User Interest and Influence Mining in Location-based Social Networks

By

Ankita Likhyani

under the supervision of

Dr. Srikanta Bedathur

Indian Institute of Technology-Delhi

January, 2022

User Interest and Influence Mining in Location-based Social Networks

By

Ankita Likhyani

submitted

in partial fulfillment of the requirements

for the award of the degree of

Doctor of Philosophy

to the

Indraprastha Institute of Information Technology-Delhi

Okhla Industrial Estate, Phase III

New Delhi, India - 110020

January, 2022

Indraprastha Institute of Information Technology, Delhi

Okhla Industrial Estate, Phase III

New Delhi-110020, India

Certificate

This is to certify that the thesis titled **User Interest and Influence Mining in Location-based Social Networks** being submitted by **Ankita Likhyani** to the Indraprastha Institute of Information Technology-Delhi, for the award of the degree of **Doctor of Philosophy**, is an original research work carried out by her under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

January, 2022

Dr. Srikanta Bedathur

Associate Professor,

Deptt. of Computer Science and Engineering,

Indian Institute of Technology-Delhi,

New Delhi-110016, India.

Indraprastha Institute of Information Technology, Delhi

Okhla Industrial Estate, Phase III

New Delhi-110020, India

Declaration

This is certified that the thesis entitled **User Interest and Influence Mining in Location-based Social Networks** being submitted by me to the **Indraprastha Institute of Information Technology-Delhi**, for the award of degree of **Doctor of Philosophy**, is a bonafide work carried out by me. This research work has been carried out under the supervision of **Dr. Srikanta Bedathur**.

The study pertaining to this thesis has not been submitted in part or in full, to any other University or Institution for the award of any other degree.

January, 2022

Ankita Likhyani

PhD student,

Deptt. of Computer Science and Engineering,

Indraprastha Institute of Information Technology-Delhi,

New Delhi-110020, India.

Small Steps leads to Big Success.

Anonymous

To my parents for believing in me.

Acknowledgment

There are a lot of people who helped me directly or indirectly in my Ph.D., to whom I wish to say, “thanks”. All of you have made me reach this point that I can write my Ph.D. thesis.

First of all, I would like to extend my heartiest gratitude towards my supervisor, Dr. Srikanta Bedathur for his constant support throughout my Ph.D. He consistently gave me time for detailed technical discussions and my paper reviews even after his busy schedule. The discussions with him have always been fruitful, insightful, and thought-provoking. I would like to thank him for his continuous technical/non-technical help, his feedback, and encouragement at all times during my Ph.D. I would like to thank him for his guidance and motivation, which enabled me to complete this thesis.

Next, I would like to thank Dr. Deepak Padmanabhan for his continuous guidance throughout my Ph.D. journey. I can not imagine myself reaching this stage without his support and suggestions. I would also like to thank Dr. PK Srijith for his guidance in the area of Point Processes and Variational Inference.

I heartily thank all my family members for their consistent support in every event of my life. I could not reach this stage without their support. I would like to thank my husband Mr. Abhinav Grover for consistently pushing me to finish my Ph.D., and having similar anticipation like me for my paper notifications.

I would like to thank my friends, Dr. Ayushi Rastogi, Dr. Venkatesh Vinayakrao, Dr.Monika Gupta, Dr. Haroon Rashid, Sangeeth Nila, Megha Agarwal, Dhriti Khana, Surabhi Garg Ishant Shanu, Pravin Nagar, and Sonia Soubam for their help and support in some way or the other. I call them my Ph.D. family members. Sonia Soubam has been a special part of my Ph.D. time. I thank her for the continuous technical/non-technical discussions and her help at all the odd paths during my Ph.D. The consistent afternoon tea discussions with her were something I looked up to, and to come to institute every day.

I would also like to thank Dr. Vikram Goyal and Dr. Rahul Purandare for providing institute-related help and support. I would also like to thank Prof. Ponnurangam Kumaraguru and Dr. Tanmoy Chakraborty for collaborating and letting me guide their students. I would also like to thank Dr. Sanjit Kaul and Dr. Saket Anand for the technical discussions in the areas of Machine Learning and Random Processes.

Lastly, I would like to thank all the members of IIIT-Delhi for providing such a great research environment, and especially thanks to Priti, Anshu, Prosenjit, Amit, and Gaurav for their friendly behavior and fast processing of every admin and finance-related issues. I would also like to thank TCS for providing financial support during my Ph.D. journey.

Ankita Likhyani

Abstract

Over the past decade with the explosion of smartphones and pervasive usage of data connectivity, location-based services have increasingly become popular. As a result, Location-based Social Network(LBSN) such as Foursquare, Facebook Places, Brightkite and Gowalla have emerged. These platforms provide users not only to connect, share, and interact but also allow users to share their check-in information with their friends. These networks typically do not expose the check-in information of users due to privacy concerns. Although, popular LBSNs such as Foursquare and Brightkite have their datasets publicly available but are anonymized, wherein only the latitude and longitude of each check-in are available. The latitude and longitude information only provides users' spatial preferences but to know their taste, knowing the location type such as restaurant, multiplex, fitness center, etc. would be essential. In the previous work, the location categories (e.g. restaurant, multiplex, etc.) were overlooked because of the unavailability of this information. In this thesis, we brought in the new direction of inferring categories and leveraging them for Location-based applications such as Location Prediction, Location Promotion, Influence Maximization, and Community Detection.

We align the location information in different networks to infer categories first at a coarse level (i.e. multiple categories for a location) for the publicly available datasets. Moreover, we also collect LBSNs data from Foursquare through Twitter spatio-temporal posts, where we also obtain the categories along with check-in location, time, and users' social connections. The dataset[1] is released publicly for researchers. We call it the fine-grained category information as we have a single category associated with each location.

The crucial task at hand is to model all this heterogeneous information i.e. spatial, temporal, and categorical to leverage it for location-based applications. We propose models that jointly model spatial, temporal, and categorical features together, and show that the use of auxiliary information and joint modeling provides improvement over state-of-the-art methods. In this thesis, we propose three semi-supervised machine learning models: 1) Category Language Model for next location prediction, 2) LoCaTe and LoCaTe+ for quantifying the influence between two users, and 3) CoLAB for determining implicit communities and to model the information diffusion process in LBSNs.

[1] https://goo.gl/ayzehx

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Location Based Social Networks

A social network is a social structure made up of individuals connected by one or more specific types of inter-dependency, such as friendship, common interests, and shared knowledge. Generally, a social networking service builds on and reflects the real-life social networks among people through online platforms such as a website, providing ways for users to share ideas, activities, events, and interests over the Internet. The increasing availability of location-acquisition technology (for example GPS[1] and Wi-Fi) empowers people to add a location dimension to existing online social networks. For example, Location-based Social Networks(LBSNs) such as Foursquare [2], and Facebook Places [3] provide platforms to users not only to connect, share and interact but also to share their check-in (or visit) information with their friends. Typical Location-Based Social Networks (LBSNs) allow users to simply share the location of their visit in a check-in post, optionally allowing one to augment the check-in with additional text and/or media. Figure 1.1 illustrates the central role played by the location information in Foursquare, a popular LBSN. The check-in history in the leftmost screen is represented as a sequence of locations along with the timestamp,

[1] `https://en.wikipedia.org/wiki/Global`
[2] `https://foursquare.com/`
[3] `https://www.facebook.com/`

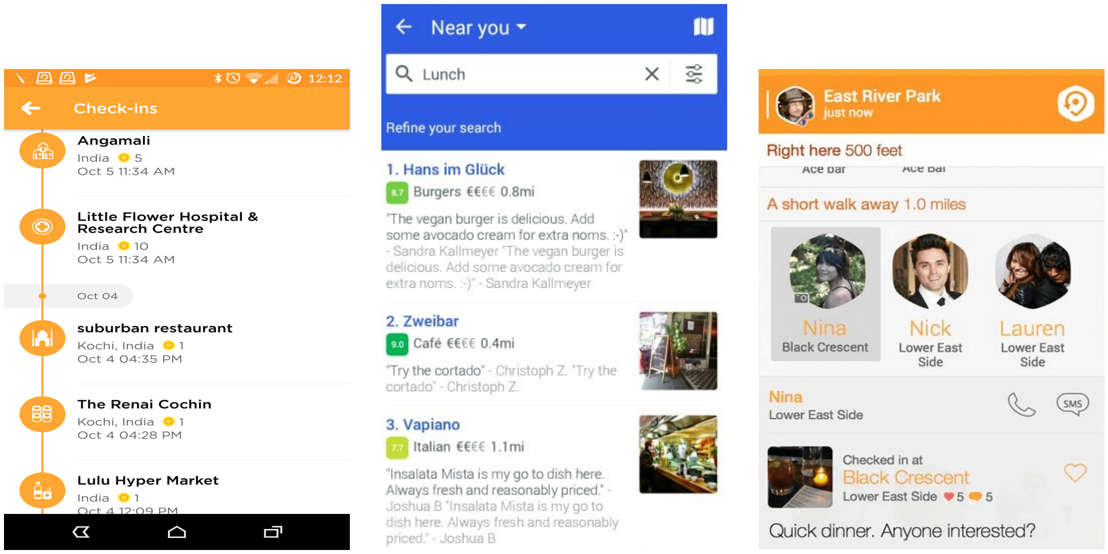


Figure 1.1: Location Usage in Foursquare

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. This implicitly become as the primary factor for the search that filters out other locations except eating out places that provide lunch. The third screen indicates a listing of connections sorted according to the distance from the user.

The three major components of LBSNs include[4]:

1. **Social** drives traffic through interaction and sharing information among connections. Smartphone users spend the bulk of their time on social networks which can drive a lot of retail traffic. 80% of smartphone users access social networking sites on their devices and 55% of smartphone users access social networking sites at least once per day on their devices. Figure 1.2 shows the average minutes per week smartphone users spend on social networking sites.

2. **Local** implies venues, check-ins, and deals. Local drives action such as users look at what's nearby and then make their visits as shown in figure 1.3

3. **Mobile** drives opportunity as anywhere, anytime an action can be taken. Most of

[4] `http://www.cruc.es/the-retailers-guide-to-solomo-infographic/`

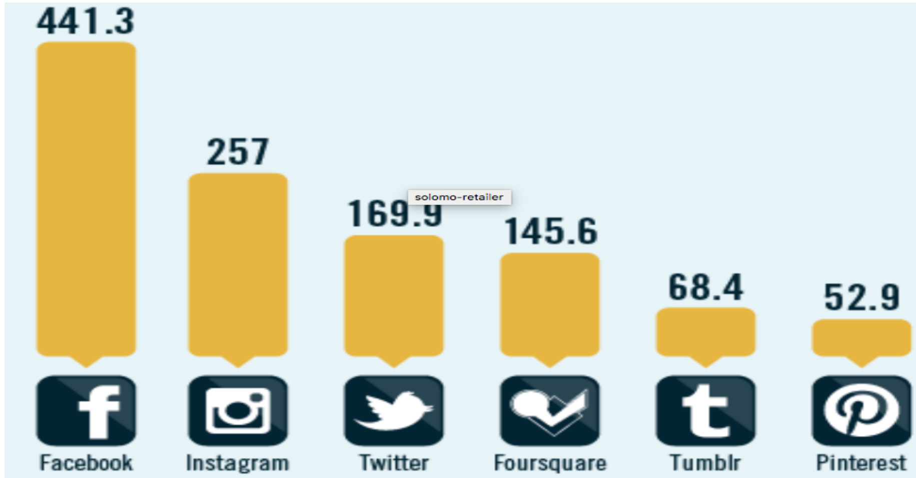


Figure 1.2: Average minutes per week smartphone users spend on social networking sites (http://www.cruc.es/the-retailers-guide-to-solomo-infographic/)

the LBSNs have mobile applications which makes it more engaging. 81.5 % of smartphone users spend time using mobile apps while only 18.5 % of smartphone users spend time using mobile web browsers. Thus, whenever a user visits a location, makes a check-in on the mobile apps of LBSNs.

The LBSNs have gained attention because 1) users' check-in bridges the Geo-social gap between the real world and the online social network 2) easy accessibility to spatio-temporal mobility data. Users choose to broadcast their check-ins to Twitter while using a mobile-based app from Foursquare, Swarmapp. This provides us an opportunity to capture their check-ins by crawling tweets with the keyword swarmapp.com on the Twitter public streaming API [5]. We collect the data by first extracting the user-IDs 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]. As a result, enormous amounts of users' spatio-temporal mobility information is available - e.g., Foursquare announced in early 2018 that it collects more than 3 billion locations "check-ins" every month from its more than 25 million users [7]

[5]https://developer.twitter.com/en/docs
[6]https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.HTML
[7]https://bit.ly/2BdhnnP (accessed in February 2019)

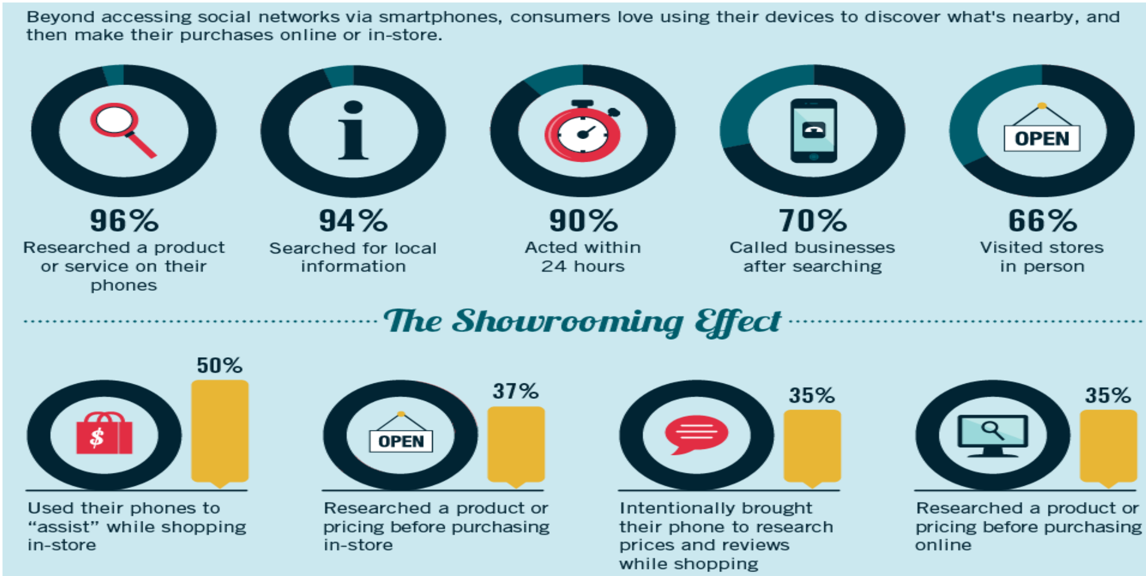


Figure 1.3: Users' search for what's nearby and then plan their visits (http://www.cruc.es/the-retailers-guide-to-solomo-infographic/)

1.2 Applications

The availability of enormous spatio-temporal mobility data has attracted a lot of researchers in mining LBSN data. Applications of mining spatio-temporal data (available from disparate sources) are divided into two broad categories based on the scope of the application i.e. for a single user or a mass of users as follows:

1. **Micro Level :**

   - Location / Event recommendation: This includes a recommendation for restaurants, museums, or other points of interest or events near the user's location. Most of the recommender systems make use of users' check-in history available from LBSNs and current location from a mobile device, for the recommendation.

   - Targeted marketing and location promotion: For example, knowing that a specific user's next check-in is likely to be at a cinema could be used to prioritize movie ticket offers to be sent to her.

- Wearable devices application: Wearable devices such as smart bands and smartwatches connected with smartphones and mobile devices are used to collect data and process it to extract information such as daily physical activities and health conditions. These include application in healthcare domain [23, 128, 142].

2. **Macro Level :** For macro level applications data is mostly collected using GIS (Geographical Information Systems) and sometimes using mobile devices as well.

   - Urban Computing : Applications such as identifying functional regions, and diagnosing transportation problems [87, 55] .
   - Disaster Management : For resource distribution and rescue planning mobile information of users through crowd sourcing is made use of [78].
   - Environmental Informatics : To detect air and noise pollution mobile information along with weather conditions are utilized [24].

In this work, we only focus on a micro-level application that focuses more on a single user rather than a mass of users. For example, we focus on applications such as next location prediction of a user, quantifying influence between two users, and community detection. Moreover, the data for all these applications is collected from LBSNs and not from social theories, geographical information systems, and sensor networks. In this work, we only focus on the users' interplay with their online social networks and their geographical footprints.

1.3 Challenges Involved

Mining spatio-temporal, contextual and social information from LBSNs creates new opportunities, but its unique properties also bring new challenges. In this section, we present some of the challenges addressed in this thesis:

1.3.1 Location Context Awareness

1. *Granularity of location information:* A user's current location can be represented at different levels of granularity (the hierarchical property of locations). Choosing a proper granularity for the recommendation scenario is important and challenging. For instance, we should use a finer granularity when recommending restaurants to a user, while a relatively coarse granularity (like in a city or state) for local news recommendations.

2. *Distance based context of location:* People are more likely to visit nearby locations than distant ones. However, the quality of location (for example, a restaurant), user preferences, and influence from friends also matter. For example, the user might be recommended beaches when traveling to Goa, even though the user prefers indoor activities more than beaches typically.

3. *Sequential property of locations:* A user's current location affects future travel decisions. For instance, the majority of people visiting Gateway Of India, Mumbai will subsequently visit Taj Hotel, Mumbai or a dessert recommendation may be appropriate after visiting certain restaurants. Discovering these sequential relations and incorporating them into prediction or recommendation tasks presents subtle challenges.

4. *User's preferences:* A user's preferences span multiple kinds of interests, such as shopping, cycling, and arts. Also, user's preferences have hierarchies and granularity, such as ”Food” → ”Italian food” → ”White Sauce pasta”. Thus, we can observe that user's preferences are constantly evolving (and are location-dependent).

5. *Data Sparsity:* As users do not share their locations everywhere, a full set of a user's location history does not exist. Learning a user's preferences from sparse location data is a challenging task.

1.3.2 Heterogeneous Feature Set

A location is not only an additional dimension of information about the user but also an important object in the LBSN. Inferring the similarity or correlation between two objects in a heterogeneous graph must incorporate the information from related nodes of other types. For instance, determining the connection between two users in an LBSN needs to involve the user-location and location-location relations in addition to the user-user relation. A location shared by two users could be an evidence of similarity, or it could simply indicate that a location is very popular. Only a careful analysis can determine which case holds, and to what extent it should influence the strength of the connection between the users.

1.4 Contributions

The key contributions made in this thesis include:

- We bring in the new direction of inferring categories (i.e. type of location such as restaurant, club, cinema, gym etc.) and leveraging it for Location based applications such as location prediction[70], location promotion[67], influence maximization[68] and community detection[69].

- We align location in different networks to infer categories at a coarse level (i.e. multiple categories for a location) for the existing publicly available data sets. Moreover, we also collect LBSNs data from Foursquare through Twitter spatio-temporal posts (as discussed in the later chapter), where we also obtain the categories along with the check-in location, time, and user's social connections. The data set is made publicly available [8]. We call it fine-grained category information as we have a single category associated with each location.

- The crucial task at hand is to model all these heterogeneous information i.e. spatial,

[8] https://goo.gl/ayzehx

temporal, and categorical to leverage it for location-based applications. We propose models that jointly model spatial, temporal, and categorical features together, and show auxiliary information and joint modeling provides improvement over state-of-the-art methods.

- In this work, we propose semi-supervised machine learning models: 1) Category Language Model for next location prediction, 2) LoCaTe and LoCaTe+ for quantifying influence between two users, and 3) CoLAB for determining implicit communities and to also model the information diffusion process. LoCaTe+ and CoLAB are mutually exciting point process[40] based techniques, using which we can model the influence of one user on another user by explicitly modeling different types of triggering events for a user to check in at a location.

1.5 Problems Addressed

At a high level, in this work, we identified the potential of using auxiliary features such as the category of location (e.g. restaurant, coffee shop) to model users' interests for a location along with spatial and temporal footprints and showed its applicability on problems such as location prediction, location promotion, location recommendation, and influence maximization. Later, we also utilize these features to come up with a model that could model the diffusion process of location adoption behavior and identify communities in LBSN. The key questions addressed in this thesis are:

- Q1. How to effectively collect and curate spatio-temporal data with auxiliary information?

- Q2. How to improve the quantification of influence strength between users through auxiliary data?

- Q3. How to identify the latent community of users based on their check-in spatio-temporal data?

1.5.1 Effective Spatio-Temporal Data Augmentation

Q1. How to effectively collect and curate spatio-temporal data with auxiliary information?

Building sophisticated models for Location-based services, by enriching the check-in data by combining them with the information from other sources is challenging due to the limited data that these LBSNs expose due to privacy concerns. In Chapter 3, we propose a framework to use the location data from LBSNs, combine it with the data from maps for associating a set of venue categories with these locations. For example, if the user is found to be checking in at a mall that has cafes, cinemas, and restaurants according to the map, all this information is associated. This category information is then leveraged to predict the next check-in location by the user. Our experiments with publicly available check-in data set show that this approach improves on the state-of-the-art methods for the location prediction task.

1.5.2 Influence Modeling with Augmented Spatio-Temporal Data

Q2. How to improve the quantification of influence strength between users through auxiliary data?

