

NANYANG
TECHNOLOGICAL
UNIVERSITY

**EXPLOITING SPATIAL, TEMPORAL,
AND SEMANTIC INFORMATION FOR
POINT-OF-INTEREST
RECOMMENDATION**

A thesis submitted
for a degree of Doctor of Philosophy
by

YUAN QUAN

School of Computer Engineering
Nanyang Technological University

September 2014

Acknowledgments

I would like to express my sincere gratitude to my supervisor Prof. Gao Cong, for his patient guidance, encouragement, immense knowledge and advice. I feel very lucky to be his student. He taught me not only the state-of-the-art data mining knowledge, but also how to do research. The knowledge and methodology I learnt are invaluable to my research, and even to my life. I feel very grateful to my co-supervisor Prof. Nadia Magnenat Thalmann, for her continuous support and insightful inter-disciplinary guidance. She taught me the right attitude to research as a Ph.D student: do not simply follow the steps of previous researchers, but try something new. Without her support and guidance, I would not be able to finish my Ph.D career. I also would like to thank Prof. Aixin Sun and Dr. Chin-Yew Lin, for sharing their extensive knowledge and experience in research.

I am truly indebted and thankful for the Institute for Media Innovation (IMI) and Data-Intensive Scalable Computing (DISCO) Lab, for the convenient space and excellent facilities provided by them.

I owe sincere gratitude to my friends for their friendship and insightful discussions. They include Xin Cao, Lisi Chen, Kaiqi Zhao, Shanshan Feng, Nguyen Tuan Anh Pham, Tao Guo, Kaiyu Feng, Zhida Chen, Yaqiong Liu, Bo Liu, Wei Wei, Zhen Hai, Ying Lu, and Chuancong Gao, who have been or being supervised by Prof. Gao Cong. I would like to appreciate all my fellow friends at the DISCO Lab and IMI for their support: Zongyang Ma, Chenliang Li, Zhiqiang Xu, Ning Chen, Wenqing Lin, Diwen Zhu, Boshen Zhang, Shumo Chu, Peilin Zhao, Shaohua Li, Yong Liu, Jun Zhang, Youze Tang, Sibowang, Hui Fang, Hongyuan Zhu, Guibing Guo, Lu Dong, Yuzhe Zhang, Peng Cheng, Wenjing Zhang, and Juzheng Zhang. I would also like to acknowledge my friends at Microsoft Research, especially, Xin Xin, Jing Liu, Xin Zhao, Lei Shi, Jingang Wang, Chung-Yi Li, Jyun-Yu Jiang, Chun-Che Wu, Yingkai Gao, and Danqing Huang.

Last but not least, I would like to express my deep gratitude to my parents, for their selfless support, encouragement, and love all the time.

Contents

| | |
|--|-----------|
| Acknowledgments | i |
| List of Figures | vi |
| List of Tables | ix |
| Summary | xi |
| 1 Introduction | 1 |
| 1.1 Geo-annotated User Generated Content | 1 |
| 1.2 Point-of-interest Recommendation | 3 |
| 1.3 Problems and Research Scope | 3 |
| 1.4 Approaches and Methodologies | 6 |
| 1.5 Research Contributions | 9 |
| 1.6 Dissertation Organization | 10 |
| 2 Literature Review | 11 |
| 2.1 Recommender Systems | 11 |
| 2.1.1 Collaborative Filtering | 11 |
| 2.1.2 Content-based Recommendation | 14 |
| 2.1.3 Hybrid Recommendation | 14 |
| 2.2 POI Recommendation | 15 |
| 2.2.1 Context-unaware POI Recommendation | 15 |
| 2.2.2 Time-aware POI Recommendation | 20 |
| 2.2.3 Group Recommendation | 23 |
| 2.3 Topic Models | 26 |
| 2.3.1 Introduction to Topic Models | 26 |
| 2.3.2 Geographical Topics Modeling | 29 |

| | | |
|----------|---|-----------|
| 2.4 | Mobility Behaviors of Users | 30 |
| 2.5 | Summary | 32 |
| 3 | Time-aware Point-of-interest Recommendation | 33 |
| 3.1 | Problem Definition | 33 |
| 3.2 | Data Analysis and Problem Definition | 34 |
| 3.2.1 | Dataset | 34 |
| 3.2.2 | Observation of Temporal Influence | 35 |
| 3.2.3 | Observation of Spatial Influence | 35 |
| 3.3 | User-based CF Approach | 36 |
| 3.3.1 | Utilizing Temporal Influence | 37 |
| 3.3.2 | Utilizing Spatial Influence | 41 |
| 3.3.3 | A Unified Framework | 43 |
| 3.4 | Graph-based Preference Propagation Approach | 44 |
| 3.4.1 | Geographical-Temporal Influences Aware Graph (GTAG) | 45 |
| 3.4.2 | Breadth-first Preference Propagation | 48 |
| 3.5 | Experiments | 54 |
| 3.5.1 | Experimental Setup | 54 |
| 3.5.2 | Tuning Parameters | 56 |
| 3.5.3 | Experimental Results | 58 |
| 3.5.4 | Efficiency of Methods | 62 |
| 3.6 | Summary | 64 |
| 4 | Group POI Recommendation | 65 |
| 4.1 | Model Description | 66 |
| 4.1.1 | Problem Statement | 66 |
| 4.1.2 | COnsensus Model for Group POI Recommendation | 67 |
| 4.1.3 | Parameter Estimation | 72 |
| 4.1.4 | Recommendation | 74 |
| 4.1.5 | Adapting COM for Movie Recommendation | 75 |
| 4.2 | Experiments | 75 |
| 4.2.1 | Experimental Setup | 76 |

| | | |
|----------|--|------------|
| 4.2.2 | Experimental Results | 78 |
| 4.2.3 | Weight of Topics in Item Selection | 84 |
| 4.2.4 | Topic Analysis | 86 |
| 4.3 | Summary | 87 |
| 5 | Requirement-aware POI Recommendation | 89 |
| 5.1 | Characteristics of Individual Mobility | 90 |
| 5.2 | Intuitions and Notation | 91 |
| 5.3 | W^4 Model Description | 95 |
| 5.4 | EW^4 Model Description | 97 |
| 5.5 | Parameter Estimation | 98 |
| 5.5.1 | Parameter Estimation for W^4 | 100 |
| 5.5.2 | Parameter Estimation for EW^4 | 102 |
| 5.6 | Applications | 108 |
| 5.7 | Experiments | 111 |
| 5.7.1 | Dataset | 111 |
| 5.7.2 | POI Prediction for Tweets | 112 |
| 5.7.3 | Requirement-aware POI recommendation | 117 |
| 5.7.4 | Example Mobility Pattern | 120 |
| 5.7.5 | Results of Example Applications | 121 |
| 5.8 | Summary | 122 |
| 6 | Conclusions and Future Work | 125 |
| 6.1 | Conclusions | 125 |
| 6.2 | Future Work | 127 |
| A | List of Publications | 129 |
| | Appendix - List of Publications | 128 |

List of Figures

| | | |
|------|---|----|
| 1.1 | Examples of Geo-annotated UGC | 2 |
| 2.1 | The SIS Model | 25 |
| 2.2 | The PIT Model | 26 |
| 2.3 | The Graphical Models of pLSA, LDA and HDP | 28 |
| 2.4 | The Graphical Models in Hong’s Paper | 31 |
| 3.1 | Returning Probability on Foursquare Dataset | 36 |
| 3.2 | Distribution of Distance between Successive Check-ins | 37 |
| 3.3 | User Behavior Similarities between a Given Hour (6:00, 8:00, and 16:00) and other Hours | 39 |
| 3.4 | The Distribution of 5 Popular POIs over Time | 43 |
| 3.5 | GTAG Constructed using 7 Sample Check-ins | 47 |
| 3.6 | Examples of Valid Propagation Paths of Lengths 3, 4, and 6 | 49 |
| 3.7 | Tuning Parameter α | 57 |
| 3.8 | Impact of Parameters η and H for TAG-BPP | 57 |
| 3.9 | Impact of Parameters τ and k for GTAG-BPP | 58 |
| 3.10 | Performance of Methods utilizing Temporal Influence | 60 |
| 3.11 | Performance of Methods with Geographical Influence | 61 |
| 3.12 | Sensitivity to Sparsity Problem | 62 |
| 3.13 | Efficiency of methods on Gowalla dataset, and the three methods BPP, DPP, RWR on different regions | 63 |
| 4.1 | The Graphical Model of COM | 71 |
| 4.2 | <i>Pre&Rec</i> under #Recommendations Made (N) | 79 |

| | | |
|-----|---|-----|
| 4.3 | <i>Pre@5&Rec@5</i> under #Topics (K) | 81 |
| 4.4 | <i>nDCG</i> under #Topics (K) | 82 |
| 4.5 | <i>Rec@5</i> and <i>nDCG</i> under Different p | 83 |
| 4.6 | <i>Rec@5</i> and <i>nDCG</i> for Different Size of Groups | 84 |
| 4.7 | Avg. λ_u under #Topics (K) | 85 |
| 4.8 | Distribution of λ_u | 86 |
| 4.9 | POI Distribution of Topics | 87 |
| 5.1 | Distribution of Distance Between Individuals' Check-ins | 91 |
| 5.2 | Visited POIs of a User | 91 |
| 5.3 | Visiting Time of POIs in Different Days | 92 |
| 5.4 | The Graphical Representation of W^4 | 96 |
| 5.5 | The Graphical Representation of EW^4 | 100 |
| 5.6 | Prediction Performance of all Methods | 115 |
| 5.7 | Recommendation Performance of all methods on both data sets | 119 |
| 5.8 | Personal Regions | 120 |
| 5.9 | Region Distribution over Time | 121 |

List of Tables

| | | |
|-----|---|-----|
| 1.1 | Overview of Studies on POI Recommendation | 6 |
| 3.1 | Symbols for Time-aware POI Recommendation | 34 |
| 3.2 | Statistics on the Datasets (after pre-processing) | 35 |
| 3.3 | The 11 Methods Evaluated in our Experiments | 55 |
| 3.4 | Statistics on the CA- $\{NV, MT, MI, ME\}$ Datasets | 64 |
| 4.1 | Symbols for Group POI Recommendation | 67 |
| 4.2 | Statistics of the Datasets | 77 |
| 4.3 | Comparison of Methods | 78 |
| 4.4 | Representative Movies for COM Topics | 86 |
| 5.1 | Symbols for Requirement-aware POI Recommendation | 93 |
| 5.2 | Symbols for Requirement-aware POI Recommendation (continue) | 94 |
| 5.3 | Statistics of Datasets | 112 |
| 5.4 | Comparison of baseline methods with $EW_{\setminus T}^4$ and EW^4 | 114 |
| 5.5 | Location prediction <i>Acc</i> of PMM, W^4 and EW^4 | 122 |
| 5.6 | User prediction <i>Acc</i> of PMM, W^4 and EW^4 | 122 |

Summary

With the prevalence of 3G & 4G services, people can easily share their opinions, moods, and activities with others via smartphones and tablets. As mobile devices are often GPS-enabled, a great quantity of user-generated content (UGC) with geographic locations has been accumulated, such as check-ins in location-based social networks (LBSNs), event-records in event-based social networks (EBSNs), and geo-annotated tweets on Twitter. Besides geographic location, UGC is often associated with timestamp and contains text content. The spatial, temporal and semantic information embedded in geo-annotated UGC can be exploited for a number of appealing applications and research problems. Point-of-interest (POI) recommendation is a representative one, which aims at recommending places that a target user has not visited before. Obviously, POI recommendation can help people explore new places and know their cities better. In addition, merchants can also benefit from it to deliver location-based advertisements and attract more customers.

In recent years, a number of POI recommendation methods have been proposed, but most of them neglect contextual information, and make recommendations only based on user-POI check-in matrix. In real life, however, a user's preference to POIs is often influenced by her surroundings or context, such as time, companions, etc. For example, a user may prefer shopping malls to pubs in the afternoon, but may prefer pubs at night. Therefore, contextual information should be an important consideration for POI recommendation. In addition, a user may have specific requirement for recommendations sometimes, which directly reveals the user's preference. Thus, in this dissertation, we exploit the contextual information and requirements to recommend POIs for users. Specifically, we study three recommendation tasks that are relevant to the spatial, temporal, and semantic information of users.

First, as human mobility is greatly influenced by time, we believe temporal influence is an important consideration for POI recommendation. We define a new problem, namely, *time-aware POI recommendation*, which aims to return a list of POIs for a user to visit at a specific time. In addition to temporal influence, human mobility is also influenced by geographic distance, *e.g.*, people often visit their nearby places. To exploit both the temporal and spatial influences, we propose two algorithms, namely, *User-based*

Collaborative Filtering with Temporal preference and smoothing Enhancement + Spatial influence with popularity Enhancement (UTE+SE) and *Geographical-Temporal influences Aware Graph+Breadth-first Preference Propagation (GTAG-BPP)*, both of which are effective in making time-aware POI recommendations. We evaluate the performance of the proposed methods on two datasets, and the results show that the proposed methods outperform the state-of-the-art baselines significantly.

Second, we observe that people often participate in activities and visit places together with others, *e.g.*, watching movies with friends, and having dinner with colleagues. Thus, *group POI recommendation* is a realistic and important task, which aims at recommending POIs for a group of people. However, group recommendation is a challenging task, since group members may have different preferences, and how to balance their preferences is still an open problem. Furthermore, groups are often ad hoc, and the number of history records of a group may be very limited. The cold-start problem caused by ad hoc groups makes group recommendations even harder. To this end, we propose a Latent Dirichlet Allocation (LDA) based *COnsensus Model (COM)* to simulate the generative process of group activities and make POI recommendations for a group of users. Extensive experimental results on four real-world datasets validate that our model COM achieves superior recommendation accuracy comparing with five baselines.

Third, when submitting recommendation requests, users may have clear requirements, *e.g.*, dining or shopping, and the requirements can be formulated as short text. To make use of such information, we define a new task, namely, *requirement-aware POI recommendation*, that generates a list of POIs for a target user based on her specific requirements. In addition, when target time is available, the recommendation results could be also time-aware. However, making time-aware and requirement-aware POI recommendations is non-trivial, as it calls for a model that can take into account the user, time, POI and words factors simultaneously. To solve this problem, we propose two frameworks, namely, a probabilistic Latent Semantic Analysis (pLSA) based model *Who+Where+When+What* (W^4) and a Hierarchical Dirichlet Process (HDP) based model *Enhanced W^4* (EW^4), to model the complex interactions among the four factors, and make time-aware and requirement-aware POI recommendations. Empirical studies on two real-world datasets demonstrate our proposals outperform state-of-the-art approaches substantially.

In summary, in this dissertation, we exploit spatial, temporal, and semantic information to recommend POIs to users, which is a natural but novel extension of exiting proposals on POI recommendation.

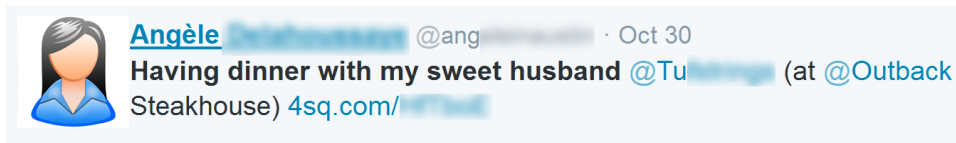
Chapter 1

Introduction

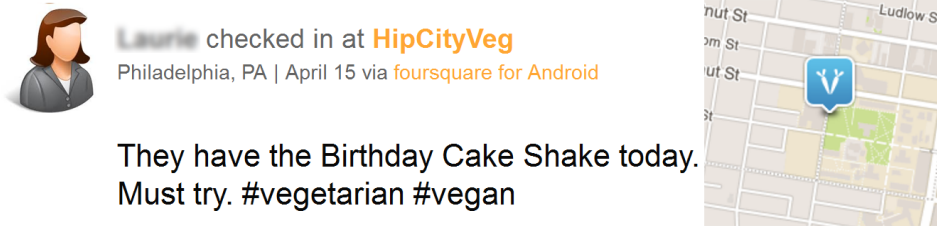
1.1 Geo-annotated User Generated Content

The past decade witnessed a vigorous development and prevalence of 3G & 4G services, which enable people to access the Internet anytime anywhere via mobile devices, such as smartphones and tablets. As mobile devices are often GPS-enabled, many social media have been equipped with spatial information: on the one hand, traditional social networks, such as Facebook [Faca] and Twitter [Twic], enable users to associate geographic locations to their posts to indicate the current surroundings (Figure 1.1.a); on the other hand, new social media that are built upon geographic locations are emerging quickly, such as location-based social networks (LBSNs) and event-based social networks (EBSNs). LBSNs, such as Foursquare [Foua] and Facebook Places [Facb], allow users to post their physical locations in the form of “check-ins”, and share their experiences and tips for points-of-interests (POIs), such as restaurants and sightseeing sites, etc. For example, the Foursquare check-in in Figure 1.1.b shows that the user visited a Vegan restaurant on April 15, and she was very satisfied with the food there. Different from LBSNs that focus on individual customers, EBSNs, such as Meetup and Plancast, are built for groups of users. On EBSNs, a user can initiate an event at a specific place and time, and others who are interested in it can join the event. For example, Figure 1.1.c shows that a BBQ party was going to be held at Golden Gate Park at 1:30 pm on October 26, and it had already attracted 19 users.

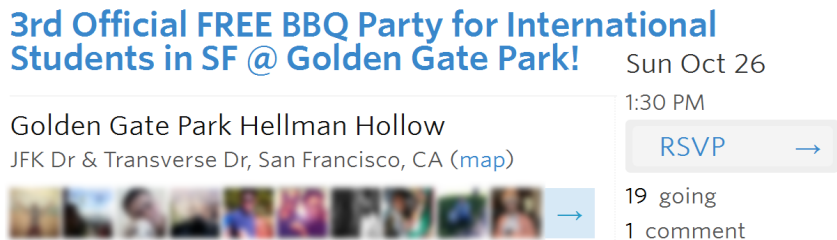
With the development of the location-aware social media, a sheer amount of user generated content (UGC) has been associated with geographic locations, *i.e.*, geo-annotated: as of January 2014, Foursquare accumulated over 5 billion check-ins that were made by 45 million users [Foub], and for Meetup, there are about 0.45 million events being held every month [Mee]. In addition, the number of tweets (short text messages with the length of 140 characters) on Twitter is increased at the speed of 1 to 2 million per hour,



1.1.a: Tweet



1.1.b: Foursquare Check-in



1.1.c: Meetup Event

Figure 1.1: Examples of Geo-annotated UGC

and about 2.7% tweets contained geographic information about users’ current surroundings according to a report dated on June 2013 [Twia]. We name such UGC with location information “geo-annotated UGC”, which is associated with posting user ID, geographic location, timestamp and sometimes text content.

Note that geographic locations may be in the form of the latitude and longitude coordinates, or POIs, *e.g.*, Times Square. While the coordinates can be captured by GPS devices, the exact POIs can be specified explicitly by users, detected by mobile devices, or annotated by geo-tagging tools. In this dissertation, we focus on POIs because of 3 reasons: 1) POIs have specific functions such as dining or shopping; 2) coordinates are less discriminant than POIs, *e.g.*, two POIs on different floors of a shopping mall may share the same coordinates; 3) coordinates are a kind of POI metadata. Thus, compared with coordinates, POIs are more important to users.

1.2 Point-of-interest Recommendation

The rich spatial-temporal information of geo-annotated UGC, along with the semantics embedded in the text content, provides exciting opportunities for a number of recommendation tasks, one representative of which is POI recommendation.

POI recommendation aims at recommending POIs that a target user has not visited, and it has been widely deployed on many websites such as Foursquare, Yelp [Yel], hungrywhere [Hun], etc. Since it is common to have thousands of POIs even in a small city, and a user may only have visited a small portion of them, POI recommendation can help both residents and visitors to explore new interesting places. For example, consider a user who has visited a number of Chinese restaurants in New York city, the POI recommender system can discover her preference in Chinese cuisine, and recommend new Chinese restaurants that she has not been to. Besides residents and visitors, business owners of POIs can also benefit from POI recommendation to launch advertisements and attract more customers.

In recent years, a great amount of research interest has been devoted to POI recommendation, and a number of algorithms have been proposed [YYLL11, CYKL12, LLAM13, CYLK13, LFYX13, KIH⁺13, YSC⁺13, HE13, NSLM12, BZM12, WTM13, FYL13]. Most of them recommend POIs only based on user-POI check-in matrix and neglect contextual information. However, in real life, users' preferences to POIs are often influenced by context, such as time, companions, etc. Consider a user Alice who wants to find a restaurant on Friday evening with her boyfriend Bob. Obviously, both the time "Friday evening" and the companion "Bob" will affect Alice's choice. As contextual information is neglected by existing proposals, it is difficult for them to generate satisfactory POI recommendations for users. In addition, sometimes users may have specific requirements for recommendations, *e.g.*, "a restaurant that serves pasta". Obviously, requirements directly reflect users' preferences, but they have not been exploited in existing proposals.

In this dissertation, we define novel POI recommendation problems that exploit the context and requirement information to generate accurate recommendations.

1.3 Problems and Research Scope

Three POI recommendation problems are defined and investigated in the dissertation, namely, time-aware POI recommendation, group POI recommendation, and requirement-aware POI recommendation.

Time-aware POI Recommendation

It has been reported that user mobility is influenced by time, and it shows a strong periodic behavior throughout a day [LDH⁺10, CML11]. Thus, when recommending POIs,

time should be an important consideration. For example, a user is more likely to go to a restaurant rather than a pub for lunch at noon, and is more likely to go to a pub rather than a library at midnight.

The key point to the solution is how to exploit the time factor. Traditional POI recommender systems rely on the user-POI check-in matrix, and neglect the temporal information. Let $|U|$ and $|L|$ be the numbers of users and POIs, respectively. The user-POI matrix C is a $|U| \times |L|$ matrix, where the element $c_{u,l} = 1$ if user u visited POI l , and $c_{u,l} = 0$ otherwise. Thus, the recommendation results are not time-specific, *e.g.*, a shopping mall might be suggested for a user at midnight, even though it is closed at that time. A straightforward solution is to rank candidate POIs based on the popularity at the target time, *i.e.*, ranking POIs based on their check-in frequencies at the target time in descending order, and returning the top-ranked ones as results. However, as the target user's check-in records are not utilized, this method cannot provide personalized results. Several proposals [LLAM13, CYLK13] try to exploit temporal information to recommend POIs for users to visit at the successive time slots. For example, given that a white collar checked in at her office before leaving the company, they can recommend a restaurant for her to have dinner. However, these proposals suffer from two limitations: 1) these models will be ineffective when the target time is far from that of the target user's last check-in; 2) they neglect users' periodic mobility behaviors.

To exploit time as additional information for recommendation, we define a new task, namely, time-aware POI recommendation, which aims to return a set of POIs for a user to visit at a specific time in a day. Formally, given the target user u and target time t , we rank candidate POIs $l \in L - L_u$ based on user u 's preference $r_{u,i,t}$ at time t in descending order, and return the top-ranked ones as the recommendations, where L and L_u are the whole POI set and set of POIs that user u has visited before, respectively.

