Location-Specific Influence Quantification in Location based Social Networks

ANKITA LIKHYANI, Indraprastha Institute of Information Technology, India
SRIKANTA BEDATHUR, IIT Delhi, India
DEEPAK P., Queen's University Belfast, UK

Location-based social networks (LBSNs) such as Foursquare offer a platform for users to share and be aware of each other's physical movements. As a result of such a sharing of *check-in* information with each other, users can be influenced to visit (or check-in) at the locations visited by their friends. Quantifying such influences in these LBSNs is useful in various settings such as location promotion, personalized recommendations, mobility pattern prediction etc. In this paper, we develop a model to quantify the influence specific to a location between a pair of users. Specifically, we develop a framework called *LoCaTe*, that combines (a) a user mobility model based on kernel density estimates; (b) a model of the semantics of the location using topic models; and (c) a user correlation model that uses an exponential distribution. We further develop *LoCaTe+*, an advanced model within the same framework where user correlation is quantified using a Mutually Exciting Hawkes Process. We show the applicability of *LoCaTe* and *LoCaTe+* for location promotion and location recommendation tasks using LBSNs. Our models are validated using a long-term crawl of Foursquare data collected between Jan 2015 - Feb 2016, as well as other publicly available LBSN datasets. Our experiments demonstrate the efficacy of the LoCaTe framework in capturing location-specific influence between users. We also show that our models improve over state-of-the-art models for the task of location promotion as well as location recommendation.

ACM Reference Format:
Ankita Likhyani, Srikanta Bedathur, and Deepak P.. 2018. Location-Specific Influence Quantification in Location based Social Networks. 1, 1 (November 2018), 29 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Determination of user influence on social networks is often seen as a tool for viral marketing [26]. Understanding of social media influence has been exploited for legitimate purposes such as promotion of health-information [27], as well as for misleading users through campaigns such as political astroturfs [33]. In the scholarly community, the problem of influence maximization has attracted much attention. Influence maximization [4, 17, 39] is the task 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 which try to maximize the reach of the campaign.

With social media yielding eminently to broad-based social campaigns such as those around health and politics, generic social networks are less suited to localized campaigns by businesses such as salons, fitness clubs, restaurants and others, since information about user locations is not as pervasive within them. Location-based social networks such as *FourSquare*, on the other hand, consider location information as a first class citizen, with most user activity within them involving

Authors' addresses: Ankita Likhyani, Indraprastha Institute of Information Technology, New Delhi, India, ankital@iiitd.ac.in; Srikanta Bedathur, IIT Delhi, New Delhi, India, srikanta@cse.iitd.ac.in; Deepak P., Queen's University Belfast, UK, deepaksp@acm.org.

XXXX-XXXX/2018/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

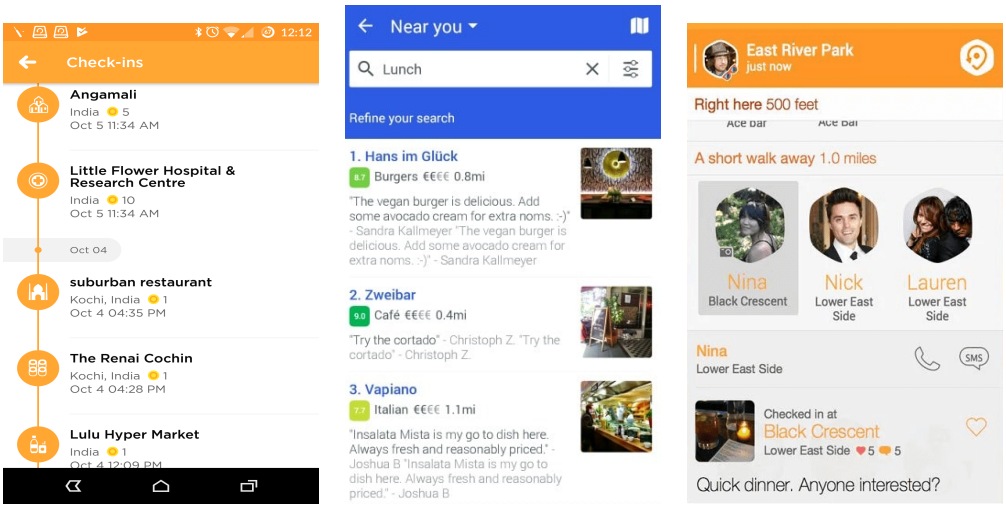


Fig. 1. Location Usage in FourSquare

the sharing of user location. This makes them a suitable platform for hosting localized marketing and advertising information, probably the category of most advertising information that we, as humans, come across in real life. The pervasiveness of GPS[1] within current-day smartphones has led to significant improvements in the penetration of location-based social networks.

Typical location-based social networks (LBSNs) allow users to simply share the location of their visit in a check-in post, optionally allowing to augment the check-in with additional text and/or media. Figure 1 illustrates the central role played by location information in FourSquare, a popular LBSN. The check-in history in the left-most screen is represented as a sequence of locations along with the timestamp, the categories of the locations indicated in the icon. The middle screen represents a typical search scenario in FourSquare, involving a purpose with the location implicitly being the primary factor that filters the search results. The third screen indicates a listing of connections sorted according to the distance from the user. As a simple example of usage of marketing campaigns within FourSquare, consider a restaurant that might want to have their business listed at the top of the search results, or as an advertisement banner along with the search results, for dining searches by users in their vicinity. At the user side, on the other hand, one may want the search to be specialized to prefer the restaurants that her friends have visited frequently and recently and also rated highly. Check-ins of connections have been shown to influence the check-in preferences of LBSN users; for example, [6, 42] have reported evidence of geographical influence over social linkages in LBSNs .

The primacy of locations in LBSNs has sparked interest in location-seeded variants of general influence problems that have been studied for generic social networks. Locations, for the purposes of LBSNs, includes any geo-localized entity that could be the subject of a check-in. This may include particular businesses, e.g., *XYZ Restaurant*, public amenities such as railway stations, as well as things such as parks that have a wider location spread. The *location promotion* problem [49] in LBSNs is the location-seeded version of the influence maximization problem. This task instantiates the influence maximization task on a specified target location (e.g., a particular restaurant), with the intent of finding a set of seed users who are well-positioned for the promotion of the business

[1]https://en.wikipedia.org/wiki/Global_Positioning_System

operating at that location [49]. Once a set of seed users is identified, it can be used to issue targeted special offers to encourage them to visit the location/business being promoted. Once these users visit the business, their check-ins would be expected to consequently attract other users, those over whom they have influence. The location promotion problem is of significant importance for launching effective campaigns to help small businesses gather more customers.

We now outline the task of *influence quantification* as a basic building block for a variety of tasks in LBSNs, including the task of location promotion. Influence quantification is the task of quantifying the influence that a user has over another user, within the context of a location, often modeled probabilistically [9, 45]. Thus, this task associates a triplet, $[u, v, l]$ with a score that indicates the influence of user u over v in the context of the location l. We now motivate as to why influence quantification may be seen as a generic building block for influence tasks in LBSNs. Once the scores for $[u, v, l]$ triplets are *rolled up* (aggregated) across various v's using a suitable aggregation function, we achieve a score for $[u, l]$ pairs that indicate the influence of u in the network, for the location l. The top-scoring u's may then be chosen as a result set for location promotion. This roll-up may be performed on different facets, leading to intuitive solutions for respective problems. For example, the scores for $[u, v, l]$ triplets may be aggregated over multiple locations in a city, to get an estimate of the influence of u over v within the city. Further, an aggregation of influence scores over multiple locations within a category (for example, restaurants or hospitals) would lead to an estimate of a category-specific influence between u and v. As an example, a user might be influenced by one connection for food recommendations, but by another for outdoor activities, and a third for medical purposes. Aggregating the $[u, v, l]$ for a particular user v over the various connections of her (as us) who have recently visited l, achieves a quantification of the likelihood of v to visit l; this could be used to order the recommendations to offer personalized LBSN search for user v. Thus, influence quantification forms a critical and basic building block for various LBSN tasks.

Influence quantification can take into consideration a variety of information that an LBSN offers:

- *geographic features*: user's mobility over different locations,
- *semantic features*: type/category of location (e.g., restaurant, cafe),
- *social correlation*: the relationship between users in the social network, and
- *temporal correlation*: the degree to which a user's movement is correlated with the movement of another user.

Previous work on influence quantification for location promotion has mostly focused on modeling geographic features and social correlation [49]. Studies on semantic features such as category have been limited primarily since datasets containing such information have been scarce [6, 8]; such deficiencies are being addressed recently (e.g., in [11, 23, 40]). The temporal correlation of users behavior has been modeled previously in online social networks, but not in LBSN as we will model in our task. The socially induced followship based on temporal correlation has been of interest in LBSN studies in other contexts [31].

1.1 Contributions

In this paper[2], we develop a novel model and framework called **LoCaTe** for quantifying the location-specific influence between a pair of users who are connected in a social network. LoCaTe combines geographic features of the location, semantics associated with the location and temporal aspects of social following. Specifically, *LoCaTe* incorporates –and derives its name from– the following aspects of check-in information in LBSNs:

[2]This paper is an extended version of [22], we include this for the review process, but this footnote will not appear in the final draft.

Technique	Spatial Target	User Location	Pairwise User Influence
Loc-IM [17]	Location	Single Location	Location-independent
Loc Promotion [49]	Location	Set of Locations	Location-dependent
Reg IM [4]	Region	Set of Locations	Location-independent
Geo Soc Inf [45]	Location	Single Location	Location-independent

Table 1. Summary of Related Work (adapted from [16]) in Influence Models for Location Promotion

- **Lo**cation affinity: The mobility patterns of users that hold cues to whether they frequent the proximity of the target location.
- **Ca**tegory affinity: The affinity of a user to the *semantic categories* of the location.
- **Te**mporal correlation: The temporal correlation of movements between the user and the candidate seedset, thus modeling time-conditioned social followship.

While the basic *LoCaTe* model makes use of exponential distribution in order to quantify temporal correlation, *LoCaTe+* uses advanced modelling (involving more parameters to learn) using mutually exciting hawkes process for the task. In order to illustrate the general purpose utility of *LoCaTe*/*LoCaTe+* for various LBSN tasks, we empirically evaluate our approach not only over the location-specific influence quantification task, but also for the more general problem of *location promotion*.

Our algorithms are evaluated over large-scale real-world LBSN data. We conduct a large-scale Foursquare check-in crawl spanning *more than one year* for use in all our experiments and also have made the collection available for other researchers. We also use the publicly available collections of LBSN data that are commonly used by others in the area. Unfortunately, these previously used collections do not have semantic category information associated with each location. We overcome this limitation by a spatial join with categorical information obtained through separate Foursquare APIs. The LBSN data collected in our crawls, as well as the category mappings to check-in locations in other crawls used in our experiments are made publicly available. Our experimental evaluation establishes the utility of our *LoCaTe* models in accurately quantifying the influence between users in the context of specific locations.

In summary, the contributions we make in this paper are three-fold:

(1) We propose a novel model that combines spatial, temporal and location semantics in the LBSN domain for location-based influence quantification.
(2) We demonstrate the applicability of our influence quantification models for identifying the k (user specified input) seed users for the promotion of a location.
(3) We conduct experimental evaluation over real datasets and show that our proposed model achieves high accuracy, outperforming state-of-the-art influence quantification models.

