art methods by more than 50% in AUC (Area Under the Curve of ROC [Bradley, 1997]).

### 2 Related Work

While influence maximization has been a well-studied problem (e.g., [Kempe et al., 2003; 2005; Chen et al., 2010; Goyal et al., 2010]), the geo-seeded instantiation motivated by LBSNs has gathered only recent attention [Li et al., 2014; Zhu et al., 2015; Bouros et al., 2014; Zhang et al., 2012; Pham and Shahabi, 2016]. Apart from the location promotion problem where we start with a specific target location, there have been studies on region promotion, where the target is a larger geo-region [Bouros et al., 2014]. Users’ geo-location affinities have been modeled by either associating one specific geo-location with each user (usually the most frequently one visited by the user) [Li et al., 2014; Zhang et al., 2012] or a set of geolocations or only the social network structure [Bouros et al., 2014; Zhu et al., 2015]. In a similar way, the user-user pairwise influence propagation probabilities are estimated either using just the (social) network structure [Bouros et al., 2014; Li et al., 2014; Zhang et al., 2012] or taking into consideration the seed location/region [Zhu et al., 2015]. To the best of our knowledge only the recent work in [Pham and Shahabi, 2016], have looked at defining user-user pairwise influence in spatio-temporal context, but for identifying followship. For our empirical evaluation, we compare against the most recent work by Zhu et al. [Zhu et al., 2015], that associates a set of locations for each user and considers the influence between two users to be dependent on the location.

**User Mobility Models**: We make use of user mobility models in our method drawing inspiration from earlier work on characterizing user behavior in LBSNs. Since LBSN data provides a trail of user’s locations, it provides a rich data platform for studying user mobility patterns; such patterns are of interest for tasks such as location prediction and personalized recommendation. In literature, mobility models that mine spatial patterns based on generative models [Gao et al., 2012], Gaussian distributions [Cho et al., 2011] and kernel density estimations [Lichman and Smyth, 2014] have been particularly successful. Accordingly, we use the kernel density mobility model [Lichman and Smyth, 2014] to model and exploit user-location affinities in our method for LBSN location promotion. The baseline technique from [Zhu et al., 2015], on the other hand, makes use of a distance-based mobility model, DMM, in their influence quantification method.

### 3 Problem Statement

Now we provide a formal definition of influence quantification problem in an LBSN for location promotion. Table 1 lists a set of notations that will be used. We model a location/venue\(^2\) as having a fixed geographic coordinate as well as a set of categories associated with it. This allows for modeling of locations such as movie multiplexes that would screen movies as well as contain eateries.

**Definition 1 (Location Promotion)** Given an LBSN \(G\), a target location \(L\), whose category set is \(C_L\), the location promotion problem is to select a small set of seed users \(S\), \(S \subseteq U\), who can lure other users to the target location \(L\) well. The hyperparameter \(\tau\) indicates the desired cardinality of \(S\).

Influence quantification, the task of quantifying the likelihood of a user visiting the location given a visit to the same location by another (i.e., seed) user, forms an important building block for location promotion problem as shown in [Zhu et al., 2015]. Given a user-user influence model, the location promotion problem reduces to finding a set of seed users which can collectively influence a large set of users to visit the location in focus.

**Definition 2 (Influence quantification.)** Influence quantification: Given an LBSN \(G\), a target location \(L\), a seed-user \(u\) (usually a user who has previously visited \(L\)), the influence quantification problem is to quantify the likelihood \(P(\text{on} \to \text{visit} | G)\), the likelihood that any user \(v\) among \(u\’s\) connections is likely to visit \(L\).

As mentioned earlier, influence quantification models typically consider the affinity of the user \(v\) to location \(L\) independent of the influence from other users, and the influence of the seed user \(u\) on \(v\) in this decision.