Group POI Recommendation

People often participate in activities together with others, *e.g.*, having dinners with colleagues, and having picnics with friends. How to find a POI that can satisfy a group of people is a realistic problem. In addition, as more and more people are willing to share their group activities on social networks, such as Facebook, Meetup, and Foursquare, a great number of group event records have been accumulated, which further promotes the research interests in group recommendation [JS07, AYRC⁺09, BMR10, YLL12, LTYL12].

However, none of existing POI recommender systems are designed for groups. As different people have different preferences, the existing approaches are not effective in making recommendations for a group of people. Recall the example in Section 1.2, suppose Alice is interested in pasta, while Bob is interested in sushi. Obviously, conventional approaches cannot balance their preferences and generate appropriate suggestions for them. Furthermore, groups are often ad hoc, and the number of history records of a

group may be very limited. For example, suppose one day Alice, Bob and their friend Lily want to find a restaurant for dinner together. Then this group is new, and no historical record can be utilized. The cold-start problem caused by ad hoc groups makes group recommendations even harder.

To address the challenges, we focus on group POI recommendation as our second research problem. Formally, given a set of users \mathbf{u} , the group POI recommendation aims to rank candidate POIs l based on the preference $r_{\mathbf{u},l}$ of users \mathbf{u} in descending order, and then returns the top-ranked ones as the recommendations. We believe group POI recommendation can not only facilitate POI selections for group activities, but also help web services improve user engagement.

Requirement-aware POI Recommendation

Users may have clear requirements before submitting the recommendation requests, *e.g.*, shopping or having dinner. Obviously, the requirements explicitly reveal users' preferences, and thus can be utilized for recommendation. However, to the best of our knowledge, none of previous studies on POI recommendation has considered users' requirements.

A straightforward solution is to formulate the requirements as categories [LLAM13]. For example, Alice who is interested in pasta may select "Italian Restaurant" category as her requirement. However, the category is too general to describe her requirement precisely, and some restaurants about Pizza may still be recommended for her. Another solution is to match the keywords of requirements with the POI descriptions, so that the POIs that do not contain the keywords will be filtered out. However, if synonyms or hyponyms of the keywords are used in POI descriptions, this approach will exclude these relevant POIs incorrectly. For example, a restaurant that is described as "spaghetti" or "macaroni" will be filtered out by keyword-matching, even though "spaghetti" and "macaroni" are two particular kinds of pasta.

To better understand users' interests for recommendation, we define a new task, namely, requirement-aware POI recommendation, which aims at predicting POIs for a target user based on her specific requirements. Formally, given a target user u and her requirement keywords \mathbf{w} , the requirement-aware POI recommender system ranks candidate POIs l based on the preference $r_{u,\mathbf{w},l}$ of user u in descending order, and then returns the top-ranked ones as the recommendations. In addition, as time is an important consideration for recommendation, the requirement-aware results can be also time-specific.

The overview of studies on POI recommendation problems is shown in table 1.1.

The relevant publications are also annotated in this table, where the papers published by the author is marked in bold font. This figure gives an overview of the POI recommendation problems, from which we can find that the three POI recommendation problems studied in the dissertation are all new that have not been investigated before.

Table 1.1: Overview of Studies on POI Recommendation

| | Time-aware | Require.-aware | Time&Require -unaware |
|------------|-----------------------------------|---|--|
| Individual | [YCM ⁺ 13a] [YCS14] | [YCM ⁺ 13b] [YCZ ⁺ 15] | [YYLL11], [CYKL12] [LSEM12], [KIH ⁺ 13] [LLAM13], <i>etc.</i> |
| Group | - | - | [YCL14] |

1.4 Approaches and Methodologies

Time-aware POI Recommendation

To address the time-aware POI recommendation problem, we explore the following two influences on users’ daily activities, namely, temporal influence and spatial influence.

For temporal influence, we assume that a user’s temporal behaviors vary in different time of a day, where the temporal behavior of a user is reflected by her check-ins to POIs over time. We exploit two temporal influences, namely, global temporal influence and personal temporal influence. The global temporal influence assumes that users with similar temporal behavior may visit the same set of POIs in the same time, and thus the temporal behavior of other similar users in the target time can be exploited to recommend POIs for the target user. For example, if two users go to the same school in the morning and go to the same library in the afternoon, then the restaurant that one user always visits in the noon can be recommended to the other user. The personal temporal influence assumes that a user’s temporal behavior at or around a specific time plays a more important role than the user’s overall behavior. For example, the temporal behavior of a user at 2-3 pm might be similar to the temporal behavior of the user at 3-4 pm, because the user is likely to stay at workplace and make check-ins around it.

To exploit the temporal influences, we develop two approaches under the frameworks of user-based collaborative filtering and preference propagation, respectively. Specifically, we split time into slots and model the temporal preference to POIs of a user in a time slot by her visited POIs in the time slot. For user-based collaborative filtering (CF), splitting time into slots will make the check-in data sparser. To tackle this problem, we enhance the proposed method with smoothing by taking advantage of user’s temporal preference in other time slots. Given a target user and target time, we first find the users sharing similar temporal preferences with her, and then generate the time-specific recommendations based on their historical check-ins made around the active time. In the graph-based preference propagation approach, we assign larger weights for the check-ins made by users in time slots close to the target time, and inject preference to the target user. The preference is then propagated to candidate POIs by several paths, and the POI with largest preference is recommended to the target user to visit at the target

time. For the sake of efficiency, a breadth-first search approach is proposed to speed up the propagation process.

Besides temporal influence, spatial influence is also an important consideration in analyzing human mobility [YYLL11]. Intuitively, users tend to visit nearby POIs, and thus the POIs visited by users often form spatial clusters. Hence, the spatial behaviors of users can be utilized to enhance POI recommendations.

To exploiting spatial influence, we assume the willingness of a user moving from a POI to another POI is influenced by the distance between the two POIs. For the user-based CF approach, we derive that the probability that a user will visit a candidate POI is influenced by both the popularity of the candidate POI and the conditional probability of visiting the historical check-in POIs of the user given the candidate one. We further extend the model to accommodate the temporal factor. For the graph-based propagation approach, the spatial influence is used to estimate the relativeness between POIs: if two POIs are close to each other, a user is more likely to visit a POI from the other one. The spatial influence is modeled as the weights of edges between POIs.

Group POI Recommendation

Making accurate recommendations for groups is not an easy task, because a group consists of multiple users who may have different preferences. How to make a trade-off among their preferences to recommend POIs is challenging. Furthermore, groups are often ad hoc, and the number of history records of a group may be very limited. For example, in a conference, some attendants may form a group to find a restaurant for dinner, but this group is new, and has no historical record. The cold-start problem caused by ad hoc groups makes group recommendations even more difficult.

To make accurate POI recommendations for groups, we propose a Latent Dirichlet Allocation (LDA) [BNJ03] based generative model, named COnsensus Model (COM), to make group POI recommendations. COM is novel because it is built based on the following three considerations that have not been exploited by previous work:

- (i) Each group is relevant to several topics, *e.g.*, a picnic group is relevant to hiking and dining topics. The POI selection of a group is influenced by both these relevant topics and group members' personal considerations of content factors, *e.g.*, the geographical distance to POIs.
- (ii) Users in a group may behave differently as group members from that as individuals, *e.g.*, a movie fan may visit cinemas when he is alone, but will go to a hill in a picnic group.
- (iii) Different users have different influences in making decisions in a group, and the influence degree of a user in a group is topic-dependent: a movie fan is probably influential in making decisions for a movie watching group, but is less influential in a picnic group, because the picnic group is less relevant to the movie topic.

Based on the three considerations, we model the generative process of a group as follows: each group has a multinomial distribution over latent topics, and these topics attract a set of users to join. The POI selection of a user is influenced by both the group topic that attracted her, and her personal considerations of content factors (Consideration i). Note that it is the topic of the group, instead of the users, that account for her item selection (Consideration ii). The final decision of a group is made by aggregating the selections of all users in the group: if a user is an expert in the relevant topics of the group, her selections will have a larger weight (Consideration iii). Based on the generative model, we propose a recommendation method to suggest items for a target group.

Requirement-aware POI Recommendation Previous studies reported that a user tends to visit nearby places, and a user might be at different places at different time in a day (e.g., at office around 4:00 PM, and at home around 11:00 PM). Thus, when recommending POIs for a user, it is important to predict the current location of the user based on his or her mobility pattern. In addition, a user might be interested in different topics at different places. Based on the given requirement (word) and time, we can infer the current location of the target user, and recommend nearby POIs that match the given requirement. Thus, in order to return accurate requirement-aware results, we need a model that can model the spatial, temporal and semantic information of each individual. In fact, these factors jointly reveal the mobility behavior of a user. It is however challenging to develop such a model, because the interdependencies among the four factors (namely, spatial, temporal, semantic and user) and the role played by each are unclear. What's more, the parameter estimation for the model would be very complex. To model user mobility behavior from the spatial, temporal, and semantic aspects, we take the following intuitions into account:

- (i) An individual's mobility usually centers at several personal geographical regions, *e.g.*, home region and work region [CML11] and users tend to visit the places within these regions. In addition, the number of personal regions is user specific, *e.g.*, some users may have additional regions for shopping and weekend activities.
- (ii) The probability that a user stays at a given region is affected by the day of the week, *e.g.*, users are more likely to stay at the work region on weekdays than weekends. Moreover, given a region, users may have different temporal patterns on different days (weekdays or weekends), *e.g.*, a user may visit her shopping region at afternoon on weekends, but at evening on weekdays.
- (iii) Users engage in different activities at different places, and the topics of a user at a place are influenced by both the user's personal topic preference and the region where the user stays. For example, a student who is interested in topics like reading and shopping will concentrate on the shopping topic rather than reading topic when she is at Times Square.

- (iv) When choosing a location to visit, a user will consider both her personal topic preference and the geographic coordinates, *i.e.*, whether the location matches her topic preference, and whether the location is within her current region of stay.
- (v) Different regions and topics lead to different word variations. Thus, the words used in a tweet posted by a user at a location are influenced by the user’s current region and her topic preference, which in turn reflect the user’s activity.

We develop a probabilistic Latent Semantic Analysis (pLSA) [Hof99] based model to model the above process in a seamless manner. Specifically, each geographical region has a Gaussian distribution over geographic coordinates, and a multinomial distribution over words, and each topic has multinomial distributions over POIs and words. We model the generative process of a piece of geo-tagged UGC as follows: a user first selects a personal geographical region, and then selects a topic that is relevant to both the region and her topic preference. Based on the region and topic, the user selects a POI and a set of words. We also propose a non-parametric Bayesian model based on hierarchical Dirichlet process (HDP) [TJBB06], which does not require any input parameters but achieves better performance.

1.5 Research Contributions

- We define the time-aware POI recommendation problem, which aims at recommending time-specific POIs for a user. We analyze the temporal influence and spatial influence from historical check-in data, and develop two POI recommendation algorithms that exploit the two kinds of influences. The proposed methods are evaluated on two real world LBSN datasets collected from Foursquare and Gowalla, respectively. The experimental results show that our methods achieve superior recommendation accuracy.
- We define the group POI recommendation problem, which aims at recommending POIs for a group of users. To solve this problem, we propose a generative model COM for modeling the process of item selection of a group, which considers members’ topic-dependent influences and members’ group behaviors. Based on COM, we develop a recommendation method to make group recommendations, which is able to exploit both users’ selection history and users’ personal considerations of content factors. We evaluate the effectiveness of the proposed method by extensive experiments on two datasets for event POI recommendation. The experimental results show that our proposed method outperforms five baselines significantly by various evaluation metrics.

- We define the requirement-aware POI recommendation problem, which aims at recommending POIs for users based on their specific needs. When the target time is available, the recommendation results could be also time-aware. To utilize the time and requirements (formulated as a set of words), we devise two probabilistic models to simulate user mobility behavior from geographic, temporal and activity aspects in a unified way. In addition to POI recommendation, the proposed models are also able to discover geographical-temporal topics for individual users. Experimental results on two real-world datasets demonstrate that our models outperform the state-of-the-art methods significantly for various applications including time-aware and requirement-aware POI recommendation.

1.6 Dissertation Organization

In Chapter 2, we review previous studies related to our research problems. In Chapter 3, we present our approaches to time-aware POI recommendation. Chapter 4 introduces our proposed framework of group recommendations. In Chapter 5, we present the proposed models about requirement-aware POI recommendation. Finally, Chapter 6 concludes this dissertation and points out several promising directions for future work.

Chapter 2

Literature Review

In this chapter, we first review three types of general recommendation techniques, namely, collaborative filtering, content-based recommendation, and hybrid recommendation. Then, we focus on the studies of POI recommendation, and survey its two variations, namely, time-aware recommendation and group recommendation. After that, we review the studies on user mobility modeling, which is closely related to POI recommendation. In the end, we review the studies on topic models, which are important techniques for recommender systems.

2.1 Recommender Systems

Recommender Systems aim at suggesting items for users based on their preferences, and they have been widely deployed to assist users to select items in various fields, such as movies (Netflix), products (Amazon), restaurants (Yelp), etc. Recommendation techniques can be divided into three categories, namely, collaborative filtering (CF), content-based approaches, and hybrid approaches. In this section, we first review them respectively, and then focus on three specific recommendation tasks, namely, POI recommendation, time-aware recommendation, and group recommendation.

2.1.1 Collaborative Filtering

CF recommends items preferred by users who have similar preferences to the target user's based on past ratings, encoded as the user-item matrix (U-I matrix). Due to its effectiveness and flexibility, CF has become the most successful and widely used recommendation technique [SLH14]. Researchers have devised a number of CF algorithms, which are categorized as memory-based CF and model-based CF.

2.1.1.1 Memory-based Collaborative Filtering

Memory-based CF approaches make recommendations based on user-user or item-item similarity derived from U-I matrix. User-based CF and item-based CF are two typical memory-based approaches.

Let U and I denote the user set and the item set, respectively, and let I_u be the items rated by user u . Given a target user $u \in U$, user-based CF first calculates the similarity $w_{u,v}$ between user u and every other user v based on their previous ratings for items, and then generates a prediction on an item $i \in I - I_u$ by a weighted combination of the other users' ratings on it. The recommendation score is formulated as follows:

$$\hat{r}_{u,i} = \frac{\sum_{v \in U} w_{u,v} \cdot r_{v,i}}{\sum_{v \in U} w_{u,v}}, \quad (2.1)$$

where $r_{v,i}$ is the rating of user v on item i , and $w_{u,v}$ is the similarity between user u and user v , which can be computed with various measures. The cosine similarity and the Pearson correlations are most widely adopted [Sin01]. In both methods, each user is represented by a rating vector over items. The former one calculates the cosine similarity between two users' rating vectors as follows:

$$w_{u,v} = \frac{\sum_{i \in I} r_{u,i} \cdot r_{v,i}}{\sqrt{\sum_{i \in I} r_{u,i}^2} \sqrt{\sum_{i \in I} r_{v,i}^2}}. \quad (2.2)$$

The latter one calculates the Pearson correlation as follows:

$$w_{u,v} = \frac{\sum_{i \in I} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{v,i} - \bar{r}_v)^2}}, \quad (2.3)$$

where \bar{r}_u is the average rating of the items rated by user u .

There are several other similarity measures, such as Proximity, Impact and Popularity measure [Ahn08], singularities measure [BOH12], Dirichlet-based measure [GZY13], etc. Since the calculation of similarity is not the focus of the dissertation, we do not introduce them in depth.

Different from user-based CF, item-based CF works by returning items that are similar to other items the target user has rated [SKKR01]. In formal, user u 's expected rating on item $i \in I - I_u$ is computed by:

$$\hat{r}_{u,i} = \frac{\sum_{j \in I_u} w_{i,j} r_{u,j}}{\sum_{j \in I_u} w_{i,j}}, \quad (2.4)$$

where $w_{i,j}$ is the similarity between item i and j , which can be obtained by the cosine similarity or adjusted cosine similarity between the item vectors containing users who have rated the items before [SKKR01].

2.1.1.2 Model-based Collaborative Filtering

Model-based CF builds models using data mining or machine learning techniques on U-I matrix to recognize complex patterns for making recommendations [SK09]. A number of model-based CF approaches have been proposed. In this section, we review them based on the techniques they employed.

- **Bayesian-based approaches** [MP00, MP02] transforms recommendation into a classification task, where each rating value corresponds to a class. They take users' ratings as features, and uses Naive Bayes strategy to predict the class with the highest posterior probability as the predicted rating.
- **Clustering-based approaches** [MOC99, SKKR00, CHW01, SKKR02, XLY⁺05] group users or items by clustering methods (*e.g.*, k-means, DBSCAN, BIRCH, *etc.*) for further recommendation processing. Typically, the clustering-based approaches first find the most relevant item cluster to the target user, and then generate recommendations within this cluster rather than the whole item collection. The clustering-based methods always achieve better efficiency, but their recommendation accuracy is often low.
- **Regression-based approaches** [PHLG00, VO05, LM07] train regression models based on U-I matrix to approximate the numerical ratings. Specifically, suppose there are n items and k latent factors, the linear factor regression model for user preference is defined as:

$$R = \Pi X + N, \quad (2.5)$$

where R is the U-I rating matrix, $X = (X_1, X_2, \dots, X_k)$ is users' preferences over items related to k factors, Π is a $n \times k$ matrix, and N is a noise matrix.

- **Matrix factorization (MF) based approaches** [SM07, KBV09] model users and items by k latent features, and decompose U-I matrix R into $|U| \times k$ user matrix P and $|I| \times k$ item matrix Q , which share the same low dimension feature space. To avoid the overfitting problem, regularization factors are introduced when estimating P and Q . The predicted rating matrix \tilde{R} is approximated by the inner product of the two matrices, *i.e.*, $\tilde{R} = P \times Q^T$. MF technique has been reported to achieve good scalability and recommendation accuracy.
- **Graphical model-based approaches** Probabilistic graphical model based approaches, such as Bayesian clustering [BHK98], aspect model [HP99], flexible mixture model, joint mixture model [SJ03] and decoupled model [JSZC03] have been shown to be effective in recommendation [Zha]. Among them, the aspect model, or topic model based approaches attract much research interest in recent years

[BNJ03, Hof04, KFN09]. These approaches make recommendations based on “latent topics” extracted from U-I matrix by topic models, such as probabilistic Latent Semantic Analysis (pLSA) [Hof99] and latent Dirichlet allocation (LDA) [BNJ03] (to be detailed in Section 2.3.1). Specifically, the topic model-based approaches assume each user u has a multinomial preference distribution $\{P(z_k|u)\}_{k=1}^K$ over K latent topics, such as “dining” and “shopping”, where each latent topic z has a multinomial distribution $\{P(i_j|z)\}_{j=1}^{|I|}$ over items, representing the relevance of item i_j to topic z . The item selection of a user is influenced by the latent topics, and the relevance of an item i to a user u is calculated by a mixture model as follows:

$$\hat{r}_{u,i} = \sum_{k=1}^K P(z_k|u)P(i|z_k). \quad (2.6)$$

2.1.2 Content-based Recommendation

Content-based recommender systems exploit content information and make recommendations by 3 steps [LDGS11]. In the first step, features are extracted to represent the content of the items by some information retrieval techniques, such as keyword-based matching and semantic analysis. The keyword-based approaches [MR00, wABG⁺07, CS08] usually build vector space models (VSMs) for items based on their descriptions, categories, tags, etc, while the semantic analysis-based approaches leverage external knowledge bases, such as ontologies [PLEV04, AZSD07, DLS07] or encyclopedias [GM07, MC07, SLBdG09] to profile items on semantic level. In the second step, the recommender systems profile user preferences through machine learning techniques, such as naive Bayes classifiers and decision trees. In the last step, the systems match the target user’s profile against candidate items by classifying the candidate items into categories “likes” and “dislikes”. There are also several adaptive learning based approaches that take feedback from users as the content information for recommendation [ZXC03].

The content-based approaches suffer from several weaknesses: (1) enough information is needed to train the classifier for each target user; (2) the performance of content-based approaches is limited by the features; (3) they cannot recommend unexpected items to users.

2.1.3 Hybrid Recommendation

Hybrid recommendation methods combine different recommendation techniques to overcome their individual limitations. Some hybrid approaches incorporate content features into CF for recommendation [AEK00, MMN02], where the content features could be text [BFZ14], user profile [LHZC10], trusted neighbors [GZT12], etc. Some approaches

combine CF and content-based approaches by different weights [MCG⁺99, PB97] or by switches [Bur02]. It has been reported that the hybrid approaches can outperform pure CF and content-based methods in terms of recommendation accuracy. In addition, there are also hybrid approaches combining memory-based CF and model-based CF [PHLG00, YST⁺04].

2.2 POI Recommendation

POI recommendation aims at recommending POIs that the target user has not visited before. In this section, we first review the context-unaware approaches to POI recommendation, and then focus on two context aware tasks, namely, time-aware POI recommendation and group recommendation.

2.2.1 Context-unaware POI Recommendation

In recent years, a number of methods have been proposed for POI recommendation. Based on the data sources used, existing proposals can be divided into three categories, namely, user profile based, user trajectory based and location history based [BZWM13]. The **user profiles based approaches** [PHC07, RDP⁺09, KIGI09] falls in the category of content-based recommender systems, and they generate recommendations based on the matching degree between user profiles (*e.g.*, age, gender, income, etc.) and the metadata of the POIs (*e.g.*, description, tags, etc.). The **trajectories based approaches** [TS06, ZZXM09a, ZZXM09b, CCJ10, LX11, LLL11, YLWT11, MRJ12, MRM12, MZRJ13, YLT13] take users' GPS trajectories as input to recommend or predict locations for users. Different from the proposals that take user profiles and trajectories as input, the **location history based approaches** [HNV06, YLL11, CYKL12, LLAM13, CYLK13, LFX13, KIH⁺13, YSC⁺13, HE13, NSLM12, BZM12, WTM13, FYL13] generate recommendations based on users visited POIs extracted from LBSNs (*e.g.*, Foursquare, Gowalla, Facebook Places) or review web services (*e.g.*, Yelp and Dianping). In this dissertation, we focus on recommending based on user check-ins, because they are publicly available online, and there is a great number of history records of more users. We do not investigate the profile based approaches, since the user profiles and location metadata may not be available online, and the performance of a method is greatly influenced by the features used. The trajectory based approaches are also out of our research scope because of three reasons: 1) GPS trajectories do not contain semantically meaningful POI information but only geographical coordinates; 2) the number of users with GPS trajectories is too small to build practical recommender systems; 3) few GPS trajectories are publicly available. There are privacy concerns when using the trajectory data. Thus, in the rest of this section, we will review existing proposals that are built on users' check-in records.

How to make use of geographic information to improve recommendation accuracy is the focus of the studies on POI recommendations. We categorize these studies based on the way of utilizing the geographic information.

- **Building independent models to leverage geographic information.** These approaches first build models to leverage geographic information, and then combine them with memory-based CF.

Ye *et al.* [YYLL11] consider the social influence under the framework of user-based CF, and models the spatial influence by a Bayesian CF model. Specifically, this work employs user-based CF to compute the recommendation score of a candidate POI for a user. To exploit the social influence, the work makes use of the users friends for recommendation rather than all the users. To explore the geographical influence, this work assumes that the probability that a user visits two POIs l_i, l_j is determined by their distance $d(l_i, l_j)$:

$$P(d(l_i, l_j)) = a \times d(l_i, l_j)^b, \quad (2.7)$$

where a and b are parameters of the power-law function that can be learned by regression models. Then, the probability of a user visiting a set of POIs is defined as the product of probabilities of visiting all the pairwise POIs in the set, and the probability that a user u checks in at a new POI l_j is estimated by the product of the probabilities of visiting all the pairwise POIs, each pair consisting of the new POI l_j and each previously visited POI $l_y \in L_u$:

$$P(l_j|L_u) = \prod_{l_y \in L_u} P(d(l_j, l_y)) \quad (2.8)$$

In this paper, the authors report that geographical influence has a significant impact on the accuracy of POI recommendations, but the social friend links themselves make little contribution.