The remainder of the paper is organized as follows: Section 3 formally defines the influence quantification problem within the larger context of location promotion. Section 2 talks about existing work in this area, and section 4 discusses the modeling methodology. Section 6 shows how to evaluate the proposed influence quantification model and the experimental results obtained. Finally, section 7 concludes the paper and outlines possible future directions.

2 RELATED WORK

While influence maximization has been a well-studied problem (e.g., [5, 9, 12, 13]), the geo-seeded instantiation motivated by LBSNs has gathered attention only recently [4, 17, 31, 34, 37, 45, 48–50]. 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 [4]. Also, there exist recent studies on determing top-k influential locations [34] and product promotion in context of location [48]. 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) [17, 37, 45] or a set of geolocations or only the social network structure [4, 49, 50]. In a similar way, the user-user pairwise influence propagation probabilities are estimated either using just the (social) network structure [4, 17, 45] or taking into consideration the seed location/region [49, 50]. To the best of our knowledge only the recent work in [31], have looked at defining user-user pairwise influence in spatio-temporal context, but for identifying followship.

A summary of important previous techniques categorized along the above dimensions appears in Table 1. In our empirical evaluation, we compare against the most recent work by Zhu et al. [49, 50], that associates a set of locations for each user and considers the influence between two users to be dependent on the location. Note that in their paper, Zhu et al., have presented results using only two popular target locations (*viz.,* the Central Park in New York City, and Cal-Train Station in San Francisco). Our evaluation, on the other hand, considers a much broader set of locations that could be the subject of user check-ins.

User Mobility Models: Capturing humans' mobility behaviors over spatial and temporal space have been studied quite intensly over past few years for applications such as Location Prediction [7, 8, 19, 23, 30], and Location Recommendation [18, 20, 24, 36, 42, 43, 47]. We utilize 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 [7, 8], Gaussian distributions [6] and kernel density based estimations [21] have been particularly successful. Accordingly, we use the kernel density based mobility model [21] to model and exploit user-location affinities. On the other hand, the baseline technique from [49, 50] uses a distance-based mobility model, DMM, in their influence quantification method for location promotion.

Other information associated with a location such as users' activities, and documents that induces spatio-temporal topics are also used for modeling user's mobility behavior [44, 46]. These induced topics can be used to extend LoCaTe further, based on the availability of the integrated data (checkins along with location and social information).

3 PROBLEM STATEMENT

Now we provide a formal definition of influence quantification problem in an LBSN. Table 2 lists a set of notations that will be used. We model a location 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. This is consistent with conventions for location representation in other domains such as OpenStreetMap[3], where multiple tags[4] may be attached to one location. In the following narrative, we use location and venue interchangeably; though we feel venue is a more appropriate word, location corresponds to the convention in existing literature.

Influence quantification is the task of quantifying the influence of a user over another in the context of a location. For most usage scenarios, we would like to quantify the influence as the

[3]https://www.openstreetmap.org/

[4]http://wiki.openstreetmap.org/wiki/Tags

Symbol	Description
G	A location based social network
U	Set of users in G
E	Set of connections from u_i to u_j s.t. $u_i, u_j \in U$ and $u_i \neq u_j$
ℓ	A location specified by a triple (x, y, C_ℓ), where x,y correspond to geo-coordinates and C_ℓ to category set of ℓ
$\langle u, \ell, t, C_\ell \rangle$	A check-in record of user u at time t at location ℓ that has a category set C_ℓ
M_u	set of check-in records $\langle u, \ell, t, C_\ell \rangle$
L	A set of locations
C	A set of categories

Table 2. Notations used in this paper

likelihood of a user visiting the location given the visit to the same location by another (i.e., seed) user. We now use this perspective to provide a formal definition.

Definition 3.1 (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 $P(\ell, u, v \mid G)$, the likelihood that any user v among u's connections is likely to visit ℓ. □

There are two implicit assumptions in this definition. First, that the seed user u has visited the location ℓ; this is typically justified since some evidence of association between u and ℓ would be necessary for the premise that u would influence v in the context of ℓ. Second, most LBSNs, like general social networks, have a timeline display where each user would be provided with (largely reverse-chronological ordering of) her connections' recent check-ins. This is in addition to a second functionality, that of location-targeted search, where a user pro-actively looks up the visitors of a particular location. With most implicit influence being through the more popular former channel, that of timelines, we will attempt to quantify the influence between connected users, since they could figure in the timelines of each other.

In many contexts, we may want to score target users within the context of the seed user and chosen location. Thus, it is appropriate to model the influence quantification as a distribution over the set of users v; accordingly, we will use $P_{\ell,u}(v|M)$ to indicate the influence quantification for the combination $[u, v, \ell]$, with M indicating the influence quantification model being employed.

4 LOCATE FRAMEWORK AND MODELS FOR INFLUENCE QUANTIFICATION

We now outline our influence quantification framework, LoCaTe, that estimates $P_{\ell,u}(v \mid M)$, a scoring that captures the likelihood that the user v from u's connections would visit the location ℓ quantified using the check-in records in the training part, denoted as M. Figure 2 shows the framework of *LoCaTe*. LoCaTe combines information from three kinds of features to arrive at a estimation as follows:

$$P_{\ell,u}(v \mid M) = \Big(\alpha \underbrace{P_L(v, \ell \mid M)}_{\text{location affinity}} + (1-\alpha) \underbrace{P_C(v, C_\ell \mid M)}_{\text{category affinity}} \Big) \times \underbrace{T(u \rightarrow v \mid M)}_{\text{temporal correlation}} \tag{1}$$

such that for all the users U, locations L and the entire time range T

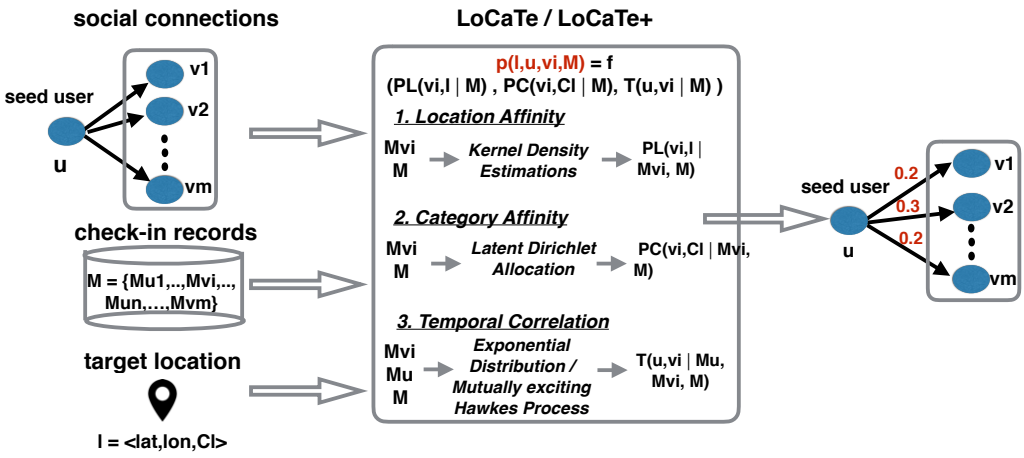


Fig. 2. LoCaTe: Framework for Influence Quantification

$$P_a = \sum_{v \in U, \ell \in L} \Big(\alpha \underbrace{P_L(v, \ell \mid M)}_{\text{location affinity}} + (1-\alpha) \underbrace{P_C(v, C_\ell \mid M)}_{\text{category affinity}} \Big) = 1 \tag{2}$$

$$P = \int_0^T P_a . T(u \rightarrow v \mid M) = 1 \tag{3}$$

$P_L(v, \ell \mid M)$ models the affinity of v to location ℓ, and $P_C(v, C_\ell \mid M)$ 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 α. Further, $T(u \rightarrow v \mid M)$ captures the temporal correlation between users u and v, a term that we model as being *independent* of the location ℓ. The first two terms quantify user's affinity for the location using mobility and categories respectively and are combined using a weighted sum. The third term quantifying location-agnostic (in the sense that the quantification is performed over all check-ins comprising a number of locations) user-user temporal affinity is merged using a product. Thus, the final scoring, due to its product form, ensures that users who are strong on both location and temporal aspects score much higher than others.

$P_{\ell,u}(v \mid M)$, being a normalized score, ranges between [0,1]. The usage of **Lo**cation affinity, **Ca**tegory affinity and **Te**mporal correlation in our model lends the name to our method. The two models that we propose based on this framework differ in the model used for the **Te**mporal correlation; while the basic model, also called *LoCaTe* uses an exponential distribution in the modelling, *LoCaTe+* makes use of mutually exciting Hawkes process.

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 [6]. Thus, a user has an inherent preference for some geo-locations. This inherent preference of number of geo-locations vary from individual to individual. Thus fixing it to two or more components can lead to inability to either capture many of high density patterns or waste considerable probability mass over certain regions. Lichman et al. in [21] addresses the limitations of fixating the densities to a specific number by introducting Kernel Density Estimates. Kernel Density Estimation is a non-parametric method for estimating the density function from

random sample of data [35], and 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.

The affinity of v to ℓ based on her own check-in history (i.e. $E = \{l_1, ..., l_n\}$, where, l_j = <x,y> is a two-dimensional location, $1 \leq j \leq n$) is modeled as the kernel density estimate that quantifies the average weighted similarity between ℓ and each checked-in location l_j, using a hyper-parameter k

$$P_L(v, \ell \mid M_v) = f_{KD}(l \mid M_v, k) = \frac{1}{|M_v|} \sum_{j=1}^{|M_v|} \kappa_{j,k}(\ell, \ell_j) \tag{4}$$

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

$$\kappa_{j,k}(\ell, \ell_j) = \frac{1}{2\pi h_{j,k}} exp\left(-\frac{1}{h_{j,k}} \|\ell - \ell_j\|\right) \tag{5}$$

Here, $h_{j,k}$ is a location-dependent scalar factor that is set to be the Euclidean distance of ℓ_j to it's k^{th} nearest neighbor, and $\|\ell - \ell_j\| = \sqrt{(\ell.x - \ell_j.x)^2 + (\ell.y - \ell_j.y)^2}$. The bandwidth $h_{j,k}$ adapts according to the k^{th} nearest neighbor, thus facilitating robustness towards varying densities. For example, setting a bandwidth value very high in urban areas where events are densely populated within a small region will lead to oversmoothing, while setting the badnwidth to a small value in sparsely populated areas will lead to overfitting. Thus, bandwidth computed using the nearest neighbors approach ensures the bandwidth computation is sensitive to differential densities of locations in urban and rural areas.

4.1.1 Mixture of Kernel Density models. The location affinity for a user v is learnt using v's check-in records. But, for some users we have very little data to make predictions. To overcome this data sparsity issue we interpolate individual user's model with the kernel density model learned over check-in records of all users, as follows:

$$P_L^k(v, \ell \mid M) = \beta_v f_{KD}(\ell \mid M_v, k) + (1 - \beta_v) f_{KD}(\ell \mid M, k), \tag{6}$$

where, β_v is a user-specific mixing weight, determining the relative influence between the user model and the global model. We will denote this as $P_L(.,.)$ when the value of k is clear. We will estimate both k and β_v using the corpus of check-in records, as we describe later in section 4.4.

Note that, in the above model we have used only two components in the mixture model, where first component models individual's check-ins and second component models full-population check-ins. But, the intermediate components between these two can be defined at different spatial scales such as neighborhoods, cities, states, and even countries. Moreover, the users' connections can also be exploited at different spatial scales. In this work, instead of fine tuning to different levels of smoothing we have kept the simplistic model with two components, since this is an orthogonal research to our current work.