**Evaluation**: Our evaluation framework follows earlier work in this area (e.g., [Zhu et al., 2015]): from a dataset of LBSN check-ins, we decide on a cut-off timestamp such that there is a sizeable amount of check-ins before and after the cut-off. All data prior to the cut-off timestamp is used as training data to learn the model. The remaining data forms the test set against which the effectiveness of the learned model is measured. Consider a particular instance of the influence quantification problem for location \(L\) and a seed-user \(u\): the

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(G)</td>
<td>A location based social network</td>
</tr>
<tr>
<td>(U)</td>
<td>Set of users in (G)</td>
</tr>
<tr>
<td>(E)</td>
<td>Set of connections from (u_i) to (u_j) s.t. (u_i, u_j \in U) and (u_i \neq u_j)</td>
</tr>
<tr>
<td>(t)</td>
<td>A location specified by a triple ((x, y, C_t)), where (x, y) correspond to geo-coordinates and (C_t) to category set of (t)</td>
</tr>
<tr>
<td>({u, t, t, C_t})</td>
<td>A check-in record of user (u) at time (t) at location (t) that has a category set (C_t)</td>
</tr>
<tr>
<td>(M_u)</td>
<td>set of check-in records ({u, t, t, C_t})</td>
</tr>
<tr>
<td>(L)</td>
<td>A set of locations</td>
</tr>
<tr>
<td>(C)</td>
<td>A set of categories</td>
</tr>
</tbody>
</table>

Table 1: Notations used in this paper

\(^2\)We use location and venue interchangeably. Though venue might be more appropriate, location is also used in literature.
influence quantification output would be an ordered list of user’s connections, ordered in the decreasing (non-increasing) order of estimated likelihood to visit $l$. This list can be cut-off using a threshold $p$ to identify a set of users who are deemed to be highly likely to visit $l$ - this set forms the predicted set, $PS(l, u, p |G)$. The ground truth activated set, $A(l, u)$, is the subset of user’s connections who have actually visited $l$ after the cut-off timestamp (i.e., from the test set). The match between $PS(l, u, p |G)$ and $A(l, u)$ measured at various values of the threshold $p$ quantifies the goodness of the influence quantification method employed. Any measure of match between sets can be aggregated over all users (i.e., by iterating $u$ over the set of LBSN users) to get a single goodness value for the combination $[l, p]$. We will use the ROC curve (generated by varying $p$) to compare our method against baselines in our empirical evaluation.

4 LoCaTe Influence Quantification

We now outline our influence quantification method, LoCaTe, that estimates $P_{l,u}(v|M)$, a scoring that captures the likelihood that the user $v$ from $u$’s connections would visit the location $l$ using the check-in records in the training part, denoted as $M$. Figure 1 shows the framework of LoCaTe. LoCaTe combines information from three kinds of features:

$$P_{l,u}(v|M) = \alpha P_{L}(v, l(M)) + (1 - \alpha) P_{C}(v, C(l|M)) \times T(u \rightarrow v|M),$$

where, $P_{L}(v, l(M))$ models the affinity of $v$ to location $l$, and $P_{C}(v, C(l|M))$ models the affinity of $v$ to the categories that are associated with the location $l$ (denoted as $C(l)$). These two terms are interpolated using an interpolation parameter $\alpha$. Further, $T(u \rightarrow v|M)$ captures the temporal correlation between users $u$ and $v$, independent of the location. The first two terms quantify users affinity for the location using mobility and category information respectively and are combined using a weighted sum. The third term quantifying location-agnostic user-user socially induced temporal affinity is merged using a product. Thus, the final scoring ensures that users who are strong on both location and temporal aspects score much higher than others.

$P_{l,u}(v|M)$ ranges between $[0,1]$, since we normalize the scores obtained for a list of influenced users $v = (v_1,..,v_m)$ keeping the seed user $u$ and target location $l$ fixed. Thus

$$\sum_{v}P_{l,u}(v|M) = 1.$$ 

The usage of Location affinity, Category affinity and Temporal correlation lends the name to our method.

4.1 Location Affinity

The mobility of each user is typically restricted to a few key locations, which would typically include the location of stay and work [Cho et al., 2011]. Thus, a user has an inherent preference for some geo-locations. We use kernel density estimates to model mobility patterns and location affinity of users. Kernel density estimates are robust to sharp transitions in spatial densities that human mobility witnesses, especially in contexts involving travels that take users far away from their usual location of residence [Cho et al., 2011; Lichman and Smyth, 2014].