Cheng *et al.* [CYKL12] employ Gaussian mixture model (GMM) trained based on the geographic information to estimate the probability that a user visits a POI in terms of the traveling distance. Then, the authors utilize Probabilistic Factor Model (PFM) [CKPC09] to model the check-in frequency, which places Beta distribution as priors on the user and POI latent matrices, and places Poisson distribution on the frequency. Finally, the scores calculated by GMM and PFM are multiplied as the final recommendation score for a candidate POI. However, the proposed method is compared with some naive MF-based model without considering the geographical influence, which makes it hard to tell its effectiveness.

Liu *et al.* [LLAM13] assume users have preference transition over POI categories, and the next check-in POIs of a user depend on her current POIs. They first cluster users based on their check-in frequency in different categories, and then predict the category of next visiting POIs by applying MF on the corresponding clusters. After getting the most likely categories, another MF is introduced to predict the venues. To utilize geographic information, they propose a power-law function to estimate the probability that the target user travels to a venue current position. The category-based and distance-based results are combined to generate the recommendations. However, this method requires a number of check-ins to cluster users. In addition, when the target time is far from that of the target user's last check-in, the model will be incapable to infer the category of the next POI.

- **Utilizing Geographic information as content features.** These algorithms incorporate the geographic information as content features to recommend POIs. We review them based on their CF techniques.

- **Memory-based CF.** Ye *et al.* [YYL10] observe that nearby friends tend to share more commonly visited POIs. They measure the similarity between friends based on their geographical distance, and employ friend-based CF to estimate the relevance of each candidate POI to the target user. Levandoski *et al.* [LSEM12] first calculate the recommendation score for each candidate POI by item-based CF, and then subtract the score by a travel penalty, which is proportional to the distance between the target user and the POI. However, these two papers concentrate on recommendation efficiency instead of accuracy.
- **Matrix (or tensor) Factorization.** Cheng *et al.* [CYLK13] focus on recommending POIs for a target user to visit at the successive time stamp given her current POI. Based on the observations that that two successive check-in venues are always correlated and located within a short distance, employ Factorized Personalized Markov Chain (FPMC) [RFST10] to factorize the User-Current venue-Next venue tensor to get the probability that a user will check in a candidate venue. To exploit the geographic information, they incorporate the localized region constraint into FPMC, *i.e.*, the authors divide the earth into grids, and only consider the venues in adjacent grids to the current venue as the candidate venues. This model suffers from with [LLAM13], *i.e.*, it can only recommend POIs for the target users to visit in a short time after their last check-in.

Liu *et al.* [LFYX13, LX13] assume each user has a multinomial distribution over a set of regions, where each region is modeled by a Gaussian distribution

over geographical coordinates. This probabilistic model is combined with MF to recommend venues, in which the preference of user for a venue is determined by the user’s and venue’s latent factors, observable properties, venue popularity and the spatial influence.

- **Topic Model.** Kurashima *et al.* [KIH⁺13] propose the “Geo Topic Model” to recommend POIs. They assume each user has a distribution over topics, where each topic has a distribution over POIs. When selecting a POI, the user will consider both the matching degree of each candidate POI to her preferred topics, and the distance to the candidate POI. The influence of distance is measured by an exponential function parameterized by the Euclidean distance between the candidate and each of her visited POI. The model can also recommend POIs that suit user’s current position, where the distance is calculated based on the candidate POI and the and the current coordinate specified by the target user. However, in this model, all users share the same sensitivity to the traveling distance, which should be diverse across users in reality different users prefer different transportation methods.

Yin *et al.* [YSC⁺13] assume that each user and each city has a distribution over topics. When selecting a POI in a city, the user will consider both her own topic preference and the topic distribution of the city, and draw a topic from the two topic distributions. The selected POI should match the topic sampled by the user. Based on these assumptions, they propose an LDA-based model to recommend POIs for a given user at a given city. However, this model does not exploit the POI-specific geographic information.

Hu *et al.* [HE13] employ topic model to recommend POIs. They assume each users has distributions over regions and topics, where the regions are model by Gaussian distributions over coordinates. The process of POI selection is similar to that of [KIH⁺13], except that the influence of distance is measured by the region-specific Gaussian distribution. This model works well on city-level dataset in which the POIs are often distributed evenly. When the scale of the dataset becomes larger, some Gaussian distributions will converge at few POIs, which will always dominate the recommendation results. In their following up work [HJE13], they propose another topic model based approach that exploits the spatio-temporal aspects of user check-ins for time-aware POI recommendation. In the model, each user has distributions over topics and regions, and each time slot has distributions over topics and POIs. The regions, topics and time together influence users’ check-in activities. However, the previous problem of Gaussian distribution still exists in this model.

- **Exploiting geographic information as pre-filtering or post-filtering.** These approaches filter out distant POIs before or after making recommendation.

Noulas *et al.* [NSLM12] divide Foursquare and Gowalla check-ins into groups based on cities, and employ personalized random walk with restart to recommend POIs. Specifically, they treat POIs and users as nodes in a graph: a user is linked to a POI if she has visited it, and a user is linked to another user if the pair are friends. The experimental results show that the random walk based approach outperforms MF, user-based CF and item-based CF for POI recommendation.

Bao *et al.* [BZM12] assume POI category is the key factor that influences user check-in activities. They first discover the experts of each category in each city by applying HITS (Hypertext Induced Topic Search) on each category-based user-POI matrix. Then, they extract the preference of each user by projecting the categories of her visited POIs onto a pre-defined category hierarchy, and apply TF-IDF scheme to each node of the hierarchy to calculate the user's preference to that category. Given a target user, the model first finds experts that are most similar to the target user, and then employs user-based CF to recommend venues. The venues that are contained specified region are considered as candidates. However, the category information is not always available, and when the number of check-ins of a user is limited, it is hard to estimate her category preference, which may bring down the recommendation accuracy.

Wang *et al.* [WTM13] propose algorithms under the framework of personalized Pagerank on a graph containing user nodes and venue nodes. A user is linked to a venue she visited it before, and two user nodes are connected with each other if they are friends or visited the same venues. Given a target user, the venues that are far away from her visited venues are filtered out from the candidate venue set.

Ference *et al.* [FYL13] employ user-based CF to recommend POIs for the target user given her current position, where only the POIs that are close to the current positions are considered as the candidates.

We can find that most of these methods do not exploit the context information for POI recommendation. However, in real life, users' preferences to POIs are often influenced by context, such as time, companions, etc (see Section 1.2). Thus, in this dissertation, we focus on three context-aware POI recommendation tasks, namely, time-aware POI recommendation, group POI recommendation, and requirement-aware POI recommendation. Among the three tasks, the requirement-aware POI recommendation has not been investigated before. Thus, in the following sections, we review the existing studies on time-aware POI recommendation and group POI recommendation.

2.2.2 Time-aware POI Recommendation

There are a number of recommender systems that take time as additional information to improve recommendation accuracy [CDC14]. We categorize time-aware recommender systems as memory-based approaches and model-based approaches, and review them respectively.

- **Memory-based approaches**

Ding *et al.* [DL05] propose an item-based CF method, which employs an exponential function to decay the weights historical ratings made long time before the target time. The rationale is that with a larger time gap the ratings would be less useful for recommendation. Specifically, given a target user u and target time t , the time-aware recommendation score for candidate item i is calculated as follows:

$$\hat{r}_{u,i} = \frac{\sum_{j \in I_u} w_{i,j} r_{u,j} \cdot f(t_j, t)}{\sum_{j \in I_u} w_{i,j} \cdot f(t_j, t)}, \quad (2.9)$$

where $w_{i,j}$ is the similarity between item i and j , I_u is the set of items rated by u , and $f(t_j, t)$ is the time function that decays the weight of rating $r_{u,j}$ based on the time gap between t_j and t , and it is defined as follows:

$$f(t_j, t) = e^{-\frac{1}{H} \cdot |t_j - t|}, \quad (2.10)$$

where H is a parameter to control the decay rate that needs to be tuned manually.

Based on similar assumption that recent ratings are more valuable to make time-aware recommendations, Campos *et al.* [CBDC10] developed a user-based CF method to incorporate the time bias. Specifically, given a target user, the proposed method first calculates the most similar users based on their previous ratings and other content information. Then, only the recent ratings of these similar users are exploited to estimate the preference of the target user. However, the performance of this method is greatly influenced by the time interval: if it is too small, there could be not enough recent ratings to make recommendations.

Baltrunas *et al.* [BA09] divide time into several categories (*e.g.*, morning, weekend, summer, etc.), and build micro-profiles for each user and each time category based on the ratings made within the corresponding time. When making recommendations, multiple micro-profiles are aggregated instead of a single profile, since multiple micro-profiles can model users more precisely.

- **Model-based approaches**

Most of the model-based approaches are built under the framework of matrix or tensor factorization. Koren *et al.* [Kor09] assume that both the user bias and

item bias change over time, *e.g.*, a movie is popular when it is just released, but may become unpopular afterwards. In addition, users' preferences to some latent factors, such as movie genres and actors, may also change over time. Based on these assumption, they propose an MF-based method, which models the rating of user u to item i at time t as follows:

$$\hat{r}_{u,i,t} = \mu + b_u(t) + b_i(t) + p_u^T(t)q_i, \quad (2.11)$$

where μ , $b_u(t)$, $b_i(t)$, $p_u^T(t)$ and q_i are the overall mean rating, time-specific user and item biases, user time-specific feature vector and item feature vector, respectively.

Xiong *et al.* [XCH⁺10] incorporate time dimension into the U-I matrix to get a three dimensional tensor, and propose a factorization method to generate time-aware recommendations. The rating of u to item i at time t is modeled as follows:

$$\hat{r}_{u,i,t} = \sum_k P_{k,u} Q_{k,i} W_{k,t}, \quad (2.12)$$

where P , Q and W are the feature vectors of users, items and time, respectively.

Koenigstein *et al.* [KDK11] propose a time-aware approach to music recommendation, where the time is modeled as consecutive ratings (or sessions). Based on the observation that a user tends to rate songs very similarly in a session, they propose an MF-based model, which considers the bias of ratings in each session and the taxonomy of music track.

Liu *et al.* [LHZ13] propose an MF-based collaborative ranking method to recommend movies. In stead of modeling the exact rating values as MF does, the ranking methods (*e.g.*, Bayesian Personalized Ranking (BPR) [RFGST12]) aim to rank the items based on users' preferences. To exploit the temporal preference, they divide time into slots, and apply the ranking model to ratings in each slot to get users' time-dependent preference, and then make time-aware movie recommendations.

Although the factorization models are effective on explicit feedbacks (*e.g.*, ratings) data such as Netflix, they acheive little improvement on implicit feedbacks (*e.g.*, binary) data [HKV08].

Xiang *et al.* [XYZ⁺10] focus on implicit feedback data. Based on the assumption that users' purchase behaviors are influenced by their long-term and short-term preferences, they divide time into bins, and propose Session-based Temporal Graph (STG) to model users' temporal preference. The graph contains three kinds of nodes: user node, item node, and session node, where a session node represents the activity of a user in a time slot. If a user purchased an item, edges will be built between the corresponding item node and user node, and between the item

node and session node. To make time-aware recommendations, they propose the algorithm Multi Source Injected Preference Fusion (MS-IPF), which first injects preference into the target user node and session node, and propagates the preference to item nodes by various paths. The items receiving most preference are returned as recommendations.

However, none of these memory-based or model-based approaches is designed for POI recommendation. In real world, users' mobility is strongly correlated with time, *e.g.*, a user may stay at office in the daytime, and go back home in the evening. Based on the time, we can infer the position of the target user, and recommend nearby POIs for him or her. Although these general time-aware recommender systems can be applied to POI recommendation directly, they only consider the correlations between time and items (POIs), but cannot capture the correlations between time and geo-locations. As a result, their accuracy for POI recommendation is limited. In addition, some time-aware recommender systems [XCH⁺10, KDK11, LHZ13] are designed for explicit feedback data (*e.g.*, ratings) instead of implicit data (*e.g.*, check-ins), and they are not effective for POI recommendation [HKV08].

In the experiments, we will compare with two approaches which share the same settings with our problem, namely, CF with time decay function [DL05], MS-IPF [XYZ⁺10]. Although MS-IPF is similar with our preference propagation based model (to be detailed in Section 3.4), our model is different from the it in at least 4 aspects:

- The item nodes in STG are connected to both user nodes and session nodes, while in our model the POI nodes are connected to session nodes and POI nodes. The edges between POI nodes in our model enable us to incorporate geographical information for recommendation.
- The session nodes in our model are bridged by user nodes while in STG they are not. The fundamental difference lies in the different intuitions in building the graphs. In STG, a user's sessions are independent. If the target user has never checked-in at the target time, the method will fail to make recommendations for the target time. In our model, check-ins from all sessions of a user are considered with different weights.
- The temporal preference considered in STG is either long-term or short-term, while in our model the temporal preference is periodic (*i.e.*, each user has 24 session nodes).
- STG is a bipartite graph, while our model is not. Thus the recommendation method MS-IPF for STG cannot be applied to our model.

Time-aware POI recommendation is defined as recommending POIs for a given user at a specified time in a day. As an important context information, time has been considered in several studies on POI recommendation. Some researchers develop models that can recommend POIs for users at the time close to the users' last check-ins [LLAM13, CYLK13]. However, most users only made a few check-ins. If the time gaps to the last check-ins are large, these methods cannot provide accurate results. Thus, different from these papers, we aim to design a model that can recommend POIs for any target time. The most relevant work that considered temporal influence is the location recommendation framework with temporal effects (LRT) proposed by Gao *et al.* [GTHL13]. LRT is an MF-based method that factorizes the user-time-POI check-in matrix to get users' temporal preference matrix U_t in time t and location characteristics matrix L . Then, the POI preference of the target user is constructed and aggregated based on L and U_t in all time slots. LRT does not exploit geographical influence for recommendation, so its recommendation accuracy is limited. In addition, LRT is not used to recommend POIs for a target time. In the experiments, we tailor it for time-aware POI recommendation by removing the aggregation step, *i.e.*, the POI preference of the target user in target time is estimated based on the POI property matrix L and time-aware user property matrix at the target time U_t only.

2.2.3 Group Recommendation

Group recommendation is defined as recommending items for a group of users, and several methods have been proposed for various domains, such as web/news pages [PDCCA05], tourism [MSC⁺06], restaurants [McC02], music [CBH02], TV programs [YZHG06], and movies [OCKR01]. Group recommendation methods in earlier studies fall into two categories [AYRC⁺09]: the *preference aggregation* approaches first aggregate the profiles of the group members into one profile, and makes recommendations based on the aggregated profile [MA98, YZHG06]. The *score aggregation* approaches, in contrast, first generate recommendations for each group member respectively, and then aggregate their recommendation results for the group [OCKR01, McC02, CBH02, PDCCA05, BMR10].

Compared with preference aggregation, score aggregation typically achieves better flexibility [OCKR01, JS07, AYRC⁺09], and thus receives more research attention. The score aggregation approaches usually employ either average (AVG) or least misery (LM) strategy to aggregate the recommendations of individuals. The AVG strategy averages the recommendation scores of all group members as the final score, aiming to maximize the overall satisfactions of a group [MA98, YZHG06]; the LM strategy takes the smallest recommendation score of group members as the final score, and tries to make everyone happy [BMR10]. For LM, the recommendation score of an item is largely influenced by the user who dislikes it most, even if all the others like it very much. For AVG,

an item’s relevance to different users might be diverse, and the recommendation results might be unfair to some users. Baltrunas *et al.* [BMR10] compare different aggregation approaches, and find that there is no clear winner, and the effectiveness of an approach depends on the group size and inner group similarity. Amer-Yahia *et al.* [AYRC⁺09] go one step further, and argue that an item is a good recommendation for a group if the group members have small disagreements but large relevance on the item. The relevance is calculated by AVG or LM strategy, and the disagreement can be either the average difference of recommendation scores of pair-wise group members, or the variance of members recommendation scores. Note that this paper mainly focuses on recommendation efficiency.

Recently, several model-based approaches have been proposed. Seko *et al.* [SYM_yM11] develop a content-based group recommendation method based on the assumption that the choice made by a group is influenced by item genres. However, this approach can be only applied to pre-defined groups, *e.g.*, couples, while in real-life, groups are often ad hoc. Carvalho *et al.* [CM13] introduce game theory into group recommendation by treating a group event as a noncooperative game among members, and transform the recommendation problem into finding the Nash equilibrium. However, this method cannot suggest a specific item, since the equilibrium contains a set of items.

Several studies employ topic model to make group recommendations. Ye *et al.* [YLL12] believe that when selecting items, a group member will follow her friends’ opinions. They propose a probabilistic generative model Social Influence Selection (SIS) to model the process of item selection. In SIS, each user has distributions over topics and friends. When selecting an item, the user u first draws a friend f from her friends set, and then draws a topic z based on the topic preference of f . Finally, one item i and its descriptions w are drawn from the item and words distributions of z . The graphical model is shown in Figure 2.1. To make recommendations for groups, the authors enhances SIS and propose the model Social Influence Group (SIG). Specifically, SIG first decomposes the target group g into a set of 2-member groups based on friendship, where one influences the other. Then, it aggregates the preferences of the 2-member groups as the final recommendation score for the group.

Liu *et al.* [LTYL12] propose the Personal Impact Topic model (PIT) to recommend items for a group of users based on the assumption that the influential users will become the representatives of all groups to make item selections. Specifically, in PIT each user u_i has an impact score γ_i , proportional to which a group G_n samples a representative r_n . Then r_n selects a topic z_n based on her topic preference θ_n , and the topic z_n samples an item s_n based on its item distribution ϕ_n . The graphical model of PIT is shown in Figure 2.2.

Gorla *et al.* [GLRW13] employ information matching technique to recommend items for group. Information matching describes users and items with two set of binary features,

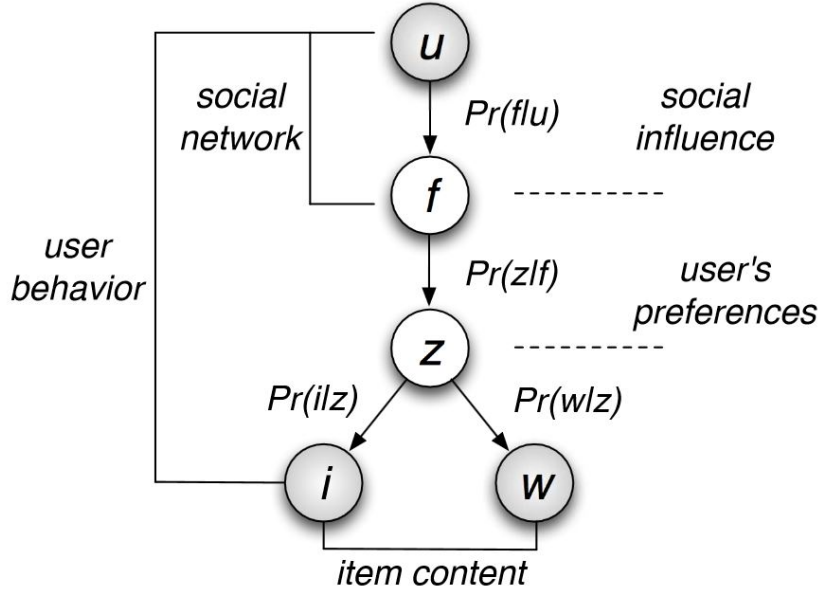


Figure 2.1: The SIS Model

namely, user features and item features, with different certainty. The user/item features have preference/appeal relations towards each other, and they are the sole results of the item/user relevance. To the problem of group recommendation, the recommendation score for an item depends on its relevance to each group member and its relevance to the group as a whole.

To the best of our knowledge, there is no existing studies that focus on group POI recommendation. Although check-in data of users are used in experiments of the papers [YLL12, LTYL12], these two methods do not exploit geographical information.

The settings of the approaches [YLL12, LTYL12, GLRW13] are similar to ours. Among them, the SIG model proposed in [YLL12] suffers from two weaknesses: 1) the strong assumption of pairwise influence in a group may not be true, especially when the group is large; 2) a group does not always consist of friends. The PIT model proposed in [LTYL12] assumes that each user has an impact score, and those with great scores are like to be representatives of groups to make item selections. However, we doubt the basic assumption of the model: the users with larger impact scores will always be the representatives, irrespective of the group topics and the influential users' expertise. In fact, the users' impact should be topic-dependent: a user may be influential in a group because of her expertise on the group's topics, but may not be in another group. The information matching based model proposed in [GLRW13] suffers from two shortcomings: first, its time complexity is $O(|U|^2|I|^2)$ for each target group ($|U|$ and $|I|$ are the sizes of user and item sets, respectively) is too high to generate results in real time; second, it cannot

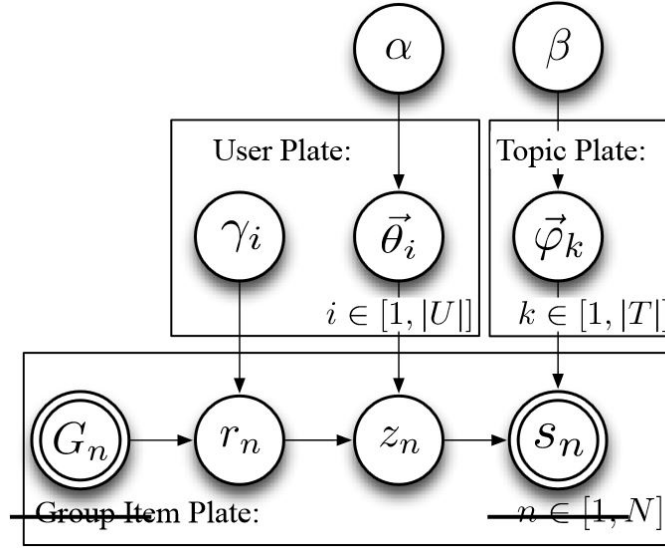


Figure 2.2: The PIT Model

provide explanations to the recommendations, because the model is build on latent dimensions. In contrast, the recommendations of our model are interpretable, *e.g.*, we can tell which user and which topics are the most influential in a group. In addition, none of these methods can incorporate content information like geographical coordinates of POIs for recommendation. Thus, we aim to design a new model to recommend POIs for a group of users, which can exploit the geographical coordinates of POIs as additional information to improve the accuracy.

2.3 Topic Models

In this section, we first introduce the topic models, and then review a specific application of topic models, namely, geographical topics modeling.

2.3.1 Introduction to Topic Models

Topic models are probabilistic models that simulate the generative process of a collection of documents. They assume there are a set of “topics” in a given corpus, *e.g.*, research papers in computer science may contains topics “information retrieval”, “database”, “computer graphics”, “system”, etc. Each topic has its own semantics, which is modeled by a multinomial distribution over the vocabulary V . Larger values in the distribution mean that the corresponding words are more relevant to the topic and thus are more likely to appear in the topic. Consider the topic “information retrieval”, the words “precision”,

“TF-IDF” may appear more frequently than the words “pixel” and “rendering”. Topic models generate the words of each document from a mixture model, where each mixture component is a topic.