4.2 Category Affinity

Locations 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 3 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, $P_C(v, C_\ell \mid M)$.

(a) User A (b) User B

Fig. 3. Category wise check-in Distribution

For topic modeling, we use Latent Dirichlet allocation (LDA) [3] which models semantic matching between text documents by learning latent topics, each of which is a probability distribution over the set of words. The LDA model 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 v_C (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 the set of categories. We then use the learned topics to estimate the user's affinity to the set of categories associated with the location of interest:

$$P_C(v, C_\ell \mid M) = \sum_{Z \in Topics(M)} P(C_\ell \mid Z) \times P(v \mid Z), \tag{7}$$

where $Topics(M)$ is the set of topics learned as described, and Z represents a topic from the learnt topic-set. $P(v \mid Z)$ and $P(C_\ell \mid Z)$ quantify how well the category distribution associated with Z match against those of the check-ins of v and the categories of ℓ respectively. High values of $P_C(v, C_\ell \mid M)$ 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, $T(u \rightarrow v \mid G)$, 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 $(u, v) \in E$. We target to arrive at a quantification based on historical check-ins of the users, so that cases where a user u has been closely followed by v historically yields a high value for the $T(u \rightarrow v \mid G)$. We first empirically analyze the behavior of general inter-arrival times (in days) of users in the LBSN at a given location, without distinguishing 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.

These two different distributions of time lags are given in Figure 4, where u_3 and u_4 are the followers of u_1. We collect these time lag distributions across all locations in the LBSN and study their frequency distribution using a histogram-style analysis. As expected, the general *across userbase* time lag distribution follows a classical Heavy Tailed distribution (see Figure 5(a)). However, the *across connections* time lag distribution (in Figure 5(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

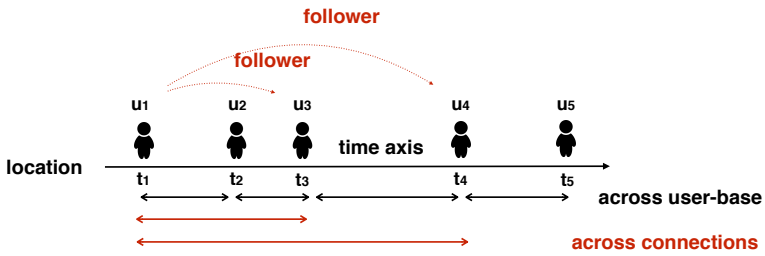


Fig. 4. Depicting the time lag between check-ins at a location for connected and non-connected users, u_3 and u_4 are followers of u_1

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.

These observations leads us to a natural and simple model of time lag distribution between users that uses an exponential distribution, used in similar settings elsewhere [31]. Despite its simplicity, this formulation surprisingly effective in practice as seen in our experiments.

However, the above model of temporal correlation or time lag distribution of checkins at a location between socially connected users using exponential distribution makes a rather strong, simplifying assumption that events (i.e., checkins) arrive at a constant rate, λ, throughout the time of observation. In reality, however, that is rarely the case. ce, when there are well-advertised promotions at a location we can expect checkin activity of each user to show a bursty behavior with higher rates of checkins, and consequently shorter time-lags, than during regular times. In our second model, we incorporate changing intensity of checkins by using nonhomogenous Poisson processes (NPP) to model the checkin behavior. Specifically, we use a class of NPPs, viz., the mutually-exciting Hawkes processes [10, 14] (*meHP*), which has been successfully used to model contagions in Financial markets [1] as well as in Social media [41]. Note that we found the use of meHP particularly attractive because it allows for a clean modeling of "self-excitation" of a user *independent* of the influence of another user in the LBSN (as in the case of a well-promoted location given above). Thus, the resulting temporal user correlation is capable of more accurately modeling the true followship strength between users.

We call the full influence quantification mdoel (Ref. Eq. 1) that uses the exponential distribution for estimating user correlation as **LoCaTe** and the one that uses mutually exciting Hawkes process modelling as **LoCaTe+**. We provide the details of the temporal user correlation models in separate subsections herein.

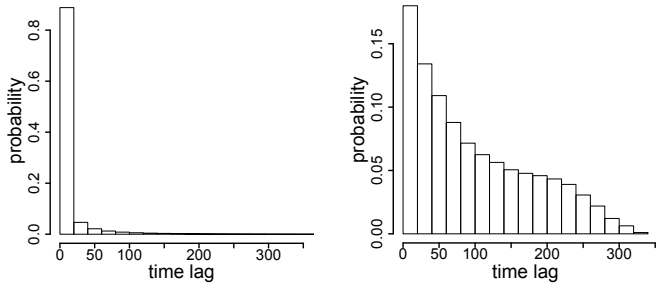
4.3.1 Modeling using exponential distribution. According to the exponential distribution modeling , the weight associated with any value of time lag, denoted δt, would be quantified as the following:

$$p(\delta t) = \lambda_t e^{-\lambda_t \delta t} \tag{8}$$

We set λ_t is the inverse of the mean time lag between check-ins by connected users:

$$\lambda_t = 1/avg\left\{|t_2 - t_1| \mid \exists\langle u, \cdot, t_1, \cdot\rangle \in M \wedge \exists\langle v, \cdot, t_2, \cdot\rangle \in M \wedge (u, v) \in E\right\}, \tag{9}$$

where the $\langle u, \cdot, t, \cdot\rangle$ 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 $T(u \rightarrow v \mid G)$ which is modeled as the cumulative weight of v checking in at a



(a) across userbase (b) across connections

Fig. 5. Time lag (in days) probability distribution plot

location visited by u after a time lag of any $t \geq t_{u,v}^{min}$:

$$T(u \rightarrow v \mid G) = \int_{t_{u,v}^{min}}^{\infty} \lambda_t e^{-\lambda_t \delta t} d(\delta t) \tag{10}$$

$$= -e^{-\infty} + e^{-\lambda_t t_0} = e^{-\lambda_t t_{u,v}^{min}} \tag{11}$$

$$t_{u,v}^{min} = \min \{(t_2 - t_1) \mid \exists \langle u, \cdot, t_1, \cdot \rangle \in M \wedge \exists \langle v, \cdot, t_2, \cdot \rangle \in M\}$$

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 $T(u \rightarrow v \mid G)$ reflects the extent of correlation between u and v, since $T(u \rightarrow v \mid G)$ would have a high value for those user pairs where the latter follows the former (temporally) closely.

4.3.2 Modeling using Mutually Exciting Hawkes Processes. We define, for a user v, the activity of checking in to location ℓ at time t as a function of three components:

(1) μ_v: user's base (location-agnostic) intensity of checking in,
(2) $\alpha_v \sum_{t_i \in H_v(t)} \exp(-\eta_{vv}(t - t_i))$: self excitation or the component that accounts for repeated check-ins by the user to the same location,
(3) $\alpha_{u \rightarrow v} \sum_{t_j \in H_u(t)} \exp(-\eta_{uv}(t - t_j))$: excitation caused by neighbors/ friends checking into the location.

In the above, $H_u(t)$ and $H_v(t)$ indicates all checkin event timestamps prior to current time t of user u and v respectively. Although we can parameterize the influence impulse responses for each pair of users, for the sake of model simplicity we set all of them to a common user-specific kernel $\exp(-\eta_v(\Delta t))$. Thus, $\lambda(t, \ell)$ can be written as:

$$\lambda_v(t, \ell) = \mu_v + \alpha_v \sum_{t_i \in H_v(t)} \exp(-\eta_v(t - t_i))\, I(l_i = \ell) + \alpha_{u \rightarrow v} \sum_{t_j \in H_u(t)} \exp(-\eta_v(t - t_j))\, I(l_j = \ell), \tag{12}$$

where the first, second and third terms account for base intensity, self-excitation and neighbors' excitation respectively. For parameter estimation under this model, we outline the likelihood expression (of a set of check-ins for the model parameters), which we would like to maximize over the entire observed set of check-ins. The design of the model allows us to break down the likelihood expressions into a product of likelihood expressions, one expression for each user that specifically deals with parameters that relate to the user.

$$L(\boldsymbol{\mu}, \boldsymbol{\alpha}, \mathbf{A}, \boldsymbol{\eta}) = \prod_v L_v(\mu_v, \alpha_v, \mathbf{A}_{* \rightarrow v}, \eta_v), \tag{13}$$

where, $\boldsymbol{\mu}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$ are vectors of user-specific parameters, $\mathbf{A}$ is a user-user influence weight (i.e. $\alpha_{u\to v}$ above) matrix and $\mathbf{A}_{*\to \mathbf{v}}$ is a row of all influence weights for a user v. Each $L_v(\cdot)$ can now be optimized separately. According to the meHP model, their construction is as follows:

$$L_v\left(\mu_v, \alpha_v, \mathbf{A}_{*\to v}, \eta_v\right) = \prod_{n=1}^{N_v} \lambda_v\left(t_n, l_n\right) \times \left(\int_0^T \exp\left(-\eta_v\left(t, l_n\right)\right) \right) dt \tag{14}$$

where product is over the N_v check-ins made by the user v, and with t_n and l_n denoting the time and location associated with the n^{th} check-in. After optimization procedure, we are ready to quantify the user correlation using the parameter estimates $\alpha_{u\to v}$. Once user u checks-in at a location, there is a time lag for the check-in information to propagate to v before the latter can make an influenced check-in. Let this time-lag be $t^{min}_{u,v}$ as in the case with exponential distribution modelling in the previous section. The temporal user correlation is simply the estimation of how likely v is, to check in at a location visited by u after a time lag of $t \geq t^{min}_{u,v}$, solely by virtue of influence from u:

$$\begin{aligned} T\left(u \to v \mid G\right) &= \int_T^\infty \alpha_{u\to v} \sum_{t_j \in H_u(t)} \exp\left(-\eta_v\left(t - t_j\right)\right) dt \\ &= \sum_{t_j < T} \left(\frac{\alpha_{u\to v}}{\eta_v}\right) \exp\left(-\eta_v (T - t_j)\right), \end{aligned} \tag{15}$$

where, T is given as:

$$T = t_u + t^{min}_{u,v} \tag{16}$$

t_u is the time u checked-in in the test data and $t^{min}_{u,v}$ is estimated as in the case of the exponential distribution-based modelling:

$$t^{min}_{u,v} = \min\left\{(t_2 - t_1) \mid \exists\langle u, \cdot, t_1, \cdot\rangle \in M \wedge \exists\langle v, \cdot, t_2, \cdot\rangle \in M\right\}.$$

4.4 Parameter Estimation

There are multiple parameters to be estimated α, β_v, k, μ_v, α_v, $\alpha_{u\to v}$, and η_v where β_v and k are the parameters specific to Location Affinity model and α is the mixing weight parameter of Location and Category affinity. The parameters (μ_v, α_v, $\alpha_{u\to v}$, and η_v) are associated to the meHP based Temporal Correlation. We use EM-algorithm for estimation of parameters β_v and α. The EM algorithm to used learn α is as follows:

- E-step: Here, we compute a data point specific α whose estimate at the i^{th} iteration is denoted as $\alpha_p^{(i)}$. Note that $\alpha_p^{(i)} \in [0, 1]$. This is done for each data point in the validation set, a held-out part of the check-ins, denoted as N.
- M-step: The data point specific weights are then aggregated to arrive at a revised overall estimate for α for this iteration, denoted as $\alpha^{(i)}$. This is done as follows:

$$\alpha^{(i)} = \frac{\sum_{p\in N} \alpha_p^{(i)}}{|N|} \tag{17}$$

- With $\alpha^{(i)}$, the new likelihood is computed. For convergence we check the difference between the old likelihood and the new likelihood is less than the threshold set to 0.01. Upon convergence, $\alpha^{(i)}$ is output as the value for the α to be used for the dataset.