Quantifying influence between users in LBSNs is useful in various settings such as location promotion, personalized recommendations, mobility pattern prediction, etc. In Chapter 4, 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 on 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.

1.5.3 Latent Community Mining from Spatio-Temporal Activity

Q3. How to identify the latent community of users based on their check-in spatio-temporal data?

The large traces of users' spatio-temporal footprints often manifest in hidden (possibly overlapping) communities of users with similar interests. Inferring these implicit communities is crucial for forming user profiles for improvements in recommendation and prediction tasks. In Chapter 5, we propose a model based on spatio-temporal point processes in continuous time but discrete space of locations that simultaneously models the implicit communities of users based on their check-in activities, without making use of their social network connections. This model captures the semantic features of the location, user-to-user influence along spatial and temporal preferences of users. To learn the latent community of users and model parameters, we propose an algorithm based on stochastic variational inference. To the best of our knowledge, this is the first attempt at jointly modeling the diffusion process with activity-driven implicit communities. We demonstrate its effectiveness through improvements in location prediction tasks over geo-tagged event traces collected from Foursquare check-ins.

Chapter 2

Related Work

2.1 Network Alignment

People nowadays usually participate in multiple online social networks simultaneously to enjoy more social network services. Besides the common users, social networks providing similar services can also share many other kinds of information entities, e.g., locations, reviews, videos, and products. The network Alignment problem aims at aligning different networks across common entities. However, these shared information entities in different networks are mostly isolated without any known corresponding connections. Thus, network alignment becomes a research problem in determining such potential corresponding connections linking multiple kinds of shared entities across networks simultaneously.

This problem, has been studied in various areas, e.g., protein-protein interaction network alignment in bio informatics [46, 53, 117], chemical compound matching in chemistry [118], data schema matching data warehouse [83], ontology alignment web semantics [22], graph matching in combinatorial mathematics [80], and figure matching and merging in computer vision [19, 2].

Several variants of the node alignment [158, 155] problem have been studied in the literature, also known as link inference/prediction [52, 113], correlating accounts [35], user identity linkage (UIL) [74], etc. In this work, we align locations based on coordinates of the location, name, and type of location. This is different from the user identity linkage problem primarily because of the fact location coordinates are available and thus similarity measures defined essentially consider the Euclidean distance as one of the factors.

The existing research efforts can be broadly classified into two main categories: profile-based and network-based identity linkage.

2.1.1 Profile-based methods

Profile-based methods leverage user's profile information (e.g., username [148], spatio-temporal patterns [108], posts [35], writing style [90], etc.) to link accounts across different sites. For instance, Reza et al. find correlating accounts using user names by modeling the naming process and rules from the perspective of information redundancy [148]. However, user names can be deliberately selected and modified at any time, which increases the difficulty of completing the User Identity Linkage (UIL) task. Writing style identification is another promising way to localize multi-account users while revealing various camouflage behaviors[90]. Arvind et al. used the writing style of user-generated text, e.g., grammatical structure and frequency of letters to identify users [90]. However, this method results in over-fitting, especially for short texts such as tweets, because it involves too many features. Recent research results investigated the security issues of multi-accounts. For instance, Jiang et al. propose a semi-supervised transfer learning method to predict cross-platform behaviors through sparse overlapped crowds [107]. Qian et al. protect against de-anonymization attack by discovering the approach for inferring privacy using knowledge graphs [103]. Luo et al. present a uni-class classification-based approach to detect multi-account users across OSN sites [135]. Riederer et al. use the timestamped location data generated by users to infer the user identities across the OSNs [108]. Zhang et al. present an energy-based model to link user identities by extracting distance-based profile features [160].

Almost all previous methods focus on either writing-style analysis or user behavior inference – however, in addition to the risk(s) of privacy leakage [103], they have a drawback in terms of coping with potential inconsistencies [115].

2.1.2 Network-based methods

Network-based methods are becoming increasingly promising in tackling the UIL problem and have received much attention because they only require structural information to align networks based on anchor nodes. For instance, BIG-ALIGN [53] introduces the problem of aligning bipartite graphs and proposes a gradient-descent-based solution. In [119], Tan et al. model user relationship using a hyper-graph and project the manifolds of two OSNs onto a common embedded space to correlate accounts. Neighborhood-based features seem like a natural choice for the UIL problem [160, 89, 149], relying on computing the Adamic/Adar scores to measure the neighborhood similarity [168]. CLF [154] predicts both anchor nodes and social links by transferring information related to social links formed by anchor nodes in the source network to the target network.

Inspired by word embedding techniques (e.g., word2vec [85] and Glove [98]) in natural language processing, a number of approaches have been proposed to embed the graph, e.g., DeepWalk [99], Line [120], SDNE [126], SDAE [9], node2vec [37], MM-DeepWalk [124], M-NMF [129], TransNet [125], CANE [123] – to name a few. Recently, some researchers have exploited Convolutional Neural Networks (CNN) and spectral graph theory to learn the representation of arbitrary networks, such as Planetoid [140], GraphCNN [21], Patchy-SAN [94], GCN [50], etc.

In the context of the UIL task, Man et al. [79] used network embedding techniques to capture latent structural regularities of observed anchor links and further learn a cross-network mapping for predicting anchor links. Liu et al. proposed IONE which also embeds two OSNs onto a common space to capture the social contacts of users [72]. PCT [156] aims at inferring potential corresponding connections linking multiple kinds of shared entities across networks simultaneously through a combination of both profile and network features.

Existing network-based methods embed structures of nodes from their local context by preserving the first and/or second-order proximity to link accounts across OSNs. The local structure of a network contains rich information for a group of nodes, but it is hard for existing UIL algorithms to discriminate the real user identity from its neighborhoods. Previous works only leverage the partial anchor nodes for supervised training, including embedding and network alignment while using the rest for testing which incurs insufficient training and inefficient linking. Moreover, many UIL algorithms including IONE use anchor nodes to embed and align non-anchor nodes [72]. However, their anchor nodes may become deviated (not aligned anymore) after training.

In this work, we do node alignment across *location* nodes to obtain location information such as categories, reviews, tips, etc. We mostly use spatial coordinates and the name as our feature set to extract the location information.

2.2 Geo-Social Link Prediction

2.2.1 Social-Link Prediction

Who is most likely to interact with a given user in the future? For example, friend suggestions on Facebook. In geo-social media, we can further improve the performance of link prediction using the info from the geographical activities of users. Scellato et al. in [111] have shown that more popular of a place visited by two users, the average probability of being friends decreases. For example, public places: touristic places, airports, stations. Unpopular places (less common number of check-ins) are likely to be of significant importance for them. For example, private houses. Wang et al. in [127] have shown that jointly using social graph and mobility features (i.e. high degree of overlap among trajectories) yields better performance. Thus, jointly using social graph features, geosocial features, and human mobility features can improve the performance of link prediction.

2.2.2 Location Prediction

POI recommendation and prediction are two different but related and extensively studied topics in LBSN: the former usually learns users' preferences over POIs while the latter is more interested in mobility pattern recognition. In this section, we first categorize the location prediction models based on the algorithms used, as described in [134]:

- **Content based Methods:** Content-based methods learn the content correlation or location transition probability, based on the assumption that the current user location is related to the previous location. To this end, researchers often construct a data structure to store content, then match the content to predict location [73]. The Markov model is a typical strategy that uses content information to predict a future state. Occasionally Markov models are combined with recommendation methods to establish a location prediction system. For example, in the Collaborative Exploration and Periodically Returning (CEPR) model, Lian et al. [63] solved the problem of location prediction based only on context.

- **Distribution based Methods:** Distribution-based methods model user movement as a distribution with location and time as two random variables. In this method, the probability of a random variable is computed and then ranked to predict a location. Cho et al. [16] discovered that human movement in daily life is influenced by the factors, of geographical limitations and social relations. The authors identified: 1) social relationships account for just 10-30% of users' daily life movement, and 50% of their periodic activity. Based on this finding the authors proposed a location prediction method: the Periodic and Social Mobility Model (PSMM) which describes human mobility based on periodic short-range travel and social network structures. 2) Most people visit their workplace during the workweek daytime hours and their place of residence on work nights and weekend day times. Based on this finding, they proposed the Periodic Mobility Model (PMM) for predicting future user location states (based on partitions of all locations visited by users, both work and home-related).

- **Preference based Methods:** User location history can be used to generate a matrix and then matrix factorization can be used to capture user movement preferences. A tensor is an extension of the matrix. Bhargava et al. [3] used tensor factorization methods in multi-dimensional collaborative filtering to predict location via user profiles, user's short message in social network, and user location and temporal information.

- **Social-relation based Methods:** Gao et al.[34] proposed a geo-social correlation(gSCorr) model to solve the cold start problem based on social information. The gSCorr model determines the correlation between a social network and geographical distance and defines a complex matrix of four relationships between social information and distance.

- **Time-dependent Methods:** Du et al.[25] proposed the Recurrent Marked Temporal Point Process (RMTPP) to simultaneously model visiting time and location. The basic concept of RMTPP is to model movement history using a nonlinear function. RMTPP also uses a recurrent neural network to automatically learn a representation of influences from a user mobility history. In addition, Zarezade et al. [150] proposed a probabilistic model based on point process, in which a periodic kernel function is used to capture the user period and multi-nominal distribution of locations. The whole framework is similar to that of the RMTPP.

- **Representation based Methods:** Noulas et al. [95] regarded the location prediction problem as a ranking problem, whereby every check-in associated location and time is defined as a tuple $< \ell, t >$, ranking at the highest possible location in a user historical list of visited locations. Prediction features can be classified as user mobility, global mobility, or temporal features. Recently, deep learning techniques – especially LSTM [112] and GRU[18] – have been widely used to capture the long-term sequential influence and mobility patterns. Spatial-Temporal Recurrent Neural Networks (STRNN) [73] extend the RNN model by incorporating temporal and spatial context in each time unit for predicting next POIs. A unified RNN-based framework jointly learning the embedding of multiple factors (e.g., user identity, location and time, etc.), was presented in [141]. However, these methods do not explicitly model user's historical visit patterns and personal preferences, but greatly focusing on current locations and short-term dependencies among POIs. More recently, [27] propose an attentional recurrent network for mobility prediction from lengthy and sparse trajectories, where two RNN models – learning the current and the historical trajectory, respectively – together exploit user's mobility and location preference with attention on multi-level periodicity of historical trajectories. However, it is complicated to train due to the relatively high density of historical trajectories.

- **Semantics based Methods:** The aforementioned approaches often focus on spatiotemporal space, however, few researchers have paid attention to the trajectory of semantic space. Semantics-based predictors enable better reasoning and therefore better location prediction results. Ying et al. [146] proposed a semantic framework for location prediction via semantic pattern mining, called SemanPredict.

2.3 Location Recommendation

Location recommendation is one of the most important tasks in LBSNs, which helps users discover new interesting locations in the LBSNs. Location recommendation typically mines users' check-in records, venue information such as categories, and users' social relationships to recommend a list of locations where users most likely check-in in the future. Location recommendation not only improves user viscosity to LBSN service providers but also benefits advertising agencies with an effective way of launching advertisements to potential consumers. Specifically, users can explore nearby restaurants and downtown shopping malls in Foursquare. Meanwhile, the merchants can make the users easily find them through location recommendations. Ye et al. [143] first propose POI recommendations for LBSNs such as Foursquare and Gowalla. After that, more than 50 papers about the problem are published in top conferences and journals, including SIGKDD, SIGIR, IJCAI, AAAI, WWW, CIKM, ICDM, RecSys, TIST, TKDE, and others. In this section, we briefly overview the existing techniques on location or POI(Point Of Interest) recommendation. Note that, similar categorization of the related work is followed for the location prediction models. Early studies in POI recommendation focus mainly on estimating users' preferences using Collaborative Filtering (CF), especially Matrix Factorization (MF) based techniques [144, 13, 64, 32, 104]. These methods can only model users' static preferences. For example, when a user living in New York travels to Hawaii for a holiday, these types of recommenders may still recommend POIs located in New York since they are unable to capture the dynamics of user preferences.

Deep Learning based Models: More recently, deep learning based methods, such as embedding learning [28, 114, 159], attention based models GeoSAN [62] and ASPPA [162], neural CF [42, 145], deep latent factor model [15], and metric learning [121] models, achieve promising performance in many recommendation systems. Researches on next-POI recommendation pay more attention to users' dynamic preference modeling. The pioneering work by Cheng et al. [14] proposes a matrix factorization method to embed the personalized Markov chains and the localized regions. Inspired by the success of RNN in sequential data modeling [10, 11, 44], RNN based methods become pervasive in the field of next-POI recommendation [73, 81, 27, 61]. For example, the ST-RNN model [73] extends RNN to model local temporal and spatial contexts. CARA [82] captures users' dynamic preferences by exploiting GRU's gate mechanism. TMCA [61] and STGN [163] adopts the LSTM-based and gated LSTM framework to learn spatial-temporal contexts, respectively.

DeepMove [27] designs a multi-modal RNN to capture the sequential transition.

2.4 Geo-Social Influence Maximization

Given a limited budget for initial advertising, the goal is to identify a small set of influential customers (as seeds), such that by convincing them to adopt the product and finally trigger a larger cascade of influence. This has application in the viral marketing of a product. This problem is different from Social Influence Maximization as spatial factors also contribute to determining the seeds.

Influence maximization is a well-studied problem (e.g., [47, 48, 12, 36]), the geo-seeded instantiation motivated by LBSNs has gathered attention only recently [130, 167, 109, 7, 59, 100, 153, 169, 170]. 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 [7]. Also, there exist recent studies on determining top-k influential locations [109] and product promotion in the context of location [167]. Recently, Jin et al. in [45] have also studied the problem of using the LBSN data to find the most influential geo-social object. 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) [59, 153, 130] or a set of geolocations or only the social network structure [7]. In a similar way, the user-user pairwise influence propagation probabilities are estimated either using just the (social) network structure [7, 59, 153] or taking into consideration the seed location/region [169, 170]. To the best of our knowledge, only the recent work in [100], has looked at defining user-user pairwise influence in spatio-temporal context, but for identifying follow-ship. A summary of important previous techniques categorized along the above dimensions appears in Table 2.1. In our empirical evaluation, we compare against the most recent work by Zhu et al. [169, 170], 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

Technique	Spatial Target	User Location	Pairwise User Influence
Loc-IM [59]	Location	Single Location	Location-independent
Loc Promotion [169]	Location	Set of Locations	Location-dependent
Reg IM [7]	Region	Set of Locations	Location-independent
Geo Soc Inf [153]	Location	Single Location	Location-independent

Table 2.1: Summary of Related Work (adapted from [58]) in Influence Models for Location Promotion

their paper, Zhu et al., have presented results using only two popular target locations (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.

2.5 Community Detection

Detecting communities has attracted much research attention within general online social networks, but there have been much fewer efforts in looking at the task within the context of LBSNs. The bulk of work on LBSNs has been on characterizing location adoption by users, across a variety of task formulations such as location prediction, recommendation, and promotion. We briefly review existing literature under the heads of community detection and characterization of location adoption.

Within general social networks, there has been much work on detecting communities as overlapping or non-overlapping clusters of users such that there is a high degree of connectedness between them. Techniques have largely considered social network connectedness as the main driver in forming communities. Community detection techniques could adopt a top-down approach starting from the entire graph and form communities by separating them into more coherent subsets that would eventually become communities. Methods from this school include those that use graph-based methods [102] and filtering out edges based on local pieces of information such as the number of mutual friends [161]. Analo-

Features	Clustering for LBSN Community Detection			
	Spectral[96]	M^2 [132]	Entropy-based [71]	CoLAB (Chapter 5)
venue-categories	✓			✓
venue-temporal		✓		✓
user geo-span		✓		✓
user social-status		✓		
structure-based			✓	
location-based			✓	✓

Table 2.2: Summary of Features used for Community Detection

gously, community discovery could proceed bottom-up by aggregating proximal nodes to form communities. Techniques within this category have explored methods such as merging proximal cliques [54] or by grouping nodes based on affinities to 'leader' nodes [49]. Point processes[20], which are popular models for sequential and temporal data, have been recently explored for community detection in general social networks[122]. NetCodec, the method proposed therein, targets to simultaneously detect community structure and network infectivity among individuals from a trace of their activities. The key idea is to leverage the multi-dimensional Hawkes process to model the relationship between network infectivity vis-a-vis community memberships and user popularity, to address the community discovery and network infectivity estimation tasks simultaneously. Our task of community detection within LBSNs is understandably more complex due to the primacy of spatial information in determining LBSN community structures.

LBSN Community Detection: There have been existing work on community detection in LBSNs that leverage a variety of user features in determining community structure within clustering formulations. Table 2.2 summarizes the feature set used by the three recent techniques proposed for community detection in LBSNs, viz., [96, 132, 71]. Our method models a wider variety of features, incorporating both spatio-temporal check-in information as well as venue category information, in inferring communities and user-user influence.

2.6 Location Adoption Characterization

With each LBSN check-in being associated with a location, the location information is central to LBSNs. There has been much research into modeling user-location correlations in various forms, which may be referred to as *Geo-Social Properties & Measures* in general.

2.6.1 Geo-Social Properties & Measures

Properties of users' check-ins

Friendship vs. Distance Cho et al. in [16] have figured out that people tend to move to nearby places on weekdays and to distant places on weekends, and the weekends' check-ins are influenced by social factors. Scellato et al. in [14] friends tend to be much closer than random users i.e. about 50 % of social links span less than 100km, while about 50 % of users are more than, 4000km apart. Thus, users who live closer have a higher probability to create friendship links.

Power Law Distribution of Check-ins Gao et al. in [33] have shown that users' check-ins follow power-law distribution i.e. a user goes to few places many a time while to many places a few times.

Temporal Periodic Patterns Cho et al in [16] and Gao et al. in [31] talks about users' temporal patterns that people commute to and from work at roughly the same time during the workdays, as opposed to the weekend when peoples travel and schedules are less predictable.

Spatio-Temporal Multi-Centre Distribution Cho et al. in [16] have also observed that Prob. of visiting regular location centers on certain time periods and decreases during other time periods, and also centers on certain location areas (for example, home and work). Lichman et al. in [66] have used kernel density-based estimations for modeling multiple centers of the mobility patterns.

Complex Sequential Transition Regularities: Feng et al. in [27] figured out the irregularities in human mobility patterns and the incapability of Markov models to capture them. For example, the probability of moving from home to office for a commuter is higher in workday mornings but often low on weekend mornings. Meanwhile, the transition may not follow the simple and exact Markov chain assumption, as people can go to different places (e.g., breakfast places) in their commute routes, which leads to high-order and irregular transition patterns. Moreover, they also mention that mobility periodicity is often complex and multi-level, involving daily routines, weekend leisure, yearly festivals, and even other personal periodic activities.

Measures of users' check-ins

Scellato et al. in [110] have defined two geo-social metrics: 1) Node Locality that measures how close are the neighbors of a given node to the node itself. 2) Geographic Clustering Coefficient that measures how spatially interconnected are the node's neighbors. Thus node locality highlight short-range social connections and geographic clustering co-efficient highlight short-range social-triangles. Node locality test on geo-social networks shows that users with more connections have friends further away, and geographic clustering co-efficient shows that users with fewer friends tend to generate social triangles on a smaller geographic scale, while users with many friends belong to triangles with longer links. This shows that there is a connection between the social properties of a given user and the geographic distance of his/her friendship connection.

2.7 Summary

In this chapter, we summarized the important related work in the area of Location-based social networks. Location-based services such as Foursquare, Brightkite, Gowalla provides users the platform to share their live location with their friends online. The location data bridges the gap between the online and offline worlds and enables a deeper understanding of users' preferences and behavior. The availability of users' geo-tagged temporal footprints has enabled the outset of research in this direction. Targeted advertisements and marketing is the primary motivation of the majority of the work in this area. Targeted advertising is a way of placing ads based on demographics, on the consumers' previous buying history, or behavior. In LBSNs based on users' locations visit, discounts on restaurant bills and movie offers are posted on their online profile wall to attract them to visit. Online location promotion and targeted advertisement have emerged over the past decade only with the widespread usage of smartphones.

In the first half of the decade (i.e. from 2011 - 2015) majority of the focus of the existing work had been on exploiting features of spatio-temporal data using statistical machine learning techniques. For example, Cho et al. in [16] used Gaussian distribution, [66] kernel density estimates, [33] power-law distribution to model users' geographical movements. collaborative filtering [165] and matrix factorization [13] based models have also been exploited for modeling users' preferences using their spatio-temporal footprints. While the latter half of the decade (i.e. from 2016 - 2020) the focus has shifted towards deep learning-based techniques such as RNN [27], LSTM, etc. There has also been some work related to point process-based techniques [17] for modeling users' check-ins. Du et al. in proposed RMTPP [25] which makes use of deep learning-based techniques along with point process-based modeling. Deep learning and point process-based techniques have shown promising results in this area.

Top conferences and journals, such as SIGKDD, SIGIR, IJCAI, AAAI, WWW, CIKM, ICDM, RecSys, TIST, TKDE, etc. have papers in this area. In this work, we have reviewed over 50 papers and summarized them in table 2.3.