The affinity of $v$ to $l$ based on her own check-in history is modeled as the kernel density estimate (KDE) that quantifies the average weighted similarity between $l$ and each location $v_j$ has checked-in using hyper-parameter $k$,

$$P_l(v, l|M_v) = f_{KD}(l|M_v, k) = \frac{1}{|M_v|}\sum_{j=1}^{|M_v|} K_{j,k}(l, \ell_j)$$

$K_{j,k}(\cdot, \cdot)$ estimates the similarity between locations as inversely related to the Euclidean distance between them:

$$K_{j,k}(\ell, \ell_j) = \frac{1}{2\pi h_{j,k}} \exp \left(-\frac{1}{h_{j,k}} ||\ell - \ell_j|| \right)$$

Here, $h_{j,k}$ is a location-dependent scalar factor that is set to be the Euclidean distance of $\ell_j$ to it’s $k^{{th}}$ nearest neighbor, thus ensuring that the similarity computation is sensitive to differential densities of locations in urban and rural areas.

Mixture of Kernel Density models:

To offset for sparsity issues in determining the location affinity using just the user’s check-in records, we use corpus smoothing by interpolating the location affinity of a user with those across all users, yielding our final formulation:

$$P^k_l(v, l|M) = \beta_v f_{KD}(l|M_v, k) + (1 - \beta_v) f_{KD}(l|M, k),$$

where, $\beta_v$ is a user-specific mixing weight, determining the relative influence between the user and global models. We will denote this as $P_l(\cdot)$ when the value of $k$ is clear.

Training:

We have two parameters that need to be estimated. First, the hyper-parameter $k$ is estimated as the value that maximizes the likelihood of check-ins in a chosen validation set, $V$ (which is a held-out part of the check-ins before the cut-off timestamp, but not included in the training). Thus, we set $k$ to the value that maximizes the following:

$$k = \arg \max_{k'} \sum_{(v, \ell, \cdot)} \log \left(P^k_l(v, l|M) \right)$$

The distribution of log-likelihood across various values of $k$ are shown in Table 2; accordingly, we chose $k = 5$ for usage in our method. Second, we choose the user-specific interpolation weights $\beta_v$ as the value that maximizes the likelihood of their check-ins in the training data itself. We do not use the validation set here since there are users who do not have any check-ins in the validation set.
Table 2: Log-likelihood at different values of k

<table>
<thead>
<tr>
<th>k</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fsqa'16</td>
<td>-2.032</td>
<td>-1.804</td>
<td>-1.704</td>
<td>-1.640</td>
<td>-1.670</td>
<td>-1.687</td>
<td>-1.722</td>
<td>-1.744</td>
<td>-1.817</td>
</tr>
<tr>
<td>Fsqa'11</td>
<td>-2.711</td>
<td>-2.640</td>
<td>-1.821</td>
<td>-1.850</td>
<td>-1.879</td>
<td>-1.897</td>
<td>-1.915</td>
<td>-1.869</td>
<td>-1.821</td>
</tr>
<tr>
<td>Gowalla</td>
<td>-1.978</td>
<td>-1.896</td>
<td>-1.847</td>
<td>-1.804</td>
<td>-1.825</td>
<td>-1.854</td>
<td>-1.877</td>
<td>-1.890</td>
<td>-1.931</td>
</tr>
</tbody>
</table>

Figure 2: Category wise check-in Distribution

4.2 Category Affinity

Locations/Venues often record correlated check-in behavior across LBSN users. For example, a restaurant might be better off targeting a user who frequently checks in to food places due to the correlation across various categories of food joints. As an example, consider two users in Figure 2 represented by the word cloud of the categories of their checked-in locations (larger font indicates higher frequency); User A evidently exhibits affinity towards visiting restaurants while user B prefers gym and fitness centers. We use topic modeling to identify such higher-level contexts, and exploit it to model the user-category affinity term, \( P_C(v, C_t|M) \).