There are two popular topic models, namely, probabilistic Latent Semantic Analysis (pLSA) [Hof99] and latent Dirichlet allocation (LDA) [BNJ03], both of which assume the number of topics K is fixed.

In pLSA, each topic z has a multinomial word distribution $\{P(w|z)\}_{w=1}^{|V|}$, where $|V|$ is the size of the vocabulary. Each document d has a multinomial distribution over topic space, *i.e.*, $\{P(z|d)\}_{z=1}^K$. For each word in document d , a topic z is first sampled based on d 's topic distribution, and then, a word w is sampled based on the word distribution of topic z . The generative process is as follows:

- For each document in the document collection
 - Select a document d with probability $P(d)$
 - For each word in d
 - * Pick z with probability $P(z|d)$
 - * Generate w with probability $P(w|z)$

The pLSA model suffers from two shortcomings [BNJ03]: (1) there is no natural way to discover the topic distribution for an unseen document; (2) the number of parameters grows linearly with the number of documents, which may lead to overfitting problem.

To overcome the two limitations, Blei *et al.* propose the LDA model, which assigns Dirichlet prior α and β for the topic distribution θ_d of document d , and word distribution ϕ_z of topic z , respectively. The generative process of LDA is as follows:

- For each topic z
 - Generate word distribution $\phi_z \sim \text{Dirichlet}(\beta)$
- For each document d
 - Generate topic distribution $\theta_d \sim \text{Dirichlet}(\alpha)$
 - For each word in d
 - * Draw $z \sim \text{Multinomial}(\theta_d)$
 - * Draw $w \sim \text{Multinomial}(\phi_z)$

The number of topics in pLSA and LDA is fixed, which needs to be set empirically. To overcome this problem, the hierarchical Dirichlet process (HDP) model is employed, which can automatically learn the number of topics from the data. Unlike LDA, the document-level topic distributions G_d are drawn from a global topic distribution G_0 by a Dirichlet process $G_d \sim \text{DP}(\alpha_0, G_0)$, where α_0 is the concentration parameter that controls the variance of G_d around G_0 . In addition, the global topic distribution G_0 is

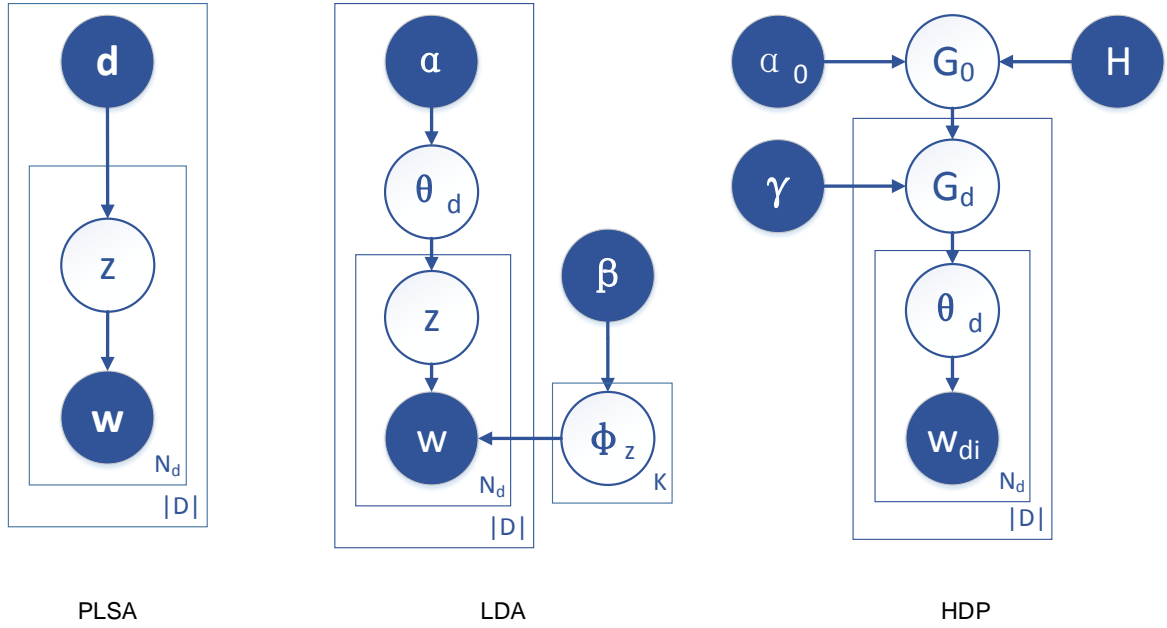


Figure 2.3: The Graphical Models of pLSA, LDA and HDP

drawn from another Dirichlet process with base distribution H on the continuous topic space, *i.e.*, $G_0 \sim DP(\gamma, H)$, where γ is another concentration parameter. Each word w_i in each document d is sampled based on the multinomial word distribution θ_z , where the topic z is sampled from the document-level topic distribution G_d . The generative process is as follows:

- Generate global topic distribution $G_0 \sim DP(\gamma, H)$
- For each document d
 - Generate document topic distribution $G_d \sim DP(\alpha_0, G_0)$
 - For each word in d
 - * Generate $\theta_{d_i} \sim G_d$
 - * Generate $w_{d_i} \sim \theta_{d_i}$

The graphical models of pLSA, LDA and HDP are shown in Figure 2.3.

Since topic models are able to incorporate heterogeneous information in one generative model, and the model is easy to follow, they have been widely used in recommender systems. Kim *et al.* [KPS13] devise a pLSA based approach to digg article recommendation, Zhu *et al.* [ZCX⁺14] propose an LDA-based model to make recommendations for mobile users based on context logs, Erosheva *et al.* [EFL04] employ LDA technique for scientific article recommendation, Liu *et al.* [LCX⁺14] develop a travel package recommender system under the framework of LDA.

2.3.2 Geographical Topics Modeling

Geographical Topic Modeling aims at discovering language variations across different geographical regions by topic models such as pLSA and LDA. How to represent locations is an essential part of these studies. Locations have two properties: the geo-locations represented by coordinates, and the functions (*e.g.*, a shop) represented by the topics. Based on the ways of representing locations, the existing studies can be divided into two categories:

First, some proposals [MLSZ06, WWXM07, HCW⁺10] represent locations by POI ids, and this enables these proposals to distinguish the functions between locations. Mei *et al.* [MLSZ06] model topics of documents from spatio-temporal aspects using pLSA. Specifically, they assume that each word of a documents are drawn based on three factors 1) document-specific topic; 2) the topic shared by all the documents with the same spatio-temporal context; 3) the background word distribution. Wang *et al.* [WWXM07] propose an LDA based model to learn the relationship between location and words. They assume that each word is associated with a location, and each topic has multinomial distributions over words and locations. When a word is sampled from a topic, its associated location is also sampled from that topic. Hao *et al.* [HCW⁺10] mine the location-representative topics from travelogues using an LDA-based model. Specifically, each travelogue is split into several segments, and the words in which are generated based on either local topics that characterize locations, or global topics that are shared by all locations. If a word is generated from a local topic, then a location will also be generated from the location distribution of that local topic. Although this modeling manner can differentiate POIs by topics, it fails to exploit the coordinate information, which is important to analyze the user mobility regions.

Second, other proposals [EOSX10, Siz10, YCH⁺11, BNS⁺12] represent locations as coordinates or grids, and they are capable of describing the mobility regions of users. Eisenstein *et al.* [EOSX10] propose regional variants of topics, which are used to generate the words of a geo-referenced document. They use bi-variant Gaussian distributions of regions to generate coordinates of locations. Sizov [Siz10] proposes GeoFolk model to manage geo-referenced documents. In addition to the word distribution, each topic in GeoFolk is also associated with two Gaussian distributions over latitude and longitude, respectively. In GeoFolk each geographic region represents a distinct topic/function. Hence, it fails to correlate the different regions with the same function; it would not be suitable to model a large area containing many topical regions since the topic model becomes computationally expensive as the number of topics is large. Yin *et al.* [YCH⁺11] propose a pLSA-based model to discover geographical topics. In the model, each region is characterized by a topic distribution, and represented by a bi-variant Gaussian distribution over coordinates. Bauer *et al.* propose an LDA-based spatio-temporal model [BNS⁺12],

where a city is divided into grids. Kling *et al.* [KKSS14] propose a multi-dirichlet process based geographical topic model, which first estimates geographical regions from the data, and then generates document content by Dirichlet process. This model uses Fisher distribution [Fis53] to model the regions, so that it can discover non-elliptic regions, *e.g.*, coast lines. We note that this modeling manner either neglects the functions of locations or assumes that nearby locations have the same functions, which are generally not true in reality. Thus, it is not a good choice to modeling locations.

2.4 Mobility Behaviors of Users

A number of studies have been proposed to model user mobility behaviors. Some of them consider the user, geographic information, and time, but overlook users' activities. Brockmann *et al.* [BHG06] investigate the trajectories of the bank note dispersals, and find that human mobility behavior can be approximated by a continuous-time random-walk model with long-tail distributions over waiting time between displacements. Gonzalez *et al.* [GHB08] study the mobility patterns of mobile phone users and find that users periodically return to a few previously visited locations, such as home or office, and the mobility of each user can be modeled by a stochastic process centered at a fixed point. Song *et al.* [SQBB10] focus on the predictability in human mobility, and report that there is a 93% predictability of human mobility, which is contributed by the high regularity of human behavior. In their following work [SC10], Song *et al.* report that users' trajectories are not strictly stochastic: when choosing next location to visit, the user will either move to a new location that she has not visited in the trajectory, or return to a location she visited frequently before. Based on the two principles, they propose a self-consistent microscopic model to predict individual mobility.

Cho *et al.* [CML11] observe that the mobility of each user is centered at several regions (representing "work" and "home"), and model each region as a Gaussian distribution over latitude and longitude. The probability that a user stays at the two regions is modeled as a function of time. They propose a generative model, Periodic Mobility Model (PMM), to predict the location of a user. PMM takes a user and time as input, and samples a region. Finally, the region generates a geo-location. They also incorporate social influence into the model: a user is more likely to visit a location that is close to the locations recently visited by her friends. Tarasov *et al.* [TKP13] follow the paper [CML11] and propose an algorithm to predict user' location, in which a region is modeled by radiation model [SGMB12] instead of Gaussian distribution. However, the above two studies take the number of regions of each user as input, which is very hard to set manually.

Some studies employ topic models to model users' mobility behaviors, but neglect the time factor.

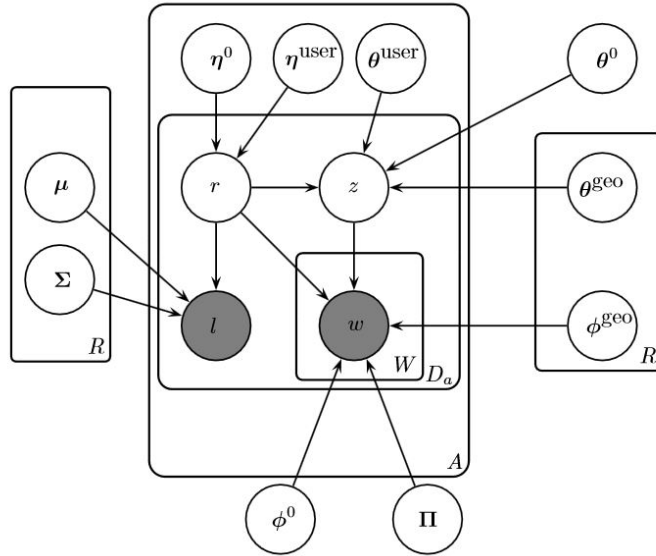


Figure 2.4: The Graphical Models in Hong’s Paper

In order to make requirement-aware POI recommendations, we need to exploit both context (spatial and temporal) and requirement (semantic) information of individuals. However, to the best of our knowledge, no studies have been proposed that can model the three aspects information (where, when and what) of individuals (who) simultaneously. The most similar studies are proposed by Hong *et al.* [HAG⁺12], Ahmed *et al.* [AHS13] and Hu *et al.* [HE13].

In the paper [HAG⁺12], Hong *et al.* present a method to learn the geographical topics for twitter users. For a user u , this method first samples a region r based on the popularity of regions η^0 and the preference of the user over the regions η^{user} . Then, a topic z is sampled dependent on both the topic distributions of the region θ^{geo} , the user θ^{user} and the background θ^0 . The words of a tweet are generated based on the word distributions of the region ϕ^{geo} , topic Π and background ϕ^0 , while the location coordinates of the tweet are sampled from the region’s gaussian distribution, which is parameterized by mean μ and covariance matrix Σ . The graphical model is shown in Figure 2.4.

In their following work [AHS13], Ahmed *et al.* assume that there exist hierarchical relationships between geographical regions, *i.e.*, regions can be organized by a hierarchy based on the geographical containment, and regions under the same parent nodes (regions) have similar semantics. Then, they propose a nonparametric hierarchical topic model to model the generative process of the content and geographic locations of documents. Specifically, this model employs the nested Chinese Restaurant Franchise Process to build the region hierarchy, over which each user has a region distribution. A tweet

is generated along a path from the root node to a leaf node, where the Gaussian distribution of the leaf node generates the coordinates of the tweet, and the topics and word distribution of the leaf node generates the content of the tweet.

Hu *et al.* [HE13] propose a model that considers both the coordinates and semantic information of locations. In this model, the topic and region of a tweet is drawn based on both global and user-specific topic and region distributions, respectively. The topic further generates the text content, where the region and topic together determine the location of the tweet.

However, these approaches suffer from four limitations: 1) the geographical regions in these studies are global, *i.e.*, sharing by every user, which leads to imprecise modeling of mobility area of individuals; 2) no temporal information is considered; 3) it is the geographical coordinates, instead of POI identifiers, that are considered in [HAG⁺12] and [AHS13], so they cannot recommend specific POIs for users; 4) there are several parameters that need to be set empirically in [HAG⁺12, HE13], such as the numbers of topics and regions.

2.5 Summary

In this chapter, we survey relevant proposals to POI recommendation. Specifically, we first review the three types of general recommendation techniques, namely, collaborative filtering, content-based recommendation and hybrid recommendation. Then, we focus on the studies on the context-unaware POI recommendation and two context-aware POI recommendations, namely, time-aware recommendation and group recommendation, which incorporate time and companions as context information, respectively. Since topic model is an important technique to make group and requirement-aware recommendations, and mobility modeling of users is closely related to requirement-aware recommendation, we review previous studies on the two topics at the end of the chapter.

Chapter 3

Time-aware Point-of-interest Recommendation

It has been reported that users tend to visit different places at different time in a day [CML11], *e.g.*, visiting a restaurant at noon and visiting a bar at night. Thus, time is an important consideration in POI recommendations. In this chapter, we define a new problem, namely, time-aware POI recommendation, to recommend POIs for a given user at a specified time in a day. To solve the problem, we develop two collaborative recommendation models under the frameworks of user-based CF and graph-based preference propagation, respectively. Moreover, based on the observation that users tend to visit nearby POIs, we further enhance the recommendation models by incorporating geographic information. Our experimental results on two real-world datasets show that our proposed approaches outperform the state-of-the-art POI recommendation methods substantially.

This chapter is organized as follows: We first report the datasets and the observations on temporal and temporal influences in Section 3.2, based on the observations, we introduce the user-based CF model and preference propagation based model in Section 3.3 and 3.4, respectively. Then, we present experimental results in Section 3.5. Finally Section 3.6 concludes this chapter. All the notations used in this chapter are listed in Table 3.1.

3.1 Problem Definition

In this chapter, we define a novel problem, namely, time-aware POI recommendation, to suggest new POIs for users based on specified time.

Formally, given the target user u and target time t , we rank candidate POIs $l \in L - L_u$ based on user u 's preference $r_{u,i,t}$ to l at time t in descending order, and return the top-ranked ones as the recommendations, where L and L_u is the whole POI set and set of POIs that user u has visited before, respectively.

Table 3.1: Symbols for Time-aware POI Recommendation

| Symbol | Description |
|----------------------------------|---|
| U, L, T, S | user set, POI set, time slot set, session set |
| $CI_l, CI_{l,t}$ | the set of check-ins at l , CI_l at time t |
| L_u | visited POI set of u |
| S_u | session node set of u . |
| A_n | adjacent node set of n . |
| $u, v, l, t, s_{u,t}$ | user $u, v \in U$, POI $l \in L$, time slot $t \in T$, session of u , $t \in S$ |
| $\mathbf{c}_u, \mathbf{c}_{u,t}$ | the binary check-in vector of u over L , and the binary check-in vector u over L at t |
| $c_{u,l}, c_{u,t,l}$ | element of \mathbf{c}_u and $\mathbf{c}_{u,t}$, respectively |
| $w_{u,v}$ | the similarity between u and v |
| $w_{u,v}^{(t)}, w_{u,v}^{(ts)}$ | time-enhanced similarity, smoothed similarity |
| $G, E, V, P_{i,j}$ | GTAG/TAG graph, edge set of G , node set of G , set of paths from node i to node j |
| $e_{i,j}, w_{i,j}$ | edge from node i to node j , weight of $e_{i,j}$ |
| $ t_i, t_j $ | time difference between t_i and t_j |
| $dis(\ell_i, \ell_j)$ | geographical distance between ℓ_i and ℓ_j |
| $wi(d)$ | the willingness a user visits a d -km far away POI |
| r_i | preference value of node i |
| η | the parameter controlling the preference of session nodes propagating to POI nodes and user nodes |
| τ | the parameter controlling the preference of POI nodes propagating to POI nodes and session nodes |
| H | the parameter controlling the time influence |
| k | the maximum number of POI nodes that a POI node can connect to |
| α, β | the parameters of power law function |

3.2 Data Analysis and Problem Definition

In this section we study the temporal and spatial influences to users' check-in behaviors on two LBSN datasets, and define the problem of time-aware POI recommendation.

3.2.1 Dataset

Two datasets are used in this chapter. We collect 342,850 check-ins from Foursquare which were made within Singapore between August 2010 and July 2011. We also set a bounding box and extracted 736,148 Gowalla check-ins from the dataset provided by [CML11], which were made within California and Nevada between February 2009 and October 2010. Each check-in is a $\langle user, POI, time \rangle$ tuple and each POI has its own

geographical coordinates. Following previous study [CYKL12], we removed users who have checked in fewer than 5 POIs, and then removed POIs which fewer than 5 users checked in. After pre-processing, the Foursquare dataset (Foursquare) contains 194,108 check-ins made by 2,321 users at 5,596 POIs, and the Gowalla dataset (Gowalla) contains 456,988 check-ins made by 10,162 users at 24,250 POIs (see Table 3.2). Note that the two datasets have different scales in terms of the size of entities (*i.e.*, users, POIs, and check-ins) and geographical range.

Table 3.2: Statistics on the Datasets (after pre-processing)

| Dataset | # Check-ins | # Users | # POIs |
|------------|-------------|---------|--------|
| Foursquare | 194,108 | 2,321 | 5,596 |
| Gowalla | 456,988 | 10,162 | 24,250 |

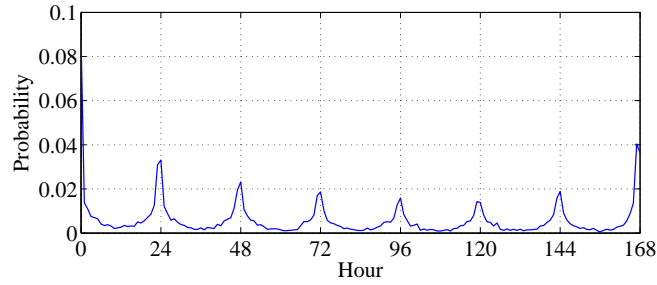
3.2.2 Observation of Temporal Influence

First, we examine the probabilities that a user returns to her firstly visited POIs after different number of hours. Specifically, if a user visits a POI multiple times, we calculate the time difference between her first check-in at the POI and each of her subsequent check-ins. The time differences of all users at all POIs are aggregated and grouped into bins on hourly basis. Figure 3.1.a plots the probability distribution obtained on the Foursquare dataset. Observe that, the returning peaks occur around every 24 hours (*e.g.*, 24, 48, 72, etc.), suggesting that users' check-in activities are daily periodic. That is, in close time periods of a day, users are likely to visit the same places, and in different time periods of a day, users visit different places. This observation is in accordance with the findings reported in [CCLS11].

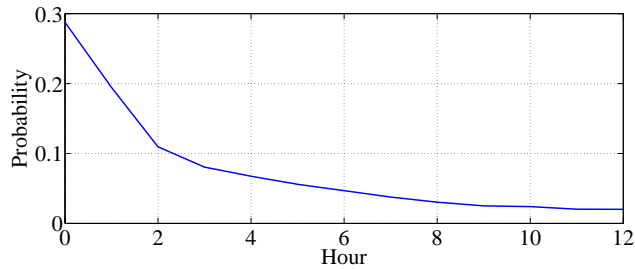
Next, to further understand the periodicity, we discard the date information of the check-ins and compute the deviation hours of each subsequent check-in of a user from the first check-in time of the user at a POI. Figure 3.1.b plots the returning probabilities at different hour deviations on the Foursquare dataset. From this probability distribution, it is clear that users tend to visit the same POIs at close time of day, and their behaviors in close time are similar. In addition, the curve fits exponential function well. Note that, similar observations are made on the Gowalla dataset. To make it concise, we choose not to plot the figures.

3.2.3 Observation of Spatial Influence

We assume that people tend to visit nearby POIs to their previous locations, and their willingness to visit a POI decreases as the distance increases. To verify this assumption, we study the effect of distance on users' check-in behaviors.



3.1.a: Hour Deviation



3.1.b: Daily Hour Deviation

Figure 3.1: Returning Probability on Foursquare Dataset

Specifically, for each user, we sort the user’s check-ins by time. For the check-ins made within a day, we calculate the distance between two POIs of every two adjacent check-ins. We aggregate the results of all users, and plot the number of check-ins as a function of the distance in Figure 3.2. Note that a larger probability value implies that users are more willing to check in POIs at that distance. Observe from Figure 3.2, the distribution of the probability values follows a power law, suggesting that users are more willing to check in nearby POIs to their current places. Moreover, observe that the slope of the curve of Singapore (Figure 3.2.a) drops quickly after 10KM. Similar observation holds for the curve of California and Nevada (Figure 3.2.b). That is, the willingness of people visiting a faraway POI (*e.g.*, 10KM) drops significantly.

3.3 User-based CF Approach

In this section, we first develop the recommendation methods incorporating temporal influence and spatial influence in Sections 3.3.1 and 3.3.2, respectively. Then, we combine the two methods in Section 3.3.3.