Conference	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
KDD	1	3			2	2	1				1
CIKM			2	2	1	1		2			
IJCAI				1		1		1	2	1	1
AAAI			1			1	1			1	
SIGIR		1							2		
ECIR											1
ICDE							2			1	
ICDM			1						2		
SDM						1		1			
WSDM	1			1				1			
WWW						2		1	2		
SIGSPATIAL	1	1			1						
Journal											
TIST						1	2		1	1	
TKDE								1		1	1
TKDD							1				

Table 2.3: Summary of papers published in LBSN area in top conferences and journals in the past decade

Chapter 3

Data Augmentation

3.1 Introduction

Location-based Social Networks have emerged as a major source of information about users and locations with the potential for various industrial-based applications. The popular LBSNs such as Foursquare, Brightkite, and Gowalla typically do not expose the check-in information of users; because of privacy concerns. Foursquare, Brightkite, and Gowalla have their data sets publicly available but are anonymized, wherein only the latitude and longitude of the region of each check-in are available, and users are completely anonymized. For example, to anonymize the location information the check-in information is available as: a user has checked in a shopping complex. The shopping complex spans a large area and also contains various types of venues such as Saloon, Gym, Mart, etc. Therefore, it becomes difficult to identify where exactly the user checked in. Moreover, the category information is also not associated with each region. Thus, the first task at hand is to combine the location information from publicly available datasets with the data from maps for associating a set of venue categories with each region/location. We call this coarse-grained category information because multiple categories are associated with a location.

Next, we consider the problem of leveraging this coarse-grained category information

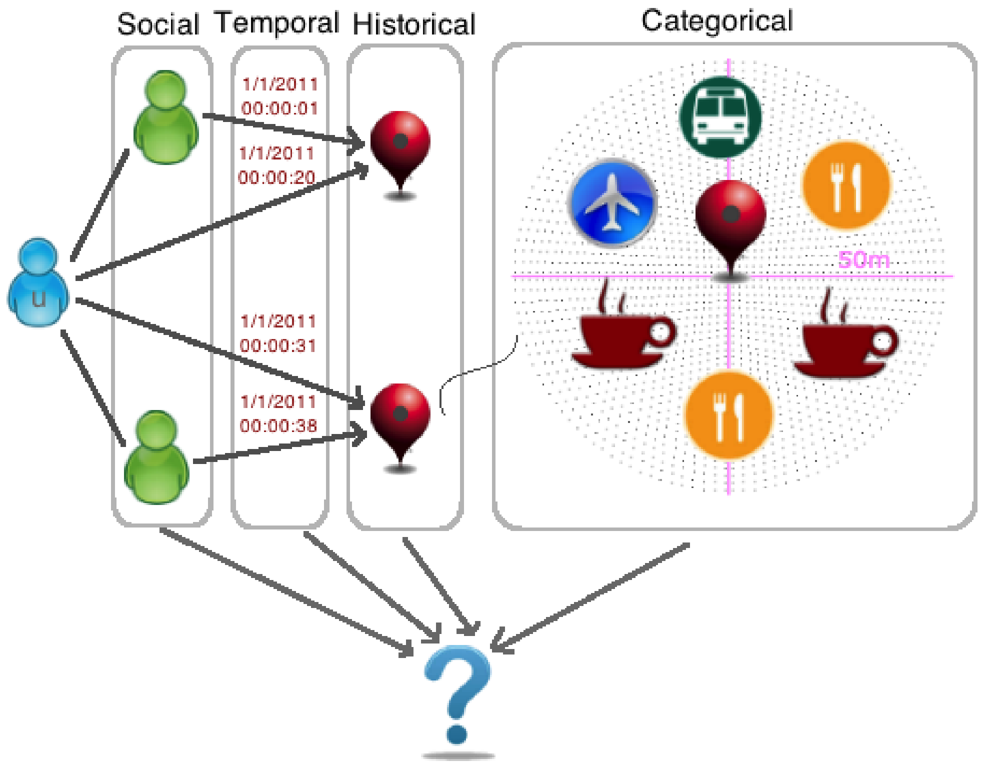


Figure 3.1: Spatial, Temporal and Categorical Features mapped together in LBSNs

extracted for locations from publicly available LBSN datasets, to improve location prediction in LBSNs as shown in figure 3.1. To the best of our knowledge, this is the first work that exploits such coarse-grained category information for the next check-in location prediction. Our experiments with publicly available check-in datasets mapped with categorical information show that this approach improves over the state-of-the-art methods for the location prediction task.

3.2 Data Augmentation and Location Alignment

We use publicly available annonymised check-in datasets [33, 16] where each check-in has only a lat-long representation. We associate each check-in $l = [lat, long]$ with a set of venue categories (i.e. gym, restaurant, coffee-shop etc.) using Foursquare APIs[1]. We map

[1]https://developer.foursquare.com/

all the venues from Foursquare that are within 50 meters radius of the location as shown in figure 3.1. i.e., $cat(l) = \{v.category|distance(v, l) \leq 50m\}$, where, v represents venue, l represents location, and $cat(l)$ represents categories associated with a location. Empirically, we found that each location gets associated with approximately k categories. The average value of k for the three data sets i.e. Foursquare (Mar'10-Jan'11), Foursquare (Jan'11-Dec'11) and Brightkite (Mar'08-Oct'10) is 22, 13 and 9, respectively.

3.3 Related Work

3.3.1 Data Curation

The check-ins data collection using Foursquare and Twitter is a commonly followed mechanism for carrying out analytics on LBSNs. Yang et al. in [137] collected data using Foursquare and Twitter for the period of April-2012 to September-2013.

Liu et al. in [75] and [76] collected dataset from Gowalla, a popular location-based social network, which has more than 600,000 users since November 2010 and was acquired by Facebook in December 2011. The authors used the Gowalla APIs to collect the user profiles, user friendship, location profiles, and users' check-in history made before June 1, 2011. Finally, they obtained 36,001,959 check-ins made by 319,063 users over 2,844,076 locations.

Liu et al. [2] collected data from Wee places, a website that aims to visualize users' check-in activities in location-based social networks (LBSN). It is now integrated with the APIs of other location-based social networking services, e.g., Facebook Places, Foursquare, and Gowalla. Users can log in at Wee places using their LBSN accounts and connect with their friends in the same LBSN who have also used this application. All the crawled data is originally generated in Foursquare. This dataset contains 7,658,368 check-ins generated by 15,799 users over 971,309 locations.

[2] https://www.yongliu.org/datasets/

The other publicly available data sets are from Foursquare[3] and BrightKite[4] data.

3.4 LBSN Data Collection

Given two LBSNs: Twitter T and Foursquare F where $V \in (T, F) = (V_u, V_\ell)$ and $E \in (T, F) = (E_{u-u}, E_{u-\ell})$, then the goal is to determine $V_u \in T(V_u) \cap F(V_u)$.

- 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 the 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 streaming API [5]. This limits our data set to Foursquare check-ins that are also posted via Twitter. We improve the coverage by first extracting the user IDs 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 a single fine-grained category for each location 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

[3]http://www.public.asu.edu/$\sim$hgao16/dataset.html
[4]http://snap.stanford.edu/data/loc-brightkite.html
[5]https://dev.twitter.com/streaming/public
[6]https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.HTML

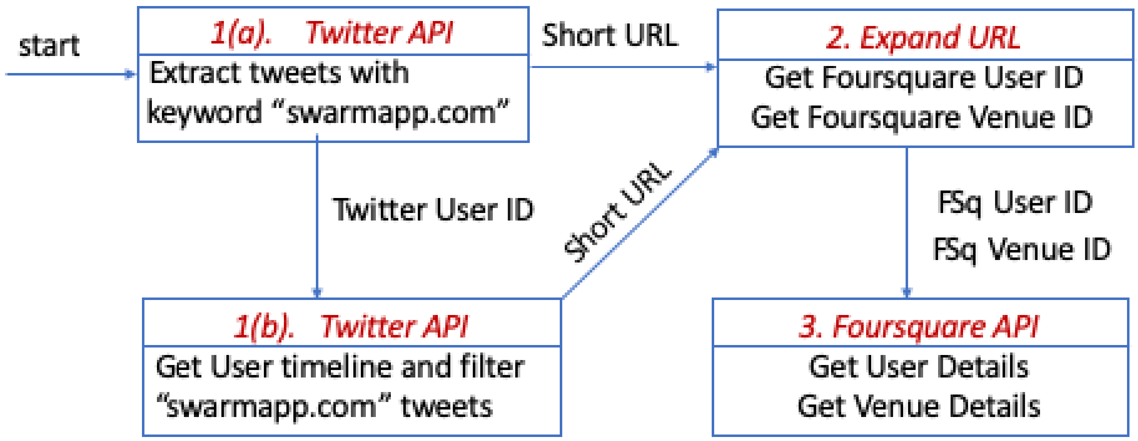


Figure 3.2: Methodology for Data Collection

information to just the users' social circles. We circumvent this again using Twitter, crawling Twitter connection information among users in our check-in data set 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. Figure 3.2 summarizes and illustrates these steps diagrammatically.

Some key characteristics of the resulting combined data set, which we denote as FSq'16, along with the other public data sets that are augmented with category information as described in section 3.2 we use, is shown in Table 3.1.

Data set	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
Mean(#categories / location)	1.00	12.12	20.47	8.38	1.28

Table 3.1: Statistical properties of the data sets

[7]https://developer.twitter.com/en/docs/accounts-and-users/follow-search-get-users/api-reference/get-friends-list

techniques / features	PMM and PSMM [16]	SHM [33]	SHM+T [31]	M5 Trees [95]	gSCorr [34]
Social Correlation	✓	✓	✓	✓	✓
Geographic Distance	✓		✓		✓
User Mobility	✓	✓	✓	✓	✓
Category Information				✓	
Periodic Patterns	✓		✓	✓	

Table 3.2: List of features used in different techniques

3.5 Leveraging Category Information for Next Location Prediction

The location prediction problem in LBSNs has been widely studied in recent years. This has applications in areas such as targeted advertisements, influence maximization and community detection. For example, knowing that a specific user's next check-in is likely to be at a cinema then this information could be used to prioritize movie ticket offers to be sent to her. In this chapter, we consider the problem of leveraging the coarse-grained category information extracted for publicly available LBSN data set (as discussed in the section above) to improve location prediction in LBSNs. The location prediction model in general does not perform well because of sparsity in the check-in data. Therefore, analysis of sparse and anonymized data is useful to achieve at least some improvement. To the best of our knowledge, this is the first work that exploits such coarse-grained category information for the next check-in location prediction.

3.6 Preliminaries for Next Location Prediction

In Table 3.2 we have broadly divided and summarized the feature space that has been used for location prediction problems by state-of-the-art Markov-based methods. It can be seen from table 3.2 that only Noulas et al in [95] have explored the category information in predicting where the user would go next. However, [95] assume that the check-in information contains not only fine-grained category information (e.g., restaurant, cinema, etc.) but also contains information about the precise venue the user has checked-in, i.e., the name of the restaurant or cinema. Thus, this technique is impractical in common cases where we have check-ins specified only at the lat-long level. The other methods listed in Table 3.2 can work with check-in data at the level of lat-long information.

3.7 Location Prediction Model

We briefly describe the SHM+T model proposed in [33] and then describe our method that extends SHM+T by exploiting category information to improve location prediction accuracy.

We denote the set of categories associated with a location l as C_l. If the last location is l', we find a likely category c' for l'. Then we determine a category c that is likely to be followed after c' (for example, restaurant after cinema, dessert after restaurant, etc.). Lastly, we find a location l that is likely for the category c. This is aggregated over multiple values for c and c' to associate a probability with each value of l, as follows:

$$P_C(l) = \sum_{c,c'} P_u(l|c).(\alpha * P_u(c|c') + (1-\alpha) * P_g(c|c')).P(c'|l'), \tag{3.1}$$

where $P_u(c|c')$ and $P_g(c|c')$ denote the probability of the user visiting a location of category c right after visiting one of category c', as estimated using the user's own check-in history, and the global check-in history across all users respectively. α is an interpolation parameter that determines the relative weighting of P_u and P_g. We estimate $P(c'|l')$ and $P(l|c)$ as follows:

$$P(c'|l') = \frac{\text{number of venues of category c close to } l}{\text{total number of venues close to } l}$$

$$P(l|c) = \frac{\text{number of venues of category c close to } l}{\sum_{loc} \text{number of venues of category c close to } loc}$$

We will call $P_C(.)$ as the category language model location prediction method; as in the case of other models, once this distribution is estimated, the most probable location could be recommended.

3.7.1 Category Language Model (CLM)

Using just the category language model for location prediction is not likely to be effective since it does not model other aspects of user movement. Thus, we devise a method to leverage CLM with SHM+T [31], a state-of-the-art model for location prediction that uses social, historical, and temporal information. The SHM+T model may be simplistically represented as a function $P(l|t, H_{u,t}, S_{u,t})$ that estimates a probability distribution of next location using: (1) t, the time of the day and week, (2) $H_{u,t}$, the user's check-in history and (3), $S_{u,t}$, the user's social circle.

We combine SHM+T and CLM by estimating the combined distribution as a weighted sum of the distributions by the individual models, with the relative weighting determined by λ as follows::

$$P_{SHM+T+CLM}(l) = \lambda * P(l|t, H_{u,t}, S_{u,t}) + (1 - \lambda) * P_C(l) \tag{3.2}$$

3.8 Experimental Evaluation

3.8.1 Implementation Details:

We have tested the proposed model over all the users who have made at least 10 check-ins. For each test user, we divide her check-in history into 4:1, where 80% of check-ins are used for training and the rest 20% are used for testing. Note that, check-ins are sorted chronologically. *Accuracy* is used as the evaluation metric.

Datasets \ Techniques	SHM	CLM	SHM +T			SHM+CLM	(SHM+T)+CLM		
			SHM+D	SHM+W	SHM+DW		(SHM+D)+CLM	(SHM+W)+CLM	(SHM+DW)+CLM
FSq'10	22.45	23.50	23.70	23.52	23.81	24.96	24.98	24.87	24.88
FSq'11	30.45	24.87	31.59	31.31	32.03	31.32	32.36	32.21	32.74
Brightkite	23.25	21.22	24.21	24.51	24.61	24.38	25.74	25.54	25.57

Table 3.3: Comparison in terms of Accuracy (in %) for different models

3.8.2 Results

The results are reported in table 3.3. Note that, we have only reported results for three datasets out of five. For FSq'16, we have results in next chapter, as this study focuses only on leveraging coarse-grained category information, and for Gowalla, the category information obtained is quite sparse (i.e. only for few locations we have categories, while for most of the locations we didn't obtain the category information from the Foursquare APIs). T in table 3.3 denotes the periodic patterns i.e. Daily(D), Weekly(W) and Daily-Weekly(DW). For CLM and SHM+T+CLM model. It can be observed that SHM+T+CLM achieves 1-2 % of improvement over SHM+T, that is similar to quantum of improvement that SHM+T has achieved over SHM in [31].

3.8.3 p-value test

In this section, we investigate the significance of adding the category language model to the existing SHM+T model by conducting the adjusted R2 and F-statistic based statistical significance test [30]. We consider two models:

- **Model-1**: includes all predictors i.e. SHM+T and CLM
- **Model-2**: includes only SHM+T as predictor

Note that, we provide as input the probability belief obtained from SHM+ T model and CLM model. Next, we compare the adjusted R2[86] and p-value[51] scores of both the models as shown in table 3.4[8]. It can be observed that Adjusted R2 values for Model-1 is

[8]The scores are calculated using broom library in R.

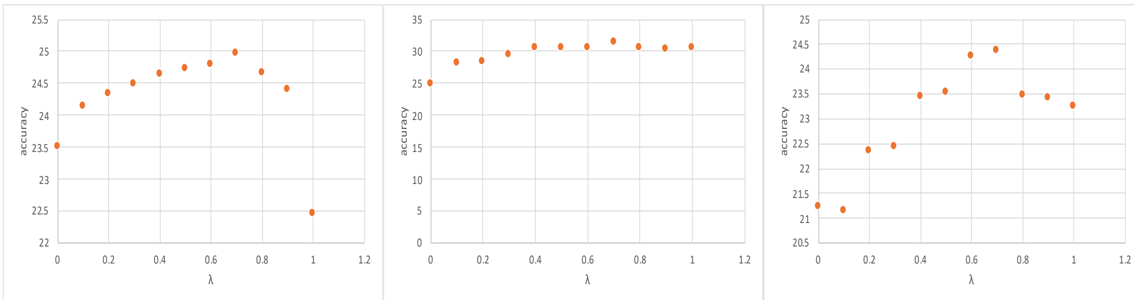
higher than Model-2 indicating that CLM also has strong correlation with the predicated values. p-values for both the models is ¡ 0.05 indicating that these are significant results.

Techniques	Test	Model	Datasets		
			FSq'10	FSq'11	Brightkite
(SHM+D)+CLM	Adjusted R2	**Model-1**	0.6713	0.7103	0.6348
		Model-2	0.4572	0.4824	0.5046
	p-value	**Model-1**	0.0253	0.0219	0.0304
		Model-2	0.0315	0.0248	0.0315
(SHM+W)+CLM	Adjusted R2	**Model-1**	0.7049	0.6261	0.7031
		Model-2	0.5281	0.4581	0.5684
	p-value	**Model-1**	0.0321	0.0272	0.02451
		Model-2	0.0314	0.0283	0.0328
(SHM+DW)+CLM	Adjusted R2	**Model-1**	0.7150	0.6319	0.8021
		Model-2	0.4826	0.5827	0.6043
	p-value	**Model-1**	0.0245	0.0264	0.1892
		Model-2	0.0389	0.0263	0.0224

Table 3.4: Comparison in terms of Adjusted R2 and p-value for **Model-1** and **Model-2**

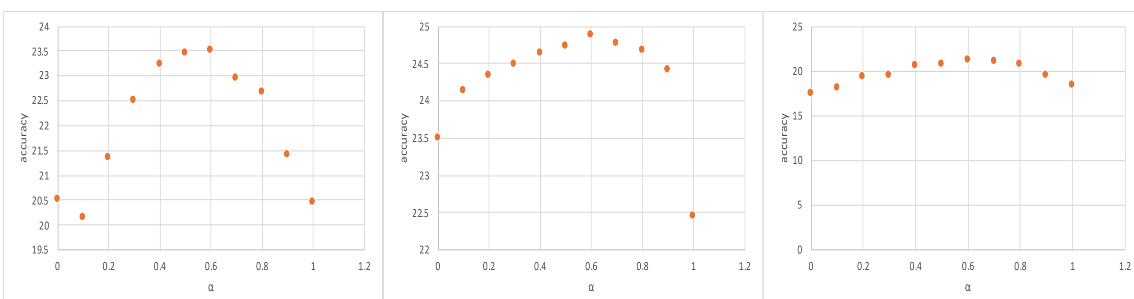
3.8.4 Tuning Parameters

We use α = 0.6 and λ = 0.7 in equations 3.1 and 3.2 respectively. At these values we obtain the best performance as shown in figure 3.4 and figure 3.3. For α the smoothing is contributing in assigning probabilities to the locations that are new and doesn't exist in the training dataset of a single user. Therefore, learning from global users' data is important, firstly for not having zero probabilities for new locations, and secondly for learning the global category sequence trend. For λ it can be observed that the distance and time are given more weightage as distance and time plays a critical role but we can observe that category information also plays an important role in prediction, because when λ = 1 (i.e. only SHM+T) the accuracy falls drastically. Hence, category information helps in modeling users' preferences, and especially for non-periodic check-ins that are apart from locations near work and home.



(a) FSq'10 (b) FSq'11 (c) Brightkite

Figure 3.3: λ at 0.7 gives highest accuracy for FSq10, FSq11 and Brightkite datasets



(a) FSq'10 (b) FSq'11 (c) Brightkite

Figure 3.4: α at 0.6 gives highest accuracy for FSq10, FSq11 and Brightkite datasets

3.9 Conclusion

In this chapter, we developed a method to infer coarse-grained category information and have leveraged this information for enhancing the performance of the existing state-of-the-art methods for the next check-in location prediction task. In continuation of this work, we would like to explore fine-grained information that can be used to improve the current state of the art. Since we observed that with coarse-grained category information we only obtained delta improvement but it shows that categorical information holds potential to model users' interest which can be leveraged for location-based applications.

Chapter 4

Influence Quantification between users in LBSNs

4.1 Introduction

Determination of user influence on social networks is often seen as a tool for viral marketing [88]. Understanding of social media influence has been exploited for legitimate purposes such as promotion of health-information [91], as well as for misleading users through campaigns such as political astroturfs [105]. In the scholarly community, the problem of influence maximization has attracted much attention. Influence maximization [7, 59, 133] 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 that 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[1], on the other hand, consider location information as a first-class citizen, with most user activity within them involving 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[2] within current-day smartphones has led to significant improvements in the penetration of location-based social networks.

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. On 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, [16, 144] 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 LBSNs, include 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 [169] 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 operating at that location [169]. 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

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

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 *influences 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 [36, 153]. Thus, this task associate 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 [169]. Studies on semantic features such as category have been limited primarily since datasets containing such information have been scarce [16, 33]; such deficiencies are being addressed recently (e.g., in [70, 41, 137]). 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 follow-ship based on temporal correlation has been of interest in LBSN studies in other contexts [100].

4.1.1 Contributions

In this thesis, we develop a novel model 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, the 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:

- **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 seed set, thus modeling time-conditioned social follow-ship.