Latent Dirichlet allocation [Blei et al., 2003] models semantic matching between text documents by learning topics, each of which is a probability distribution over the set of words. LDA ensures that words that are semantically related would have high probabilities associated with the same topic(s). In our adaptation of LDA for modeling topical contexts across check-in categories, each user \( v \) is treated as a document constructed as a bag of categories \( C_v \) (i.e., each category as a word) of checked-in locations. These documents across the users in the population form a document corpus. We apply LDA on this document corpus, to learn topics which are probability distributions over categories. We then use the learnt topics to estimate the user’s affinity to the set of categories associated with the location of interest:

\[
P_C(v, C_t|M) = \sum_{Z \in \text{Topics}(M)} P(C_t|Z) \times P(v|Z),
\]

where Topics(M) is the set of topics learned as described, and Z represents a topic from the topic-set learned. \( P(v|Z) \) and \( P(C_t|Z) \) quantify how well the category distribution for Z matches against those of the check-ins of v and the categories of t respectively. High values of \( P_C(v, C_t|M) \) are achieved when the user’s category distribution and that of the location under consideration are correlated with the same set of topics.

4.3 Temporal User Correlation

We now turn our attention to the temporal correlation term, \( T(u \rightarrow v|G) \), that quantifies the extent of influence that \( u \) has over \( v \). This primarily accounts for the socially induced followship in our Influence Quantification model. The task at hand is to quantify the chance that \( v \) will follow \( u \) in checking-in to a location, such that \( (u, v) \in E \). Towards modeling this, we first analyze the behavior of general inter-arrival times of users in the LBSN at a given location, without considering whether they are connected to each other in the LBSN network or not; we call this the time lag distribution across userbase. The analogous time lag distribution across connections considers the distribution of the time duration elapsed between two users who are connected to each other, visiting the location in question. A visual depiction of the time lags that go into either of these distributions appears in Figure 3. We collect these time lag distributions across all locations in the LBSN and analyze their frequency distribution using a histogram. As expected, the general across userbase time lag distribution follows a power law distribution, as illustrated in Figure 4(a). However, the across connections time lag distribution in 4(b) does not quite follow a power law distribution despite exhibiting a monotonic decay with increasing values of time lag. It may also be noted that the across connections data is much sparser than across userbase; this is so since there are a significantly fewer number of occurrences of connected users visiting the same location. Drawing cues from the trends across the two time lag distributions, we choose to model the time lag distribution between users using an exponential distribution, given it’s popularity in similar contexts [Pham and Shahabi, 2016].

Modeling using exponential distribution

According to the exponential distribution modelling, the weight associated with any value of time lag, denoted \( \delta t \), would be quantified as the following:

\[
p(\delta t) = \lambda_t e^{-\lambda_t \delta t}
\]

We set \( \lambda_t \) to be the mean time lag between check-ins by connected users:

\[
\lambda_t^{-1} = \frac{1}{\text{avg}(|t_2 - t_1| : \exists(u, t, v, t_{1, t}, \cdot) \in M \land \exists(v, t, 2, \cdot) \in M \land (u, v) \in E)}
\]

where the \( (u, ., t, ., \cdot) \) implies that we consider all check-ins by \( u \) at time \( t \) irrespective of the location of the check-in or the set of categories associated with the location. This feeds into our user correlation estimate \( T(u \rightarrow v|G) \) which is modelled...
as the cumulative weight of v checking in at a location visited by u after a time lag of any \( t \geq t_{u,v}^{\min} \):

\[
T(u \rightarrow v|G) = \int_{t_{u,v}^{\min}}^{\infty} \lambda e^{-\lambda t} dt = e^{-\lambda t_0} \prod_{t_{u,v}^{\min} \leq t < \infty} e^{-\lambda t}
\]

As indicated above, we set \( t_{u,v}^{\min} \) to be the earliest time that \( v \) has checked in after \( u \) at the same location, according to training data; this ensuring that \( T(u \rightarrow v|G) \) reflects the extent of correlation between \( u \) and \( v \), since \( T(u \rightarrow v|G) \) would have a high value for those user pairs where the latter follows the former (temporally) closely.