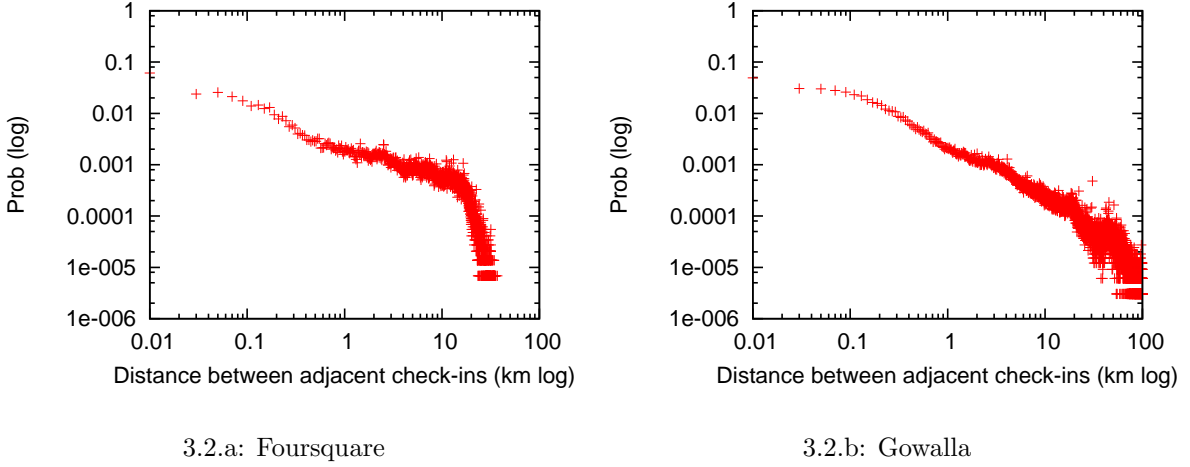


Figure 3.2: Distribution of Distance between Successive Check-ins

3.3.1 Utilizing Temporal Influence

In this section, we first present the method of incorporating time influence in the user-based CF in Section 3.3.1.1. The enhancement to the method by smoothing is then presented in Section 3.3.1.2.

3.3.1.1 Incorporating Temporal Influence

Check-in Representation As reported in 3.2.2, human show strong periodic behavior throughout a day. Thus, we split a day into multiple equal time slots based on hour. Then, we represent the behavior of a user at a specific time by the set of check-ins that user has made at that time. For ease of presentation, *time* and *time slot* are used interchangeably in this chapter unless noted otherwise.

To represent the temporal check-in behavior of users, we introduce the time dimension into the conventional user-POI matrix. Specifically, we use user-time-POI cube (UTP) to represent the temporal check-in records. In the UTP cube, each element $c_{u,t,l}$ represents the check-in activity of a user u , at a POI l at time slot t , where $c_{u,t,l} = 1$ if user u has checked in POI l at time t , and $c_{u,t,l} = 0$ otherwise.

Recommendation To make use of the time influence for time-aware POI recommendations, we extend the user-based CF model in two aspects: (i) we leverage the time factor when computing the similarity between two users; (ii) we consider the historical check-ins at time t in the repository, rather than at all time, during recommendation.

Given a user u and time t , the recommendation score that the user will check-in an unvisited POI l at t is defined as follows:

$$\hat{c}_{u,t,l}^{(t)} = \frac{\sum_v w_{u,v}^{(t)} c_{v,t,l}}{\sum_v w_{u,v}^{(t)}}, \quad (3.1)$$

where $w_{u,v}^{(t)}$ is the temporal behavior similarity between u and v . Next, we detail the computation of the temporal behavior similarity $w_{u,v}^{(t)}$.

Similarity Estimation

In our method, the similarity between two users is estimated based on their temporal behaviors over all time. Specifically, if two users always check-in the same POIs at the same time, the similarity value between the two will be high, and one user’s check-in history has a large impact on the POI recommendation for the other user. Therefore, we extend the cosine similarity measure to calculate the similarity between u and v as follows:

$$w_{u,v}^{(t)} = \frac{\sum_{t=1}^T \sum_{l=1}^L c_{u,t,l} \cdot c_{v,t,l}}{\sqrt{\sum_{t=1}^T \sum_{l=1}^L c_{u,t,l}^2} \sqrt{\sum_{t=1}^T \sum_{l=1}^L c_{v,t,l}^2}} \quad (3.2)$$

Note that, the similarity might be calculated in an alternative manner based on the check-in history at the *exact time*, which is similar to the general idea of context-aware recommendation [ASST05]. We did not adopt this approach because of two reasons: (i) There is a high possibility that the user has never made any check-in at the time. In our two datasets, on average the check-ins of each user fall into 9.97 and 13.58 hour slots, respectively. Thus, if we only consider the time, no similar users may be found for the user; (ii) Even if the user has check-ins at the time, the number of these check-ins is still too small for meaningful similarity computation (on average, only 4.17 and 5.81 check-ins in each hour for the two datasets, respectively).

3.3.1.2 Enhancement by Smoothing

The straightforward extension of incorporating temporal influence presented in Section 3.3.1.1 has its weakness in handling data sparsity. In this section, we first explain the reason of the weakness and then present two novel enhancements to overcome this weakness that have not been used in previous work. First, we smooth the similarity estimation using the similarity values of the other time slots. Second, we consider the POIs visited by similar users at different time slots, and weight them with the estimated similarity for recommendation.

The similarity incorporating temporal influence in Equation 3.2 is estimated based on the UTP cube, which is much sparser than the user-POI matrix. The sparsity could easily make it fail to characterize the users’ similarity when a POI is visited at different time slots by different users. For example, consider two POIs and two time slots. User u checked in l_1 and l_2 at t_1 and t_2 , respectively, while user v checked in l_1 and l_2 at t_2 and

t_1 , respectively. If we do not consider time, the similarity between the two users will be 1 by Equation 2.2, since both users checked in both POIs. However, if the time is taken into account, their similarity becomes 0 according to Equation 3.2, since their check-ins to the two POIs are made at different time. Obviously, the time-aware similarity is not desirable in the case, particularly when t_2 is very close to t_1 , *e.g.*, the next hour to t_1 .

We proceed to discuss how to address this problem caused by data sparsity. A straightforward method is to use a decaying parameter to give a higher weight to the POIs checked in at close time slots, and a smaller weight to those checked in at distant time slots. However, this method faces the following challenge: A user’s behavior at a time is described by the check-in records of the user at that time, and the user behavior at different time may be similar. For example, the check-in behavior of a user at 9-10 am might be similar to the check-in behavior of the user at 3-4 pm, because the user is likely to stay at workplace and make check-ins around it. Such similarity cannot be easily captured by a decaying parameter.

To further illustrate the point, we analyze the check-in data in Foursquare made within Singapore for the similarity patterns between check-in behaviors of users at different time. Note that, the results on the other data used in our experiment are qualitatively similar to this dataset, and thus are omitted here.

Let $\mathbf{c}_{u,t} = \{c_{u,t,1}, c_{u,t,2}, \dots, c_{u,t,L}\}$ be the check-in vector of user u at time t , which is extracted from the UTP cube. For each user u , we calculate the cosine similarity between every pair of check-in vectors \mathbf{c}_{u,t_i} and \mathbf{c}_{u,t_j} at time t_i and t_j respectively. Then, we calculate the similarity value between two time slots t_i and t_j to be the average of the similarity values of all users between these two time slots t_i and t_j .

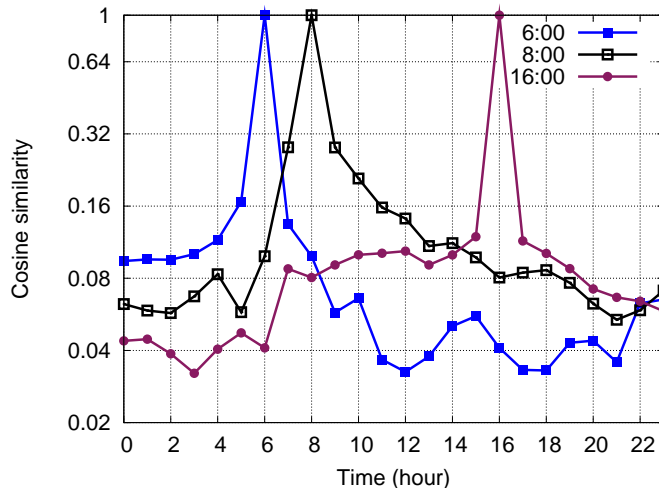


Figure 3.3: User Behavior Similarities between a Given Hour (6:00, 8:00, and 16:00) and other Hours

Figure 3.3 shows the similarity curves for three time slots (6:00, 8:00, and 16:00) over the Singapore data. The similarity curve for 6:00 shows the check-in similarity between 6:00 and every other hour in a day, similarly for the other two curves. Observe that the check-in behavior at a time is similar to the check-in behavior of its close time slots. For example, the check-in similarity between 6:00 and 5:00 is much higher than the similarity between 6:00 and 4:00. Nevertheless, although 6:00 and 8:00 are close to each other in terms of time, their *similarity curves* are in quite different shapes. For instance, the check-in behavior of 6:00 is similar to that at its previous hours (0:00 – 5:00), but quite different from the behavior at the later hours (8:00 – 23:00). The curve of time 8:00 displays an opposite shape. In contrast, the curve of 16:00 drops with equal speed on both sides. Observe that the curve of 16:00 almost does not decrease from 8:00 to 14:00, and it even increases around 12:00 to 13:00. This is, users might stay at workplace during that period, and hence have similar check-ins (while people tend to have lunch about 12:00, making the curve drops to some extent). Instead of using cosine similarity to compute the user check-in behavior, we have also tried other metrics, such as Pearson correlation and Total Variation Distance, but observed similar results. In summary, the check-in behavior at one time may be more similar to some time slots than others. This motivates us to develop a method to utilize the check-in behavior similarity for POI recommendations.

We propose to smooth the UTP cube based on the check-in similarity between different time slots. For each check-in vector in the cube, we smooth it using the vectors of those similar time slots. Specifically, we compute a new value for element $c_{u,t,l}$ using Equation 3.3, where $\rho_{t,t'}$ is the similarity between time slots t and t' . Accordingly, the smoothing enhanced similarity between two users u and v is calculated using Equation 3.4.

$$\tilde{c}_{u,t,l} = \sum_{t'=1}^T \frac{\rho_{t,t'}}{\sum_{t''=1}^T \rho_{t,t''}} c_{u,t',l} \quad (3.3)$$

$$w_{u,v}^{(te)} = \frac{\sum_{t=1}^T \sum_{l=1}^L \tilde{c}_{u,t,l} \cdot \tilde{c}_{v,t,l}}{\sqrt{\sum_{t=1}^T \sum_{l=1}^L \tilde{c}_{u,t,l}^2} \sqrt{\sum_{t=1}^T \sum_{l=1}^L \tilde{c}_{v,t,l}^2}} \quad (3.4)$$

With the similarity formulation in Equation 3.4, if two users u and v have visited the same POIs at *the same* or *similar* time slots, their similarity value will be high. Otherwise, if they have visited the same POIs at dissimilar time slots, their similarity value will be low.

The check-in behavior similarity between two time slots also enables us to introduce another enhancement in recommending POIs for user u at a given time t . Recall that in the method described in Section 3.3.1.1, only the POIs visited by a similar user v at

the time t are considered in recommendation. The check-in behavior similarity between time slots makes it possible to consider any candidate POI l visited by a similar user v , irrespective if the check-in time t' by v is the same as the time t . More specifically, in the enhancement method, a candidate POI l is weighted by the similarity value between two time slots, t' and t , if the historical check-ins are in different from the time t . The recommendation score, denoted by $\hat{c}_{u,t,l}^{(te)}$, that user u will check-in l at t is then updated as follows:

$$\hat{c}_{u,t,l}^{(te)} = \frac{\sum_v w_{u,v}^{(te)} \sum_{t'} \tilde{c}_{v,t',l} \cdot \rho_{t,t'}}{\sum_v w_{u,v}^{(te)}} \quad (3.5)$$

3.3.2 Utilizing Spatial Influence

The geolocation of POI is an important factor affecting human’s check-in behavior and has been exploited in earlier studies [YLL11], namely, the probability of a user visiting a set of POIs is the product of probabilities of visiting all the pairwise POIs in the set, and the probability that a user checks in a new POI is estimated by the product of the probabilities of visiting all the pairwise POIs, each pair consisting of the new POI and each previously visited POI.

In this work, we assume users tend to visit POIs that are close to their current positions. This assumption has been verified in Section 3.2.3. Based on the assumption, we propose a new method and show that POI recommendation can be improved by considering spatial influence in an alternative manner.

3.3.2.1 Incorporating Spatial Influence

Based on the observations made in Section 3.2.3, we use a power law distribution to model the willingness of a user moving from one place to another as a function of their distance. Defined in Equation 3.6, $wi(dis)$ is the willingness of a user to visit a dis km far away POI, and a , k are parameters of the power law function.

$$wi(dis) = a \cdot dis^k \quad (3.6)$$

Maximum likelihood estimation [Arn83] is used to estimate the two parameters a and k . More specifically, we take logarithmic on both side of Equation 3.6, and get the following equation.

$$\ln(wi(dis)) = \ln(a) + k \ln(dis). \quad (3.7)$$

The above linear function over $\ln(dis)$ can be easily learned by the least-square regression. As the result, we learn the parameters in Equation 3.6. Note that, in learning

the two parameters, irregular portion in Figures 3.2.a and 3.2.b (*i.e.*, data points having distance larger than 10km) is not considered. These data points represent fewer than 15% of the total number of check-ins.

Consider a user is currently at POI l_i and POI l_j is a candidate POI to check in next, at distance $dis(l_i, l_j)$ from l_i . We model the probability that the user will check in l_j to be proportional to the user's willingness to check in a POI at distance $dis(l_i, l_j)$. The conditional probability is computed using the following equation.

$$p(l_j|l_i) = \frac{wi(dis(l_i, l_j))}{\sum_{l_k \in L, l_k \neq l_i} wi(dis(l_i, l_k))} \quad (3.8)$$

Observe that, as the distance increases, the conditional probability decreases, which reflects that the user is less likely to visit a faraway candidate POI.

Given a user u , and his/her historical POIs L_u , we calculate $P(l|L_u)$ as the ranking score for each candidate POI l , and then recommend the top-ranked POIs to the user. Based on the Bayes rule, this score is calculated as follows.

$$\hat{c}_{u,l}^{(s)} = P(l|L_u) \propto P(l)P(L_u|l) \quad (3.9)$$

$$= P(l) \prod_{l' \in L_u} P(l'|l) \quad (3.10)$$

Note that the derivation from Equation 3.9 to Equation 3.10 is based on the assumption that for a given l , the check-in probabilities of POIs in L_u are independent from each other.

3.3.2.2 Enhancement by Temporal Popularity

In Equations 3.9 and 3.10, $P(l)$ is the prior probability that POI l is checked in by all users in the dataset. However, the popularity of a POI varies over time, *e.g.*, a restaurant is more popular around noon and evening, and a workplace is more popular during working hours. To illustrate this point, we plot the check-in probabilities of the top-5 most popular POIs in Figure 3.4, based on the check-in data of Singapore, collected from Foursquare. The check-in probability at a given time is computed by the ratio of the number of check-ins at that time to the total number of check-ins at the POI. Observe that the popularity of each POI varies greatly over time, and different POIs become popular at different time.

Reconsider Equations 3.9 and 3.10. The probability of checking in a POI should reflect both its popularity at the specific time and the distance to the user's current location. In other words, the probability user u will check in POI l at time t depends on the probability (or popularity) of l at t , along with the distance between l and u 's visited

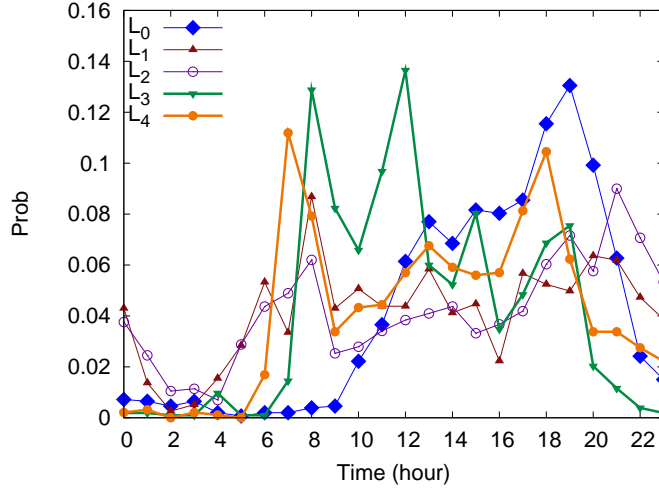


Figure 3.4: The Distribution of 5 Popular POIs over Time

POIs. It provides us a way to enhance this model by adjusting $P(l)$ with the temporal popularity of POI l , denoted by $P_t(l)$.

$$P_t(l) = \beta \frac{|CI_{l,t}|}{\sum_{l' \in L} |CI_{l',t}|} + (1 - \beta) \frac{|CI_l|}{\sum_{l' \in L} |CI_{l'}|} \quad (3.11)$$

In the above equation, $|CI_l|$ is the number of check-ins at l ; $|CI_{l,t}|$ is the number of check-ins at l at time t , and β tunes the weight between l 's temporal popularity and long-term popularity.

By replacing $P(l)$ in Equation 3.10 with $P_t(l)$, the resultant method incorporates the temporal popularity. The recommendation score, denoted by $\hat{c}_{u,t,l}^{(se)}$, is then computed by the following equation:

$$\hat{c}_{u,t,l}^{(se)} \propto P_t(l) \prod_{l' \in L_u} P(l'|l) \quad (3.12)$$

We emphasize that the temporal information here is used in a different way from what we do in Section 3.3.1. In Section 3.3.1, the temporal information is employed to discover personalized temporal preference on POIs. In contrast, here we use the temporal preference on POIs in a collective manner (*i.e.*, by all users).

3.3.3 A Unified Framework

Given a user u , time t and a candidate POI l , we can get a recommendation score $\hat{c}_{u,t,l}^{(t)}$ that user u will check in l at t using either of the two methods incorporating temporal influence in Section 3.3.1 (*i.e.*, with or without smoothing). Similarly, we can also compute a score

$\widehat{c}_{u,t,l}^{(s)}$ using either of the two methods incorporating spatial influence in Section 3.3.2 (*i.e.*, with or without considering temporal popularity).

We use linear interpolation to weight the two scores to compute the final recommendation score for POI l . However, the two scores are measured by different methods, and have different value ranges. Thus, we first normalize the two scores using min-max normalization before we combine them.

$$\bar{c}_{u,t,l}^{(t)} = \frac{\widehat{c}_{u,t,l}^{(t)} - \min_{l'}(\widehat{c}_{u,t,l'}^{(t)})}{\max_{l'}(\widehat{c}_{u,t,l'}^{(t)}) - \min_{l'}(\widehat{c}_{u,t,l'}^{(t)})} \quad (3.13)$$

$$\bar{c}_{u,t,l}^{(s)} = \frac{\widehat{c}_{u,t,l}^{(s)} - \min_{l'}(\widehat{c}_{u,t,l'}^{(s)})}{\max_{l'}(\widehat{c}_{u,t,l'}^{(s)}) - \min_{l'}(\widehat{c}_{u,t,l'}^{(s)})} \quad (3.14)$$

In the above equations, $\max_{l'}(\cdot)$ and $\min_{l'}(\cdot)$ denote the maximum and minimum check-in scores of u at t across all POIs.

After normalization, we compute the combined score that user u will check-in POI l at time t using the following equation, where α is a tuning parameter.

$$c_{u,t,l} = \alpha \times \bar{c}_{u,t,l}^{(t)} + (1 - \alpha) \times \bar{c}_{u,t,l}^{(s)} \quad (3.15)$$

By this framework, we calculate the check-in score for each candidate POI, and return the top-ranked POIs as the recommendation results.

3.4 Graph-based Preference Propagation Approach

The model proposed in last section makes time-aware POI recommendations by linearly combining the results of two independent components, which exploit the temporal and spatial influences independently. This model has two shortcomings. 1) The influences of the two factors to the check-in behavior of users interact with each other in real-life, which is not captured in the last model. Intuitively, a user may stay at different places at different time, *e.g.*, staying round workplace in the daytime, and staying home at night. When recommending new POIs for a user at a specific time, the candidate POIs close to the POIs that were visited by the user round the target time, rather than all the visited POIs, should be recommended. 2) The temporal check-in behaviors of two users might be similar at two different time slots, namely, similar set of POIs might be visited by user u_1 at time t_1 , but visited by user u_2 at time t_2 . Thus, when making recommendation for u_1 at t_1 , the POIs visited by u_2 at t_2 should be recommended. However, our last model cannot exploit such information, because it only considers the similarity between two users based on the POIs visited at the same time slots.

To solve these two problems, in this section, we propose a new time-aware POI recommendation algorithm, namely, Geographical-Temporal Influences Aware Graph + Breadth-first Preference Propagation (GTAG+BPP).

3.4.1 Geographical-Temporal Influences Aware Graph (GTAG)

In this section, we detail the Geographical-Temporal influences Aware Graph (GTAG) to exploit the observations of geographical influence and temporal influence in time-aware POI recommendation.

3.4.1.1 GTAG Structure

We build the GTAG based on the following intuitions:

- (i) **Intuition 1:** Users' interests vary with time, and a user may visit different POIs at different time [GTHL13]. The temporal interests of a user in a time is reflected as the POIs she visited in that time.
- (ii) **Intuition 2:** The check-in interests of a user in the time closer to the target time are more relevant, and thus more important, for recommendation.
- (iii) **Intuition 3:** If two users have similar temporal interests in two time, they tend to visit the same POIs in the two time.
- (iv) **Intuition 4:** Users tend to visit their nearby POIs [YYLL11].

In GTAG, users and POIs are represented by user nodes $u \in U$ and POI nodes $\ell \in L$, respectively. To represent users' check-in interests at different time (**Intuition 1**), we use a session node $s_{i,j} \in S$ to relate the POIs visited by user u_i in time $t_j \in T$. Here U , L and S are the user node set, POI node set, and session node set, respectively. These three types of nodes are connected by weighted directed edges, namely, $E_{U,S}$, $E_{S,U}$, $E_{S,L}$, $E_{L,S}$ and $E_{L,L}$, where $E_{X,Y}$ denotes the set of edges from nodes in set X to nodes in set Y . Edges in the five sets compose two types of links, namely, check-in link and POI link, which embed the temporal and geographical influences, respectively. A check-in link represents a check-in record, and it consists of edges between user node and session node, and between session node and POI node. The edges $e_{u_i,s_{i,j}} \in E_{U,S}$, $e_{s_{i,j},u_i} \in E_{S,U}$, $e_{s_{i,j},\ell_k} \in E_{S,L}$, and $e_{\ell_k,s_{i,j}} \in E_{L,S}$ form a check-in link, which corresponds to user u_i 's check-in on POI ℓ_k in time t_j . Since the check-in interests of a user in the time closer to the target time are more relevant (**Intuition 2**), the edges connecting to the session nodes that are close to the target time will be assigned with larger weights (to be explained in Section 3.4.1.2). The edges from POI nodes to session nodes bridge sessions of users that

share similar POI interests, which enables us to exploit other users' temporal interests for recommendation (**Intuition 3**).

To incorporate the intuition that users tend to visit their nearby POIs (**Intuition 4**), we use a POI link $e_{\ell_k, \ell_m} \in E_{L,L}$ to connect ℓ_k to ℓ_m if ℓ_m is close to ℓ_k in distance. Theoretically, there are $|L||L - 1|$ edges that link every pair of POIs. Incorporating all of them into GTAG will greatly deteriorate the recommendation efficiency. In addition, if two POIs are far from each other, users are less likely to travel from one to the other, and thus the correlation between them is small. We therefore set a threshold k to limit the number of edges starting from one POI to other POIs, *i.e.*, for each POI, only the k nearest POIs are connected with it.