While in its basic form the *LoCaTe* model uses exponential distribution to quantify temporal correlation across all locations checked in by the user, we also propose the *Lo-*

CaTe+ model which makes use of advanced, location-specific modeling of temporal correlations (involving more parameters to learn) based on mutually exciting Hawkes processes (meHP) [38, 138]. In order to illustrate the general-purpose utility of *LoCaTe* and *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 datasets. We conduct a large 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 work 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 chapter is organized as follows: Section 4.2 formally defines the influence quantification problem within the larger context of location promotion. Section 4.3 discusses the modeling methodology, and section 4.6 shows how to evaluate the proposed influence quantification model and the experimental results obtained. Finally, section 4.7 concludes the work and outlines possible future directions.

4.2 Problem Statement

In this section, we provide a formal definition of the influence quantification problem in an LBSN. Table 4.1 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 the venue is a more appropriate word, location corresponds to the convention in the 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 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 4.2.1 (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 $P(\,\ell, u, v|G\,)$, 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 an association be-

[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 check-in records of user u $\langle u, \ell, t, C_\ell \rangle$
M	set of all check-in records
L	A set of locations
C	A set of categories

Table 4.1: Notations used in this chapter

tween 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 the 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 check-in records employed to train the influence quantification model.

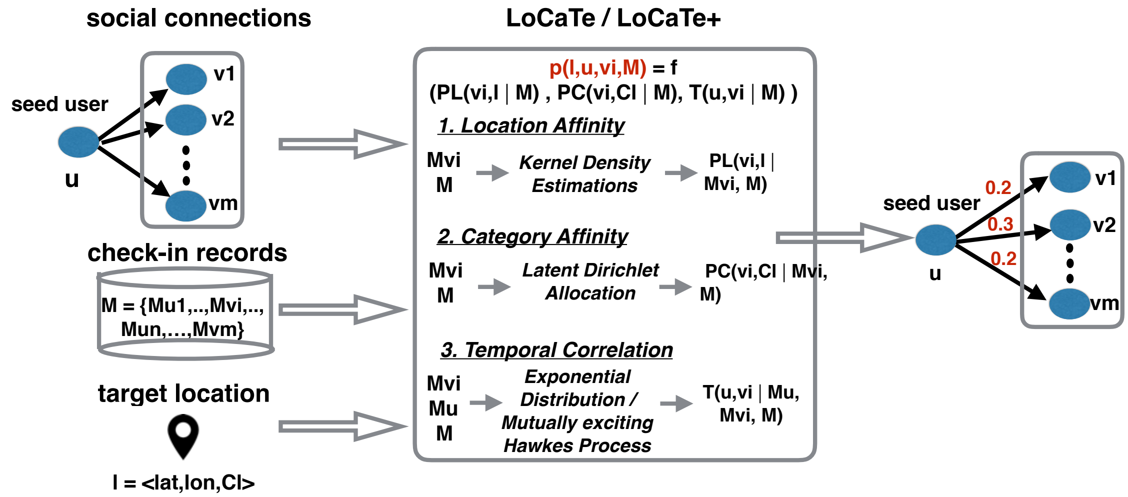


Figure 4.1: LoCaTe: Framework for Influence Quantification

4.3 LoCaTe Framework and Models for Influence Quantification

We now outline our influence quantification framework, LoCaTe, that estimates $P_{\ell,u}(v|M)$, 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 4.1 shows the framework of *LoCaTe*. LoCaTe combines information from three kinds of features to arrive at a estimation as follows:

$$P_{\ell,u}(v|M) = \left(\alpha \underbrace{P_L(v,\ell|M)}_{\text{location affinity}} + (1-\alpha) \underbrace{P_C(v,C_\ell|M)}_{\text{category affinity}}\right) \times \underbrace{T(u \rightarrow v|M)}_{\text{temporal correlation}} \tag{4.1}$$

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

$$P_a = \sum_{v \in U, \ell \in L} \left(\alpha \underbrace{P_L(v,\ell|M)}_{\text{location affinity}} + (1-\alpha) \underbrace{P_C(v,C_\ell|M)}_{\text{category affinity}}\right) = 1 \tag{4.2}$$

$$P = \int_0^T P_a.T(u \rightarrow v|M) = 1 \tag{4.3}$$

$P_L(v, \ell|M)$ models the affinity of v to location ℓ, and $P_C(v, C_\ell|M)$ 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, $T(u \to v|M)$ 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.

$P_{\ell,u}(v|M)$, 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.

4.3.1 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 [16]. 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 [66] addresses the limitations of fixating the densities to a specific number by introducing Kernel Density Estimates. Kernel Density Estimation is a non-parametric method for estimating the density function from random sample of data [116], 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 signifying the number of nearest neighbors.

$$P_L\left(v, \ell|M_v\right) = f_{KD}\left(l|M_v, k\right) = \frac{1}{|M_v|}\sum_{j=1}^{|M_v|}\kappa_{j,k}\left(\ell, \ell_j\right) \tag{4.4}$$

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

$$\kappa_{j,k}\left(\ell, \ell_j\right) = \frac{1}{2\pi\, h_{j,k}} exp\left(-\frac{1}{h_{j,k}}\|\ell - \ell_j\|\right) \tag{4.5}$$

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 over smoothing, while setting the bandwidth to a small value in sparsely populated areas will lead to over fitting. Thus, bandwidth computed using the nearest neighbors approach ensures the bandwidth computation is sensitive to differential densities of locations in urban and rural areas.

Mixture of Kernel Density models

The location affinity for a user v is learned 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:

$$P_L^k\left(v, \ell|M\right) = \beta_v f_{KD}\left(\ell|M_v, k\right) + \left(1 - \beta_v\right) f_{KD}\left(\ell|M, k\right), \tag{4.6}$$

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 4.3.4.

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 thesis, instead of fine tuning to different levels of smoothing we have kept a simplified model with two components, since this is an orthogonal research to our current work.

4.3.2 Category Preference

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 4.2 represented by the word cloud of the categories of their checked-in locations (larger font indicates higher frequency); User A evidently exhibits affinity towards visiting restaurants while user B prefers gym and fitness centers. We use topic modeling to identify such higher-level contexts, and exploit it to model the user-category affinity term, $P_C\left(v, C_\ell | M\right)$.

For topic modeling, we use Latent Dirichlet allocation (LDA) [6] 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

(a) User A (b) User B

Figure 4.2: Category wise check-in Distribution

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:

$$P_C\left(v, C_\ell | M\right) = \sum_{Z \in Topics(M)} P\left(C_\ell | Z\right) \times P\left(v | Z\right), \tag{4.7}$$

where $Topics(M)$ is the set of topics learned as described, and Z represents a topic from the learnt topic-set. $P\left(v|Z\right)$ and $P\left(C_\ell|Z\right)$ 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 $P_C\left(v, C_\ell|M\right)$ are achieved when the user's category distribution and that of the location under consideration are correlated with the same set of topics.

4.3.3 Temporal User Correlation

We now turn our attention to the temporal correlation term, $T(u \to v|G)$, that quantifies the extent of influence that u has over v. This primarily accounts for the socially induced follow-ship 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 $T(u \to v|G)$. 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 4.3, 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 4.4(a)). However, the *across connections* time lag distribution (in Figure 4.4(b)) does not quite follow a power law distribution despite exhibiting a monotonic decay with increasing values of time lag. It may also be noted that the across connections data is much sparser than across userbase; this is so since there are a significantly fewer number of occurrences of connected users visiting the same location.

These observations lead us to a natural model of time lag distribution between users that uses an exponential distribution, used in similar settings elsewhere [100]. Despite its simplicity, this formulation is surprisingly effective in practice as seen in our experiments.

However, the above model of temporal correlation or time lag distribution of check-

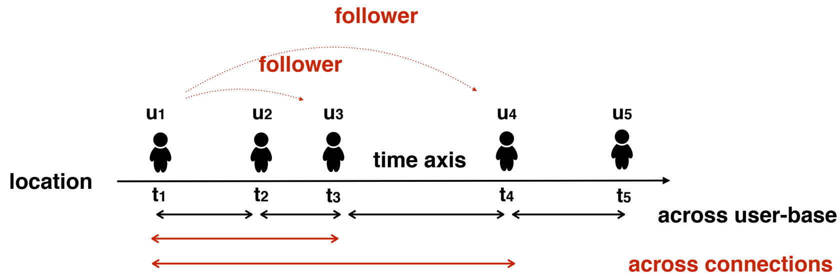
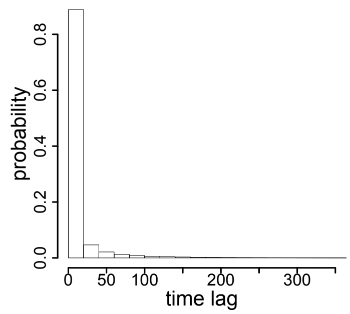


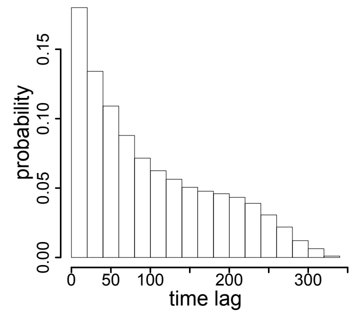
Figure 4.3: Depicting the time lag between check-ins at a location for connected and non-connected users, u_3 and u_4 are followers of u_1

ins at a location between socially connected users using exponential distribution makes a rather strong assumption that events (i.e., check-ins) arrive at a constant rate, λ, throughout the time of observation. In reality, however, that is rarely the case. For instance, 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 check-ins, and consequently shorter time-lags, than during regular times. In our second model, we incorporate changing intensity of check-ins by using nonhomogenous Poisson processes (NPP) to model the check-in behavior. Specifically, we use a class of NPPs, viz., the mutually-exciting Hawkes processes [38, 138] (*meHP*), which has been successfully used to model contagions in Financial markets [8] as well as in Social media [138]. 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 follow-ship strength between users.

We call the full influence quantification model (Ref. Eq. 4.1) that uses the exponential distribution for estimating user correlation as **LoCaTe** and the one that uses mutually exciting Hawkes process modeling as **LoCaTe+**. We provide the details of the temporal user correlation models in separate subsections herein.



(a) across userbase



(b) across connections

Figure 4.4: Time lag (in days) probability distribution plot

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:

$$p(\delta t) = \lambda_t e^{-\lambda_t \delta t} \tag{4.8}$$

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

$$\lambda_t = 1/avg \left\{ |t_2 - t_1| | \exists \langle u, \cdot, t_1, \cdot \rangle \in M \ \wedge \ \exists \langle v, \cdot, t_2, \cdot \rangle \in M \ \wedge \ (u, v) \in E \right\}, \tag{4.9}$$

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 $T\left(u \rightarrow v | G\right)$ which is modeled as the cumulative weight of v checking in at a location visited by u after a time lag of any $t \geq t_{u,v}^{min}$:

$$\begin{aligned} T(u \rightarrow v|G) &= \int_{t_{u,v}^{min}}^{\infty} \lambda_t e^{-\lambda_t \delta t} d(\delta t) && (4.10) \\ &= -e^{-\infty} + e^{-\lambda_t t_0} \ = e^{-\lambda_t t_{u,v}^{min}} && (4.11) \end{aligned}$$

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

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

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 check-in 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:

$$\lambda_v(t, \ell) = \mu_v + \alpha_v \sum_{t_i \in H_v(t)} \exp(-\eta_v(t - t_i))\, I(l_i = \ell) + \alpha_{u \rightarrow v} \sum_{t_j \in H_u(t)} \exp(-\eta_v(t - t_j))\, I(l_j = \ell), \tag{4.12}$$

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.

$$L\left(\mu, \alpha, \mathbf{A}, \eta\right) = \prod_{v} L_v\left(\mu_v, \alpha_v, \mathbf{A}_{*\to v}, \eta_v\right), \tag{4.13}$$

where, μ, α and η 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:

$$L_v\left(\mu_v, \alpha_v, \mathbf{A}_{*\to v}, \eta_v\right) = \prod_{n=1}^{N_v} \lambda_v\left(t_n, l_n\right) \times \left(\int_0^T \exp\left(-\eta_v\left(t, l_n\right)\right)\right) dt \tag{4.14}$$

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 $t_{u,v}^{min}$ as in the case with exponential distribution modeling 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 $t \geq t_{u,v}^{min}$, solely by virtue of influence from u:

$$\begin{aligned} T\left(u \to v|G\right) &= \int_T^\infty \alpha_{u\to v} \sum_{t_j \in H_u(t)} \exp\left(-\eta_v\left(t - t_j\right)\right) dt \\ &= \sum_{t_j < T} \left(\frac{\alpha_{u\to v}}{\eta_v}\right) \exp\left(-\eta_v(T - t_j)\right), \end{aligned} \tag{4.15}$$

where, T is given as:

$$T = t_u + t_{u,v}^{min} \tag{4.16}$$

t_u is the time u checked-in in the test data and $t_{u,v}^{min}$ is estimated as in the case of the

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 4.2: Log-likelihood at different values of k

exponential distribution-based modeling:

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

4.3.4 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 and are learned jointly by maximizing the likelihood function for each user v given in Equation 4.14 using the simplex method [92]. The parameters β_v and k are learnt using the likelihood function defined in equation 4.6. The hyper-parameter k is estimated as the value that maximizes the likelihood of check-ins in the training set. Thus, we set k to the value that maximizes the following:

$$k = \arg\max_{k'} \sum_{\langle v,\ell,\cdot,\cdot\rangle \in V} \log\left(P_L^{k'}\left(v, \ell | M\right)\right) \tag{4.17}$$

The distribution of log-likelihood across various values of k are shown in Table 4.2; accordingly, we chose $k = 5$ for usage in our method.

Note that, k is estimated based on all data points in the training set (i.e., not an individual-level model).

We use EM-algorithm for estimation of parameters β_v and α. The EM algorithm to

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 4.3: α and β_v values

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. The other parameters i.e. β_v and k are kept as constant and are learnt separately using the likelihood function defined in equation 4.6

- 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:

$$\alpha^{(i)} = \frac{\sum_{p \in N} \alpha_p^{(i)}}{|N|} \tag{4.18}$$

- 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 and k is treated as constant. 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 4.3 shows values of β_v and α learned for different datasets.

The parameters of the meHP model of temporal user correlation (i.e., μ_v, α_v, $\alpha_{u \to v}$, and η_v) are learned jointly by maximizing the likelihood function for each user v given in Equation 4.14 using the simplex method [92].

4.4 Applications Using the LoCaTe Influence Quantification Models

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.

4.4.1 Location Promotion

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

Definition 4.4.1 (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. □

Figure 4.5 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

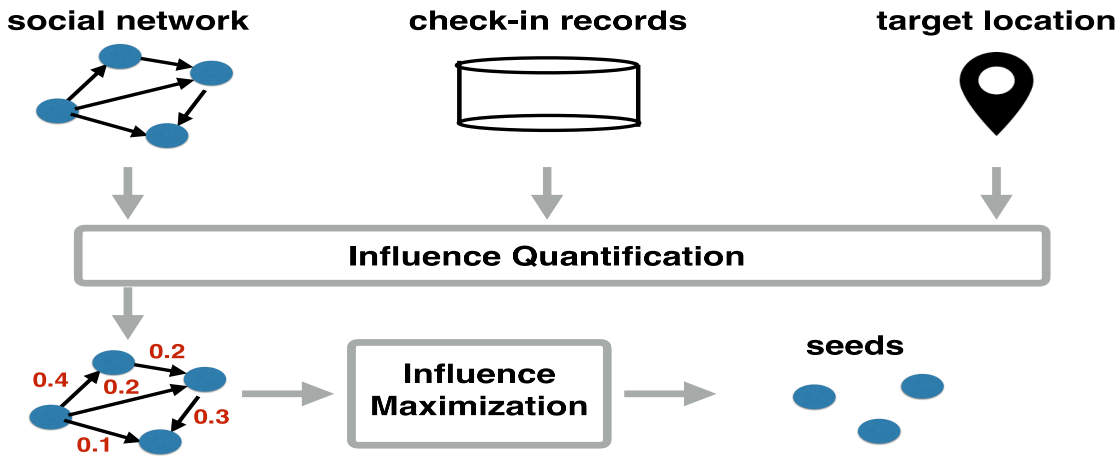


Figure 4.5: Location Promotion Framework

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' $\leftarrow \arg\max_{\forall u \in visited(\ell)} \{v | v \notin I \wedge P_{\ell,u}(v|M) \geq \rho\}$;
  $S \leftarrow S \cup u'$;
  $I \leftarrow I \cup \{v | P_{\ell,u'}(v|M) \geq \rho\}$;
end
return S

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 $P_{\ell,u}(v|M) \geq \rho$ 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.

4.4.2 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:

$$\begin{aligned}\mathcal{S}_{u,p}(\ell) = \gamma_1 \times Proximity\,(\ell, p)\, + \\ \gamma_2 \times \left(\alpha P_L\,(u, \ell|M) + (1-\alpha)\, P_C\,(u, C_\ell|M)\right) + \\ \gamma_3 \times \sum_{v,[u,v]\in E} P_{\ell,v}\,(u|M)\end{aligned} \qquad (4.19)$$

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 4.3.4. 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.

4.5 Influence Maximization

In this section, we outline the Influence Maximization step (as shown in figure 4.5) in detail. Once we have the scores or propagation probabilities between each and every pair of users in the social network (from the influence quantification step). The next step is to determine the τ seed users set for the promotion such that the total number of users influenced is maximized. Kempe et al. in [47] has shown the finding the optimal solution for the IM problem is NP-Hard, and proved that simple Greedy-algorithm can provide best approximation in polynomial time. They incorporated the use of diffusion models for information propagation, followed by most of the subsequent work. The two most well studied information diffusion models are: (1) Linear Thresholding (LT) and (2) Independent Cascade (IC).

1. Linear Thresholding (LT): Under the LT model, every node v contains an activation threshold θ_v, which is chosen uniformly at random from the interval [0, 1]. Further, LT dictates that the summation of all incoming edge weights is at most 1, i.e., $\forall u \in In(v) W(u,v) \leq 1$. v gets activated if the sum of weights $W(u,v)$ of all the incoming edges (u,v) originating from active nodes exceeds the activation threshold θ_v. Mathematically,

$$\forall_{u \in In(v)} W(u,v) \geq \theta_v$$

2. Independent Cascade (IC): Under the IC model, time unfolds in discrete steps. At any time-step i, each newly activated node $u \in V_a$ gets one independent attempt to activate each of its outgoing neighbors $v \in Out(u)$ with a probability $W(u,v)$. In other words, $W(u,v)$ denotes the probability of u influencing v

Using the LT/IC model, the influence spread i.e. the expected number of activated users are computed given a seed users set.

4.6 Experimental Evaluation

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 [169, 170]. 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 addition to the comparative evaluation, we also present trends across varying values of cut-off thresholds used to discretize the influence scoring to the sets of influenced and other users.

4.6.1 Datasets

We tested over 5 datasets as shown in Table 4.4, of which FSq'16 is the one that we collected using Twitter and Foursquare APIs, and rest are publicly available datasets [16, 33], for which the category information is inferred as explained in chapter 3. For the ease of reference we are reporting the full table with some key characteristics of the datasets in Table 4.4.

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. There are some recently released Foursquare datasets (e.g.,[137]) 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.

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.

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 4.4: Statistical properties of the datasets

4.6.2 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 modeling implementation from Mallet (`http://mallet.cs.umass.edu/topics-devel.php`). We 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 are made publicly available at `https://goo.gl/ayzehx`.

4.6.3 Influence Quantification Models

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

1. Distance-based mobility models (DMM) [169, 170],
2. Gaussian-mixture models (GMM) [16, 169, 170] 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 modeling, 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 [16, 169]

$$\mathcal{N}\left(x|\mu,\Sigma\right) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}\ \exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$
$$p(x) = \sum_{k=1}^{K}\pi_k\mathcal{N}\left(x|\mu_k,\Sigma_k\right)$$

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

   GMM-category: Zhu et. al. in [170] extends the basic GMM model to incorporate category information as follows:

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

   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., $p(x,y|C_\ell,R_{x,y,r})$, is used instead of directly calculating $P(x,y|C_\ell,u)$, where $R_{x,y,r}$ is a circular region with center (x,y) and radius r.

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

θ_{C_ℓ} and r is set to 10 and 1 km, respectively as used in [170].

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

   DMM_Basic: Pareto distribution [93] is used for modeling the distances between the checked-in locations of a user.

$$\begin{aligned} p_u(\ell) &= \sum_l P(u \text{ is at } l)\ P(u \text{ moves distance } d(l,\ell) \text{ from } l) \\ &= \sum_l \frac{p_{u,l}\alpha_M}{\left(d\left(l,\ell\right)+1\right)^{\alpha_M}} \end{aligned}$$

   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:

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

3. **Baseline:** In equation (1) in section 4.3 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.

4.6.4 Evaluation on Influence Quantification Task

For evaluation on Influence Quantification task, we use the same framework as used in an earlier work [169]. 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*, $PS(\ell, u, \rho \,|\, G)$. 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 $PS(\ell, u, \rho \,|\, G)$ 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 4.4 shows the number of test cases, $A(\ell, u)$, along with the cut-off timestamp for each dataset.