For β_v, similar procedure is followed. The only difference is that it is done over the training dataset, and not on validation set since there are many users who do not have any check-ins in the validation set (i.e. the heldout part from training and testing). Table 3 shows values of β_v and α learned for different datasets.

dataset	FSq'16	FSq'11	FSq'10	BrightKite	Gowalla
α	0.90	0.95	0.92	0.93	0.94
β_v	0.78	0.86	0.85	0.91	0.90

Table 3. α and β_v values

k	2	3	4	**5**	6	7	8	9	10
Fsq'16	-2.032	-1.804	-1.704	**-1.640**	-1.670	-1.687	-1.722	-1.744	-1.817
Fsq'11	-2.711	-2.640	-2.063	-1.726	-0.939	-0.738	**-0.677**	-0.794	-0.851
Fsq'10	-1.283	-1.251	-1.233	**-1.211**	-1.225	-1.231	-1.246	-1.260	-1.278
Brightkite	-1.915	-1.869	-1.836	-1.789	**-1.779**	-1.821	-1.850	-1.879	-1.897
Gowalla	-1.978	-1.896	-1.847	**-1.804**	-1.825	-1.854	-1.877	-1.890	-1.931

Table 4. Log-likelihood at different values of k

The hyper-parameter k is estimated as the value that maximizes the likelihood of check-ins in a chosen validation set. Thus, we set k to the value that maximizes the following:

$$k = \arg\max_{k'} \sum_{\langle v,\ell,\cdot,\cdot\rangle \in V} \log\left(P_L^{k'}(v,\ell \mid M)\right) \tag{18}$$

The distribution of log-likelihood across various values of k are shown in Table 4; accordingly, we chose $k = 5$ for usage in our method.

The parameters of the meHP model of temporal user correlation (i.e., μ_v, α_v, $\alpha_{u\to v}$, and η_v) are learnt jointly by maximizing the likelihood function for each user v given in Equation 14 using the simplex method [28].

5 APPLICATIONS USING THE LOCATE INFLUENCE QUANTIFICATION MODELS

The quality of the LoCaTe models may be used for the fine-grained task of predicting the set of v's connections who would check-in into a location ℓ shortly after v's check-in. However, this task in itself is not of enough utility to allow for practical use cases such as those allowing businesses to intervene into the market and focus their activities towards achieving desirable effects on their clientele. The estimates from the influence quantification model, as observed in the introduction, could be aggregated along different facets, for a variety of interesting tasks in LBSNs, including those that allow for interventions. We consider the usage of influence quantification models such as LoCaTe/LoCaTe+ in two scenarios: location promotion and personalized location recommendations. Our empirical evaluation is limited to the location promotion task since that can be evaluated using the datasets without expensive user studies.

5.1 Location Promotion

We first start with a definition of the location promotion problem.

Definition 5.1 (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, $S \subseteq U$, such that seed users corresponding to S lure other users to the target location ℓ maximally. The task typically uses a hyper-parameter τ, that limits the number of seed users in the output, to τ. □

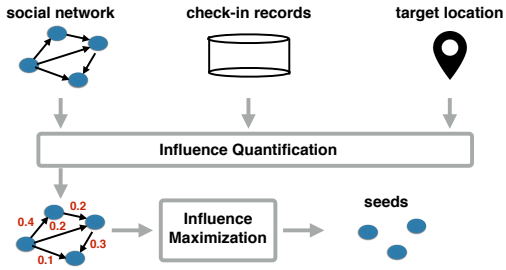


Fig. 6. Location Promotion Framework

ALGORITHM 1: Influence Maximization

Data: Target Location ℓ, τ, Influence Quantification Model M, threshold ρ
Result: τ seed users, denoted as S
initialize $S \leftarrow \phi$;
initialize $I \leftarrow \phi$;
while $|S| < \tau$ **do**
  u' $\leftarrow \arg\max_{\forall u \in visited(\ell)} \{v \mid v \notin I \wedge P_{\ell,u}(v \mid M) \geq \rho\}$;
  $S \leftarrow S \cup u'$;
  $I \leftarrow I \cup \{v \mid P_{\ell,u'}(v \mid M) \geq \rho\}$;
end
return S

Figure 6 illustrates the schematic of a location promotion framework using the LoCaTe models. We first localize our interest to the location that forms the target, i.e., the one to be promoted. The chosen LoCaTe model is run just for the location of interest, to arrive at a set of user-user edge-weights represented in the bottom right corner. These weights can then be consumed by a greedy algorithm for influence maximization that we outline in Algorithm 1. In Algorithm 1, we use a threshold ρ to determine the users who influence others; in other words, we estimate v to be influenced by u for the location ℓ if $P_{\ell,u}(v \mid M) \geq \rho$ is satisfied. The greedy strategy is then straightforward in that it builds a set S of potential seed users, and the corresponding set of influenced users I. Both these sets are initialized to null; at each iteration, the user who can bring in the largest number of new users to I is chosen for inclusion in S. This seed user accumulation stops on reaching the desired output size τ, upon which the set of chosen seed-users S is output.

5.2 Personalized Location Recommendations

A user u, who is at a particular geo-position p, may be interested in getting a list of personalized recommendations of locations to visit, based on her interests and the interests of her connections in the LBSN. In such a scenario, it is likely that the user is interested in locations that are (i) proximal (i.e., geographically closer), (ii) in line with her interests, and (iii) are aligned with the interests of her connections. Accordingly, the scoring for a location may be arrived at using separate modeling of each of these factors, and then aggregated using a weighted sum; this, followed by the choice of the top-k scored locations, would complete a solution to the location recommendation problem. This leads to the following scoring function:

$$\mathcal{S}_{u,p}(\ell) = \gamma_1 \times Proximity\,(\ell, p) + \gamma_2 \times \Big(\alpha P_L\,(u, \ell \mid M) + (1-\alpha)\,P_C\,(u, C_\ell \mid M)\Big) + \gamma_3 \times \sum_{v,[u,v]\in E} P_{\ell,v}\,(u \mid M) \quad (19)$$

The first term quantifies the proximity between the location and the user's position using a suitable geo-similarity measure, whereas the second term uses the same models as in LoCaTe/LoCaTe+ to quantify the user's likely interest in the location ℓ using both location and category affinities. The third term is where the influence quantification model gets plugged in, whereby the scoring is boosted based on the influence from connections of the user who have previously visited ℓ, the extent of the boosting determined by the estimate from the influence quantification. The parameters γ_1, γ_2 and γ_3 are estimated in the same manner as we estimate the weight parameters for the chosen LoCaTe model using EM-algorithm, as described in section 4.4. This is followed by choosing the locations with the top-k scores to be displayed to the user in a scored list. The usage of the influence models is intuitively expected to cause desirable deviations from a simple scoring such as one based on just the user interests and proximity, leading to enhanced user satisfaction and reliance on the search interface.

6 EXPERIMENTAL DETAILS

In this section, we evaluate the effectiveness of both our proposed models, viz., LoCaTe and LoCaTe+, against state-of-the-art influence quantification models, those from [49, 50]. We perform empirical evaluation over the influence quantification task, as well as over the more coarse-grained tasks of location promotion, and location recommendation. In additional to the comparative evaluation, we also present trends across varying values of cut-off thresholds used to discretize the influence scoring to sets of influenced and other users.

6.1 Datasets used

We tested over 5 datasets as shown in Table 5, of which FSq'16 is the one that we collected using Twitter and FourSquare APIs, and rest are publicly available datasets [6, 8]. In our method, we make use of check-in histories, social connections as exemplified in the social graph, as well as the categories associated with each location. However, the publicly available datasets, from [6, 8], do not have category information associated with locations. With the location names being anonymized as well, there is no possibility of inferring the category from the location name. Thus, we follow the approach outlined in [23] – from FourSquare location API, we obtain for each location (specified using its GPS coordinates) a set of categories that correspond to actual venues within a distance threshold (we use 50m) of the location. Note that this *spatial join* can be noisy, particularly in urban centres where venues with diverse categories may be located within a 50m radius of the location (e.g., in large malls or shopping districts). Nevertheless, this is the only way we were aware of that circumvents the lack of category information. There are some recently released FourSquare datasets (e.g.,[40]) which could not be used in our experiments since they do not have even the social graph information, making them unsuitable in tasks relating to social influence.

6.1.1 Data Collection. We now describe the data collection process used for compiling the FSq'16 dataset that comprises check-ins, location information and social graphs.

First, for check-in information, it may be noted that FourSquare users' check-in information is visible only within their respective social circles. However, users can choose to broadcast their check-ins to Twitter while using mobile-based app from Foursquare, Swarm app. This provides us an opportunity to capture their check-ins by crawling tweets with keyword *swarmapp.com* on the Twitter public stream API [5]. This limits our dataset to FourSquare check-ins that are also posted via Twitter. We improve the coverage by first extracting the userIDs from these check-in tweets and using it to harvest more check-ins of the user by crawling their tweet timelines with Twitter API[6].

In the second step, we get the location information by following the FourSquare URL in the tweet that leads to the FourSquare location page. We parse this web page to get the information about the checked-in location. Specifically, we scrape the category information from this page, and augment it to the location. Thus, we were able to get single fine-grained category for each location as against the others for which we use approximate spatial joins to infer categories.

Thirdly and lastly, for gathering social graph information, FourSquare poses the same restriction, due to privacy reasons, as for check-ins, since it limits the connection information to just the users' social circles. We circumvent this again using Twitter, crawling Twitter connection information among users in our check-in dataset by using Twitter API[7]. While the resulting social graph is not expected to be identical to the original Foursquare graph, but it is a subset where each user has their Twitter profile public and have linked with the FourSquare profile. To extract the check-in details of friends we crawl tweets on their timeline in the same manner as above.

Some key characteristics of the resulting combined dataset, which we denote as FSq'16, along with those of other public datasets we use, is shown in Table 5.

Dataset	FSq'16	FSq'11	FSq'10	Brightkite	Gowalla
Duration	Jan'15 - Feb'16	Jan'11 - Dec'11	Mar'10 - Jan'11	Apr'08 - Oct'10	Feb'09 - Oct'10
#users	119, 756	11, 326	18, 107	58, 228	196, 591
#check-ins	9, 317, 276	1, 385, 223	2, 073, 740	4, 491, 143	6, 442, 890
#unique locations	183, 225	187, 218	43, 064	772, 966	1, 280, 970
#unique categories	734	638	624	683	680
#friendship-links	1, 308, 337	47, 164	115, 574	214, 078	950, 327
avg. degree	21.85	8.33	12.76	7.35	9.66
#users (training records > 10)	78, 312	11, 324	17, 369	23, 356	72, 925
$A(\ell, u)$	55, 884	15, 951	4, 056	2, 642	88, 865
cut-off timestamp	1/12/2015	1/10/2011	1/12/2010	1/5/2010	1/6/2010
Mean(#categories / location)	1.00	12.12	20.47	8.38	1.28
Mean(#categories / topic)	330.23	305.12	319.45	249.92	352.56

Table 5. Statistical properties of the datasets

6.1.2 Train-Test Partitioning. For each dataset, we assign a cut-off timestamp, the data prior to it is used for training the influence models and rest of the check-ins for testing the validity of their predictions. The cut-off timestamp is chosen such that 80% of total checkins are used for training.