Figure 3.5 gives a sample set of 7 check-in records made by two users (u_1 and u_2) on four POIs (ℓ_1 to ℓ_4) during two time slots (t_1 and t_2). The GTAG constructed using this sample set of check-in records is illustrated on the right hand side of the figure. Since u_2 visited POI ℓ_4 in time t_2 (*i.e.*, the last one in the 7 sample check-in records), there are directed edges between u_2 and $s_{2,2}$, and between $s_{2,2}$ and ℓ_4 in the graph. ℓ_3 is connected to ℓ_4 , since ℓ_4 is assumed to be close to ℓ_3 in distance. Note that k is set to 1 in this example.

Generally, GTAG makes recommendations as follows: given a target user u_i and target time t_q , we first set the weights of links from user nodes to session nodes adaptively based on the target time. We inject preference to the user node u_i , and then propagate the preference to candidate POI nodes via various paths. During the propagation, both the geographical and temporal influences are exploited. In the end, the POIs that receive larger preference will be recommended. Before detailing the preference propagation, we first introduce how to set the weights for edges in GTAG.

3.4.1.2 Weight Computation of the Edges

The initial weights of all edges in $E_{S,U} \cup E_{S,L} \cup E_{L,S}$ are set to 1. We have also considered setting the weights of edges in $E_{S,L}$ and $E_{L,S}$ based on the number of visits. However, poorer recommendation results were obtained in our experiments compared with using 1 as the weights. This is consistent with the result reported for graph-based recommendation in [NSLM12].

Next, we detail how to set the initial weights for edges in $E_{U,S}$ and $E_{L,L}$, and how to normalize the weights for GTAG.

Weights of Edges in $E_{U,S}$. Recall that our time-aware POI recommendation task is to recommend POIs for a target user u to visit at a target time t_q . Naturally, the session node whose time is close to the target time t_q is more important to the recommendation

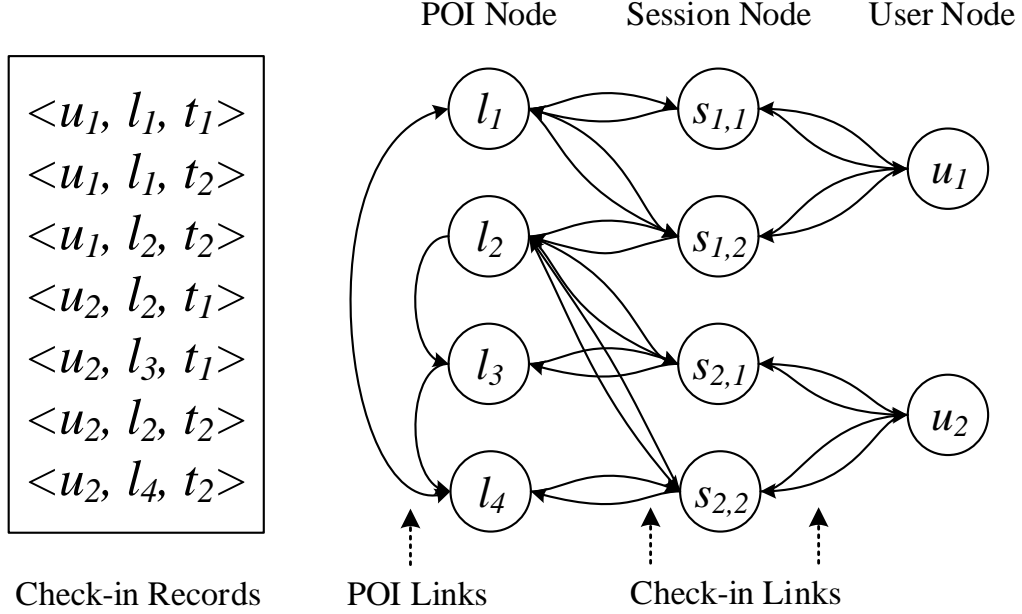


Figure 3.5: GTAG Constructed using 7 Sample Check-ins

task (**Intuition 2**). Following previous work [DL05], we use an exponential function to model the importance of the session node for time slot t to the target time t_q :

$$f(t, t_q) = \exp\left(-\frac{1}{H} \cdot |t, t_q|\right), \quad (3.16)$$

where $|t, t_q|$ is the time difference between t_q and t , and H is a parameter controlling the extent of temporal influence. A smaller H leads to smaller weight for the sessions far from t_q . Note that, if the time slot of session node t matches t_q , then $f(t, t_q) = 1$.

We use the importance values of session nodes computed above to initialize the weights of edges from user nodes to session nodes. Specifically, given a target time t_q for recommendation, the weight of the edge from user u_i to session node $s_{i,j}$ is computed by

$$w_{u_i, s_{i,j}} = f(t_j, t_q), \quad (3.17)$$

where t_j is the time slot of session node $s_{i,j}$.

We argue that the above weighting scheme offers at least two advantages. First, the weight between a user node and a session node is *adaptively* set based on the target time for recommendation. That is, the GTAG adjusts the importance of session nodes based on the target time for recommendation, so that the temporal influence is considered. Second, the above weighting scheme alleviates the data sparsity problem: when we recommend POIs for target time t_q , check-in records in all time sessions are considered with different weights. If we only consider the check-ins during the target time slot, the data will

become much more sparse, and it is well known that sparsity is a major challenge in recommendation.

Weights of Edges in $E_{L,L}$. The observation on graphical influence states that users tend to visit nearby places (**Intuition 4**). The willingness of visiting a place decays with the increase of distance from the current location. Here, we adopt a power-law function of distance to model the willingness of a user moving from one place to another as in Section 3.3.2. We employ the willingness as the weight of edge in $E_{L,L}$:

$$w_{\ell_i, \ell_j} = \text{wi}(\text{dist}(\ell_i, \ell_j)), \quad (3.18)$$

where $\text{dist}(\ell_i, \ell_j)$ is the geographical distance between POIs ℓ_i, ℓ_j .

Edge Weight Normalization. After setting the initial weights for edges in GTAG, we normalize the edges' weights as follows:

$$w_{i,j} = \begin{cases} \frac{1}{\tau \sum_{k \in A_i \cap L} \text{wi}(\text{dis}(i, k)) + |A_i \cap S|} & \text{if } i \in L \text{ and } j \in S, \\ \frac{\tau \text{wi}(\text{dis}(i, j))}{\tau \sum_{k \in A_i \cap L} \text{wi}(\text{dis}(i, k)) + |A_i \cap S|} & \text{if } i \in L \text{ and } j \in L, \\ \frac{1}{\eta |A_i \cap L| + 1} & \text{if } i \in S \text{ and } j \in U, \\ \frac{\eta}{\eta |A_i \cap L| + 1} & \text{if } i \in S \text{ and } j \in L, \\ \frac{w_{i,j}}{\sum_{k \in A_i} w_{i,k}} & \text{if } i \in U \text{ and } j \in S. \end{cases}$$

In these equations, A_i is the set of adjacent nodes of node i , $|A_i \cap S|$ is the number of adjacent session nodes of node i , and $|A_i \cap L|$ is the number of adjacent POI nodes of node i . τ is a parameter to balance the propagation preference of a location node to its adjacent location nodes and session nodes. A larger τ indicates that geographical distance plays a more important role in preference propagation. η is another parameter that balances the importance of POI nodes and user nodes to the preference propagation of a session node.

3.4.2 Breadth-first Preference Propagation

The basic idea of the preference propagation is to first inject initial preference on the target user node u , and then propagate the preference to candidates POI nodes through various paths over the graph [XYZ⁺10]. Defined in [XYZ⁺10], the preference propagated

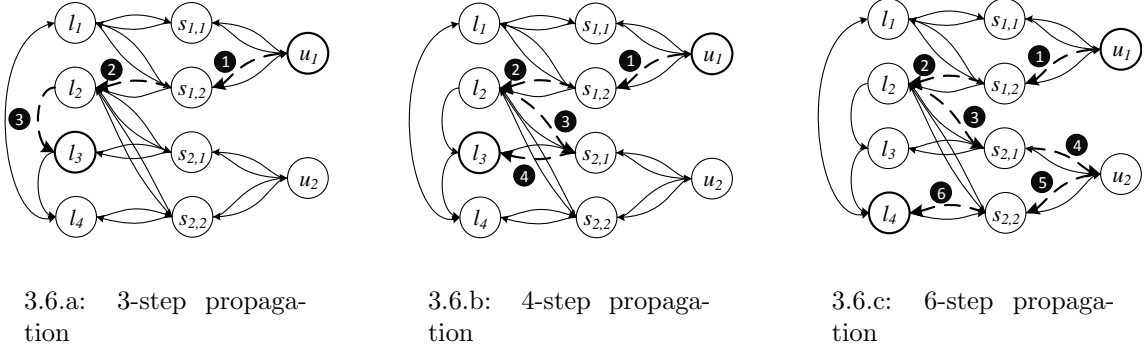


Figure 3.6: Examples of Valid Propagation Paths of Lengths 3, 4, and 6

by each path p is the production of the initial preference r_u assigned to target user node u and the weights of all edges on the path:

$$r_p^{(p)} = \prod_{e_{i,j} \in p} w_{i,j} \cdot r_u, \quad (3.19)$$

where $w_{i,j}$ is the weight of the edge $e_{i,j}$ contained in path p . For each candidate POI ℓ , its preference value is the sum of all preference propagated to it through all paths from target user node u :

$$r_\ell = \sum_{p \in P_{u,\ell}} r_p^{(p)}, \quad (3.20)$$

where $P_{u,\ell}$ is the set of paths from u to ℓ . The top-ranked POIs sorted by preference value are then recommended.

Two *key elements* need to be considered for effective and efficient preference propagation: (i) the selection of the paths among all possible paths from the target user node to POI nodes, and (ii) the algorithm for efficient preference propagation along the selected paths.

In this section, we first present the constraints in selecting paths for preference propagation, and then present an efficient algorithm named *Breadth-first Preference Propagation* (BPP) for preference propagation. Finally, we analyze the time complexity of BPP in Section 3.4.2.3.

3.4.2.1 Path Selection

There exist many possible paths between a user node and a POI node. Enumerating all possible paths is computationally expensive, and may introduce noise that will deteriorate the recommendation accuracy [BSS⁺08, XYZ⁺10]. How to select a subset of paths from all possible ones for preference propagation is a key challenge to efficient and accurate recommendation. In our design, we select paths based on the following three criteria:

- (i) The path must be a simple path, *i.e.*, there is no repeated node in a path. This constraint eliminates loop(s) in a path. For example, in Figure 3.5, $u_1 \rightarrow s_{1,1} \rightarrow \ell_2 \rightarrow s_{1,2} \rightarrow u_1 \rightarrow \dots$ is forbidden (for looping back to u_1).
- (ii) The path can contain only one visited POI node and session node of the target user. This constraint avoids generating very long propagation paths. For example, in Figure 3.5, $u_1 \rightarrow s_{1,1} \rightarrow \ell_1 \rightarrow s_{1,2} \rightarrow \ell_2$ is forbidden.
- (iii) The path terminates when an unvisited POI node is met. Without this constraint, the preference will be propagated from an unvisited POI to another unvisited POI, which will amplify the uncertainty in recommendation. In fact, this constraint is in accordance to the user-based CF method, in that only the items purchased by the users who share at least one item with the target user are considered as the candidate set.

The above three criteria determine that a valid path must be of length 3, 4 or 6, as shown in Figures 3.6.a, 3.6.b and 3.6.c, respectively. Specifically, a preference propagation always starts from a target user node (*e.g.*, u_1), and then visits one of the user's session nodes (*e.g.*, $s_{1,2}$), followed by a visited POI node (*e.g.*, ℓ_2). After that, the preference can be directly propagated to an unvisited POI (*e.g.*, ℓ_3), forming a 3-step path (illustrated in Figure 3.6.a). On the other hand, after reaching a visited POI node (*e.g.*, ℓ_2), the next node to visit in a path could be a session node of another user (*e.g.*, $s_{2,1}$, illustrated in Figures 3.6.b and 3.6.c). At this point, there are two types of propagations: one is to reach an unvisited POI node (*e.g.*, ℓ_3) and stops, which forms a 4-step path as illustrated in Figure 3.6.b; the other one is to visit the user node of that session node (*e.g.*, u_2), then to distribute the preference to the user's other session node (*e.g.*, $s_{2,2}$), and to reach an unvisited POI node (*e.g.*, ℓ_4). This propagation follows a 6-step path (see Figure 3.6.c).

In fact, the 3-step path exploits the geographical influence and solves the first problem mentioned in Section 3.4: the target user u is likely to check-in POIs that are close to her visited POIs at the time slots close to the target time. The 4-step path considers the temporal interests of other users and solves the second problem mentioned in Section 3.4: if the set of visited POIs of user u in time t_1 is similar with that of user v in time t_2 , then in t_1 , u is likely to be interested in the POIs visited by v in t_2 . Finally, the 6-step path explores the correlations between time slots: if v shares similar temporal interests with u , then v 's temporal interests that are close to t_q are also important for recommendation.

3.4.2.2 Breadth-first Preference Propagation

With all valid paths selected based on the three criteria, a straightforward propagation method is to adopt the depth-first search (DFS) strategy. Specifically, based on DFS

we propagate the preference from the target user node through all possible paths that satisfy the three criteria in Section 3.4.2.1. The propagation of a path stops when an unvisited POI node is reached. We name this algorithm *Depth-first Preference Propagation* (DPP). However, the DPP algorithm has high time complexity (See the analysis in Section 3.4.2.3) because the same edge may be visited multiple times along different paths. For example, in Figure 3.6.c, the edge e_{ℓ_1, ℓ_4} will be visited twice by following the paths $u_1 \rightarrow s_{1,1} \rightarrow \ell_1 \rightarrow \ell_4$ and $u_1 \rightarrow s_{1,2} \rightarrow \ell_1 \rightarrow \ell_4$.

Comparing with DFS strategy, the breath-first search (BFS) strategy is much more efficient for preference propagation. In BFS, a node n first collects preference from all of its precedent nodes, and then propagates the preference to its subsequent nodes of all paths involving n in a batch manner. For example, following BFS, ℓ_1 first aggregates the preference from $s_{1,1}$ and $s_{1,2}$, and then propagates its received preference to ℓ_4 via edge e_{ℓ_1, ℓ_4} . As a result, the edge e_{ℓ_1, ℓ_4} is visited only once, whereas is visited twice in DFS.

However, BFS cannot be directly applied for preference propagation, because it is designed for graph traversal and each node can be visited only once. In contrast, when propagating preference, some nodes need to be visited multiple times. For example, consider the two 6-step paths in Figure 3.6.c:

- Path p_1 : $u_1 \rightarrow s_{1,2} \rightarrow \ell_2 \rightarrow s_{2,2} \rightarrow u_2 \rightarrow s_{2,1} \rightarrow \ell_3$
- Path p_2 : $u_1 \rightarrow s_{1,2} \rightarrow \ell_2 \rightarrow s_{2,1} \rightarrow u_2 \rightarrow s_{2,2} \rightarrow \ell_4$

If we follow BFS exactly, p_2 is not valid as $s_{2,2}$ has already been visited in p_1 at the third step. As a result, it cannot be visited at the fifth step in p_2 .

To solve this problem, we relax the constraint on the times of visiting a node, and allow a node being visited multiple times. In fact, the multi-times-visiting problem only happens to the session nodes of users who share at least one POI with the target user (*e.g.*, $s_{2,2}$), because these nodes will be visited at different steps in different paths.

However, after allowing a node being visited multiple times, a new problem arises: the preference a node n' receives from a precedent node n will be propagated back to n , which violates the simple path constraint. For example, u_2 in Figure 3.6.c receives preference from $s_{2,2}$ and $s_{2,1}$, and propagates the preference to $s_{2,1}$ and $s_{2,2}$, respectively. However, if we employ BFS, a part of the preference from one node (*e.g.*, $s_{2,1}$) will be propagated back to itself, because at u_2 we cannot differentiate the preference from $s_{2,1}$ between that from $s_{2,2}$. We could solve this problem by keeping a table at each node to record the amount of preference the node receives from each of its precedent nodes, but it will increase space cost significantly and incur extra computation. In this paper, we propose a more efficient solution.

Consider a node n that propagates its preference $r_n \cdot w_{n,n'}$ to its adjacent node $n' \in A_n$. After collecting the preference from its adjacent nodes, n' propagates its preference

$r_{n'} = \sum_{n'' \in A_{n'}} r_{n''} \cdot w_{n'',n'}$ to its adjacent nodes $A_{n'}$, one of which is n . Then, the amount of preference originally propagated from n to n' needs to be excluded from the preference to be propagated from n' to n , *i.e.*, the amount of preference that n receives from n' should be:

$$\begin{aligned}
 & \left(\sum_{n'' \in A_{n'}} r_{n''} \cdot w_{n'',n'} - r_n \cdot w_{n,n'} \right) \cdot w_{n',n} \\
 = & \sum_{n'' \in A_{n'}} r_{n''} \cdot w_{n'',n'} \cdot w_{n',n} - r_n \cdot w_{n,n'} \cdot w_{n',n} \\
 = & r_{n'} \cdot w_{n',n} - r_n \cdot w_{n,n'} \cdot w_{n',n}, \tag{3.21}
 \end{aligned}$$

where the second part is the portion of preference of n that will be propagated back to itself through n' .

This inspires the following method to address the problem. We pre-subtract the amount of preference from n that will be propagated back to it through its neighbors to be visited (*e.g.*, $r_n \cdot w_{n,n'} \cdot w_{n',n}$) after propagating the preference from it (n) to its neighbors. Then, the relaxed BFS can be applied for preference propagation. For example, after performing pre-subtraction for $s_{2,1}$ in Figure 3.6.c, we can propagate the preference of u_2 to its neighbouring node $s_{2,1}$, since the invalid preference that will be propagated back to it (*i.e.*, $r_{s_{2,1}} \cdot w_{s_{2,1},u_2} \cdot w_{u_2,s_{2,1}}$) has already been subtracted.

We name the proposed propagation approach *Breadth-first Preference Propagation* (BPP), for it is designed based on BFS strategy. The algorithm is shown in Algorithm 2. Line 8 relaxes the constraint on the times of visiting a node. If a node n' is not in the set of candidate POI nodes (\tilde{L}_u), and needs to be visited in subsequent steps (not in N), we put it into the queue Q (lines 12-13), which contains the nodes to be visited. Line 14 propagates the preference from n to each of its neighbor nodes n' , and lines 15-16 perform the pre-subtraction operation, *i.e.*, if n 's neighbor n' will be visited, and n' ' preference will be propagated back to n ($n \in A_{n'}$), we subtract this part of preference from n 's preference.

3.4.2.3 Time Complexity Analysis

Time Complexity of DPP. We first look at the 6-step path for DPP. In the first step, one of target user u 's session nodes is selected, which has $|T|$ choices. Then, for each session node, we select one POI node ℓ , which has $|L^{(s)}|$ choices. Here $|L^{(s)}|$ is defined as the maximum number of POIs a session node connects to. In the third step, a session node of other users' connection with ℓ is reached, and there are $|S^{(\ell)}|$ possibilities, where $|S^{(\ell)}|$ is the maximum number of session nodes having links with a POI node. Then, the user node is visited with the cost 1, followed by other session nodes ($|T| - 1$). Finally,

Algorithm 1: Breadth-first Preference Propagation (BPP)

Input: user u , time t , GTAG graph G_t , recommendation size k
Output: Top- k POIs as recommendation results

- 1 $Q \leftarrow$ queue of nodes to be visited;
- 2 $N \leftarrow$ visited node set;
- 3 push u into Q ;
- 4 **while** Q is not empty **do**
- 5 $n \leftarrow$ pop head node of Q ;
- 6 **if** $n \in N$ **then**
- 7 **continue**;
- 8 **end**
- 9 **if** n is not a session node of those users who share at least one POI with user u **then**
- 10 add n into N ;
- 11 **end**
- 12 $\hat{r}_n \leftarrow 0$;
- 13 **foreach** $n' \in A_n$ **do**
- 14 **if** $n' \notin \tilde{L}_u$ **and** $n' \notin N$ **and** $n' \notin Q$ **then**
- 15 $Q.push(n')$;
- 16 **end**
- 17 $r_{n'} \leftarrow r_{n'} + r_n \cdot w_{n,n'}$;
- 18 **if** $n' \in Q$ **and** $n \in A_{n'}$ **then**
- 19 $\hat{r}_n \leftarrow \hat{r}_n - r_n \cdot w_{n,n'} \cdot w_{n',n}$;
- 20 **end**
- 21 **end**
- 22 $r_n \leftarrow \hat{r}_n$;
- 23 **end**
- 24 **return** top- k POIs in $L - L_u$ based on recommendation scores;

a POI node is visited from the session node, where there are $|L^{(s)}|$ choices. Based on the product rule, the complexity of DPP is $O(|T||L^{(s)}||S^{(l)}||T||L^{(s)}|)$. In the worst case where $|L^{(s)}| = |L|$, $|S^{(l)}| = |U||T|$, the complexity becomes $O(|T|^3|L|^2|U|)$. Obviously, the complexities of 3 and 4-step propagations are much lower.

Time Complexity of BPP. For the BPP method, the propagation starts from the target user node u . In the first step, all u 's session nodes are visited, so the complexity is $|T|$. The second steps propagates the preference from each u 's session node $s_{u,t}$ to the POI nodes it connects with by the cost of $|T||L^{(s)}|$. Then, the preference is transformed from u 's visited POI nodes to other users' session nodes they connect with. The cost of the third step is $|L^{(u)}||S^{(l)}|$, where $L^{(u)}$ is the maximum number of POIs a user has

visited. We define $S_\ell^{(\ell)}$ as the set of the session nodes having link with POI ℓ , and $|N_u|$ as the number of users who co-visited at least one POI with u . Then the cost of the fourth step (propagating from other users' session nodes to their corresponding user nodes) and the fifth step (propagating from the other user nodes to their session nodes) will be $|\bigcup_{\ell \in L_u} S_\ell^{(\ell)}|$ and $|N_u||T|$. Finally, the preference reaches POI nodes with the cost of $|N_u||T||L^{(s)}|$. For BPP, the total complexity of 6-step propagation is the sum of complexity of each step, which is $O(|T| + |T||L^{(s)}| + |L^{(u)}||S^{(\ell)}| + |\bigcup_{\ell \in L_u} S_\ell^{(\ell)}| + |N_u||T| + |N_u||T||L^{(s)}|)$. In the worst case ($|L^{(s)}| = |L|$, $|S^{(\ell)}| = |N_u| = |U|$, $|\bigcup_{\ell \in L_u} S_\ell^{(\ell)}| = |U||T|$), the complexity is $O(|T||U||L|)$, which is much lower than that of DPP.

Time Complexity of MS-IPF. The worst case time complexity of MS-IPF method proposed in [XYZ⁺10] is $O(|T||U||L|^2)$, which is higher than that of BPP.

3.5 Experiments

We systematically evaluate the proposed approaches and compare with the state-of-the-art methods on 2 real-world datasets. We first detail the experimental setting, then compare the effectiveness of these methods. The efficiency of different methods are studied in the end.

3.5.1 Experimental Setup

We first introduce the datasets and metrics, and then list the methods to be evaluated. In the end, we explain the design of the experiments.