5 Experiments

5.1 Experimental Setting

We tested over 5 datasets as shown in table 3, of which Fsq’16 is the one that we collected using Twitter and FourSquare APIs, and rest are publicly available datasets [Cho et al., 2011; Gao et al., 2012]. The publicly available datasets do not provide category information, and moreover, the locations are also anonymized, they only expose the latitude, longitude information of the location. A recent work uses approximate spatial joins across checked-in locations and map data (available at Foursquare itself) to infer category information for each check-in and the data is made available by the authors [Likhyani et al., 2015]. Such joins provide a large set of categories for each location as there could be multiple types of locations (e.g., cafe, hospital) at the same geo-location. Although another large collection of Foursquare check-in data with category information is released, it does not have the social graph of users essential for the location promotion problem addressed in this paper [Yang et al., 2016]. To the best of our knowledge, there does not exist any publicly available dataset that contain category information of check-in locations along with the social graph.

Data Collection: We use Twitter for getting check-ins of users, because user’s check-in information in Foursquare can only be accessed from her own social circle, and it is not available publicly. However, Foursquare users can choose to post their check-ins via Twitter when they check in at a place. Hence, we capture check-ins by crawling Foursquare-tagged tweets from the Twitter Public Stream API \(^3\) and REST API\(^4\). The Foursquare-tagged check-in tweet crawled contains a unique URL that points to a Foursquare web page containing the geographical information of this check-in location. We parse this web page to get check-in location along with the timestamp. Further to obtain category of this location, we pass this location to FourSquare API\(^5\) and get access of the location information. Lastly, to acquire the social information of users (i.e., friendship links), we again use Twitter, because Foursquare does not provide public access to user’s friends list. Thus, we acquired friendship ties that Foursquare users have among them on Twitter, where they are publicly available. To extract the check-in details of friends we again crawl their tweets in the same manner as above. While the resulting social graph is not identical to the original Foursquare graph, it encompasses the subset of users who are on Twitter and have linked their Twitter and FourSquare accounts. In our experiment, we consider the users who have at least 10 check-ins. Some key statistics of the dataset are shown in table 3. Fsq’16 is the dataset that we curated, rest are public datasets for which we collected only the category information.

Test Set: For each dataset, we assign a cut-off timestamp, the data prior to it is used for training and rest of the check-ins for testing. The cut-off timestamp is chosen such that 80% of total check-ins are used for training. The target locations are identified as locations where a user’s check-in precedes its follower’s check-ins and both the check-ins are made on or after the cut-off timestamp. The user who checks-in first is the seed user and its followers who checks-in after him, are the activated users (by the seed user). Section 2 describes ground truth construction set of activated users and also the evaluation. Table 3 shows the number of test cases (A(f, u)), and cut-off timestamp for each dataset. Note that there is a high variability in the number of categories per location across the datasets, thus enabling the empirical evaluation to cover scenarios across a wide spectrum of semantic information availability.

Parameter Estimation: We also construct a validation set in the same manner as test set is built from the training set for learning the parameters \( \beta \) and \( \alpha \); mixing the kernel density estimations for spatial density and the topic model for category information, respectively. We use EM algorithm to learn this. Table 4 shows values of \( \beta \) and \( \alpha \); learned for different datasets. The parameter \( Z \) (number of topics) is set to 50.

Implementation: We coded all the models including the baselines (DMM\(^6\)_Basic and DMM\(^6\)_Social), in Java. For KDE we made use of the source code available at UCI Datalab website (http://www.datalab.uci.edu/resources) and for LDA based topic model we made use of the source code available at http://mallet.cs.umass.edu/topics-devel.php. We ran the code on a 6-core 2.5GHz Intel Xeon CPU with 64GB of RAM. The source code and the datasets used are available publicly\(^6\).