6.1.3 Implementation Details. We implemented our model and the baselines in Java. Whenever specific building blocks were available off-the-shelf, we made use of those; this includes the kernel density estimation code from the UCI Datalab website (http://www.datalab.uci.edu/resources) and the topic modelling implementation from Mallet http://mallet.cs.umass.edu/topics-devel.php. We

[5] https://dev.twitter.com/streaming/public
[6] https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.html
[7] https://developer.twitter.com/en/docs/accounts-and-users/follow-search-get-users/api-reference/get-friends-list

ran all algorithms on a server with 6-core 2.5GHz Intel Xeon CPU with 64GB of RAM. The source code and the datasets used will all be made publicly available through https://goo.gl/ayzehx.

6.2 Influence Quantification Models

We compare our proposed **LoCaTe** models with three baseline methods;

(1) Distance-based mobility models (DMM) [49, 50],
(2) Gaussian-mixture models (GMM) [6, 49, 50] and
(3) a *Baseline* model that brings together mobility, categorical and temporal features using a simple aggregation.

The first and second methods yield variants based on the usage of social connections and location categories; however, they do not use any form of user correlation information. Thus, we compare against the third method that uses a simplistic temporal user correlation modelling, to illustrate the effectiveness of our method.

(1) **GMM:** It models user's mobility patterns using a Gaussian mixture model. Each user's check-in records can be represented using several states, and each state can be modeled using Gaussian distribution. In our experiments we choose two states: home and work states as suggested in [6, 49]

$$\mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right\}$$

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

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.

GMM-category: Zhu et. al. in [50] extends the basic GMM model to incorporate category information as follows:

$$p(\ell \mid u) = P(x, y, C_\ell \mid u) = p(x, y \mid C_\ell, u)\, p(C_\ell \mid u)$$

To derive $p(x, y|C_\ell, u)$, u's check-in records that belong to C_ℓ are selected to build the Gaussian distribution if u has a sufficient number (i.e., larger than θ_{C_ℓ}) of check-in records that belong to C_ℓ. Otherwise the check-ins under category C_ℓ in the region $R_{x,y,r}$, i.e., $p(x, y \mid C_\ell, R_{x,y,r})$, is used instead of directly calculating $P(x, y \mid C_\ell, u)$, where $R_{x,y,r}$ is a circular region with center (x, y) and radius r.

$$p(x, y \mid C_\ell, u) = \begin{cases} \mathcal{N}\left(\mu_{u,C_\ell}, \Sigma_{u,C_\ell}\right) & \text{if } \left|\{(u, \ell = (x, y, C_\ell), t) \mid u, C_\ell\}\right| > \theta_{C_\ell} \\ p\left(x, y \mid C_\ell, R_{x,y,r}\right) = \mathcal{N}\left(\mu_{R_{x,y,r,C_\ell}}, \Sigma_{R_{x,y,r,C_\ell}}\right) & \text{otherwise} \end{cases}$$

θ_{C_ℓ} and r is set to 10 and 1km, respectively as used in [50].

(2) **DMM:** Distance based mobility model, models the probability of a user moving from visited locations to the target location.

DMM_Basic: Pareto distribution [29] is used for modeling the distances between the checked-in locations of a user.

$$p_u(\ell) = \sum_l P(u \text{ is at } l)\; P(u \text{ moves distance } d(l, \ell) \text{ from } l)$$

$$= \sum_l \frac{p_{u,l}\, \alpha_M}{(d(l, \ell) + 1)^{\alpha_M}}$$

DMM_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:

$$P_u(\ell) = \sum_l p_{u,l} \left[\frac{p(M)\alpha_M}{(d(l,\ell)+1)^{\alpha_M}} + \frac{p(S)\alpha_S}{(d(l,\ell)+1)^{\alpha_S}} \right]$$

where, $p(M)$ and $p(S)$ are mixing components and α_M and α_S are the Pareto distribution parameters learned using individual and social data, respectively.

DMM_Category: Similar to GMM_Category, DMM_Category is adopted from DMM_Basic as follows:

$$p(x,y|C_\ell,u) = \begin{cases} \sum_l \frac{p_{u,l}\alpha_{u,C_\ell}}{(d(l,\ell)+1)^{\alpha_{u,C_\ell}}} & \text{if } \left|(u,\ell=(x,y,C_\ell),t) \mid u,C_\ell\right| > \theta_{C_\ell} \\ p(x,y|C_\ell,R_{x,y,r}) = \sum_l \frac{p_{u,l}\alpha_{R_{x,y,r,C_\ell}}}{(d(l,\ell)+1)^{\alpha_{R_{x,y,r,C_\ell}}}} & \text{otherwise} \end{cases}$$

(3) **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. These are combined in exactly the same way as the analogous terms are combined within the LoCaTe model, i.e.:

$$P_{\ell,u}(v|M) = \left(\alpha \frac{I_\ell}{|M_u|} + (1-\alpha) \frac{I_{C_\ell}}{\sum_{i=1}^{|M_u|} |C_i|} \right) \times \exp\left(-\overline{\Delta t_{u,v}}\right),$$

where, I_ℓ is the number of instances when u has checked-in at ℓ, I_{C_ℓ} is the number of instances when u has checked-in at category set C_ℓ, and $\overline{\Delta t_{u,v}}$ is the average of time lag between u and v check-ins in the training data.

6.3 Evaluation on Influence Quantification Task

For evaluation on Influence Quantification task, we use the same framework as used in an earlier work [49]. Consider a particular instance of the influence quantification problem for location ℓ and a seed-user u; the influence quantification output would be an ordered list of u's connections, ordered in the decreasing (non-increasing) order of estimated likelihood to visit ℓ. This list can be cut-off using a threshold ρ to identify a set of users who are deemed to be highly likely to visit ℓ - this set forms the *predicted set*, $PS(\ell, u, \rho \mid G)$. The ground truth activated set, $A(\ell, u)$, is the subset of u's connections who have actually visited ℓ after the cut-off timestamp (i.e., from the test set). The match between $PS(\ell, u, \rho \mid G)$ and $A(\ell, u)$ measured at various values of the threshold ρ 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 $[\ell, \rho]$. We use the ROC curve (generated by varying ρ) to compare our method against baselines in our empirical evaluation.

Now, to arrive at a set of target locations for ℓ to perform the aforementioned ROC curve evaluation, we identify a set of locations from the dataset where there are many users checking-in before the train/test cut-off timestamp, and their followers checking-in after the cut-off timestamp. This will ensure that there are enough users in the respective $A(\ell, u)$ sets formed for the location, to alleviate sparsity issues in the evaluation. Table 5 shows the number of test cases, $A(\ell, u)$, along with the cut-off timestamp for each dataset.

6.3.1 ROC and AUC. Figure 7 shows ROC curves and table 6 shows AUC (Area Under the Curve) of different influence quantification models on different datasets. It can be observed that the *LoCaTe* models outperform DMM_Basic, DMM_Social and DMM_Category models quite significantly on FSq'16 dataset, where we have accurate location category information. Even on the other datasets,

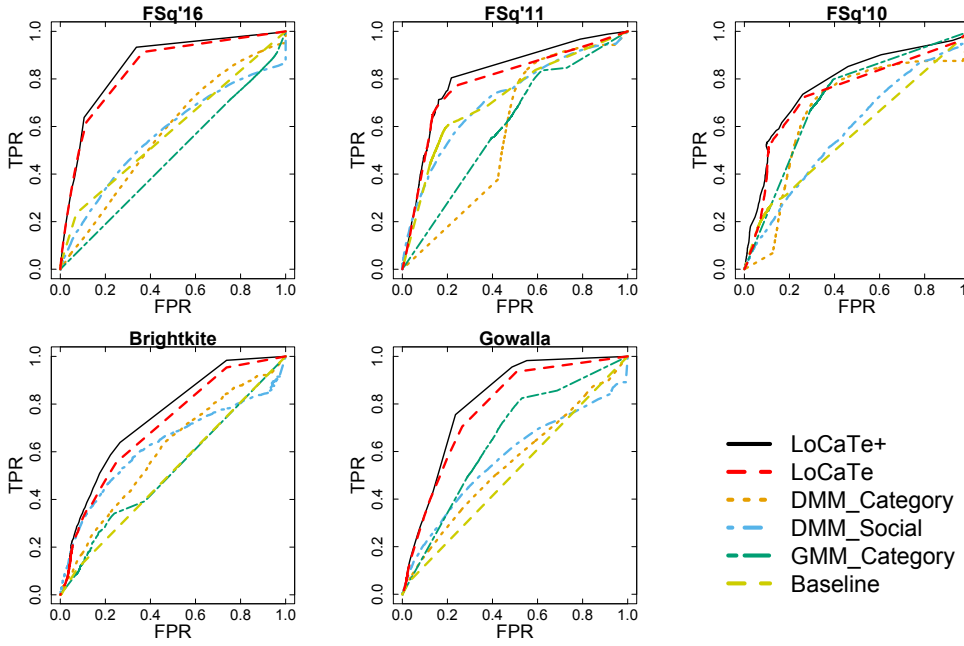


Fig. 7. ROC for different influence quantification models (AUC is in table 6)

we observe that *LoCaTe* models outperform DMM_Basic, DMM_Social and DMM_Category models by moderate to large margins, illustrating the effectiveness of our influence modelling framework. Moreover, the LoCaTe+ model further outperforms LoCaTe model, as the temporal correlation modeled in LoCaTe+ is specific to the location thus it is better in capturing the influence as compared to LoCaTe. The efficacy of the *LoCaTe* models is not only contributed by additional knowledge we gain from categories, but also due to the usage of temporal user-user correlation, modeled using exponential distributions and mutually exciting hawkes processes respectively. The Temporal correlation captures the social influence by modeling the time lag between checkins of the connected users. To verify this claim we computed the AUC with and without **Te** model (i.e. Temporal modeling). For FSq'16 the AUC for *LoCaTe* (*LoCaTe+*) is 0.839 (0.857) and **LoCa**(without Temporal modeling) is 0.752, this shows that **Te** model indeed captures social followship and that the mutually exciting Hawkes process modelling delivers improvements over the simpler exponential distribution based modelling. For the sake of brevity, we have not provided the expanded results, although similar trends were observed across all datasets. From these results, it may also be inferred that our **Lo**cation 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. The Lo and Te components along with semantic location modelling using category information is seen to provide significant gains in accuracy of influence quantification.