Dataset. The two datasets used in our experiments have been introduced in Section 3.2.1. Following the settings of [YYLL11], for each user we randomly mark off 12.5% of his or her visited POIs as development data to tune parameters, and mark off another 25% of POIs as testing data to evaluate the effectiveness of the recommendation methods.

The densities of the training data of Foursquare and Gowalla are 6.35×10^{-3} and 9.85×10^{-4} , respectively. As expected, after splitting a day into 24 slots by hours, the data becomes much sparser. The densities of the two datasets after splitting become 4.82×10^{-4} and 6.65×10^{-5} , respectively.

Metrics. To study the effectiveness of the proposed methods, *i.e.*, how well the methods can recover the hold-off POIs in the testing data for a given user at a given time, we use four metrics, namely, *precision@N*, *recall@N*, *meanaverageprecision@N*, and *bDCG@N* (denoted by *Pre@N*, *Rec@N*, *MAP@N*, *nDCG@N*, respectively), where N is the number of recommendation results, following the work [YYLL11].

Table 3.3: The 11 Methods Evaluated in our Experiments

| Method | Description | T | G |
|----------|--|---|---|
| U | User-based CF (Section 2.1.1.1) | - | - |
| G-BPP | BPP on TAG without temporal influence | - | - |
| G | Geo. influence based rec. method [YYLL11] | | |
| UG | U with geographical influence [YYLL11] | - | ✓ |
| UT | U with Temporal preference | ✓ | - |
| TS | Two-stage method | ✓ | - |
| UTF | U with time function [DL05] | ✓ | - |
| MS-IPF | Multi Source IPF on STG [XYZ ⁺ 10] | ✓ | - |
| UCLAF | user-centered col. loc. and act. filtering [ZCZ ⁺ 10] | ✓ | - |
| LRT | Location rec. with temporal effects [GTHL13] | ✓ | - |
| SE | Spatial influence rec. with pop. Enhancement (Section 3.3.2.2) | | |
| UTE | Smoothing enhanced time-aware CF (Section 3.3.1.2) | ✓ | - |
| UTE-SE | UTE with geographical influence (Section 3.3.3) | ✓ | ✓ |
| TAG-RWR | Random Walk with Restart on TAG | ✓ | - |
| TAG-BPP | BPP on TAG. | ✓ | - |
| GTAG-BPP | BPP on GTAG (Section 3.4) | ✓ | ✓ |

Recall that both the Foursquare and Gowalla datasets have a low density, which usually results in relatively low precision and recall values [KSJ09, YYLL11]. In addition, the POIs in the test data of each user may represent only a small portion of POIs that the user may be interested in. Thus, although the relatively low precision and recall values are common and reasonable, in this chapter, we focus on the relative improvements we achieved, instead of the absolute values.

Recommendation Methods. We evaluated 11 methods as listed in Table 3.3. The ✓ mark in the table indicates whether a method utilizes temporal and/or geographical influences.

U is the basic CF method without utilizing temporal or geographical influence. We use it to evaluate the improvement brought by considering the temporal/geographical influence. **UG** is the method proposed in [YYLL11], which linearly combines the results of user-based CF and bayesian geographical model **G** to make recommendation. This method does not utilize temporal influence. **TS** is a two-stage method, which first calculates the recommendation score of each POI to user, and then multiplies it with the popularity of the POI at target time as the final score. **UTF** (user-based CF with time function) is the user-based version of the algorithm proposed in [DL05]. It estimates the similarity between users as the conventional user-based CF does, but weights the check-ins according to the gaps between their time slots and the target time slot by an exponential time function, which is also used in our method. We also tried the original

item-based method in [DL05], but very poor results were obtained; hence we choose not to report the results. **MS-IPF** [XYZ⁺10] and **LRT** [GTHL13] are the recommendation methods using STG graph and MF, detailed in Section 2.2. We also compare with the tensor based **UCLAF** (user-centered collaborative location and activity filtering) model [ZCZ⁺10]. UCLAF is proposed to recommend locations and activities. However, UCLAF does not consider how to make time-aware POI recommendations. We apply UCLAF method by utilizing Laplacian regularization terms on temporal influence, where adjacent time slots are connected to form the Laplacian matrices.

Methods **UTE** is the time-aware user-based CF described in Section 3.3.1.2. **UTE-SE** is UTE incorporating popularity enhanced geographical influence **SE** described in Section 3.3.3. **TAG-RWR**, **TAG-BPP**, and **GTAG-BPP** are all based on techniques described in Section 3.4. Specifically, TAG (temporal-aware graph) is a simplified version of GTAG by removing the POI links (*i.e.*, $E_{L,L}$). TAG is employed to fairly compare our preference propagation methods with other methods that do not consider geographical influence. The preference is propagated with Random Walk with Restart (RWR) and BPP algorithms respectively for TAG-RWR and TAG-BPP. To study the contribution of smoothing, we include **UT** (U with Temporal preference), a simplified version of UTE without the smoothing enhancement. To study the performance of our model without temporal influence, we also use a simplified version named **G-BPP**, in which each user only has 1 (24-hour) session node.

Parameters in all methods are tuned to their optimal values by grid search on the development set. For RWR, we set the stop criteria to 0.0001 L-1 distance between vectors of two successive iterations, and the restart probability is 0.15. For UTE-SE, the optimal values of α are 0.5 and 0.8 on Foursquare and Gowalla, respectively. For GTAG-BPP, the optimal values of η , H , τ and k are 0.2, 3, 0.1 and 450 on Foursquare, and are 0.09, 3, 0.01 and 450 on Gowalla.

3.5.2 Tuning Parameters

Before comparing with the baseline methods, we tune the parameters of the proposed methods using the development set and examine their impacts.

In UTE+SE, a parameter α is used to control the weights of the temporal part UTE and the spatial part SE (Equation 3.15). We tune α on the development data, and plot the average $Pre@5$ and $Rec@5$ on both datasets with different α values in Figure 3.7. From the figures, it is observed that best precision and recall are reached when $\alpha = 0.5$ and 0.8 on Foursquare and Gowalla, respectively.

In GTAG-BPP, there are four parameters, namely, η , H , τ and k in the proposed GTAG-BPP method. Because TAG is a simplified version of GTAG, we first tune η and H for TAG-BPP on the development set. Then we apply the obtained η and H to

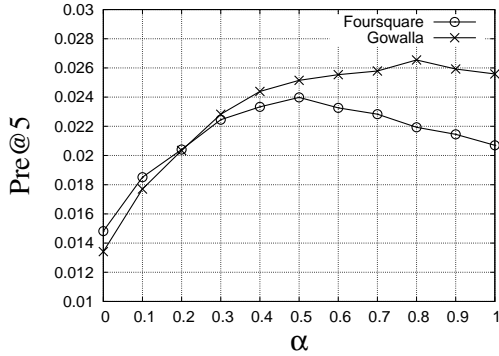
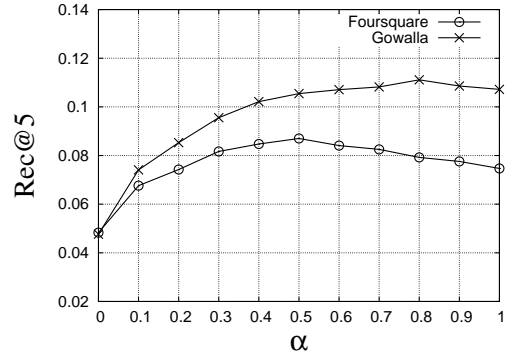

 3.7.a: $Pre@5$

 3.7.b: $Rec@5$

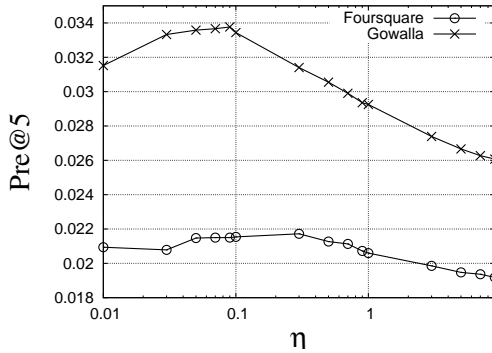
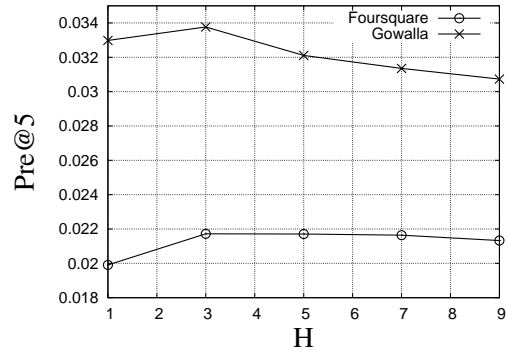
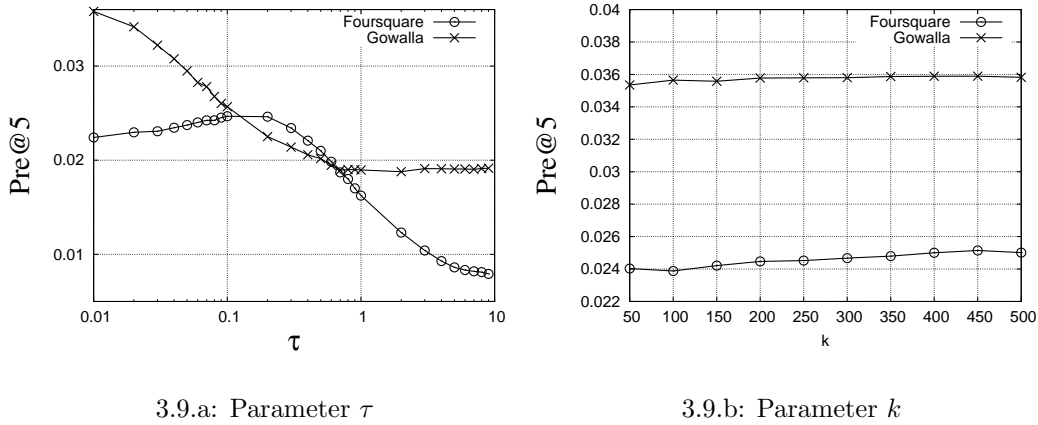
 Figure 3.7: Tuning Parameter α

 3.8.a: Parameter η

 3.8.b: Parameter H

 Figure 3.8: Impact of Parameters η and H for TAG-BPP

GTAG-BPP, and adjust τ and k . The default values of H and k are empirically set to 3 and 500, respectively, when tuning η and τ . Figures 3.8 and 3.9 show the $Pre@5$ of TAG-BPP and GTAG-BPP with varying parameter settings, respectively. We only plot $Pre@5$. Similar observations hold for $Rec@5$.

Figure 3.8 shows that the optimal value of η for Foursquare and Gowalla datasets are 0.3 and 0.09, respectively. Recall that η balances the importance of user nodes and POI nodes when propagating preference from session nodes; a smaller η means more preference will be propagated to user nodes, which will be further distributed to session nodes that are close to the target time. Gowalla dataset is much sparser than Foursquare, and utilizing other session interests can help mitigate the sparsity problem. This might explain why the optimal η for Gowalla data is smaller than that for Foursquare data.

Figure 3.9: Impact of Parameters τ and k for GTAG-BPP

Parameter H controls the importance of session nodes with respect to the target time. If H is small, the session nodes that are far from the target time will contribute less. The optimal H for both datasets is 3, and accuracy decreases as increasing H , showing that the session interests close to target time are more important. In addition, when H is smaller than 3, poorer accuracy is observed on both datasets, because small H leads to less importance of other session nodes and worsens the sparsity problem.

In GTAG-BPP, parameter τ controls the amount of preference in POI node that will be propagated to its nearby POI nodes. As shown in Figure 3.9, GTAG-BPP achieves the best accuracy when $\tau = 0.2$ and 0.01 on Foursquare and Gowalla datasets, respectively. To find out the reason for the difference, we compute the average of willingness of visiting from one POI node to its nearby POIs (*i.e.*, the sum weight of POI edges from a POI node). The value is 95.98 and 238.37 respectively on Foursquare and Gowalla datasets. A possible reason is that, in California, users' activities are less sensitive to geographical distance because most people drive cars. Consequently, users are more likely to visit distant POIs. Thus, a smaller τ can weaken the geographical influence. The best k on the two datasets are both 450.

3.5.3 Experimental Results

We conduct two sets of experiments. The first set of experiments evaluates the accuracy of the methods utilizing temporal influence (U, G-BPP, UT, UTF, TS, MS-IPF, UCLAF, LRT, UTE, TAG-RWR, TAG-BPP). The second set of experiments evaluate the effectiveness of the methods utilizing geographical influence (G, SE, UG, UTE-SE and GTAG-BPP).

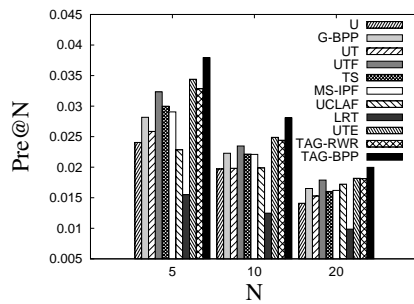
Methods Utilizing Temporal Influence. The precision and recall of the 11 methods (U, G-BPP, UT, UTF, TS, MS-IPF, UCLAF, LRT, UTE, TAG-RWR, TAG-BPP) are reported in Figure 3.10, from which we observe that:

- The proposed method TAG-BPP performs the best w.r.t. all metrics at different N values on both datasets.
- Among the memory-based CF methods, our proposed model UTE achieves the best performance, followed by UTF. UTE achieves better performance than UT, which shows the effectiveness of the proposed smoothing enhancement. TS does not consider the personalized temporal influence, so its performance is worse than that of UTF. U does not exploit temporal influence, and performs the worst. The low accuracy delivered by U suggests that time is an important factor in POI recommendation. TAG-BPP outperforms UTE w.r.t. all evaluation metrics.
- Among the graph-based CF methods, G-BPP performs the worst because it does not exploit the temporal influence. TAG-RW achieves the best accuracy, probably because of the effectiveness of the TAG graph. Compared against TAG-RW, TAG-BPP improves $Pre@5$ by 15% and 10% on the two datasets, respectively. In addition, TAG-BPP beats MS-IPF by 19% and 13% w.r.t. $Pre@5$ on the two datasets. Note that TAG-DPP returns the same results as does TAG-BPP.
- The performance of LRT is not satisfactory, probably because MF method does not handle the datasets in low density well. The bad result is also in accordance with that reported in [NSLM12]. The performance of UCLAF is worse than other baselines, probably because it approximates the check-in tensor by fitting both zero and nonzero entries during the factorization. However, some zero entries in the tensor are actually missing values, and fitting them directly is not reasonable.

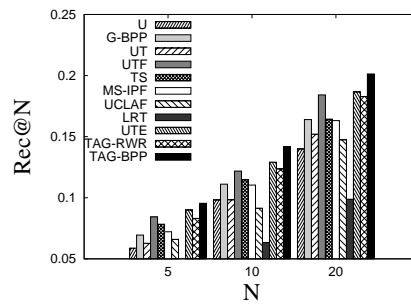
Methods Utilizing Geographical Influence. The precision and recall of the 6 methods utilizing geographical influence (G, SE, UG, UTE-SE, and GTAG-BPP) are plotted in Figure 3.11.

G and SE only exploit the geographical information. Between the two methods, SE achieves much better recommendation accuracy than G, which demonstrates the effectiveness of the integration of temporal popularity.

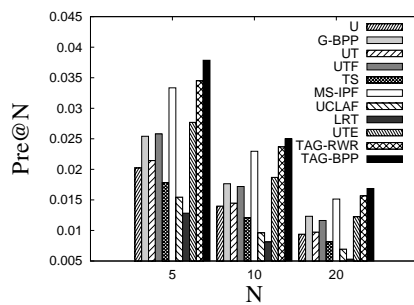
Among the remaining methods, UG delivers the worst results because it does not exploit temporal influence. Our proposed method UTE-SE achieves much better accuracy than UG. GTAG-BPP achieved the best accuracy among the three methods. Compared against TAG-BPP (reported in Figure 3.10), GTAG-BPP further improves the accuracy by 4% to 10%. The improvement results from the incorporation of geographical influence.



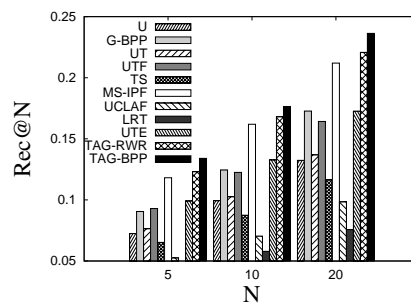
3.10.a: $Pre@N$ - Foursquare



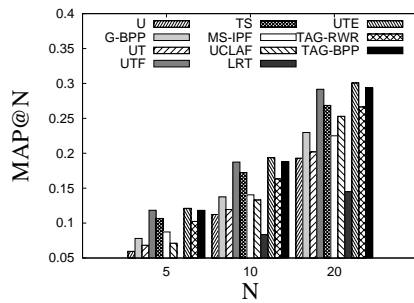
3.10.b: $Rec@N$ - Foursquare



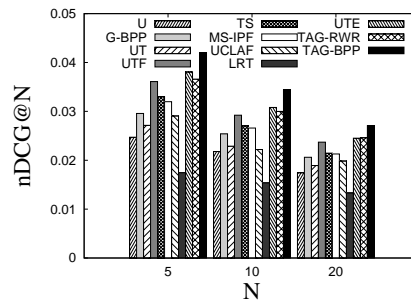
3.10.c: $Pre@N$ - Gowalla



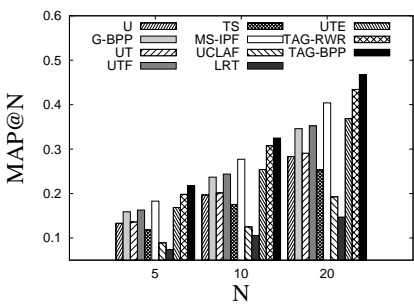
3.10.d: $Rec@N$ - Gowalla



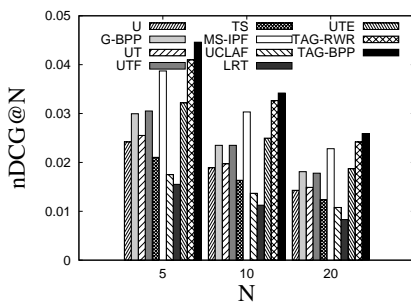
3.10.e: $MAP@N$ - Foursquare



3.10.f: $nDCG@N$ - Foursquare

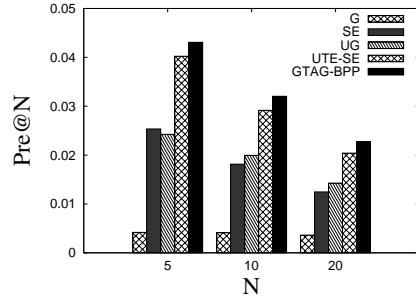


3.10.g: $MAP@N$ - Gowalla

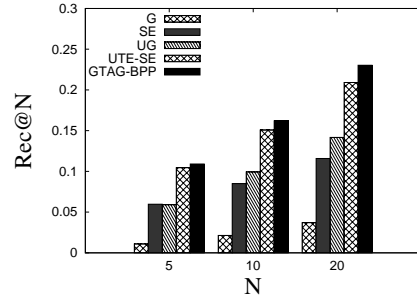


3.10.h: $nDCG@N$ - Gowalla

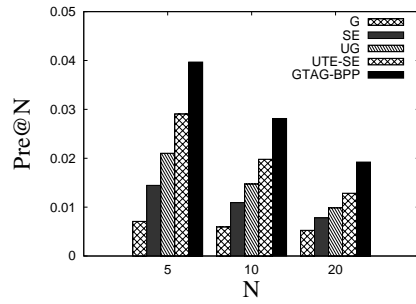
Figure 3.10: Performance of Methods utilizing Temporal Influence



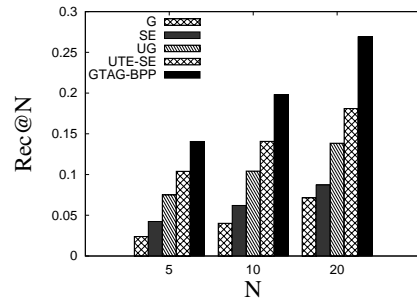
3.11.a: $Pre@N$ - Foursquare



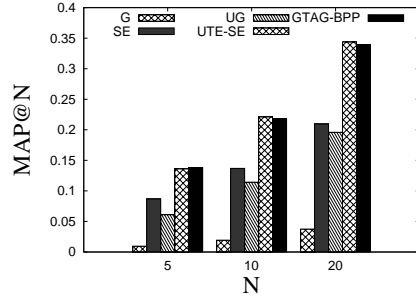
3.11.b: $Rec@N$ - Foursquare



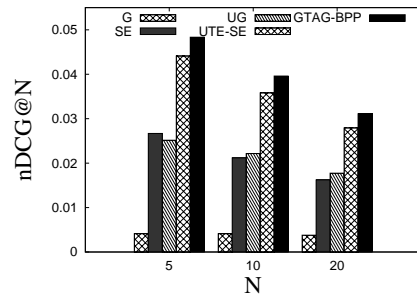
3.11.c: $Pre@N$ - Gowalla



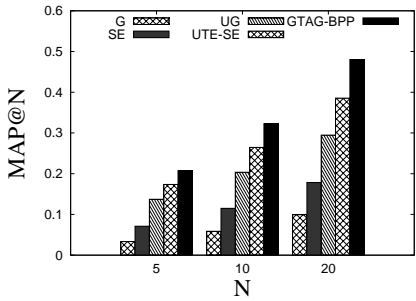
3.11.d: $Rec@N$ - Gowalla



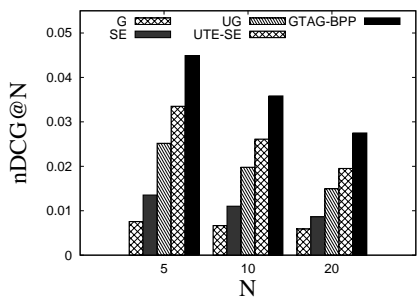
3.11.e: $MAP@N$ - Foursquare



3.11.f: $nDCG@N$ - Foursquare

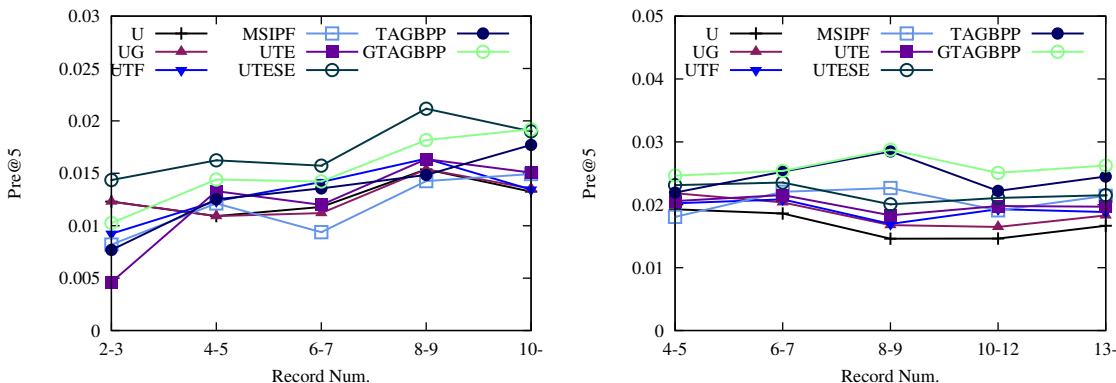


3.11.g: $MAP@N$ - Gowalla



3.11.h: $nDCG@N$ - Gowalla

Figure 3.11: Performance of Methods with Geographical Influence



3.12.a: Foursquare

3.12.b: Gowalla

Figure 3.12: Sensitivity to Sparsity Problem

Note that all improvements reported in this sections are significant according to t-test with p-value < 0.01 .