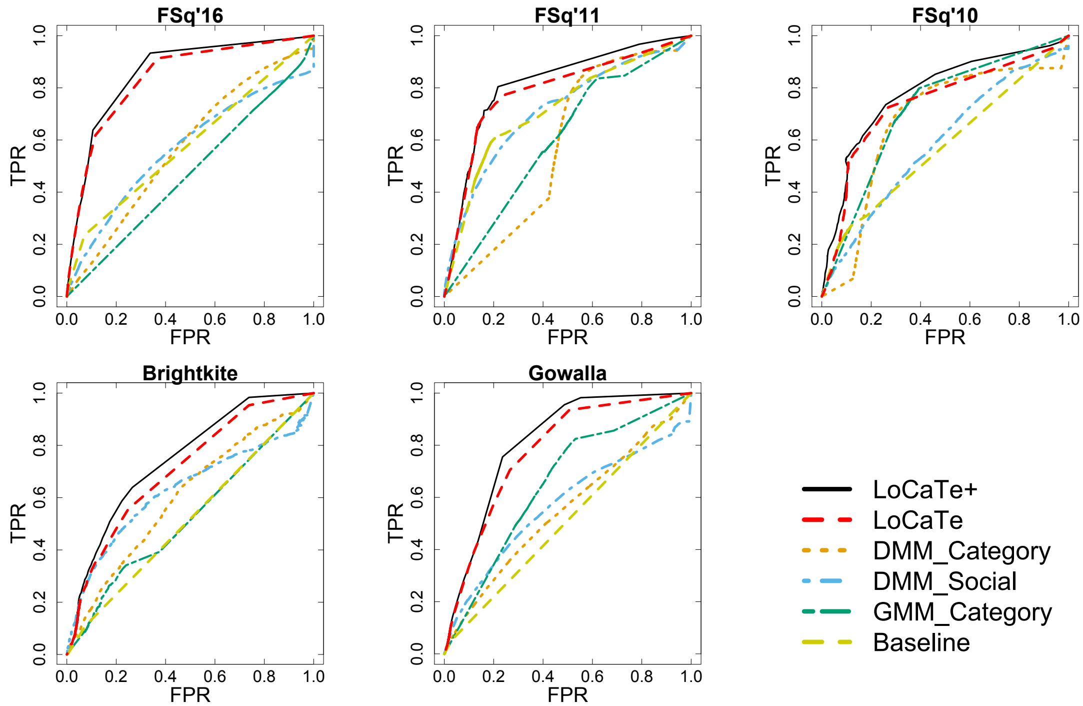


Figure 4.6: ROC for different influence quantification models (AUC is in table 4.5)

ROC and AUC

Figure 4.6 shows ROC curves and table 4.5 shows AUC (Area Under the Curve) of different influence quantification models on different datasets along with the F-statistic based significance test [30] where, p-value < 0.05 indicating that the results are significant. 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, 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 modeling 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.

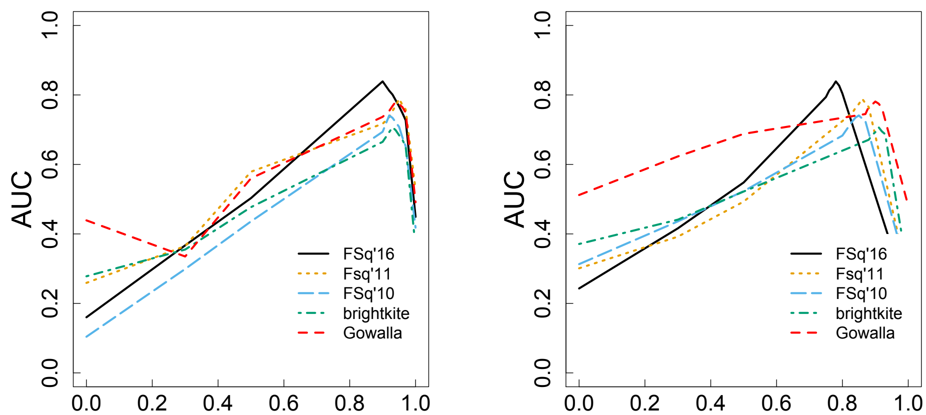
Datasets	Techniques							
	Baseline	GMM	GMM_category	DMM_basic	DMM_social	DMM_Category	LoCaTe	LoCaTe+
Fsq'16	0.582	0.599	0.473	0.521	0.568	0.573	0.839	**0.857****
Fsq'11	0.721	0.716	0.605	0.727	0.716	0.579	0.789	**0.816****
Fsq'10	0.575	0.718	0.717	0.699	0.588	0.671	0.741	**0.781***
Brightkite	0.517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

Table 4.5: AUC (Area Under the Curve) of different influence quantification models over different datasets along with $^{**}p < 0.01$ and $^{*}p < 0.05$

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 modeling delivers improvements over the simpler exponential distribution based modeling. 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 neighbors (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 modeling using category information is seen to provide significant gains in accuracy of influence quantification.

Parameter Tuning

Figure 4.7 (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 (4.1)) and β_v (weighted parameter for user and global KDE model in eq (4.6)) varies, respectively. It can be observed that the highest value of AUC is achieved close to 0.90 for all the datasets, giving less weightage to topic model. But, at α = 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. The peak is observed close to 0.9 because there are fewer number of categories as compared to locations 4.4, thus probability values for category affinity is high as compared to location affinity.



(a) α (b) β_v

Figure 4.7: AUC (Area Under the Curve) with varying α and β_v between $0.0 - 1.0$

F-measure

$PS(\ell, u, \rho \,|\, G)$ and $A(\ell, u)$, both being sets, allow comparing the methods based on the F-measure [101]. The Table 4.6 shows F-measure of different influence quantification models on different datasets along with the F-statistic based statistical significance test. 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 modeling in LoCaTe+ lead to very significant gains in F-measure over the basic LoCaTe model. Note that, although the F-measure is less than 0.01 but the p-value < 0.05 indicates that the results are significant.

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 4.6: F-measure of different influence quantification models along with p-value significance test where $^{**}p < 0.01$ and $^{*}p < 0.05$

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 4.7: Average time taken in execution of a testcase in (ms) for different influence quantification methods

Execution Time

Table 4.7 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).

4.6.5 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:

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

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 sizable 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.

Results

Table 4.8 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 [12]. We observe that both the

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 4.8: $|I(S)|$ at ρ = 0.003 and τ = 5

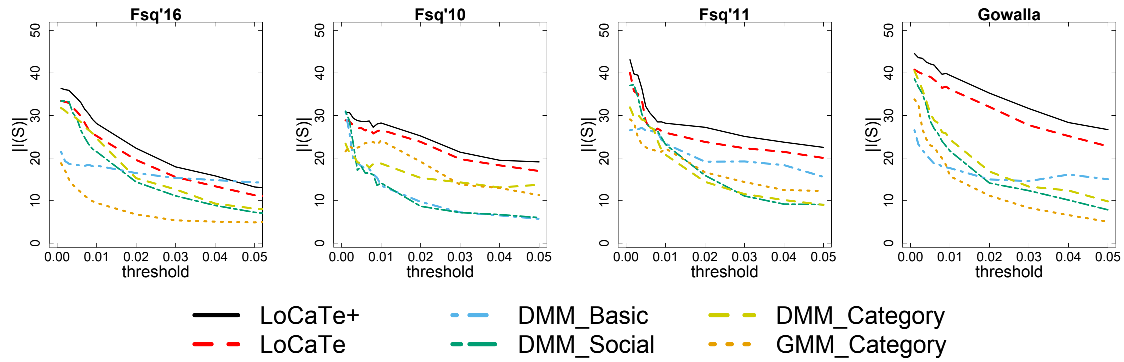


Figure 4.8: The size of $I(S)$ at different thresholds with τ = 5

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.

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 4.8, and 4.9. 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 capability 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.

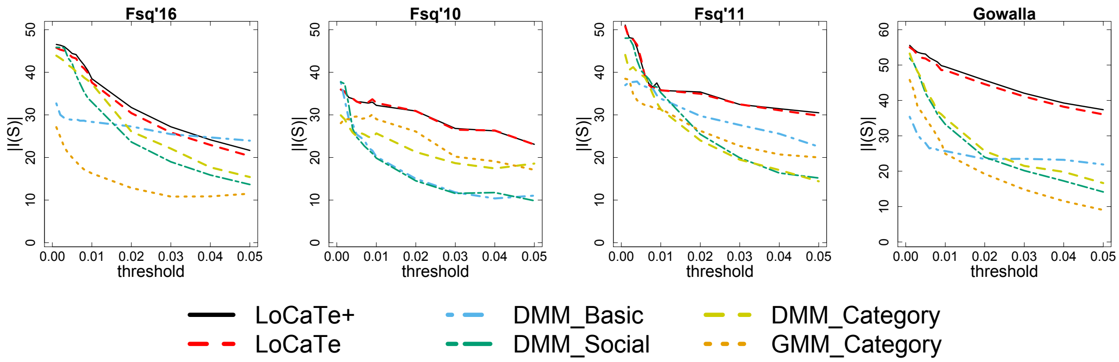


Figure 4.9: The size of $I(S)$ at different thresholds with τ = 10

4.6.6 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

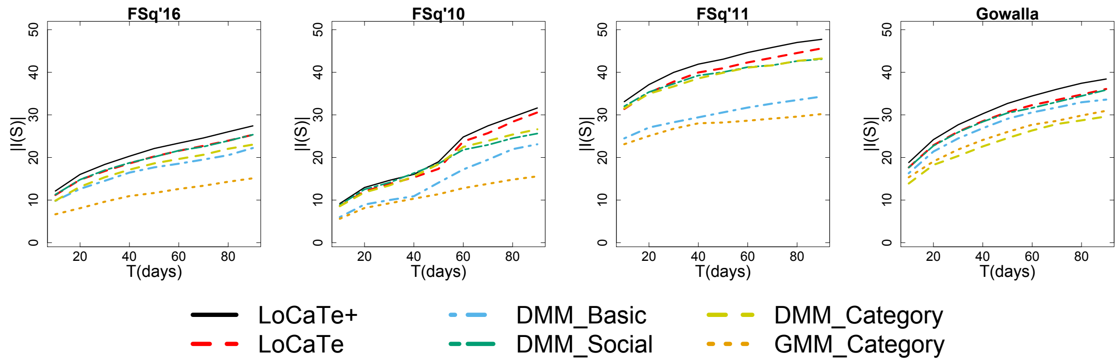


Figure 4.10: The size of $I(S)$ at time_windows with $\rho = 0.003$ and $\tau = 5$

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, 2015 and the time window T is 20 days. A candidate seed user u visits the target location ℓ on Sep 20, 2015 (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, 2015; 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 4.10, 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 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

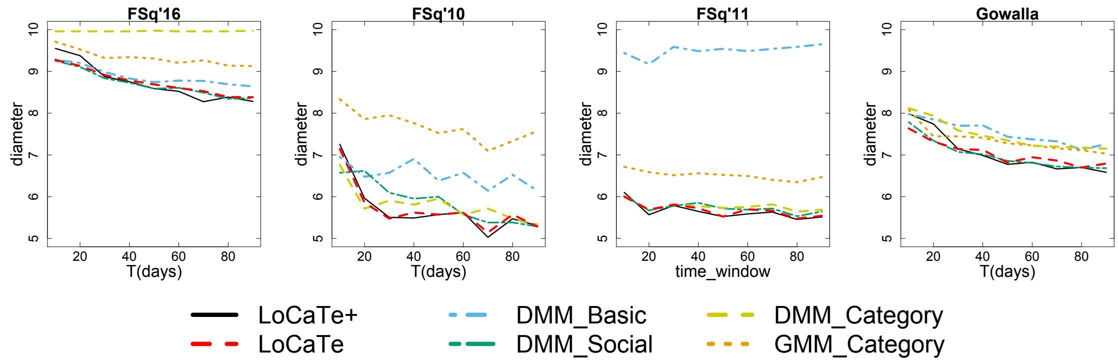


Figure 4.11: The diameter, ϕ, at different time_windows with $\rho = 0.003$ and $\tau = 5$

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 [1, 131, 77, 56].

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 4.11 and 4.12 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 4.9, 4.11, 4.10, and 4.12 shows that the LoCaTe models are able to render better average degree centrality of G_T. Note that, G_T is an unobserved graph

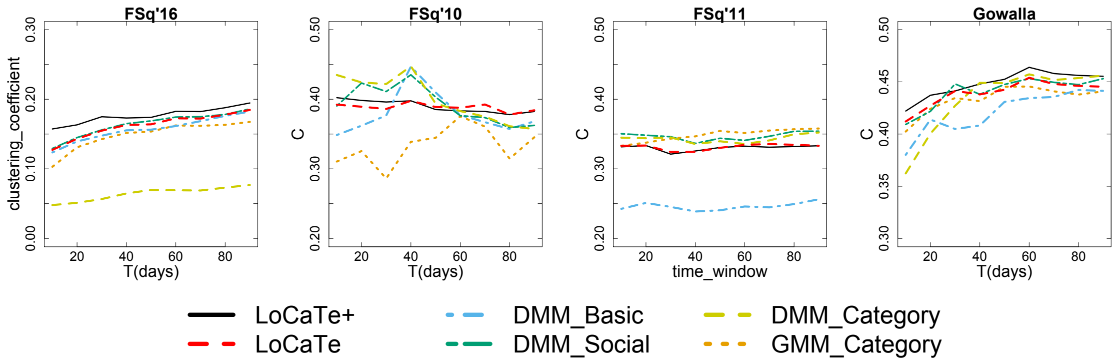


Figure 4.12: The clustering coefficient, C, at different time_windows with $\rho = 0.003$ and $\tau = 5$

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 4.9: Average degree centrality, C_D, of the graph of influenced users for FSq'16

and is formed while testing. Thus, we can conclude that LoCaTe models provide us with more cohesive unobserved graph as compared to other quantification models.

4.6.7 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 4.4.2. 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. Note that, for computing we compare single ground truth location with the recommended rank of the location. For measuring the efficiency of our model we compare our results against

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 4.10: Average degree centrality, C_D, of the graph of influenced users 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 4.11: Average degree centrality C_D of the graph of influenced users for FSq'10

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 4.12: Average degree centrality C_D of the graph of influenced users 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 4.13: Recall at different values of top-k for different datasets

	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 4.14: NDCG at different values of top-k for different datasets

GeoMF++ (Joint Geographical model and Matrix Factorization for Location Recommendation) [65]. 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 4.13 and table 4.14 reports the Recall and NDCG obtained on the Personalized Location Recommendation Task at different $top-k$ values using LoCaTe+ model and GeoMF++ [65], 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. Table 4.15 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 4.16 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.

4.7 Conclusion

In this chapter, 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

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 4.15: 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 4.16: Recall at top-10 for different values of weight parameters for FSq'10 dataset

well as the affinity of users to a location-based on the 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 the 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 some 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 next chapter, we further explore the diffusion process of location-based influence, and also the communities.

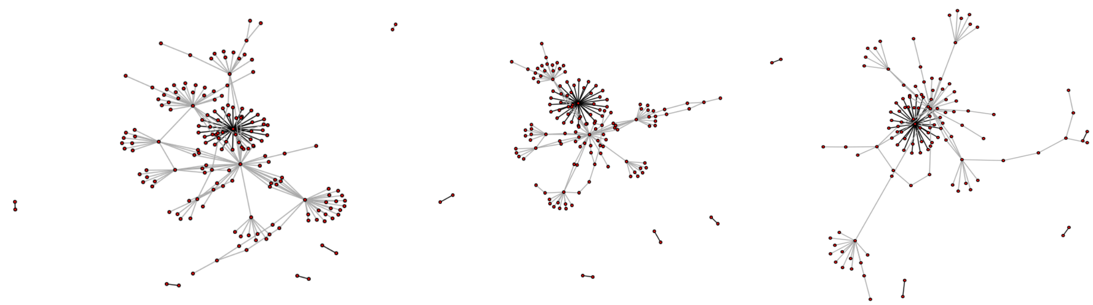
Chapter 5

Information Diffusion and Community Detection

5.1 Introduction

Proliferation of smartphone usage and pervasive data connectivity have made it possible to collect enormous amounts of mobility information of users with relative ease. Foursquare announced in early 2018 that it collects more than 3 billion events every month from its 25 million users[1]. These events generate a location information diffusion process through an underlying –possibly hidden– network of users that determines an location adoption behavior among users. Location adoption primarily depends upon user's spatial, temporal and categorical preferences. For instance, one user's check-in at a newly opened jazz club could inspire another user to visit the same club or a similar club in her vicinity depending upon the distance from the club and time of the day/week. These users might not be having a social connection but it's an implicit influence because of similar choices. This often leads to the formation of –possibly overlapping– communities of users with similar behavior. Detecting community of such like minded people from large geo tagged events

[1] https://bit.ly/2BdhnnP (accessed in February 2019)



(a) threshold = 0.5 (b) threshold = 0.7 (c) threshold = 0.9

Figure 5.1: COLAB extracts underlying diffusion network over US data using geotagged checkin traces. This Maximum Weighted Spanning Forest (MWSF) is constructed by varying the threshold for edge weights (i.e. influence score computed using COLAB). The inferred influence network depicts a tree-like structure of influence.

can benefit applications of various domains such as targeted advertisements and friend recommendation. Prior work [157] also suggests that social connections are not as effective factors for prediction tasks.

In this chapter, we move away from communities derived purely from social network links, and instead explicitly identify spatio-temporal activity-driven communities. Eschewing the reliance on social connections alone allows us to learn communities that are not overly biased towards connected users. At the same time, it supports the use of COLAB under settings where social network is not available – either due to privacy settings of users, or due to other restrictions from platform.

Example 5.1.1. *We illustrate the spatio-temporal activity-based network and communities that* COLAB *derives using the US dataset we collected, shown in Figure 5.1. We observe that explicit social connections, shown using black edges, are very sparse and are unable to capture the implicit influence network. On the other hand, the latent network derived by* COLAB*, shown using gray edges, can identify significantly higher number of relations in the latent influence network. It can be observed that it not only capture the clusters but also identifies potential influencers (and their influence networks) which can critically help in prediction tasks.*

In this work, we determine that the location adoption process and community formation among users can be explained by the same latent factors underlying the observed behavior of users without considering their social network information. We propose **COLAB** (Communities of Location Adoption Behavior) that focuses on jointly inferring location specific influence scores between users and the communities they belong to, based solely on their activity traces. **COLAB** completely disregards the social network information, which makes it suitable for scenarios where only activity traces are available. Note that if social network is available, it can be used as a prior or a regularization over the influence matrix we derive. Unlike the current best-performing models for location prediction task (e.g., [151]), COLAB avoids the community formation from being biased only by the availability of social connections. Thus we generate better communities even for users with few or no social connections as shown in figure 5.1.

Further, we generate overlapping communities of users that take into account the spatio-temporal patterns, and the shared special interests over location categories. Although a user may be part of multiple communities with different special interests, we assume each event or check-in to be associated with only one of those communities. Prior works have focused on modeling temporal + textual together [157] or temporal + spatial features together [147], none of the techniques to the best of our knowledge have modeled the entire combination of temporal, spatial and location semantics in a spatio-temporal point process to infer the underlying influence network.

Our main contributions are as follows:

1. We propose a novel model called COLAB to model activity patterns over geo-tagged event traces. It leverages spatio-temporal Hawkes process [17] to not only construct a information diffusion-based latent network but also recover overlapping community structures within it.

2. We develop a novel stochastic variational inference technique to learn the latent communities and model parameters.

3. As our target is to identify communities that comprise users who share interests as opposed to just being socially connected, there is unfortunately no gold-standard community information to evaluate our results. Therefore, we first empirically evaluate our method over synthetic data; further, results shows our inference algorithm can accurately recover the model parameters. For communities evaluation on real data, we make use of a joint loss function that evaluates on the basis of intra-community properties defined in terms of users' category affinity and spatial dispersment in their checkin characteristics.

4. We evaluate on three real-world geo-tagged event traces collected from three countries – viz., SA (Saudi Arabia), Brazil and US. The experimental results demonstrates that we achieve upto 27% improvement over neural network based models.

5.2 Problem Statement

Consider an geo-tagged event trace dataset S over a set of L locations $\mathbb{L} = \{\ell_k\}_{k=1}^{L} = \{(x_k, y_k)\}_{k=1}^{L}$, a set of I users $\mathbb{U} = \{i_k\}_{k=1}^{I}$ and V categories (restaurants, entertainments etc.). Let us consider there are N events, with the n^{th} check-in denoted as $E_n = (t_n, \ell_n, c_n, i_n, g_n)$ and $\mathbb{E} = \{E_n\}_{n=1}^{N}$. The notation denotes that E_n is the check-in event involving the user i_n checking in to location ℓ_n at time t_n, with the category associated with the location being c_n and the latent community of the user associated with the check-in is g_n. The task is to learn the latent community associated with the users and effectively model the diffusion of information among users. Towards this, we aim to learn a matrix ϕ of size $|\mathbb{U}|$ x $|M|$ where i^{th} row represents community participation for the i^{th} user (assuming M communities). In addition, to model the diffusion process, we estimate matrix A_{ij}, where an element a_{ij} represents the influence of i^{th} user on j^{th} user.