6.3.2 Parameter Tuning. Figure 8 (a) and (b) shows the variation in the AUC (Area Under the Curve) as the tuning parameter α (weighted parameter for Lo and Ca in eq (1)) and β_v (weighted parameter for user and global KDE model in eq (6)) varies, respectively. It can be observed that the

Datasets	Techniques							
	Baseline	GMM	GMM_category	DMM_basic	DMM_social	DMM_Category	LoCaTe	LoCaTe+
Fsq'16	0.582	0.599	0.473	0.521	0.568	0.573	0.839	**0.857**
Fsq'11	0.721	0.716	0.605	0.727	0.716	0.579	0.789	**0.816**
Fsq'10	0.575	0.718	0.717	0.699	0.588	0.671	0.741	**0.781**
Brightkite	0.517	0.526	0.534	0.601	0.627	0.494	0.707	**0.746**

Table 6. AUC (area under the curve) of different influence quantification models over different datasets

Datasets	Techniques							
	Baseline	GMM	GMM_category	DMM_basic	DMM_social	DMM_Category	LoCaTe	LoCaTe+
Fsq'16	0.008	0.035	0.031	0.027	0.036	0.035	0.038	**0.040**
Fsq'11	0.003	0.016	0.018	0.014	0.022	0.021	0.023	**0.033**
Fsq'10	0.006	0.065	0.060	0.086	0.021	0.020	0.093	**0.112**
Brightkite	0.008	0.031	0.030	**0.036**	0.031	0.024	0.032	**0.036**
Gowalla	0.007	0.028	0.021	0.012	0.027	0.025	0.032	**0.035**

Table 7. F-measure of different influence quantification models over different datasets

highest value of AUC is achieved close to 0.90 for all the datasets, giving less weightage to topic model. But, at $\alpha = 1$ the performance decreases sharply, thus it shows topic model is essential as it covers the zero probability cases and improves the overall performance of the LoCaTe (LoCaTe+) model.

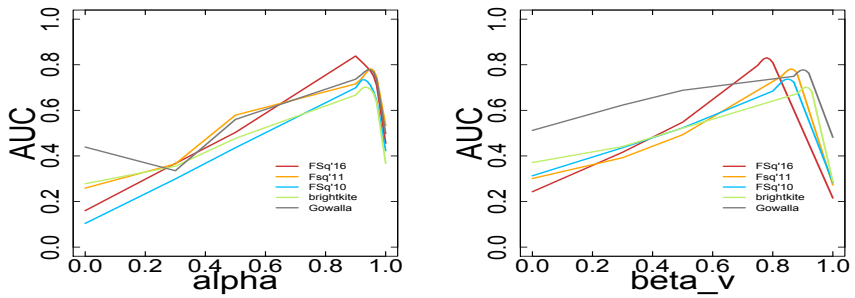


Fig. 8. AUC(Area Under the Curve) varies as the tuning parameter α and β_v varies from 0.0 to 1.0

6.3.3 F-measure. $PS(\ell, u, \rho \mid G)$ and $A(\ell, u)$, both being sets, allow comparing the methods based on the F-Measure [32]. The Table 7 shows F-measure of different influence quantification models on different datasets. F-measure is computed as follows:

$$F - measure = \frac{2.Precision.Recall}{Precision + Recall}$$

Overall, we observed that both the LoCaTe models perform better in terms of F-measure over other influence quantification models for all the datasets, except on Brightkite where DMM_Basic is seen to be neck-to-neck with LoCaTe+. It is notable that the temporal modelling in LoCaTe+ lead to very significant gains in F-measure over the basic LoCaTe model.

Datasets	Techniques							
	Baseline	GMM	GMM_category	DMM_basic	DMM_social	DMM_Category	LoCaTe	LoCaTe+
Fsq'16	19.2	3.5	446.8	**3.1**	5.1	508.3	3.5	4.2
Fsq'11	70.2	6.9	35.5	**3.2**	16.7	50.2	31.5	35.3
Fsq'10	17.0	6.1	151.5	**3.2**	10.8	15.6	5.4	7.5
Brightkite	1003.7	46.9	2626.6	**26.1**	154.4	1937.8	129.4	140.8
Gowalla	24.9	3.7	130.1	**2.8**	27.0	141.8	5.0	8.9

Table 8. Average time taken in execution of a testcase in (ms) for different influence quantification methods over different datasets

6.3.4 Execution Time. Table 8 shows average execution time (in milli seconds) of each test case using different influence quantification models on all the datasets. Overall, we observed that the LoCaTe models run slightly slower than the simple DMM_Basic, but remains faster than other methods considered. Moreover, LoCaTe+ is further slower because of the extra sophistication involved in modeling. On the other hand, GMM_Category and DMM_Category are significantly slower. It may be noted that within the LoCaTe framework, the location-affinity terms are user-specific and thus can be maintained in current state as the stream of check-ins arrive, and they simply need to be looked up at query time; this opens up possibilities for further efficiency improvements for the LoCaTe models, in real-time usage scenarios (our timings were based on an offline evaluation).

6.4 Evaluation on Location Promotion Task

In evaluating the location promotion task, our interest is in the quality of the set computed using Algorithm 1 based on using various underlying influence models. Unlike the influence quantification, this task is just location-specific (and not user-specific). For each location, based on the training data, location promotion is the task of finding a set of good seed users S, who are likely to lure a lot of their connections to the location. More formally, consider a target location ℓ, and the input parameter τ (the desired size of the output seed set, S), the influence quantification model M and the influence quantification threshold ρ that are passed to the location promotion algorithm. The goodness of S, as estimated from the test data, are the set of connections of S who visit the location ℓ, in test data. This is computed as:

$$I(S) = \{v \mid (u, v) \in E, u \in S \ \& \ v \text{ has visited } \ell \text{ in test data}\}$$

The size of the set $I(S)$ indicates the amount of collective influence that users across S have, in luring their connections to the target location. Accordingly, we simply use the size of $I(S)$, i.e., $|I(S)|$, as a measure of quality to evaluate the seed sets output by the various methods for the location promotion task.

For constructing the test set of target locations ℓ, we choose those locations that have a sizeable number of users checking-in, in the test set. This ensures that a reasonable sized $I(S)$ may be achieved, for good quality estimates of S, thus alleviating sparsity issues in the evaluation.

6.4.1 Results. Table 9 reports the results of $|I(S)|$ computed in the test data where ρ is set to 0.003. The threshold value of 0.003 is determined using the knee-point in the curve of $|I(S)|$ as we vary the value of ρ, following the method suggested in [5]. We observe that both the LoCaTe models perform better than the baselines in terms of $|I(S)|$ on all datasets but for FSq'16 where DMM_Social scores slightly better than the basic model but is overshadowed by LoCaTe+. The overall trends underline the effectiveness of LoCaTe framework in the location promotion task.

Datasets	Techniques						
	GMM	GMM_category	DMM_basic	DMM_social	DMM_Category	LoCaTe	LoCaTe+
Fsq'16	18.11	14.98	18.66	33.24	30.32	32.95	**35.91**
Fsq'11	20.42	24.91	26.74	34.35	30.17	35.06	**39.48**
Fsq'10	16.52	22.05	19.38	23.14	20.67	27.61	**29.33**
Gowalla	7.51	26.84	21.94	35.18	36.12	39.78	**43.44**

Table 9. $|I(S)|$ at $\rho = 0.003$ and $\tau = 5$

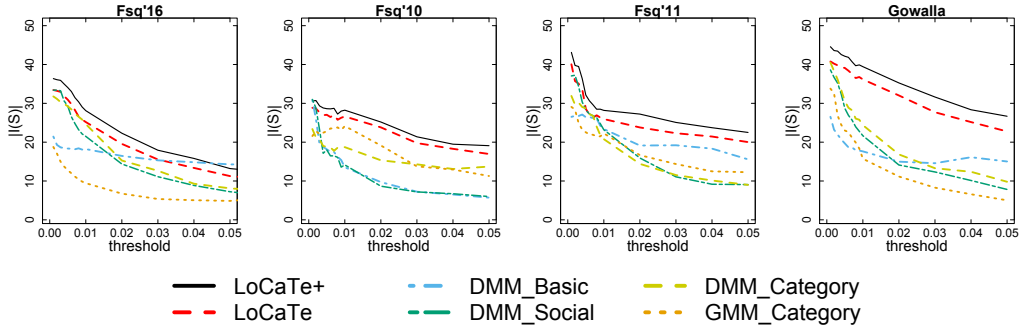


Fig. 9. number of influenced users at different thresholds at $\tau = 5$

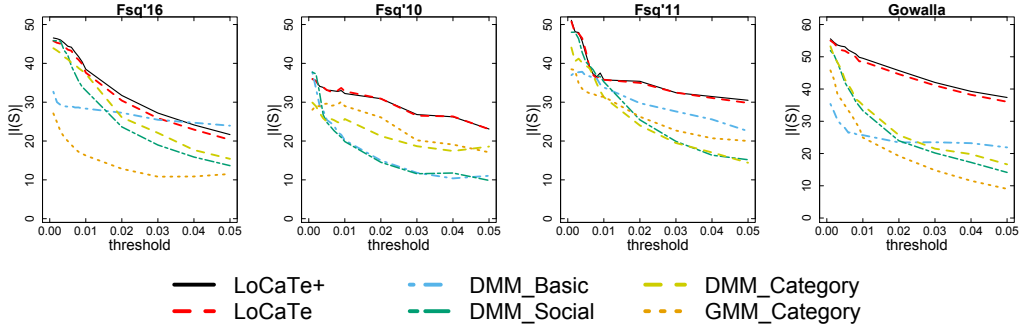


Fig. 10. number of influenced users at different thresholds at $\tau = 10$

6.4.2 Varying ρ. To understand the trends over varying ρ, we evaluate at different values of ρ (the influence quantification threshold) ranging from 0.001 to 0.05 at two different settings of seed set size, τ. The $|I(S)|$ numbers are plotted in Figures 9, and 10. It can be observed that LoCaTe models perform consistently better at all the threshold values, with the difference being exceedingly pronounced in the Gowalla dataset. This consistent performance is contributed to LoCaTe's capabilty to capture the influence in a better way. The trends were found to be similar for other values of τ; thus, we omitted those graphs for brevity.

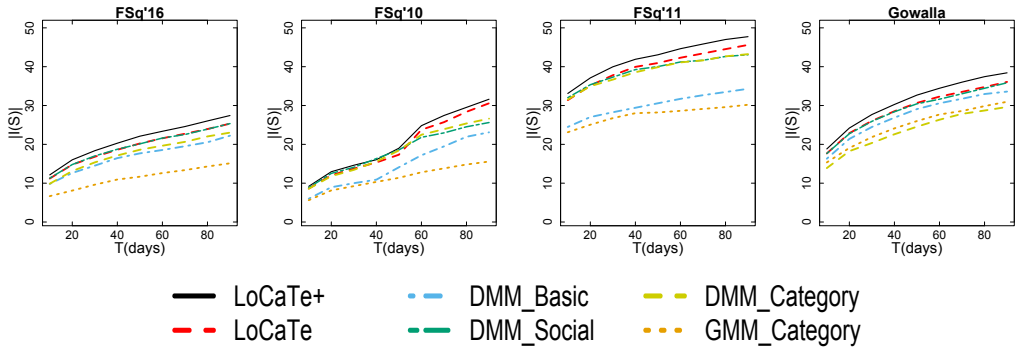


Fig. 11. number of influenced users at different time_windows at $\rho = 0.003$ and $\tau = 5$

6.5 Impact of Time Window on Location Promotion

Social Networks in general are dynamic in nature, and users' influence strength changes over time. For example, consider the seed user u visited the target location at time t_u and her follower visits the target location at time t_v such that $t_v > t_u$. It may happen that as $t_v \rightarrow \infty$, and the seed user does not contribute anymore towards the influence process. As a consequence of this assumption, we will end up getting seed users set which does not hold much value in influencing and activating their followers. In the previous section, the evaluation technique described does not consider temporal dynamics. Since the check-in activity that we consider is time based and it is possible that a user at some time in future may become useful/useless for the promotion of a specific location. Thus, while computing the set of influenced users I in the algorithm 1 we consider the time-window T upto which the influence persists, and the set of influenced users is computed as:

$$I = \{v | P_{u,\ell}(v) > \rho \ \& \ t_v - t_u < T\},$$

such that $(u, v) \in E$ and v has visited the target location ℓ. $P_{u,\ell}(v)$ is the influence score between u and v, and t_v and t_u are the timestamps when v and u visited the target location ℓ.