5.2 Comparative Evaluation

We compare our proposed model LoCaTe with two state of the art users’ mobility based influence quantification models, viz., the DMM\(^6\)_Basic and the DMM\(^6\)_Social models proposed in [Zhu et al., 2015] and two baseline methods: GMM (Gaussian Mixture Model) and a baseline method that plugs in mobility, categorical and temporal features in a simplistic manner (described as follows).

DMM\(^6\)_Basic: It models the probability of a user moving from visited locations to the target location. Pareto distribution [Newman, 2005] is used for modeling the distances between the checked-in locations of a user.

\[
p_u(l) = \sum_l p(u \text{ is at } l) \cdot p(u \text{ moves distance } d(l, l') \text{ from } l)
\]

\[
= \sum_l \frac{p_u^{(l)}}{d(l, l')} \cdot \alpha
\]

DMM\(^6\)_Social: It models user’s and user’s friends mobility patterns using Pareto distribution as above and the resulting
model is the mixture of individual’s distance density and social distance density as follows:

\[ p_u(t) = \Sigma_i \pi_i \left( \frac{p(M)}{(d(t, l) + 1)^{\alpha_M}} + \frac{p(S)}{(d(t, l) + 1)^{\alpha_S}} \right) \]

where, \( p(M) \) and \( p(S) \) are the Pareto distribution parameters learned using individual and social data, respectively.

**GMM:** It models user’s mobility patterns using the Gaussian Mixture Model. Each user’s check-in records are represented using several states, each state modeled using Gaussian distribution. In our experiments we choose two states: home and work [Cho et al., 2011; Zhu et al., 2015].

\[
N(x|\mu_x, \Sigma_x) = \frac{1}{(2\pi)^{D/2}} |\Sigma_x|^{1/2} \exp \left( -\frac{1}{2} (x - \mu_x)^T \Sigma_x^{-1} (x - \mu_x) \right)
\]

\[ p(x) = \sum_k \pi_k N(x|\mu_k, \Sigma_k) \]

where, \( \pi_1, \ldots, \pi_k \), are the mixture weights of the states, \( \mu_1, \ldots, \mu_k \), the mean of each state and \( \Sigma_1, \ldots, \Sigma_k \), the variance of each state.

**Baseline:** In equation (1) in section 4 we plugin Most Frequent Checkins as the location model, Simple Category Distribution as the category model and average time lag based exponential distribution as the temporal model.

\[ P_{t,u}(v|M) = \left( \alpha \frac{I_{|v|}}{|M|u} + (1 - \alpha) \frac{I_{C_t,v}}{|C_t|} \right) \times \exp(-\Delta_{t,v}), \]

where, \( I_{|v|} \) is the number of instances when \( v \) has checked-in at \( t \), \( I_{C_t,v} \) is the number of instances when \( u \) has checked-in at category set \( C_t \), and \( \Delta_{t,u,v} \) is the average of time lag between \( u \) and \( v \) check-ins in the training data.

**ROC and AUC:** Figure 5 shows ROC curves and table 5 shows AUC (area under the curve) of different influence quantification models on different datasets. It can be observed that LoCaTe model outperforms DMM_Basic and DMM_Social models quite significantly on FSq’16 dataset, where we have a single category with each location. On the other datasets where we have a set of categories with each location, although we observe that LoCaTe model outperforms DMM_Basic and DMM_Social models, but we are able to leverage the category information better in the case of a single category with each location. The efficacy of the LoCaTe model is not only contributed by additional knowledge we gain from categories, but also the temporal based user-user correlation modelling, and the adaptive KDE based mobility model (Lo). The Lo model provides a better fit to the mobility data as for each testing location the distance around it is determined using the k nearest neighbours (from the training data). On the other hand, the distance based mobility model (DMM) is sensitive to short distances and thus assigns a low probability to locations at larger distances.

### 6 Conclusion

In this paper, we addressed the problem of influence quantification in the location promotion setting, and proposed a model that incorporates not only the traditional user mobility models but also socially induced temporal correlation of users as well as the affinity of users to a location based on semantics of the location. The resulting LoCaTe model has demonstrated more than 54% improvements over state-of-the-art methods over a number of real-world LBSN data with a large number of users and spanning more than a year.
References