Symbol	Description
$\mathbb{U}$	set of I users
$\mathbb{L}$	set of L locations
$\mathbb{G}$	set of M communities (latent)
$\lambda_i(t, x, y)$	intensity at time t and spatial coordinate x, y of user i
μ_i	base rate of check-ins of user i
A_{ij}	influence of user i on user j
$\kappa(t, x, y)$	triggering / self exciting kernel

Table 5.1: Terminology

5.3 Preliminaries

5.3.1 Hawkes Process

Point processes provides a mathematical framework to specify a distribution over points located in space or time. They are characterized by a latent intensity function which determines the occurrence of points in space or time. Temporal point processes consider the intensity to be a function of time $\lambda(t)$, and it provides instantaneous probability of occurrence of an event at time t. In social networks, events such as posts trigger more future posts. It has been found that Hawkes process[39, 40] , a point process with self triggering property, is more suited to model the occurrences of events in social media [164]. Here, the conditional intensity as a function of time is defined as follows,

$$\lambda(t) = \mu + \sum_{t_k < t} \kappa(t - t_k)$$

where μ is the base intensity with which some events occur, and $\kappa()$ (kernel function) consider the influence of the past events on the current event. Typically, an exponentially decaying function is considered, as the influence of previous events decays exponentially over time.

5.3.2 Multidimensional Hawkes Process

Events can belong to different individuals/entities (dimensions) and one entity can trigger events belonging to other entities. Multidimensional Hawkes process models this mutually exciting property by considering time-stamped events from multiple entities [166]. It allows explicit representation of infectivity among entities. The intensity function for an entity i at time t depends on past events as following

$$\lambda_i(t) = \mu_i + \sum_{t_k < t} A_{i_k i} \kappa(t - t_k),$$

where (t_k, i_k) represents time and entity pair associated with past events, $\mu_i > 0$ is the spontaneous intensity for the i^{th} entity and the non-negative coefficient A_{ij} captures the mutually-exciting property (influence) of the i^{th} entity on the j^{th} entity. Larger values of A_{ij} indicates that events associated with i^{th} entity are likely to trigger more events in the j^{th} entity.

5.3.3 Spatio-Temporal Hawkes Process

The Spatio-temporal Hawkes process is an extension of temporal Hawkes process where the events are modeled as a function of space, time and previous history of events[17, 106]. These models naturally capture triggering and clustering behavior, and have been widely used in fields where spatio-temporal clustering of events is observed, such as earthquake modeling, infectious disease, and crime. The conditional intensity of self exciting spatio-temporal Hawkes process is defined as:

$$\lambda(\ell, t | H_t) = \mu + \sum_{t_k < t} \kappa(\ell - \ell_k, t - t_k)$$

where $\{\ell_1, \ell_2, ..., \ell_n\}$ denotes the observed sequence of locations of events ordered chronologically by time and $\{t_1, t_2, ..., t_n\}$ the observed times of these events.

5.4 CoLAB Model

In this section we describe our model to infer the ϕ matrix i.e. the communities vector inferred from check-in activities and information diffusion over the users in the network.

5.4.1 Spatio-Temporal Data Modeling

Given the check-in events E as defined earlier, for modeling time and spatial components w.r.t. communities, we define the Hawkes process based model as follows:

Intensity Function

We model the user's community-specific intensity using the multi-dimensional spatio-temporal Hawkes process. Multi-dimensional because influence from other users also contribute in the intensity of a user [166]. Consider the task of estimating the community-specific intensity of a user i_n towards generating a check-in $E_n = (t_n, \ell_n, c_n, i_n, g_n)$; the multi-dimensional spatio-temporal Hawkes process formulation yields the following:

$$\lambda_{i_n,g_n}(t_n, \ell_n) = \mu_{i_n}\eta_{g_n} + \sum_{t_k<t_n} A_{i_k i_n}\kappa(t_n - t_k, \ell_n - \ell_k)\mathbb{I}(g_k = g_n) \tag{5.1}$$

where μ_{i_n} is the base intensity of user i_n and η_{g_n} is weight associated to g_n towards a community g_n, and $\boldsymbol{\eta} = \{\eta_g | g = 1, ..., M\}$ with $\eta_g \geq 0$. $\lambda_{i_n,g_n}(t_n, \ell_n)$ is community specific intensity of user i at the n^{th} instance. We allow historical check-ins to contribute to the intensity - proportionate to their temporal and spatial proximity to t_n and ℓ_n respectively, and weighted using the influence between user i_n and i_k (i.e., $A_{i_k i_n}$) - as long as they belong to the same community, enforced by the indicator function $\mathbb{I}(g_k = g_n)$. Here, $\kappa(t_n - t_k, \ell_n - \ell_k)$ is the triggering exponential kernel which factorizes over time and location.

$$\kappa(t_n - t_k, \ell_n - \ell_k) = \kappa(t_n - t_k) * \kappa(\ell_n - \ell_k), \tag{5.2}$$

where, $\kappa(t_n - t_k) = \exp(-\nu(t_n - t_k))$ is the time specific triggering kernel with ν decay and $\kappa(\ell_n - \ell_k) = \frac{1}{2\pi h} exp\left(-\frac{||\ell_n - \ell_k||}{2h}\right)$ is the location specific triggering kernel with h bandwidth. When the decay parameter is low, the influence of the previous events is high and similarly when the bandwidth parameter is high the influence of previous locations is high.

In general, the intensity of a particular user i at some time t and location ℓ, is given as the sum of intensities that are estimated at the level of each community i.e. total intensity $\lambda_i(t, \ell) = \sum_g \lambda_{i,g}(t, \ell)$

5.4.2 Category Distribution

The category c associated with a check-in is represented as a $|V|$-length vector and it represents one of V possible categories[2] such as restaurant, entertainment etc. associated with the check-in. Also, we assume that the category depends on the underlying latent community associated with this check-in. For example, some community may be more inclined towards restaurants while another community is oriented towards sports. The category is modeled as a sample from a Multinomial (categorical) distribution,

$$c \sim Multinomial(\boldsymbol{\theta_g}) \tag{5.3}$$

where $\boldsymbol{\theta_g}$ is a $|V|$-length vector whose elements encode probability of each category and which depends on the community g that the check-in belongs to. We assume a prior over $\boldsymbol{\theta_g}$ as a sample from Dirichlet distribution with parameters $\boldsymbol{\theta_0}$. We write the conditional distribution $p(c, \boldsymbol{\theta_g}|\boldsymbol{\theta_0})$ as:

$$p(c, \boldsymbol{\theta_g}|\boldsymbol{\theta_0}) = p(c|\boldsymbol{\theta_g})p(\boldsymbol{\theta_g}|\boldsymbol{\theta_0}) = \theta_{g,c} \frac{\Gamma(\sum_j \theta_{0,j})}{\prod_j \Gamma(\theta_{0,j})} \prod_j \theta_{g,j}^{\theta_{0,j}-1} \tag{5.4}$$

j runs over the V categories and $p(c|\theta_g)$ is given as $\theta_{g,c}$

[2] We overload the notation c to also represent a scalar categorical value in the set $\{1, \ldots, V\}$

5.4.3 Distribution over communities

We assume the latent variable g (the communities) associated with the user for some check-in, is distributed as multinomial distribution parameterized by π_i for a user i. π_{ig} represents the probability user i belongs to community g.

$$g \sim Multinomial(\boldsymbol{\pi_i}) \tag{5.5}$$

5.4.4 Generative Process

Algorithm 2: Generative Process

Initialize the number of communities M, and number of checkins N_i for each user;
Set μ_i proportional to N_i;
Initialize A_{ij} as column normalized matrix;
Initialize π, η and θ as Dirichlet-Multinomial distribution ;
Initialize $\lambda_i(t_0, x_0, y_0) = \mu_i \quad \forall i = 1, \ldots, U$;
for n = 1 to N **do**
  Sample (t_n, ℓ_n) from $\sum_{i=1}^{U} \lambda_i(t, \ell)$;
  Sample i_n from Multinomial $(\lambda_1(t_n, \ell_n)$, $\lambda_2(t_n, \ell_n)$, ... , $\lambda_U(t_n, \ell_n))$;
  Sample g_n from a Multinomial (π_{i_n});
  Sample c_n from $Multinomial(\theta_{g_n})$ (θ_g is defined in section 5.4.2);
end

Note that, for sampling (t, x, y), the thinning algorithm proposed in [57] is modified in order to sample location coordinates from discrete "venue" set rather the continuous space. First we consider a discrete set of locations L for the user based on her region. We sample (x', y') at n^{th} iteration, from a Gaussian distribution centered at the previous coordinates in the $(n-1)^{th}$ iteration: (x_{n-1}, y_{n-1}). Once (x', y') is sampled, the nearest coordinate in the L is determined and returned as (x_n, y_n).

5.5 Estimation and Inference

Given the multi-dimensional Hawkes process model defined above, the joint probability density function over the check-in events $\mathbb{E}$ is given as:

$$\prod_{n=1}^{N} p(t_n, l_n, c_n, g_n|i_n) = \prod_{n=1}^{N}\Big((p(t_n, l_n|i_n, g_n) \times p(g_n|i_n)) \times p(c_n|g_n, \theta)\Big) \tag{5.6}$$

Here, $p(c_n|g_n, \theta) = \theta_{g_n,c_n}$, where c_n represents the category associated with the n^{th} check-in, and $p(g_n|i_n) = \pi_{i_n,g_n}$ the probability that user i_n belong to the community g_n.

$$\prod_{n=1}^{N} p(t_n, \ell_n|i_n, g_n) = \prod_{n=1}^{N} \lambda_{i_n,g_n}(t_n, \ell_n) \exp\Big(-\sum_{i=1}^{U}\int_{0}^{T}\int_{\ell_{min}}^{\ell_{max}} \lambda_i(t, \ell)dtd\ell\Big) \tag{5.7}$$

is the likelihood (event density) of generating the observations given the community and users in the interval $[(0, \ell_{min}), (T, \ell_{max})]$. The first term in (5.7) provides the instantaneous probability of occurrence of the observed events and the second term provides the probability that no event happens outside these observations (survival probability) [20]. Thus, the complete joint log likelihood is:

$$\mathcal{LL} = \sum_{n=1}^{N}\Big(\log \lambda_{i_n,g_n}(t_n, \ell_n) + \log \pi_{i_n,g_n} + \log \theta_{g_n,c_n}\Big) - \sum_{i=1}^{U}\int_{0}^{T}\int_{\ell_{min}}^{\ell_{max}} \lambda_i(t, \ell)dtd\ell \tag{5.8}$$

Assuming communities are known, we can estimate the model parameters μ, η,A, θ_g's and π's by maximum likelihood estimation. We treat the kernel parameters, and the Dirichlet parameters as the hyper-parameters which we initialize to some fixed values. However, the communities are latent and the maximum likelihood estimation cannot be applied directly. This calls for the expectation maximization algorithm, where the parameters are estimated after integrating out the latent variables from the joint likelihood using the posterior distribution over the latent variables. In our case, the posterior distribution over latent communities is given as

$$p(g_1, \ldots, g_n | \{t_n, \ell_n, c_n, i_n\}_{n=1}^{N}) \\ = \frac{\prod_{n=1}^{N} p(t_n, \ell_n | i_n, g_n) \times p(c_n | g_n, \theta) \times p(g_n | i_n)}{\sum_{g_1,\ldots,g_n} \prod_{n=1}^{N} p(t_n, \ell_n | i_n, g_n) \times p(c_n | g_n, \theta) \times p(g_n | i_n)} \quad (5.9)$$

The posterior distribution over the latent communities cannot be obtained in closed form due to the intractable normalization constant (denominator term) which involves an exponential number of summation terms. Markov chain Monte Carlo methods [4] can be used to obtain samples from the posterior. However, these approaches are not scalable to large datasets [5] and becomes computationally expensive for use in LBSNs. To overcome this, we use a variational expectation maximization algorithm where we approximate the posterior over communities using a variational distribution and estimate the model parameters and variational parameters by maximizing a variational lower bound [152].

5.5.1 Variational Expectation Maximization

The latent variables g_n's dependent on different types of feature set i.e. space, time through $p(t_n, \ell_n | i_n, g_n)$ and semantics through $p(c_n | g_n, \theta)$. Though the prior over g_n is conjugate to $p(c_n | g_n, \theta)$, it is not with respect to $p(t_n, \ell_n | i_n, g_n)$ and hence the posterior over g_n cannot be computed in closed form. Moreover, g_n's are inter-dependent i.e. at current step g_n it is dependent on history from g_1 to g_{n-1} as well as the future ones i.e. g_{n+1}. Thus marginalizing out over such interconnected latent variables to compute the normalization constant for the posterior is intractable. To this end we assume a variational distribution over g_n's conditioned on the user i_n. The conditional variational distribution over g_n is considered to be a multinomial distribution with parameters $\boldsymbol{\phi}_{\boldsymbol{i_n}}$. The variational parameter

$\boldsymbol{\phi_i}$ for a user i represents the posterior probability distributions over the communities for the user as observed from the data.

$$q(g_n|i_n) = Multinomial(g_n|\boldsymbol{\phi_{i_n}}) \tag{5.10}$$

The variational parameters can be learned by minimizing the KL divergence between the variational posterior (5.10) and the exact posterior (5.9). However, a direct minimization of KL divergence is not possible due to the intractable posterior. Following variational inference approach [5], the variational parameters are learned by maximizing a variational lower bound, Evidence Lower Bound (ELBO), which indirectly minimizes the KL divergence. ELBO is obtained by considering an expected value of the complete joint log likelihood w.r.t the variational distribution [5] and acts as a lower bound to the marginal likelihood or evidence (normalization constant of the posterior). Hence, ELBO is useful to learn the model parameters also in addition to the variational parameters. Using the variational distribution defined in (5.10) and the complete joint log likelihood (5.8), we obtain the ELBO as:

$$\begin{aligned}\mathscr{L} = \sum_{n=1}^{N} \left(\mathbb{E}_q[\log \lambda_{i_n,g_n}(t_n, \ell_n)] + \sum_{m=1}^{M} \phi_{i_n,m} \log \pi_{i_n,m} + \sum_{m=1}^{M} \phi_{i_n,m} \log \theta_{m,c_n} \right) \\ - \sum_{i=1}^{U} \int_0^T \int_{\ell_{min}}^{\ell_{max}} \mathbb{E}_q[\lambda_i(t,\ell)] d\ell dt - \mathbb{E}_q[\log q]\end{aligned} \tag{5.11}$$

Here, $\mathbb{E}_q$ represents the expectation with respect to the variational distribution q defined in (5.10). We learn the variational parameters and the model parameters by maximizing the ELBO. Table 5.2 lists the model parameters and variational parameters to be learned using ELBO. All the terms in the ELBO except the first term can be computed in closed form.

Since the first term in (5.11) cannot be computed in closed form, we approximate it

Table 5.2: Parameters to be estimated and whether a Hyper parameter

Par	**Description**	**H**
$\boldsymbol{\mu}$	Base Intensity	
$\boldsymbol{\eta}$	Weight associated towards community	
A_{ij}	Influence Matrix	
h	Bandwidth (KDE)	✓
ν	Temporal Decay Parameter	✓
$\boldsymbol{\theta_0}$	Dirichlet Prior: Category	✓
$\boldsymbol{\theta_g}$	Multinomial Prior: Categories / community	
π	Multinomial Prior: Communities (All users)	
ϕ	Variational Parameters: Communities (All users)	

using the samples from the variational posterior (5.10) (Monte-Carlo approximation).

$$\mathbb{E}_q[\log \lambda_{i_n,g_n}(t_n, \ell_n)] \approx \frac{1}{S}\sum_{s=1}^{S} \log \lambda_{i_n,\boldsymbol{g}^{(s)}}(t_n, \ell_n) \tag{5.12}$$

where $\boldsymbol{g}^{(s)}$ represent the vector of N samples sampled from the joint variational distribution over all the g_n's, *i.e.* $q(\boldsymbol{g}^{(s)}) = \prod_{i=1}^{N} q(g_n^{(s)}|\phi_{i_n})$. This results in a stochastic variational lower bound where the stochasticity arises due to the approximation of expectation using Monte Carlo sampling [152]. We learn the model parameters μ, η, A_{ij}, θ by maximizing the stochastic variational lower bound [97] using gradient based methods. However learning the variational parameters is problematic as the variational parameters does not appear explicitly in the stochastic term but only through the samples. For determining gradient w.r.t. ϕ we apply the Reinforcement trick to the stochastic term and compute the gradient as follows [29]:

$$\nabla_\phi \, \mathbb{E}_q[\log \lambda_{i_n,g_n}(t_n, \ell_n)] \approx \frac{1}{S}\sum_{s=1}^{S} \log \lambda_{i_n,\boldsymbol{g}^{(s)}}(t_n, \ell_n) \, \nabla_\phi \log q(\boldsymbol{g}^{(s)}) \tag{5.13}$$

5.6 Experiments

5.6.1 Baselines

Compared Baselines We empirically evaluate the performance of CoLAB[3] over synthetic and real datasets with the following baselines:

- **STHP**: Spatio-Temporal Hawkes Process models the diffusion process across spatial and temporal dimensions but ignores the *category* associated location feature. This is the baseline derived from CoLAB ignoring the categories.
- **Sequence Mining** [60]: First extracts frequent occuring venue category sequences and assigns communities based on clusters with similar patterns.
- **DH** [26]: Dirichlet-Hawkes, clusters continuous time event streams using a modified Hawkes model with preferential cluster assignment through Dirichlet Process.
- **RMTPP**:[25] Recurrent Marked Temporal Point Process model the time and the marker information by learning a general representation of the nonlinear dependency over the history based on recurrent neural networks. In this model, event history is embedded into a compact vector representation which is then used for predicting the next event time and marker type.
- **LBSN2Vec**: [136] is a hypergraph embedding approach designed specifically for LBSN data. It performs random-walk-with-stay to jointly sample user mobility patterns and social relationships from the LBSN hypergraph, and then learns node embeddings from the sampled hyperedges by preserving the n-wise node proximity captured by a hyperedge containing n nodes. For next location prediction, the similarity between user embedding and location embedding along with time embedding and location embedding is computed for the obtaining ranked list of locations for a user.

[3]We pledge to make our codes and datasets public.

- **LSTM**:[43] Long Short Term Memory model the location sequence of users' trajectories

For an even comparison, we feed the check-in events to all the baselines and evaluate the community quality formed by DH and Sequence Mining along with location prediction for STHP and RMTPP.

5.6.2 Implementation Details

The code is based on Python v2.7 in a standard conda environment. Due to the limited scalability of Hawkes process and its inference process, we parallelize each of the simulation on the basis if A_{ij} dimensions across each core. The results presented were obtained by running on a linux server with 64GB memory with 36 cores: Intel(R) Xeon(R) Gold 6140 CPU @ 2.30GHz.

5.6.3 Datasets

Synthetic Data

We generate synthetic data using algorithm 2 (statistics in Table 5.3). The set of locations, #users to categories i.e. (U:V) and to the number of check-ins i.e. (U:N) are kept similar to the real data collected from Brazil with ν (temporal decay parameter) is set to 0.01 [139] and h is picked up from the bandwidth values learned from users' check-ins. During inference, we use <u>true values</u> for all the parameters, except for the influence matrix A_{ij} and the user-community posterior ϕ which are estimated. We use $RelErr(A_{ij}, \hat{A}_{ij}) = \frac{1}{I^2}\sum_{i,j=1}^{I}\frac{|a_{ij}-\hat{a}_{ij}|}{|a_{ij}|}$ (and similarly for ϕ), as the metric to evaluate the ability to recover the true values. Table 5.4 indicates that the stochastic variational inference technique offers considerably better reconstruction of the parameters, recording significant reductions in the error.

Table 5.3: Dataset Properties

Property	**#Users(U)**	**#Communities(M)**	**#Categories(V)**	**#Check-ins(N)**
Synthetic	100	10	200	9777
SA (Real)	95	-	314	15110
US (Real)	133	-	524	22059
BR (Real)	157	-	381	29354

Table 5.4: $RelErr$ on A and ϕ and True Positives for Location Prediction results at Top-K on Synthetic Data

	RelErr		Top K	
Technique	A	ϕ	5	10
STHP	0.99174	0.13807	681	1206
CoLAB	**0.04813**	**0.07216**	**972**	**1677**

Real Data

For real data, we use our crawls over Foursquare conducted between January-2015 and March-2016 and construct three collections consisting of check-ins from Saudi Arabia (SA), Brazil (BR) and United States (US), with details given in Table 5.3. We allocated first 80% (as per check-in timestamp) of each dataset to *training* and remaining for *testing*. Here, we also use temporal decay parameter as 0.01 [139] and h is learned for each user based on Silverman's rule of thumb in kernel density estimation [4] and is fixed during the joint estimation.

5.6.4 Location Prediction with CoLAB

For location prediction task, we predict the next location from the previously seen locations in the training set at $M(\#communities) = 10$ and at various top-K ranked (eq. 5.8) cutoffs. For instance, consider a user u and L_u is the set of locations u has visited in the training set. For predicting the next location in the test set given previous N latest check-ins we determine the likelihood score of all the locations in set L_u and rank them based on the scores obtained. Note that, N is set to 10 for our experiments.

Since, DH is for clustering event streams, therefore we make use of STHP, LBSN2Vec, LSTM and RMTPP as baselines for the Location Prediction task. Table 5.5 shows that CoLAB is able to offer significant improvements (18-37% in the top-5). As we increase K, these gains diminish because the number of candidate locations saturates. We also study the effect of A_{ij}(influence matrix) and μ (base intensity) over CoLAB's performance. It can be observed that without A_{ij}, the CoLAB's performance degrades signifying CoLAB's ability to capture the underlying diffusion process well.