For the evaluation of the time window impact, we observe that with the time window constraint the influence period of a seed user may intersect with the test set. Thus, in order to make sure that the entire training data with the influence period lie within the global cut-off timestamp limits, we use last checked-in time stamp of the seed user for training without compromising on the test set. For instance, consider a target location ℓ, the global cut-off timestamp as Oct 1 and the time window T is 20 days. A candidate seed user u visits the target location ℓ on Sep 20 (this is the last check-in in the training data at ℓ by u) and u's follower v visits the ℓ on Oct 5; the influence period of u is till Oct 10, which intersects with the test data. If we want to choose a global cut-off timestamp where this intersection doesn't happen then how far we have to go backward in the training data is a question and if we go forward in the test data then we may have to compromise the size of the test data. Thus, for each candidate seed user u and its followers (like v) we use the training data until last checked-in time stamp of u, as it ensures sufficient and also same amount of data for training at all the time window sizes.

Figures 11, shows results of time based evaluation at different values of time window sizes, here ρ is set to 0.003. It can be observed that as the time window size increases from 10 to 90 days, the number of influenced users has also increased; the relative trends show that the LoCaTe models record higher number of influenced users consistently. To understand the phenomenon

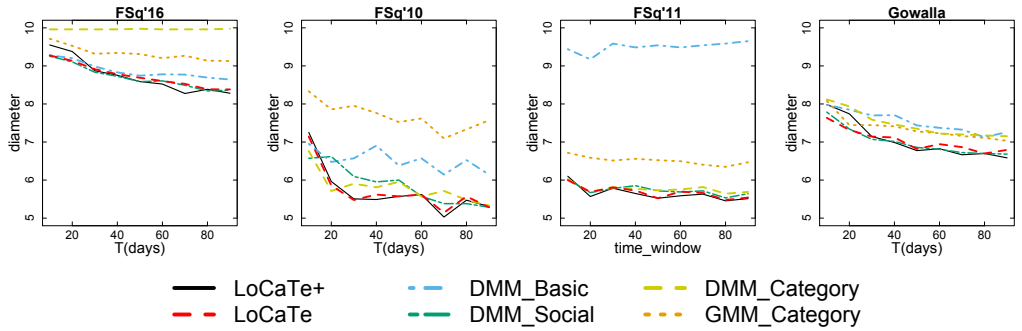


Fig. 12. ϕ at different time_windows atρ = 0.003 and τ = 5

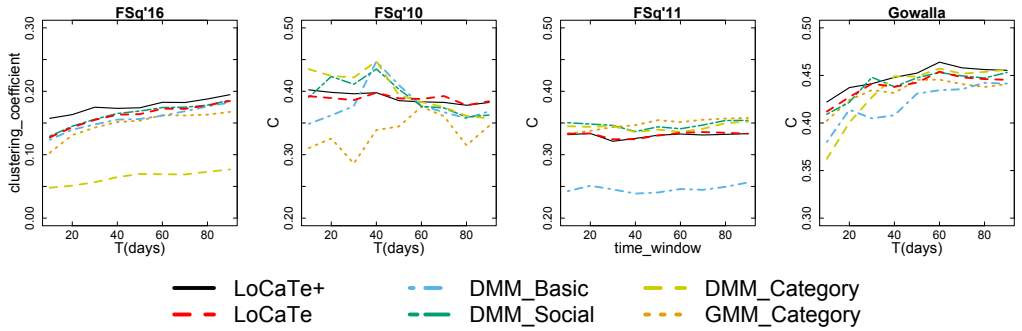


Fig. 13. C at different time_windows atρ = 0.003 and τ = 5

that *why the number of influenced users has increased with time window sizes*, we analyze the graph structural properties of the graph formed using the set of influenced users and seed users. We analyze the graph structural properties because we know that there exists community formation with information propagation in social networks [2, 15, 25, 38].

6.5.1 Graph Structural Analyses with Time. In this section, to analyze whether a certain location becomes prevalent in a community or does the check-in activities leads to community formation, we determine the diameter ϕ, clustering coefficient C, and average Degree Centrality C_D of the influencers (S) and influenced users'($I(S)$) graph $G_T(V_T, E_T)$, where $V_T = I(S) \bigcup S$ and $E_T = \{(u, v)|(u, v) \in E\}$.

ϕ and C: Figures 12 and 13 shows the results for diameter and clustering coefficient of the graph G_T with respect to the time window size T. It can be observed that as T increases the clustering coefficient C increases and the diameter ϕ decreases. Thus, with time as the influence propagates there exist community formation. Hence, a location becomes prevalent amongst a group of users.

Degree Centrality Test: We analyze the average Degree Centrality C_D of G_T computed using different quantification models to understand how much cohesive G_T does each model renders. Tables 10, 12, 11, and 13 shows that the LoCaTe models are able to render better average degree centrality of G_T. Note that, G_T is an unobserved graph and is formed while testing. Thus, we can conlude that LoCaTe models provide us with more cohesive unboserved graph as compared to other quantification models.

Time Window	Techniques						
	GMM	GMM_category	DMM_basic	DMM_social	DMM_category	LoCaTe	LoCaTe+
10	0.234	0.199	0.243	0.258	0.258	0.253	0.263
20	0.262	0.224	0.270	0.282	0.284	0.267	0.280
30	0.267	0.243	0.278	0.284	0.283	0.275	0.288
40	0.276	0.258	0.294	0.296	0.297	0.291	0.301
50	0.285	0.267	0.301	0.302	0.300	0.294	0.304
60	0.289	0.277	0.307	0.306	0.303	0.303	0.313
70	0.291	0.288	0.306	0.299	0.299	0.312	0.322
80	0.286	0.299	0.313	0.306	0.306	0.325	0.325
90	0.285	0.308	0.311	0.309	0.310	0.326	0.328

Table 10. Average Degree Centrality C_D of Influenced Users graph for FSq'16

Time Window	Techniques						
	GMM	GMM_category	DMM_basic	DMM_social	DMM_category	LoCaTe	LoCaTe+
10	0.327	0.376	0.397	0.400	0.385	0.403	0.423
20	0.339	0.386	0.395	0.399	0.391	0.409	0.419
30	0.347	0.393	0.401	0.399	0.403	0.404	0.424
40	0.346	0.383	0.391	0.390	0.406	0.400	0.420
50	0.346	0.388	0.392	0.394	0.408	0.416	0.422
60	0.343	0.376	0.387	0.389	0.407	0.416	0.426
70	0.346	0.374	0.385	0.389	0.403	0.413	0.424
80	0.343	0.374	0.388	0.391	0.401	0.417	0.427
90	0.340	0.373	0.391	0.390	0.401	0.416	0.425

Table 11. Average Degree Centrality C_D of Influenced Users graph for FSq'11

Time Window	Techniques						
	GMM	GMM_category	DMM_basic	DMM_social	DMM_category	LoCaTe	LoCaTe+
10	0.391	0.325	0.378	0.458	0.401	0.466	0.476
20	0.363	0.431	0.421	0.452	0.417	0.480	0.490
30	0.353	0.393	0.407	0.418	0.403	0.458	0.479
40	0.363	0.401	0.429	0.413	0.392	0.429	0.449
50	0.380	0.363	0.390	0.387	0.420	0.431	0.441
60	0.357	0.351	0.390	0.393	0.417	0.425	0.435
70	0.348	0.327	0.377	0.376	0.416	0.402	0.412
80	0.338	0.354	0.372	0.359	0.409	0.425	0.435
90	0.342	0.356	0.365	0.356	0.418	0.418	0.428

Table 12. Average Degree Centrality C_D of Influenced Users graph for FSq'10

6.6 Evaluation on Location Recommendation Task

For the evaluation of the Location Recommendation Task, first of all we consider all the locations in the training set as the candidate locations (that can be recommended), then we assign score to each candidate location using the scoring method as described in section 5.2. Next, we rank the locations based on the scores obtained and compare it against the actual checked-in location. *Recall* and *NDCG* are used as the evaluation metrics. For measuring the efficiency of our model

Time Window	Techniques						
	GMM	GMM_category	DMM_basic	DMM_social	DMM_category	LoCaTe	LoCaTe+
10	0.111	0.331	0.342	0.342	0.320	0.329	0.349
20	0.131	0.346	0.347	0.352	0.336	0.348	0.368
30	0.151	0.353	0.368	0.373	0.347	0.370	0.380
40	0.165	0.346	0.350	0.350	0.327	0.373	0.383
50	0.174	0.341	0.351	0.353	0.335	0.366	0.376
60	0.182	0.345	0.357	0.357	0.336	0.369	0.379
70	0.187	0.349	0.351	0.352	0.332	0.365	0.375
80	0.198	0.349	0.351	0.354	0.332	0.369	0.379
90	0.211	0.354	0.348	0.347	0.325	0.380	0.370

Table 13. Average Degree Centrality C_D of Influenced Users graph for Gowalla

	Datasets									
top-k	FSq'16		FSq'10		FSq'11		Brightkite		Gowalla	
	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++
5	0.163	0.084	0.420	0.399	0.168	0.124	0.378	0.288	0.203	0.078
10	0.392	0.254	0.613	0.564	0.330	0.289	0.420	0.355	0.270	0.135
20	0.602	0.482	0.692	0.667	0.556	0.467	0.480	0.417	0.366	0.224

Table 14. Recall at different values of $top - k$ for different datssets

	Datasets									
top-k	FSq'16		FSq'10		FSq'11		Brightkite		Gowalla	
	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++	LoCaTe+	GeoMF++
5	0.122	0.084	0.160	0.138	0.118	0.093	0.151	0.126	0.111	0.089
10	0.130	0.112	0.212	0.189	0.150	0.122	0.211	0.174	0.132	0.106
20	0.157	0.130	0.252	0.220	0.208	0.187	0.228	0.208	0.151	0.127

Table 15. NDCG at different values of $top - k$ for different datasets

we compare our results against GeoMF++ (Joint Geographical model and Matrix Factorization for Location Recommendation) [20]. The evaluation is performed over 2608178, 495616, 148424, 324091, and 386203 number of test cases for FSq'16, FSq'11, FSq'10, Brightkite, and Gowalla dataset, respectively. Table 14 and table 15 reports the Recall and NDCG obtained on the Personalized Location Recommendation Task at different $top - k$ values using LoCaTe+ model and GeoMF++ [20], respectively. It can be observed that LoCaTe performs significantly better than GeoMF++ over all the datasets. This we believe is because LoCaTe incorporates additional information i.e. Category Affinity and Temporal Information, while GeoMF++ only models users' location preferences based on its mobility. The trends were found similar for the LoCaTe model, thus we omitted those results for brevity. Table 16 report values of tuning parameters γ_1, γ_2 and γ_3 used for the above evaluation for all the datasets learned for the LoCaTe+ model. We also performed grid search using grid sizes of 0.01 to demonstrate the chosen parameter values return the best performance. Table 17 reports the Recall result for the FSq'10 dataset of the grid search at different values of γ_1, γ_2 and γ_3. We only report few values, although an exhaustive grid search was performed.