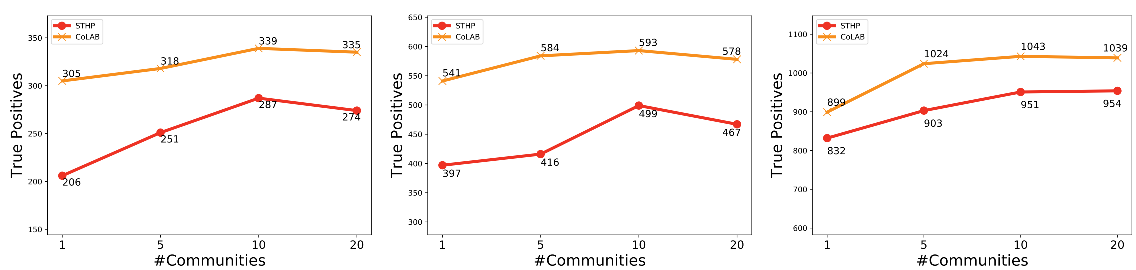
Neural Network based Methods: The pure neural network based methods such as LSTM learns from the sequence of locations and predicts the next location. In case of LBSN the checkin data for each user is sparse, thus for a single user there are fewer data points to train upon. Therefore, these models become unsuitable for such scenarios. In case if we use the entire data i.e. all users' trajectories for learning as we used here to train these models it introduces lot of noise as users' preferences are different. For example, some users go for dessert after a meal while others go for coffee after a meal. These discrepancies leads to poor location prediction results. To conclude, for these model to work well we certainly need large amount of data. Therefore, for these scenarios point process based models seems to be promising, not only because of good results but they also offer flexibility to model contribution towards check-in through different sources i.e. self excitation or through influence of others.

Impact of #Communities

In figures 5.2, 5.3 and 5.4 we study the impact of M, over SA, US, and BR data we observe that with increasing M the prediction accuracy improves and then diminishes, signifying optimal value of M for better predictions. CoLAB performs significantly better than STHP, primarily due to the better estimation of A_{ij} (influence matrix), because of the presence of category information in the CoLAB model.

Table 5.5: Comparison of CoLAB with other baselines and CoLAB without A_{ij} and μ

Dataset	K	STHP	RMTPP	LBSN2Vec	LSTM	CoLAB w/o Aij	CoLAB w/o μ	CoLAB
SA	5	287	250	293	173	279	331	**339**
(#test-	10	455	499	456	256	478	589	**593**
cases =	20	950	951	897	741	911	1038	**1043**
2805)	50	1539	1539	1574	1024	1520	1664	**1666**
US	5	153	172	206	87	172	231	**237**
(#test-	10	456	467	**520**	240	445	507	512
cases =	20	870	888	908	608	919	920	**927**
4395)	50	1700	1691	1667	1012	**1759**	1668	1673
BR	5	256	223	289	186	250	278	**312**
(#test-	10	432	445	545	234	450	512	**589**
cases =	20	984	973	1047	756	989	1088	**1123**
3828)	50	1445	1432	1390	978	1467	1464	**1534**



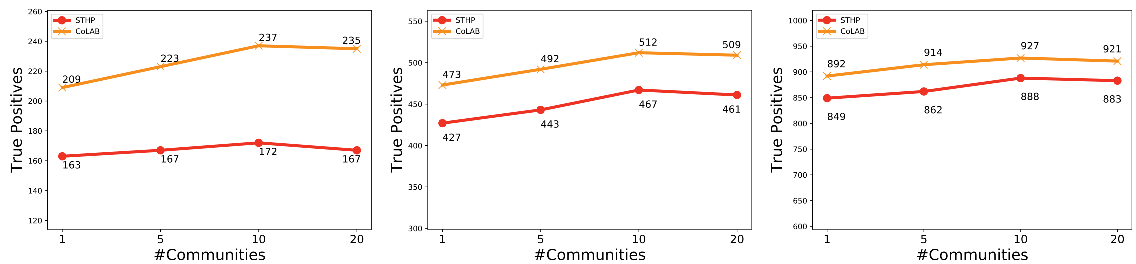
(a) M=5 (b) M=10 (c) M=20

Figure 5.2: Location Prediction Results for Varying M over SA

5.6.5 Community Assessment

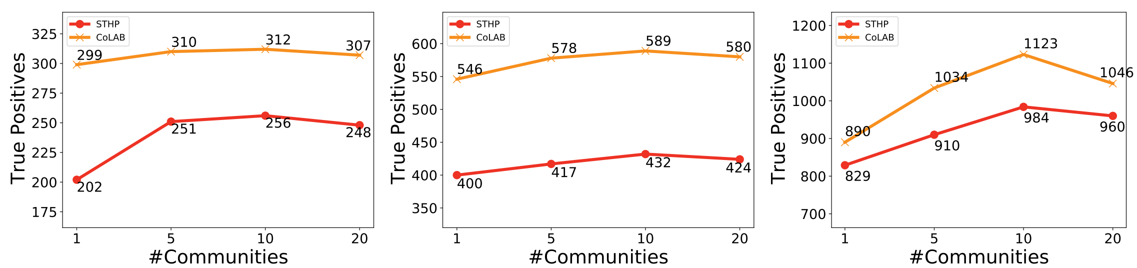
We plot communities over SA and US data in Figure 5.5, where colored dots represents a check-in. In figure 5.5, it can be observed that; (i) *Overlap* between communities due to data concentration in cities. (ii) CoLAB is able to capture communities across cities (e.g. users with frequent visits to Jazz Clubs). As a consequence, clustering quality metrics such as Silhouette score that penalize the overlapping clusters are not a meaningful option.

Unfortunately, we lack the community ground truth for users, making communities assessment a non-trivial task. Thus, we use a metric a joint loss function for the intra-



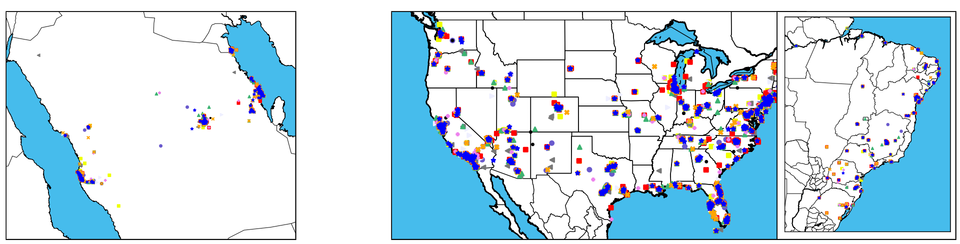
(a) M=5 (b) M=10 (c) M=20

Figure 5.3: Location Prediction Results for Varying M over US



(a) M=5 (b) M=10 (c) M=20

Figure 5.4: Location Prediction Results for Varying M over BR



(a) SA (b) US (c) BR

Figure 5.5: Spatio-Temporal Activity-driven Latent Communities Captured by CoLAB

community properties through a mixture of (i) category loss ($\mathcal{L}_{cat}$) and (ii) location loss ($\mathcal{L}_{loc}$). **Category Loss:** We consider all categories associated to locations as independent *marks* of a point process. Although this is quite unrealistic as some categories do have more association between them (e.g. Jazz Club & Record Shop and Steakhouse & Pizza Place). Hence to judge whether the assigned communities really do capture the category affinity of a user, we try to estimate the similarity among the categories within each community. Hence to estimate the category affinity in a community, we consider similarity among the check-in categories using pre-trained word embeddings [84] and devise a *loss*

Table 5.6: Results for Category Loss

Dataset	K_{cat}	Sequence Mining		DH	STHP	COLAB
		Daily	Weekly			
SA	10	250.24	236.19	**103.47**	125.17	119.41
	50	1118.23	1089.27	862.05	842.63	**826.72**
	100	2007.93	2120.76	1983.61	1784.04	**1749.37**
US	10	248.45	217.56	**98.37**	118.32	113.86
	50	956.87	990.45	781.83	793.07	**771.15**
	100	1907.84	2020.49	1901.37	1605.64	**1583.02**

function.

$$\mathcal{L}_{cat} = \frac{1}{|\mathbf{T}|} \sum_{E_n \in \mathbf{T}} \sum_{g \in \mathcal{M}} \left\{ 1 - \frac{\mathbf{v}_{E_n} \cdot \mu_g}{||\mathbf{v}_{E_n}||_2 ||\mu_g||_2} \right\} \cdot \Phi(E_n, g) \tag{5.14}$$

where $\mathbf{T}$ represents test data, μ_g is category mean for a community g using K_{cat} frequent categories, $\mathbf{v}_{E_n}$ is the category vector for event E_n; $\Phi(E_n, g)$ indicates whether E_n is assigned to community g, with $\mathcal{M}$ as all communities. Table 5.6 demonstrates COLAB's ability to capture category dynamics across communities. Although, it can be observed that at K_{cat} = 10 Dirichlet Hawkes performs better because with most frequent categories like restaurant, coffee shop etc. DH assigns it most of the communities. Note that, DH is unable to capture communities with varied categories as seen at K_{cat} = 50 and 100, that COLAB performs better.

Location Loss:[16] show that users tend to visit nearby locations given we ignore the bias of loyalty. Hence ideally, a community of users not spatially dispersed in their check-in characteristics should be distinct from another community with check-ins spanning large distances. We capture this through a distance based *k-means* loss ($\mathcal{L}_{loc}$) with cluster means(μ_l) for check-in coordinates for each community. In table 5.7 we can see COLAB performs significantly better than other baselines because COLAB can better capture the geographical dispersion in communities.

Table 5.7: Results for Location Loss

Datasets	Sequence Mining		DH	STHP	CoLAB
	Daily	Weekly			
SA	600.67	547.56	413.16	306.73	**298.09**
US	1127.34	1067.50	1039.08	849.92	**834.64**

5.6.6 Qualitative Assessment

We claim that a user in a community will display an affinity towards certain categories. For US data, figure 5.6 shows even with highly overlapping venue categories, our model finds the intricate differences between a community with affinity to music (a) and a community with affinity towards food/bar joints. The word clouds for SA and BR dataset shows similar properties and have been avoided for brevity.

(a) Community A (b) Community B

Figure 5.6: Word Cloud of Categories in Two Communities from US

5.7 Conclusion

In this chapter, we presented CoLAB that uses spatio-temporal Hawkes process to infer the implicit communities using a novel stochastic variational inference technique. Empirical evaluations over synthetic as well as real-world datasets highlight its prowess with significant improvements in location and community detection tasks. This illustrates the effectiveness of the modeling used in CoLAB in generating user communities even in the absence of social connectedness information. In future work, we would like to explore scalability of COLAB through sample-based inference techniques.

Chapter 6

Conclusion and Future Directions

6.1 Summary of Contributions

In this thesis, we have brought out the potential of exploiting a location's categorical features such as restaurant, club etc and its modeling aspects jointly along with spatial and temporal features for three different problems.

- First, we exploited coarse grained category information by mapping annonymised location information (i.e. publicly available) with category information using Foursquare APIs. For the proof of concept: Will inferring categories would be useful? We modeled the coarse grained category information for next location prediction application, and obtained improvements over state-of-the-art methods.

- Next we curated users' spatio-temporal footprints along with categories at a fine grained level using FourSqaure and Twitter users' alignment, and developed a model to quantify the influence specific to a location between a pair of users. We collected Foursquare data between Jan 2015- Feb 2016 using a long-term crawl.

  Specifically, we developed 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 developed **LoCaTe+**, an advanced model within the same framework where user correlation is quantified using a Mutually Exciting Hawkes Process. We showed the applicability of LoCaTe and LoCaTe+ for location promotion and location recommendation tasks over the above crawl as well as other publicly available LBSN datasets. Our experiments demonstrated the efficacy of the LoCaTe framework in capturing location-specific influence between users. We also showed that our model improve over state-of-the-art models for the task of location promotion as well as location recommendation.

- The location check-ins of users event-traces often manifest in hidden (possibly overlapping) communities of users with similar interests. Inferring these implicit communities is crucial for forming user profiles for improvements in recommendation and prediction tasks. Given only time-stamped geo-tagged traces of users, can we find out these implicit communities, and characteristics of the underlying influence network? Can we use this network to improve the next location prediction task? We focused on the problem of community detection as well as capturing the underlying diffusion process and propose a model COLAB based on spatio-temporal point processes in continuous time but discrete space of locations that simultaneously models the implicit communities of users based on their check-in activities, without making use of their social network connections. This model captures the semantic features of the location, user-to-user influence along with spatial and temporal preferences of users. To learn the latent community of users and model parameters, we propose an algorithm based on stochastic variational inference. To the best of our knowledge, this is the first attempt at jointly modeling the diffusion process with activity-driven implicit communities. We demonstrated COLAB achieves upto 27% improvements in location prediction task over recent deep point-process based methods on geo-tagged event traces collected from Foursquare check-ins.

6.2 Future Directions:

We see number of open issues that can be addressed in future work:

- **Set of Influential Users and Locations:** Most of the existing work as well as in this thesis we looked at the problem of influence maximization and location promotion only by identifying influential users. An interesting direction for future work will be to jointly identify users and locations for geo-social influence maximization and recommendation. For example, for the promotion of a ”saloon” online, it can be made through influencers identified and offline, it can be done at the hot locations where the saloon clientele is expected to visit such as a cafe.

- **Face-to-face Co-location Prediction:** Another interesting future work would be to predict whether, where and when two users will have offline meetings. Given the sparsity of LBSN data this will open up further directions of curating the data from different sources. For example, the current LBSN data has very few connections and also trajectories of users are quite sparse therefore services such as Instagram and Flickr can be used to extract trajectories and social connections by mining photos and videos which users haven’t explicitly marked.

- **Social Link Prediction:** In this thesis, for identifying implicit communities we didn’t make use of the social connections, and computed influence score between two users using COLAB model. An interesting direction would be to extend this to identify social ties. The main challenge here is the non existence of ground truth. Therefore, the first task in hand is to curate the dataset that has rich set of connections, which can be used as a ground truth for evaluation.

Publications

Journal Articles

- **Ankita Likhyani**, Srikanta Bedathur and Deepak P. *Location-Specific Influence Quantification in Location-Based Social Networks*. In ACM Transactions on Intelligent Systems and Technology (TIST 2019). (https://dl.acm.org/doi/10.1145/3300199)

Conference Articles

- **Ankita Likhyani**, Vinayak Gupta, Srijith PK, Deepak P and Srikanta Bedathur. *Modeling Implicit Communities using Spatio-Temporal Point Processes from Geo-tagged Event Traces*. In Proceedings of the 21st International Conference on Web Information Systems Engineering (WISE 2020). (https://arxiv.org/abs/2006.07580)

- **Ankita Likhyani**, Srikanta Bedathur and Deepak P. *LoCaTe: Influence Quantification for Location Promotion in Location Based Social Networks*. In Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017). (https://www.ijcai.org/proceedings/2017/0314.pdf)

- **Ankita Likhyani**, Deepak P, Srikanta Bedathur and Sameep Mehta. *Inferring and Exploiting Categories for Next Location Prediction*. In Proceedings of the 24th International Conference on World Wide Web Companion (WWW 2015). (https://dl.acm.org/citation.cfm?id=2742770)

Bibliography

[1] N. Barbieri, F. Bonchi, and G. Manco. Cascade-based community detection. In WSDM, pages 33–42. ACM, 2013.

[2] M. Bayati, M. Gerritsen, D. F. Gleich, A. Saberi, and Y. Wang. Algorithms for large, sparse network alignment problems. In 2009 Ninth IEEE International Conference on Data Mining, pages 705–710, 2009.

[3] P. Bhargava, T. Phan, J. Zhou, and J. Lee. Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In Proceedings of the 24th International Conference on World Wide Web, WWW '15, page 130–140, 2015.

[4] C. M. Bishop. Pattern Recognition and Machine Learning. Springer, 2006.

[5] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. Journal of the American Statistical Association, 112(518):859–877, 2017.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research, 3:993–1022, 2003.

[7] P. Bouros, D. Sacharidis, and N. Bikakis. Regionally influential users in location-aware social networks. In Proceedings of the 22nd ACM SIGSPATIAL International

Conference on Advances in Geographic Information Systems, SIGSPATIAL '14, 2014.

[8] C. Bowsher. Modelling security market events in continuous time: Intensity based, multivariate point process models. Journal of Econometrics, 141(2):876–912, 2007.

[9] S. Cao, W. Lu, and Q. Xu. Deep neural networks for learning graph representations. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, page 1145–1152, 2016.

[10] T. Chen, H. Yin, H. Chen, L. Wu, H. Wang, X. Zhou, and X. Li. Tada: Trend alignment with dual-attention multi-task recurrent neural networks for sales prediction. In 2018 IEEE International Conference on Data Mining (ICDM), pages 49–58, 2018.

[11] T. Chen, H. Yin, H. Chen, R. Yan, Q. V. H. Nguyen, and X. Li. Air: Attentional intention-aware recommender systems. In 2019 IEEE 35th International Conference on Data Engineering (ICDE), pages 304–315, 2019.

[12] W. Chen, C. Wang, and Y. Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In KDD, pages 1029–1038. ACM, 2010.

[13] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, AAAI'12, page 17–23. AAAI Press, 2012.

[14] C. Cheng, H. Yang, M. R. Lyu, and I. King. Where you like to go next: Successive point-of-interest recommendation. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, IJCAI '13, page 2605–2611, 2013.

[15] W. Cheng, Y. Shen, Y. Zhu, and L. Huang. Delf: A dual-embedding based deep latent factor model for recommendation. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, page 3329–3335, 2018.

[16] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11, pages 1082–1090, 2011.

[17] Y.-S. Cho, A. Galstyan, P. J. Brantingham, and G. Tita. Latent self-exciting point process model for spatial-temporal networks. In Discrete ande Continuous Dynamical Systems - B, volume 19, 2014.

[18] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In NIPS 2014 Workshop on Deep Learning, December 2014, 2014.

[19] D. CONTE, P. FOGGIA, C. SANSONE, and M. VENTO. Thirty years of graph matching in pattern recognition. International Journal of Pattern Recognition and Artificial Intelligence, 18(03):265–298, 2004.

[20] D. J. Daley and D. Vere-Jones. An introduction to the theory of point processes. Vol. I. Probability and its Applications (New York). Springer-Verlag, New York, second edition, 2003. Elementary theory and methods.

[21] M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16, page 3844–3852, 2016.

[22] A. Doan, J. Madhavan, P. Domingos, and A. Halevy. Ontology Matching: A Machine Learning Approach, pages 385–403. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[23] E. Dorsey, C. Y. Yvonne, M. McConnell, S. Shaw, A. Trister, and S. Friend. The use of smartphones for health research. pages 157–160.

[24] G. Droj. Gis and remote sensing in environmental management. Journal of environmental protection and ecology, 2, 01 2012.

[25] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song. Recurrent marked temporal point processes: Embedding event history to vector. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, page 1555–1564, 2016.

[26] N. Du, M. Farajtabar, A. Ahmed, A. J. Smola, and L. Song. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In SIGKDD, 2015.

[27] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In Proceedings of the 2018 World Wide Web Conference, WWW '18, page 1459–1468, 2018.

[28] S. Feng, X. Li, Y. Zeng, G. Cong, Y. M. Chee, and Q. Yuan. Personalized ranking metric embedding for next new poi recommendation. In Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI'15, page 2069–2075, 2015.

[29] Y. Gal. Uncertainty in Deep Learning. PhD thesis, University of Cambridge, 2016.

[30] R. Ganesan and S. P V. Text Book of Statistics (as per ICAR UG Syllabus). 01 2016.

[31] H. Gao, J. Tang, X. Hu, and H. Liu. Modeling temporal effects of human mobile behavior on location-based social networks. In Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management, CIKM '13, pages 1673–1678, 2013.

[32] H. Gao, J. Tang, X. Hu, and H. Liu. Content-aware point of interest recommendation on location-based social networks. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI'15, page 1721–1727, 2015.

[33] H. Gao, J. Tang, and H. Liu. Exploring social-historical ties on location-based social networks. In ICWSM. The AAAI Press, 2012.

[34] H. Gao, J. Tang, and H. Liu. gscorr: Modeling geo-social correlations for new check-ins on location-based social networks. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pages 1582–1586, 2012.

[35] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira. Exploiting innocuous activity for correlating users across sites. In Proceedings of the 22nd International Conference on World Wide Web, WWW '13, page 447–458, 2013.

[36] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. In WSDM, pages 241–250. ACM, 2010.

[37] A. Grover and J. Leskovec. Node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, page 855–864, 2016.

[38] A. G. Hawkes. Point spectra of some mutually exciting point processes. Biometrika, 58(1):83–90, 1971.

[39] A. G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. Biometrika, 58(1):83–90, 1971.

[40] A. G. Hawkes and D. Oakes. A cluster process representation of a self-exciting process. Journal of Applied Probability, 11(3):493–503, 1974.

[41] T. He, H. Yin, Z. Chen, X. Zhou, S. W. Sadiq, and B. Luo. A spatial-temporal topic model for the semantic annotation of pois in lbsns. ACM TIST, 8(1):12:1–12:24, 2016.