7 CONCLUSION

In this paper, we proposed a framework *LoCaTe* that incorporates not only the traditional user mobility models but also temporal correlation within the social network of users as well as the

Datasets	γ_1	γ_2	γ_3
FSq'16	0.05	0.68	0.28
FSq'11	0.10	0.65	0.25
FSq'10	0.10	0.70	0.20
Brightkite	0.15	0.70	0.15
Gowalla	0.12	0.70	0.18

Table 16. γ_1, γ_2 and γ_3 used for different datasets

γ_1	γ_2	γ_3	Recall
0.2	0.3	0.5	0.601
0.5	0.3	0.2	0.578
0.5	0.1	0.4	0.570
0.2	0.6	0.2	0.605
0.1	0.7	0.2	**0.613**
0.2	0.7	0.1	0.604

Table 17. Recall at different values of weight parameters at $top - k$ = 10 for FSq'10 dataset

affinity of users to a location based on semantics of the location (i.e., categories). We developed two models based on the framework; a basic model, also called *LoCaTe*, that uses exponential distributions to model temporal correlation between users, and a more advanced model, called *LoCaTe+* that makes use of mutually exciting hawkes processes. We empirically evaluated our approaches using the influence quantification task, and the more general problem of location promotion over a number of real-world LBSN data with a large number of users and spanning more than a year. For the influence quantification task we observed that *LoCaTe* models demonstrated more than 54% improvements over state-of-the-art methods. Further for the location promotion setting, *LoCaTe* models were seen to be able to predict the graph of influenced users with better degree centrality. The gains transferred nicely over to the location recommendation task as well, where *LoCaTe* models provided more than 50% improved recommendation over existing methods. In our future work, we would like to further explore the diffusion process of location based influence. Moreover, we would also like to enhance the LoCaTe framework to encompass location attributes from other sources which could be integrated and leveraged.

REFERENCES

[1] [n. d.]. Journal of Futures Markets: Volume 38, Number 11, November 2018. *Journal of Futures Markets* 38, 11 ([n. d.]), 1283–1283. https://doi.org/10.1002/fut.21878 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/fut.21878

[2] Nicola Barbieri, Francesco Bonchi, and Giuseppe Manco. 2013. Cascade-based community detection. In *WSDM*. ACM, 33–42.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.

[4] Panagiotis Bouros, Dimitris Sacharidis, and Nikos Bikakis. 2014. Regionally influential users in location-aware social networks. In *SIGSPATIAL/GIS*. ACM, 501–504.

[5] Wei Chen, Chi Wang, and Yajun Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *KDD*. ACM, 1029–1038.

[6] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *KDD*. ACM, 1082–1090.

[7] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2013. Modeling temporal effects of human mobile behavior on location-based social networks. In *CIKM*. ACM, 1673–1678.

[8] Huiji Gao, Jiliang Tang, and Huan Liu. 2012. Exploring Social-Historical Ties on Location-Based Social Networks. In *ICWSM*. The AAAI Press.

[9] Amit Goyal, Francesco Bonchi, and Laks V. S. Lakshmanan. 2010. Learning influence probabilities in social networks. In *WSDM*. ACM, 241–250.

[10] Alan G. Hawkes. 1971. Point Spectra of Some Mutually Exciting Point Processes. *Biometrika* 58, 1 (1971), 83–90.

[11] Tieke He, Hongzhi Yin, Zhenyu Chen, Xiaofang Zhou, Shazia W. Sadiq, and Bin Luo. 2016. A Spatial-Temporal Topic Model for the Semantic Annotation of POIs in LBSNs. *ACM TIST* 8, 1 (2016), 12:1–12:24.

[12] David Kempe, Jon M. Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *KDD*. ACM, 137–146.

[13] David Kempe, Jon M. Kleinberg, and Éva Tardos. 2005. Influential Nodes in a Diffusion Model for Social Networks. In *ICALP (Lecture Notes in Computer Science)*, Vol. 3580. Springer, 1127–1138.
[14] Patrick J Laub, Thomas Taimre, and Philip K Pollett. 2015. Hawkes Processes. *arXiv.org* (2015). arXiv:1507.02822v1
[15] Jure Leskovec, Lada A. Adamic, and Bernardo A. Huberman. 2007. The dynamics of viral marketing. *TWEB* 1, 1 (2007), 5.
[16] Cheng-Te Li and Hsun-Ping Hsieh. 2015. Geo-Social Media Analytics. In *WWW (Companion Volume)*. ACM, 1533–1534.
[17] Guoliang Li, Shuo Chen, Jianhua Feng, Kian-Lee Tan, and Wen-Syan Li. 2014. Efficient location-aware influence maximization. In *SIGMOD Conference*. ACM, 87–98.
[18] Xutao Li, Gao Cong, Xiao-Li Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. 2015. Rank-GeoFM: A Ranking Based Geographical Factorization Method for Point of Interest Recommendation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 433–442. https://doi.org/10.1145/2766462.2767722
[19] Defu Lian, Xing Xie, Vincent W. Zheng, Nicholas Jing Yuan, Fuzheng Zhang, and Enhong Chen. 2015. CEPR: A Collaborative Exploration and Periodically Returning Model for Location Prediction. *ACM Trans. Intell. Syst. Technol.* 6, 1, Article 8 (April 2015), 27 pages. https://doi.org/10.1145/2629557
[20] Defu Lian, Kai Zheng, Yong Ge, Longbing Cao, Enhong Chen, and Xing Xie. 2018. GeoMF++: Scalable Location Recommendation via Joint Geographical Modeling and Matrix Factorization. *ACM Trans. Inf. Syst.* 36, 3, Article 33 (March 2018), 29 pages. https://doi.org/10.1145/3182166
[21] Moshe Lichman and Padhraic Smyth. 2014. Modeling human location data with mixtures of kernel densities. In *KDD*. ACM, 35–44.
[22] Ankita Likhyani, Srikanta Bedathur, and Deepak P. 2017. LoCaTe: Influence Quantification for Location Promotion in Location-based Social Networks. In *IJCAI*. ijcai.org, 2259–2265.
[23] Ankita Likhyani, Deepak Padmanabhan, Srikanta J. Bedathur, and Sameep Mehta. 2015. Inferring and Exploiting Categories for Next Location Prediction. In *WWW (Companion Volume)*. ACM, 65–66.
[24] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. 2013. Learning Geographical Preferences for Point-of-interest Recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '13)*. ACM, New York, NY, USA, 1043–1051. https://doi.org/10.1145/2487575.2487673
[25] Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King. 2008. Mining social networks using heat diffusion processes for marketing candidates selection. In *CIKM*. ACM, 233–242.
[26] Rohan Miller and Natalie Lammas. 2010. Social media and its implications for viral marketing. *Asia Pacific Public Relations Journal* (2010).
[27] Brad L Neiger, Rosemary Thackeray, Sarah A Van Wagenen, Carl L Hanson, Joshua H West, Michael D Barnes, and Michael C Fagen. 2012. Use of social media in health promotion: purposes, key performance indicators, and evaluation metrics. *Health promotion practice* (2012).
[28] John A. Nelder and Roger Mead. 1965. A simplex method for function minimization. *Computer Journal* 7 (1965), 308–313.
[29] M. E. J. Newman. 2005. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics* (2005).
[30] Anastasios Noulas, Salvatore Scellato, Neal Lathia, and Cecilia Mascolo. 2012. Mining User Mobility Features for Next Place Prediction in Location-Based Services. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining (ICDM '12)*. IEEE Computer Society, Washington, DC, USA, 1038–1043. https://doi.org/10.1109/ICDM.2012.113
[31] Huy Pham and Cyrus Shahabi. 2016. Spatial influence - measuring followship in the real world. In *ICDE*. IEEE Computer Society, 529–540.
[32] D. M. W. Powers. 2011. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies* (2011).
[33] Jacob Ratkiewicz, Michael Conover, Mark R. Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. 2011. Detecting and Tracking Political Abuse in Social Media. In *ICWSM*. The AAAI Press.
[34] Muhammad Aamir Saleem, Rohit Kumar, Toon Calders, Xike Xie, and Torben Bach Pedersen. 2017. Location Influence in Location-based Social Networks. In *WSDM*. ACM, 621–630.
[35] B. W. Silverman. 1986. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, London.
[36] Weiqing Wang, Hongzhi Yin, Shazia Wasim Sadiq, Ling Chen, Min Xie, and Xiaofang Zhou. 2016. SPORE: A sequential personalized spatial item recommender system. In *ICDE*. IEEE Computer Society, 954–965.
[37] Xiaoyang Wang, Ying Zhang, Wenjie Zhang, and Xuemin Lin. 2016. Distance-aware influence maximization in geo-social network. In *ICDE*. IEEE Computer Society, 1–12.
[38] Yu Wang, Gao Cong, Guojie Song, and Kunqing Xie. 2010. Community-based greedy algorithm for mining top-K influential nodes in mobile social networks. In *KDD*. ACM, 1039–1048.
[39] Hao-Hsiang Wu and Mi-Yen Yeh. 2013. Influential Nodes in a One-Wave Diffusion Model for Location-Based Social Networks. In *PAKDD (2) (Lecture Notes in Computer Science)*, Vol. 7819. Springer, 61–72.

[40] Dingqi Yang, Daqing Zhang, and Bingqing Qu. 2016. Participatory Cultural Mapping Based on Collective Behavior Data in Location-Based Social Networks. *ACM TIST* 7, 3 (2016), 30:1–30:23.
[41] Shuang-Hong Yang and Hongyuan Zha. 2013. Mixture of Mutually Exciting Processes for Viral Diffusion. In *Proceedings of the 30th International Conference on Machine Learning, ICML*. 1–9.
[42] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*. ACM, 325–334.
[43] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. 2013. LCARS: a location-content-aware recommender system. In *KDD*. ACM, 221–229.
[44] Quan Yuan, Gao Cong, Kaiqi Zhao, Zongyang Ma, and Aixin Sun. 2015. Who, Where, When, and What: A Nonparametric Bayesian Approach to Context-aware Recommendation and Search for Twitter Users. *ACM Trans. Inf. Syst.* 33, 1 (2015), 2:1–2:33.
[45] Chao Zhang, Lidan Shou, Ke Chen, Gang Chen, and Yijun Bei. 2012. Evaluating geo-social influence in location-based social networks. In *CIKM*. ACM, 1442–1451.
[46] Kaiqi Zhao, Lisi Chen, and Gao Cong. 2016. Topic Exploration in Spatio-Temporal Document Collections. In *SIGMOD Conference*. ACM, 985–998.
[47] Yu Zheng and Xing Xie. 2011. Learning travel recommendations from user-generated GPS traces. *ACM TIST* 2, 1 (2011), 2:1–2:29.
[48] Tao Zhou, Jiuxin Cao, Bo Liu, Shuai Xu, Ziqing Zhu, and Junzhou Luo. 2015. Location-Based Influence Maximization in Social Networks. In *CIKM*. ACM, 1211–1220.
[49] Wen-Yuan Zhu, Wen-Chih Peng, Ling-Jyh Chen, Kai Zheng, and Xiaofang Zhou. 2015. Modeling User Mobility for Location Promotion in Location-based Social Networks. In *KDD*. ACM, 1573–1582.
[50] Wen-Yuan Zhu, Wen-Chih Peng, Ling-Jyh Chen, Kai Zheng, and Xiaofang Zhou. 2016. Exploiting Viral Marketing for Location Promotion in Location-Based Social Networks. *TKDD* 11, 2 (2016), 25:1–25:28.