[42] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web, WWW '17, page 173–182, 2017.

[43] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural Comput., 9(8):1735–1780, Nov. 1997.

[44] J. Huang, W. X. Zhao, H. Dou, J.-R. Wen, and E. Y. Chang. Improving sequential recommendation with knowledge-enhanced memory networks. In The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '18, 2018.

[45] P. Jin, Z. Liu, and Y. Xiao. Discovering the most influential geo-social object using location based social network data. In 2020 IEEE International Conference on Knowledge Graph (ICKG), pages 607–614, 2020.

[46] M. Kalaev, V. Bafna, and R. Sharan. Fast and accurate alignment of multiple protein networks. In Research in Computational Molecular Biology, pages 246–256. Springer Berlin Heidelberg, 2008.

[47] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In KDD, pages 137–146. ACM, 2003.

[48] D. Kempe, J. M. Kleinberg, and É. Tardos. Influential nodes in a diffusion model for social networks. In ICALP, volume 3580 of Lecture Notes in Computer Science, pages 1127–1138. Springer, 2005.

[49] R. R. Khorasgani, J. Chen, and O. R. Zaïane. Top leaders community detection approach in information networks. In 4th SNA-KDD workshop on Social Network mining and Analysis. Citeseer, 2010.

[50] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In 5th International Conference on Learning Representations, ICLR 2017, 2017.

[51] P. Koehn. Statistical significance tests for machine translation evaluation. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pages 388–395, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[52] X. Kong, J. Zhang, and P. S. Yu. Inferring anchor links across multiple heterogeneous social networks. In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM '13, page 179–188, 2013.

[53] D. Koutra, H. Tong, and D. Lubensky. Big-align: Fast bipartite graph alignment. In 2013 IEEE 13th International Conference on Data Mining, pages 389–398, 2013.

[54] J. M. Kumpula, M. Kivelä, K. Kaski, and J. Saramäki. Sequential algorithm for fast clique percolation. Physical Review E, 78(2):026109, 2008.

[55] M. Kutz. McGraw-Hill Education, 2011.

[56] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. ACM Trans. Web, 1(1), May 2007.

[57] P. A. W. Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. Naval Research Logistics Quarterly, 26(3):403–413, 1979.

[58] C.-T. Li and H.-P. Hsieh. Geo-social media analytics. In Proceedings of the 24th International Conference on World Wide Web, WWW '15 Companion, 2015.

[59] G. Li, S. Chen, J. Feng, K. Tan, and W. Li. Efficient location-aware influence maximization. In SIGMOD Conference, pages 87–98. ACM, 2014.

[60] H. Li, K. Deng, J. Cui, Z. Dong, J. Ma, and J. Huang. Hidden community identification in location-based social network via probabilistic venue sequences. Information Sciences, 422:188 – 203, 2018.

[61] R. Li, Y. Shen, and Y. Zhu. Next point-of-interest recommendation with temporal and multi-level context attention. In 2018 IEEE International Conference on Data Mining (ICDM), pages 1110–1115, 2018.

[62] D. Lian, Y. Wu, Y. Ge, X. Xie, and E. Chen. Geography-aware sequential location recommendation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '20, page 2009–2019, 2020.

[63] D. Lian, X. Xie, V. W. Zheng, N. J. Yuan, F. Zhang, and E. Chen. Cepr: A collaborative exploration and periodically returning model for location prediction. ACM Trans. Intell. Syst. Technol., 6(1), Apr. 2015.

[64] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui. Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, page 831–840.

[65] D. Lian, K. Zheng, Y. Ge, L. Cao, E. Chen, and X. Xie. Geomf++: Scalable location recommendation via joint geographical modeling and matrix factorization. ACM Trans. Inf. Syst., 36(3):33:1–33:29, 2018.

[66] M. Lichman and P. Smyth. Modeling human location data with mixtures of kernel densities. In SIGKDD, 2014.

[67] A. Likhyani, S. Bedathur, and D. P. Locate: Influence quantification for location promotion in location-based social networks. In IJCAI, 2017.

[68] A. Likhyani, S. Bedathur, and D. P. Location-specific influence quantification in location-based social networks. ACM Trans. Intell. Syst. Technol., 10(3), 2019.

[69] A. Likhyani, V. Gupta, P. K. Srijith, P. Deepak, and S. Bedathur. Modeling implicit communities from geo-tagged event traces using spatio-temporal point processes. In Z. Huang, W. Beek, H. Wang, R. Zhou, and Y. Zhang, editors, Web Information Systems Engineering – WISE 2020, pages 153–169, Cham, 2020. Springer International Publishing.

[70] A. Likhyani, D. Padmanabhan, S. J. Bedathur, and S. Mehta. Inferring and exploiting categories for next location prediction. In WWW (Companion Volume), pages 65–66. ACM, 2015.

[71] J. Liu, Y. Li, G. Ling, R. Li, and Z. Zheng. Community detection in location-based social networks: An entropy-based approach. In IEEE CIT, 2016.

[72] L. Liu, W. K. Cheung, X. Li, and L. Liao. Aligning users across social networks using network embedding. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, page 1774–1780, 2016.

[73] Q. Liu, S. Wu, L. Wang, and T. Tan. Predicting the next location: A recurrent model with spatial and temporal contexts. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, page 194–200, 2016.

[74] S. Liu, S. Wang, F. Zhu, J. Zhang, and R. Krishnan. Hydra: Large-scale social identity linkage via heterogeneous behavior modeling. In Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, SIGMOD '14, page 51–62, 2014.

[75] X. Liu, Y. Liu, K. Aberer, and C. Miao. Personalized point-of-interest recommendation by mining users' preference transition. In Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management, CIKM '13, pages 733–738, 2013.

[76] Y. Liu, W. Wei, A. Sun, and C. Miao. Exploiting geographical neighborhood characteristics for location recommendation. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM '14, pages 739–748, 2014.

[77] H. Ma, H. Yang, M. R. Lyu, and I. King. Mining social networks using heat diffusion processes for marketing candidates selection. In CIKM, pages 233–242. ACM, 2008.

[78] J. Maantay and A. Maroko. Mapping urban risk: Flood hazards, race, and environmental justice in new york. Applied Geography, 29(1):111 – 124, 2009.

[79] T. Man, H. Shen, S. Liu, X. Jin, and X. Cheng. Predict anchor links across social networks via an embedding approach. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, page 1823–1829, 2016.

[80] F. Manne and M. Halappanavar. New effective multithreaded matching algorithms. In 2014 IEEE 28th International Parallel and Distributed Processing Symposium, pages 519–528, 2014.

[81] J. Manotumruksa, C. Macdonald, and I. Ounis. A deep recurrent collaborative filtering framework for venue recommendation. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM '17, page 1429–1438, 2017.

[82] J. Manotumruksa, C. Macdonald, and I. Ounis. A contextual attention recurrent architecture for context-aware venue recommendation. In The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '18, page 555–564, 2018.

[83] S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity flooding: a versatile graph matching algorithm and its application to schema matching. In Proceedings 18th International Conference on Data Engineering, pages 117–128, 2002.

[84] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin. Advances in pre-training distributed word representations. In LREC, 2018.

[85] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13, page 3111–3119, 2013.

[86] J. Miles. R Squared, Adjusted R Squared. American Cancer Society, 2014.

[87] H. Miller and S.-L. Shaw. Geographic information systems for transportation (gis -t): Principles and applications. 01 2001.

[88] R. Miller and N. Lammas. Social media and its implications for viral marketing. Asia Pacific Public Relations Journal, 2010.

[89] X. Mu, F. Zhu, E.-P. Lim, J. Xiao, J. Wang, and Z.-H. Zhou. User identity linkage by latent user space modelling. In Proceedings of the 22nd ACM SIGKDD International

Conference on Knowledge Discovery and Data Mining, KDD '16, page 1775–1784, 2016.

[90] A. Narayanan, H. Paskov, N. Z. Gong, J. Bethencourt, E. Stefanov, E. C. R. Shin, and D. Song. On the feasibility of internet-scale author identification. In 2012 IEEE Symposium on Security and Privacy, pages 300–314.

[91] B. Neiger, R. Thackeray, S. van Wagenen, C. Hanson, J. West, M. Barnes, and M. Fagen. Use of social media in health promotion: Purposes, key performance indicators, and evaluation metrics. Health Promotion Practice, 13(2):159–164, 3 2012.

[92] J. A. Nelder and R. Mead. A simplex method for function minimization. Computer Journal, 7:308–313, 1965.

[93] M. E. J. Newman. Power laws, pareto distributions and zipf's law. Contemporary Physics, 2005.

[94] M. Niepert, M. Ahmed, and K. Kutzkov. Learning convolutional neural networks for graphs. In Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML'16, page 2014–2023, 2016.

[95] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. Mining user mobility features for next place prediction in location-based services. In ICDM, ICDM '12, pages 1038–1043, 2012.

[96] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. Exploiting semantic annotations for clustering geographic areas and users in location-based social networks. In The Social Mobile Web, ICWSM Workshop, 2011.

[97] J. W. Paisley, D. M. Blei, and M. I. Jordan. Variational bayesian inference with stochastic search. In ICML, 2012.

[98] J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, 2014.

[99] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, page 701–710, 2014.

[100] H. Pham and C. Shahabi. Spatial influence - measuring followship in the real world. In ICDE, pages 529–540. IEEE Computer Society, 2016.

[101] D. M. W. Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. Journal of Machine Learning Technologies, 2011.

[102] A. Prat-Pérez, D. Dominguez-Sal, and J.-L. Larriba-Pey. High quality, scalable and parallel community detection for large real graphs. In Proceedings of the 23rd international conference on World wide web, pages 225–236. ACM, 2014.

[103] J. Qian, X. Li, C. Zhang, and L. Chen. De-anonymizing social networks and inferring private attributes using knowledge graphs. In IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, pages 1–9, 2016.

[104] H. A. Rahmani, M. Aliannejadi, M. Baratchi, and F. Crestani. Joint geographical and temporal modeling based on matrix factorization for point-of-interest recommendation. In J. M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. J. Silva, and F. Martins, editors, Advances in Information Retrieval. ECIR 2020. Springer International Publishing, 2020.

[105] J. Ratkiewicz, M. Conover, M. R. Meiss, B. Gonçalves, A. Flammini, and F. Menczer. Detecting and tracking political abuse in social media. In ICWSM. The AAAI Press, 2011.

[106] A. Reinhart. A review of self-exciting spatio-temporal point processes and their applications. Statist. Sci., 33(3):299–318, 08 2018.

[107] Y. Ren, Y. Zhang, M. Zhang, and D. Ji. Context-sensitive twitter sentiment classification using neural network. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, page 215–221, 2016.

[108] C. Riederer, Y. Kim, A. Chaintreau, N. Korula, and S. Lattanzi. Linking users across domains with location data: Theory and validation. In Proceedings of the 25th International Conference on World Wide Web, WWW '16, page 707–719. International World Wide Web Conferences Steering Committee, 2016.

[109] M. A. Saleem, R. Kumar, T. Calders, X. Xie, and T. B. Pedersen. Location influence in location-based social networks. In WSDM, pages 621–630. ACM, 2017.

[110] S. Scellato, C. Mascolo, M. Musolesi, and V. Latora. Distance matters: Geo-social metrics for online social networks. In Proceedings of the 3rd Wonference on Online Social Networks, WOSN'10, pages 8–8, 2010.

[111] S. Scellato, A. Noulas, and C. Mascolo. Exploiting place features in link prediction on location-based social networks. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11, pages 1046–1054, 2011.

[112] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 45(11):2673–2681, Nov 1997.

[113] Y. Shen and H. Jin. Controllable information sharing for user accounts linkage across multiple online social networks. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM '14, page 381–390, 2014.

[114] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu. Heterogeneous information network embedding for recommendation. IEEE Trans. on Knowl. and Data Eng., 31(2):357–370, 2019.

[115] K. Shu, S. Wang, J. Tang, R. Zafarani, and H. Liu. User identity linkage across online social networks: A review. SIGKDD Explor. Newsl., 18(2):5–17, 2017.

[116] B. W. Silverman. Density estimation for statistics and data analysis. Chapman & Hall, London, 1986.

[117] R. Singh, J. Xu, and B. Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In T. Speed and H. Huang, editors, Research in Computational Molecular Biology, pages 16–31. Springer Berlin Heidelberg, 2007.

[118] A. Smalter, J. Huan, and G. Lushington. Gpm: A graph pattern matching kernel with diffusion for chemical compound classification. In 2008 8th IEEE International Conference on BioInformatics and BioEngineering, pages 1–6, 2008.

[119] S. Tan, Z. Guan, D. Cai, X. Qin, J. Bu, and C. Chen. Mapping users across networks by manifold alignment on hypergraph. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI'14, page 159–165, 2014.

[120] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, WWW '15, page 1067–1077, 2015.

[121] Y. Tay, L. A. Tuan, and S. C. Hui. Latent relational metric learning via memory-based attention for collaborative ranking. In P. Champin, F. L. Gandon, M. Lalmas, and P. G. Ipeirotis, editors, Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW, pages 729–739, 2018.

[122] L. Q. Tran, M. Farajtabar, L. Song, and H. Zha. Netcodec: Community detection from individual activities. In SDM. SIAM, 2015.

[123] C. Tu, H. Liu, Z. Liu, and M. Sun. CANE: Context-aware network embedding for relation modeling. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1722–1731, 2017.

[124] C. Tu, W. Zhang, Z. Liu, and M. Sun. Max-margin deepwalk: Discriminative learning of network representation. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, page 3889–3895.

[125] C. Tu, Z. Zhang, Z. Liu, and M. Sun. Transnet: Translation-based network representation learning for social relation extraction. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, pages 2864–2870, 2017.

[126] D. Wang, P. Cui, and W. Zhu. Structural deep network embedding. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, page 1225–1234, 2016.

[127] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A.-L. Barabasi. Human mobility, social ties, and link prediction. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11, pages 1100–1108, 2011.

[128] R. Wang, C. Yu, X.-D. Yang, W. He, and Y. Shi. Eartouch: Facilitating smartphone use for visually impaired people in mobile and public scenarios. In Proceedings of

the 2019 CHI Conference on Human Factors in Computing Systems, CHI '19, page 1–13, 2019.

[129] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang. Community preserving network embedding. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI'17, page 203–209, 2017.

[130] X. Wang, Y. Zhang, W. Zhang, and X. Lin. Distance-aware influence maximization in geo-social network. In 2016 IEEE 32nd International Conference on Data Engineering (ICDE), pages 1–12, 2016.

[131] Y. Wang, G. Cong, G. Song, and K. Xie. Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. In KDD, pages 1039–1048. ACM, 2010.

[132] Z. Wang, D. Zhang, D. Yang, Z. Yu, and X. Zhou. Detecting overlapping communities in location-based social networks. In Social Informatics, 2012.

[133] H. Wu and M. Yeh. Influential nodes in a one-wave diffusion model for location-based social networks. In PAKDD (2), volume 7819 of Lecture Notes in Computer Science, pages 61–72. Springer, 2013.

[134] R. Wu, G. Luo, J. Shao, L. Tian, and C. Peng. Location prediction on trajectory data: A review. Big Data Mining and Analytics, 1(2):108–127, 2018.

[135] Xucheng Luo, Fan Zhou, Mengjuan Liu, Yajun Liu, and Chunjing Xiao. Efficient multi-account detection on ugc sites. In 2016 IEEE Symposium on Computers and Communication (ISCC), pages 450–455, 2016.

[136] D. Yang, B. Qu, J. Yang, and P. Cudre-Mauroux. Revisiting user mobility and social relationships in lbsns: A hypergraph embedding approach. In The World Wide Web Conference, WWW '19, page 2147–2157, 2019.

[137] D. Yang, D. Zhang, and B. Qu. Participatory cultural mapping based on collective behavior data in location-based social networks. ACM TIST, 7(3):30:1–30:23, 2016.

[138] S. Yang and H. Zha. Mixture of mutually exciting processes for viral diffusion. In Proceedings of the 30th International Conference on Machine Learning, ICML, pages 1–9, 2013.

[139] S.-H. Yang and H. Zha. Mixture of mutually exciting processes for viral diffusion. In Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ICML'13, pages II–1–II–9, 2013.

[140] Z. Yang, W. W. Cohen, and R. Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML'16, page 40–48, 2016.

[141] D. Yao, C. Zhang, J. Huang, and J. bi. Serm: A recurrent model for next location prediction in semantic trajectories. In CIKM, pages 2411–2414, 2017.

[142] H. Ye, M. Malu, U. Oh, and L. Findlater. Current and future mobile and wearable device use by people with visual impairments. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, page 3123–3132, 2014.

[143] M. Ye, P. Yin, and W.-C. Lee. Location recommendation for location-based social networks. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '10, page 458–461, 2010.

[144] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In SIGIR, 2011.

[145] H. Yin, W. Wang, H. Wang, L. Chen, and X. Zhou. Spatial-aware hierarchical collaborative deep learning for poi recommendation. IEEE Transactions on Knowledge and Data Engineering, 29(11):2537–2551, 2017.

[146] J. J.-C. Ying, W.-C. Lee, T.-C. Weng, and V. S. Tseng. Semantic trajectory mining for location prediction. In Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11, page 34–43, 2011.

[147] B. Yuan, H. Li, A. L. Bertozzi, P. J. Brantingham, and M. A. Porter. Multivariate spatiotemporal hawkes processes and network reconstruction. SIAM Journal on Mathematics of Data Science, 2019.

[148] R. Zafarani and H. Liu. Connecting users across social media sites: A behavioral-modeling approach. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13, page 41–49, 2013.

[149] R. Zafarani, L. Tang, and H. Liu. User identification across social media. ACM Trans. Knowl. Discov. Data, 10(2), 2015.

[150] A. Zarezade, S. Jafarzadeh, and H. Rabiee. Spatio-temporal modeling of check-ins in location-based social networks. PLOS ONE, 13, 11 2016.

[151] A. Zarezade, S. Jafarzadeh, and H. R. Rabiee. Recurrent spatio-temporal modeling of check-ins in location-based social networks. PLOS ONE, 13:1–20, 05 2018.

[152] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt. Advances in variational inference. CoRR, abs/1711.05597, 2017.

[153] C. Zhang, L. Shou, K. Chen, G. Chen, and Y. Bei. Evaluating geo-social influence in location-based social networks. In CIKM, pages 1442–1451. ACM, 2012.

[154] J. Zhang and P. S. Yu. Integrated anchor and social link predictions across social networks. In Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI'15, page 2125–2131, 2015.

[155] J. Zhang and P. S. Yu. Multiple anonymized social networks alignment. In 2015 IEEE International Conference on Data Mining, pages 599–608, 2015.

[156] J. Zhang and P. S. Yu. Pct: Partial co-alignment of social networks. In Proceedings of the 25th International Conference on World Wide Web, WWW '16, page 749–759. International World Wide Web Conferences Steering Committee, 2016.

[157] P. Zhang, X. Wang, and B. Li. On predicting twitter trend: Factors and models. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '13, page 1427–1429, 2013.

[158] S. Zhang and H. Tong. Final: Fast attributed network alignment. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, page 1345–1354, 2016.

[159] S. Zhang, H. Yin, Q. Wang, T. Chen, H. Chen, and Q. V. H. Nguyen. Inferring substitutable products with deep network embedding. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pages 4306–4312, 2019.

[160] Y. Zhang, J. Tang, Z. Yang, J. Pei, and P. S. Yu. Cosnet: Connecting heterogeneous social networks with local and global consistency. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, page 1485–1494, 2015.

[161] F. Zhao and A. K. Tung. Large scale cohesive subgraphs discovery for social network visual analysis. Proceedings of the VLDB Endowment, 6(2):85–96, 2012.

[162] K. Zhao, Y. Zhang, H. Yin, J. Wang, K. Zheng, X. Zhou, and C. Xing. Discovering subsequence patterns for next poi recommendation. In C. Bessiere, editor, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, pages 3216–3222, 2020.

[163] P. Zhao, H. Zhu, Y. Liu, J. Xu, Z. Li, F. Zhuang, V. S. Sheng, and X. Zhou. Where to go next: A spatio-temporal gated network for next POI recommendation. In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, pages 5877–5884, 2019.

[164] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec. Seismic: A self-exciting point process model for predicting tweet popularity. In KDD, 2015.

[165] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang. Collaborative filtering meets mobile recommendation: A user-centered approach. In Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI'10, page 236–241, 2010.

[166] K. Zhou, H. Zha, and L. Song. Learning triggering kernels for multi-dimensional hawkes processes. In ICML, volume 28, 2013.

[167] T. Zhou, J. Cao, Bo, S. Xu, Z. Zhu, and J. Luo. Location-based influence maximization in social networks. In CIKM, pages 1211–1220. ACM, 2015.

[168] X. Zhou, X. Liang, H. Zhang, and Y. Ma. Cross-platform identification of anonymous identical users in multiple social media networks. IEEE Transactions on Knowledge and Data Engineering, 28(2):411–424, 2016.

[169] W. Zhu, W. Peng, L. Chen, K. Zheng, and X. Zhou. Modeling user mobility for location promotion in location-based social networks. In KDD, pages 1573–1582. ACM, 2015.

[170] W. Zhu, W. Peng, L. Chen, K. Zheng, and X. Zhou. Exploiting viral marketing for location promotion in location-based social networks. TKDD, 11(2):25:1–25:28, 2016.