We also study the sensitivity of different methods to the sparsity problem. Specifically, we group users into bins based on the numbers of check-ins the users have made, and compare the effectiveness of U, UG, UTF, MSIPF, UTE, UTESE, TAGBPP and GTAGBPP. The $Pre@5$ and $Rec@5$ of these methods on the two datasets are plotted in Figure 3.12. From the figure, we can observe that the proposed model TAGBPP and GTAGBPP outperform other baselines for users in all bins, which show that these two methods are less sensitive to the sparsity problem. The performance of UTE and UTF is similar, because both of them exploit the smoothing technique to make the check-in matrix densier.

3.5.4 Efficiency of Methods

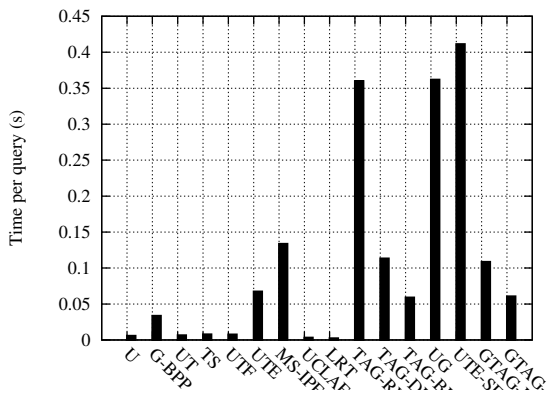
Next, we look into the efficiency of different methods. We plot the average running time of a recommendation query of all 16 methods on the Gowalla dataset in Figure 3.13.a. Since the Gowalla dataset has much more recommendation queries (34,353) comparing with the Foursquare dataset, we choose it to evaluate the efficiency.

- LRT, UCLAF, U, UT, TS, and UTF are more efficient due to their simple MF, TF or CF techniques. However, their accuracy is among the worst. UTE is slower than UTF, because the smoothing techniques in UTE calls for additional computation. MS-IPF is even much slower, which is consistent with the complexity analysis(See Section 3.4.2.3). G-BPP is more efficient than TAG-BPP, since its graph contains fewer nodes and edges. However, its accuracy is much worse than that of TAG-BPP.

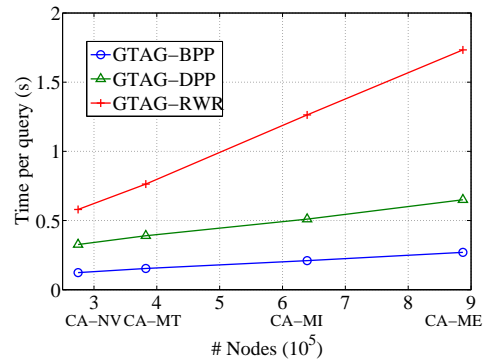
- Among methods based on TAG (*i.e.*, TAG-RWR, TAG-DPP, and TAG-BPP), TAG-RWR is the slowest because it needs a number of iterations to converge. TAG-DPP is much faster than TAG-RWR, since it employs preference propagation, and stops in a fix number of steps. TAG-BPP achieves the best efficiency for its BFS-based propagation strategy avoids multiple propagation on each edge.
- Among the methods exploiting geographical influence, UG and UTE-SE are much slower, because the computation of distance between each candidate POI and each visited POI of the target user is costly. The additional time required by BPP and DPP on GTAG is very small.

To examine the scalability of the proposed methods with the size of TAG, we extract another four datasets from Gowalla [CML11] based on region, namely, CA-NV, CA-MT, CA-MI, and CA-ME, which cover the area in the bounding boxes with *the southwest point of California as the lower left point, and the northeast point of Nevada, Montana, Michigan and Maine as the upper right point, respectively*. Among the four datasets, the former ones are contained in the latter ones. The information of each dataset after removing users who checked-in fewer than 5 POIs and POIs that were checked-in by fewer than 5 users is shown in Table 3.4.

We randomly select 300 user-time pairs as queries, and report the average running time of a query for GTAG-BPP, GTAG-DPP and GTAG-RWR on each dataset in Figure 3.13.b. We observe that GTAG-BPP has the best scalability.



3.13.a: Methods on Gowalla dataset



3.13.b: Regions in different scale

Figure 3.13: Efficiency of methods on Gowalla dataset, and the three methods BPP, DPP, RWR on different regions

Table 3.4: Statistics on the CA-{NV, MT, MI, ME} Datasets

| Dataset | # Check-ins | # Users | # POIs | # Nodes |
|----------------|--------------------|----------------|---------------|----------------|
| CA-NV | 418,842 | 10,657 | 18,947 | 274,715 |
| CA-MT | 572,122 | 14,827 | 26,673 | 382,521 |
| CA-MI | 986,349 | 24,668 | 47,126 | 639,158 |
| CA-ME | 1,342,131 | 34,250 | 65,029 | 887,029 |

3.6 Summary

In this chapter, we present the first analysis of temporal influence to users' check-in behaviors, and observe that users tend to visit different POIs at different time, and the check-in behavior of a user is daily periodic. To exploit the time information, we define a novel problem time-aware POI recommendation, which is an extension of the conventional POI recommendation problem that considers the temporal influence in user activities. To solve this problem, we propose two approaches, namely, UTE+SE and GTAG-BPP, to exploit both the temporal influence and spatial influence. UTE+SE utilizes the temporal and spatial influences by user-based CF and Bayes model, respectively, and generates the recommendation results by a linear combination of the two component. In contrast, GTAG-BPP is an integrated graph based model that incorporates the two influences simultaneously. As a result, it can model the sophisticated interactions of the two influences and provide more accurate recommendations. We conduct extensive experiments over two real-world LBSN datasets, namely, Foursquare check-ins and Gowalla check-ins. The experimental results show that the time information contributes a lot to the recommendation accuracy, and our proposed UTE-SE outperforms all baselines significantly. In addition, our second model, GTAG-BPP further improves the recommendation accuracy against UTE-SE. In addition, the GTAG-BPP shows good efficiency performance among the methods compared.

Chapter 4

Group POI Recommendation

With the rapid development of online social networks, a growing number of people are willing to share their group activities, *e.g.*, having dinners with colleagues, and having picnics with friends. This motivates the studies on group POI recommendation, which aims to recommend POIs for a group of users. Group recommendation is a challenging problem because different group members have different preferences, and how to make a trade-off among their preferences for recommendation is still an open problem.

Previous solutions to group recommendation can be divided into two types: memory-based and model-based approaches. Memory-based approaches further fall in two categories based on the aggregation strategy: *preference aggregation* strategy first aggregates the profiles of group members into a new profile, and then employs recommendation techniques designed for individuals to make group recommendations [MA98, YZHG06]; *score aggregation* strategy first calculates a recommendation list for each group member, and then aggregates these lists for group recommendation [OCKR01, McC02, CBH02, PD-CCA05, BMR10]. However, both strategies overlook the interactions between group members, and use trivial methods to aggregate members' preferences. Different from memory-based approaches, model-based methods exploit the interactions among members by modeling the generative process of a group [YLL12, LTYL12, GLRW13]. Among them, the model proposed in [YLL12] requires each group consists of friends, and the group members influence the choice of each other even in a large group. Obviously, the assumptions are too strong for groups in real-life. The PIT model proposed in [LTYL12] assumes that each user has an impact score, and those with great scores are like to be representatives of groups, who select the items for the groups based on their own topic preferences. Obviously, these assumptions may not hold in real world, because users' influences should be topic-dependent. For example, a user who is an expert in movie may not be influential in a dining group. The information matching based model proposed in [GLRW13] suffers from two shortcomings: first, its time complexity is too high to generate results in real time; second, it cannot provide explanations to the recommendations,

because the model is build on latent dimensions. In addition, none of these model-based approaches can incorporate content information like geographical coordinates of POIs for recommendation.

In this chapter, we propose a probabilistic model named COM (COnsensus Model) to model the generative process of group activities, and make group POI recommendations. Intuitively, users in a group may have different influences, and those who are expert in topics relevant to the group are usually more influential. In addition, users in a group may behave differently as group members from that as individuals. COM is designed based on these intuitions, and is able to incorporate both users' selection history and personal considerations of content factors. When making recommendations, COM estimates the preference of a group to a POI by aggregating the preferences of the group members with different weights. Besides recommending POIs, our proposed model can be adapted to other recommendation tasks, *e.g.*, movie recommendation. We conduct extensive experiments on four datasets, and the results show that the proposed model is effective in making group recommendations, and outperforms baseline methods significantly.

The rest of this chapter is organized as follows. Section 4.1 introduces the proposed COM model and the recommendation method. We present experimental results in Section 4.2. Finally, Section 4.3 concludes this chapter.

4.1 Model Description

We first define the group POI recommendation problem in Section 4.1.1, and then introduce the proposed COnsensus Model (COM) in Section 4.1.2. After that, the inference algorithm and the recommendation method are presented in Section 4.1.3 and 4.1.4, respectively. Finally, we present how to adapt the proposed model for the movie recommendation tasks in Section 4.1.5. All the notations used in this chapter are listed in Table 4.1.

4.1.1 Problem Statement

Let U , I , G be the user, POI and group sets, respectively. A *group* $g \in G$ consists of a set of users (group members) $\mathbf{u}_g = \{u_{g,1}, u_{g,2}, \dots, u_{g,|g|}\}$, where $u_{g,i} \in U$, and $|g|$ is the size of the group, *i.e.*, the number of users in g . In addition, a group g is associated with a POI $i_g \in I$, if i_g is selected by group g . We define a *group event* by $\langle \mathbf{u}_g, i_g \rangle$, *i.e.*, the POI selection event of the group members as a whole. For example, the members of a picnic group selecting a POI for picnic is a group event.-

Then, given a target group g_t , the problem of group recommendation is defined as recommending a list of POIs that users in g_t may be interested in.

Table 4.1: Symbols for Group POI Recommendation

| Symbol | Description |
|-----------------------------|---|
| U, I, G | user set, POI set, group set |
| K | the number of latent topics |
| N | the size of recommendation list |
| $g, g $ | the group g , the size of group g |
| \mathbf{u}_g | the members in group g |
| i_g | the POI selected by group g |
| u | user $u \in U$ |
| z_j, c_j | the topic and switch of the user-POI pair j |
| $\mathbf{i}^{(\cdot)}$ | the set of POIs that are generated when $c = (\cdot)$ |
| θ_g | distribution of topics specific to group g |
| ϕ_z^{ZU} | distribution of users specific to topic z |
| ϕ_z^{ZI} | distribution of POIs specific to topic z |
| ϕ_u^{UI} | distribution of POIs specific to user u |
| λ_u | the parameter of Bernoulli distribution specific to user u for sampling the binary switch c |
| $\alpha, \beta, \rho, \eta$ | Dirichlet prior vector for $\theta, \phi^{ZU}, \phi^{UI}$ and ϕ^{ZI} |
| γ | Beta prior for λ , where $\gamma = \{\gamma, \gamma_t\}$ |
| $n_{g,z,\neg j}^{GZ}$ | number of times topic z is assigned to group g , excluding the j^{th} user-POI pair |
| $n_{z,u,\neg j}^{ZU}$ | number of times user u is drawn from topic z , excluding the j^{th} user-POI pair |
| $n_{z,i,\neg j}^{ZI}$ | number of times POI i is drawn from topic z , excluding the j^{th} user-POI pair |
| $n_{u,i,\neg j}^{UI}$ | number of times POI i is drawn from user u , excluding the j^{th} user-POI pair |
| $n_{u,c,\neg j}^{UC}$ | number of times switch c is drawn for user u , excluding the j^{th} user-POI pair |

4.1.2 Consensus Model for Group POI Recommendation

We model the generative process of a group event based on the following intuitions:

- **Intuition 1:** Each group is relevant to several topics with different degrees of match, *e.g.*, a picnic group is more relevant to the hiking and dining topics than to the body-building topic. The topics of a group attract users to join the group.
- **Intuition 2:** When selecting a POI, users in a group have two considerations. The first is topics, *i.e.*, a user tends to select the POIs that are related to the group topic, which attracted her to join the group. The second is users' personal considerations of content factors, such as the geographical distance to a POI. These

factors are user-specific, and cannot be captured by topics. In addition, different users make different trade-offs between group topics and personal considerations of content factors: some users tend to select the POIs that match the group topics best, while some may select POIs near them.

- **Intuition 3:** Users behave differently when selecting POIs as members in a specific group and when selecting POIs as individuals. In a group, a user tends to match her preference to the topics to the topics of the group.
- **Intuition 4:** The preference of a group to a candidate POI is determined by the preferences of the group members [AYRC⁺09, GLRW13]. In addition to this, we exploit the following new intuition: the influence of each member on the POI selection of the group is topic-dependent.

These intuitions can solve the weaknesses of existing group recommendation methods. First, since groups may often happen to be formed, we do not assume any friendships and pair-wise influences among the group members. Thus, the problems of [YLL12] do not exist in our model. Second, we assume each group, instead of user, has a topic proportion, and the topics of the group attract users to join. Thus, if a user is an expert in the relevant topics of a group, the user is more influential in making item selection. In other word, the user influence is topic dependent, which is different from the assumption of [LTYL12]. In addition, when selecting item, a user will either consider the group topics and the content information. Thus, the selected items will be relevant to the group instead of the influential users, and the geographical information can be easily incorporated. To verify this intuition, on the Jiepang dataset (to be detailed in the experiments) we represent each POI as a vector of visitors, and study the average pairwise similarity between POIs visited by users as individuals (individual check-ins), and the average pairwise similarity between POIs visited by users as group members (group check-ins). We find the similarity of POIs of individual check-ins is 0.0194, while the similarity of group check-ins is only 0.01034. The difference between the similarity values may be caused by the behavior changes of users. Last but not least, our model is built under the framework of topic models, thus the recommendation results are interpretable, and the online recommendation is very efficient.

We use a multinomial distribution θ_g over latent topics to model the topic preferences of group g (**Intuition 1**). In addition, each latent topic z has a multinomial distribution ϕ_z^{ZU} over user set, which represents the relevance of users to the topic z , and a multinomial distribution ϕ_z^{ZI} over POI set, which represents the relevance of POIs to the topic z . Here $\phi_{z,i}^{ZI}$ reflects given a topic z , how likely the POI i is selected; $\phi_{z,u}^{ZU}$ reveals the appealing degree of topic z to the user u , or the user u 's expertise on topic z . To model **Intuition 1** that users join a group because of different topics, for each member in group g , a latent

topic z is sampled from its topic distribution θ_g , and then a user u is drawn according to ϕ_z^{ZU} .

A user in a group selects POIs either based on the group topics that attracted her to join the group, or her personal considerations of content factors such as distance (**Intuition 2**). We use a switch c to decide which one accounts for the POI selection of a user, *i.e.*, if $c = 1$, the POI is sampled based on the topic-specific multinomial distribution over POIs ϕ^{ZI} ; if $c = 0$, the POI is drawn from the user-specific multinomial distribution of POIs ϕ^{UI} . Since different users will make different trade-offs between group topics and distance (**Intuition 2**), in our model, the switch c is drawn from a user-specific Bernoulli distribution with parameter λ_u . In other words, user u is influenced by group topics with probability λ_u , and is influenced by distance with probability $1 - \lambda_u$. Note that the Bernoulli distribution for λ_u has a Beta prior $\gamma = \{\gamma, \gamma_t\}$.

Next we illustrate the model using an example. Suppose a picnic group is more relevant to both hiking and dining topics than body-building topic. The three topics are sampled from the topic distribution of the picnic group, which attract users u_1 , u_2 and u_3 , respectively. Then, these three users determine which POI to visit based on the group topics and their personal considerations of content factors such as traveling distance. Suppose u_1 does not mind traveling, and the topic “hiking” has a more significant influence to his selection. Then she may select a distant POI that matches the hiking topic best. Thus, the switch c for u_1 is more likely to be 1. u_2 and u_3 will also make trade-offs between group topics and distance to select the POI for picnic.

Different from COM, previous topic model based approaches [YLL12, LTYL12] assume that when selecting items, a group member only considers her own topic preference. The assumption may not hold, because users in a group may behave differently as group members from that when they make choices as individuals (**Intuition 3**). For example, suppose u_1 is interested in both hiking and movie topics. In previous approaches, u_1 may select a theater for the picnic group because of her interest in movie topic. In contrast, in our model, u_1 join the picnic group because of the hiking topic, and thus her selection will be related to hiking rather than movie.

In summary, the generative process of a collection of group events is as follows:

- For each topic z_k , $k = 1, \dots, K$
 - Draw $\phi_k^{ZU} \sim \text{Dirichlet}(\beta)$;
 - Draw $\phi_k^{ZI} \sim \text{Dirichlet}(\eta)$;
- For each user u_v , $v = 1, \dots, |U|$
 - Draw $\phi_v^{UI} \sim \text{Dirichlet}(\rho)$;

- Draw $\lambda_v \sim \text{Beta}(\gamma)$;
- For each group g
 - Draw $\boldsymbol{\theta}_g \sim \text{Dirichlet}(\boldsymbol{\alpha})$;
 - For each group member
 - * Draw $z \sim \text{Multinomial}(\boldsymbol{\theta}_g)$;
 - * Draw $u \sim \text{Multinomial}(\boldsymbol{\phi}_z^{ZU})$;
 - * Draw switch $c \sim \text{Bernoulli}(\lambda_u)$;
 - * If $c = 0$
 - Draw $i \sim \text{Multinomial}(\boldsymbol{\phi}_u^{UI})$;
 - * If $c = 1$
 - Draw $i \sim \text{Multinomial}(\boldsymbol{\phi}_z^{ZI})$;

The graphical model is shown in Figure 4.1.

For POI recommendation, geographical distance is an important factor to consider. Previous studies reported that users tend to visit nearby POIs, and the willingness of visiting a POI decreases with the increase of distance from their current locations. We can incorporate the user-specific influence of geographical distance into our model by modifying the Dirichlet prior $\boldsymbol{\rho}$ to $\boldsymbol{\phi}_u^{UI}$. Here, we adopt a power-law function of distance to model the willingness $wi(d)$ of a user moving from one POI to another d km far away POI as Chapter 3.3.2 does.

Then, given a user u , the set of POIs that she has visited I_u , we calculate $P(i|I_u)$ for each candidate POI i according to the geographical distance, and use this value for $\rho_{u,i}$. Based on the Bayes rule, $P(i|I_u)$ is calculated as follows:

$$\begin{aligned} \rho_{u,i} = P(i|I_u) &\propto P(i)P(I_u|i) \\ &= P(i) \prod_{i' \in I_u} P(i'|i) \end{aligned} \quad (4.1)$$

where $P(i'|i)$ is proportional to the willingness value in Equation 3.6, in which d is the distance between POIs i' and i .

Note that different users in a group will sample different POIs in the model, which is in accordance with our experience: users may have different preferences over POIs, and thus are likely to make different choices. In fact, the POI selection of a group is often made by two steps: group members express their own opinions on POI selections first, and then these selections are weighted and a consensus is reached. As to be detailed in Section 4.1.4, we propose a recommendation method that can aggregate the selections of group members based on their topic-dependent influences, and generate a single recommendation for the target group.

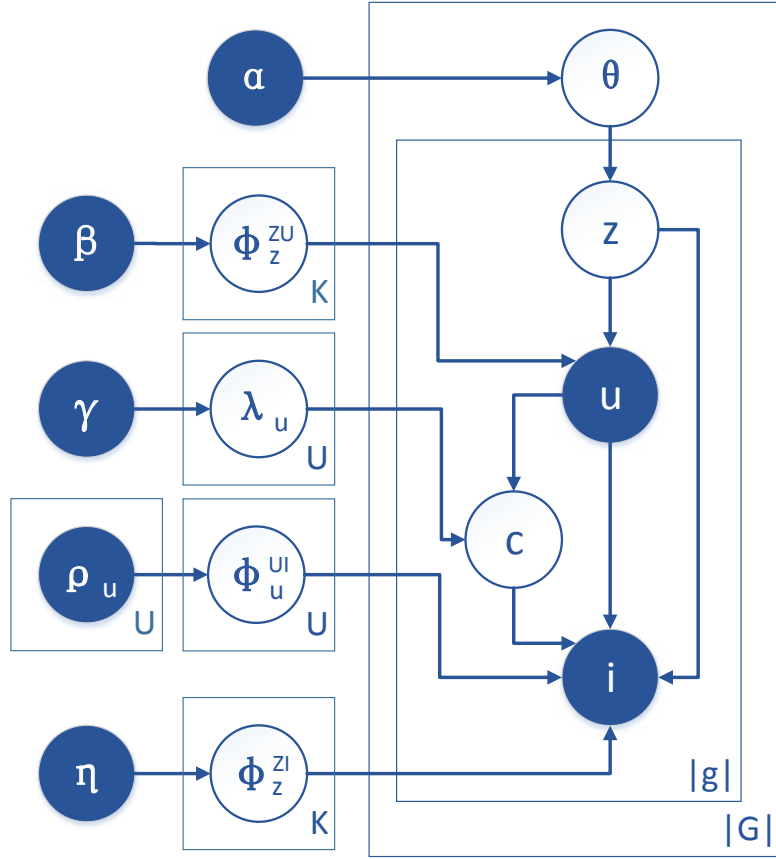


Figure 4.1: The Graphical Model of COM

We remark that the aforementioned generative process is also applicable to the groups with pre-defined members, since these groups also have topic distributions. Consider some students plan to form a club group and the topics of the group are dining, hiking, etc. The group is formed because its topics attract the members. If someone is not interested in any of the group topics, she will not join the group. Thus, the generative process of the club can also be explained by the proposed model, where the group members are sampled from the students.

4.1.3 Parameter Estimation

The total likelihood of the group event corpus is:

$$\begin{aligned}
& P(\mathbf{z}, \mathbf{u}, \mathbf{c}, \mathbf{i} | \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\rho}, \boldsymbol{\eta}, \boldsymbol{\gamma}) \\
= & \int P(\mathbf{c} | \boldsymbol{\lambda}) P(\boldsymbol{\lambda} | \boldsymbol{\gamma}, \boldsymbol{\gamma}_t) d\boldsymbol{\lambda} \cdot \int P(\mathbf{z} | \boldsymbol{\theta}) P(\boldsymbol{\theta} | \boldsymbol{\alpha}) d\boldsymbol{\theta} \cdot \\
& \int P(\mathbf{u} | \mathbf{z}, \boldsymbol{\phi}^{ZU}) P(\boldsymbol{\phi}^{ZU} | \boldsymbol{\beta}) d\boldsymbol{\phi}^{ZU} \cdot \\
& \int \int P(\mathbf{i} | \mathbf{u}, \mathbf{z}, \mathbf{c}, \boldsymbol{\phi}^{UI}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{UI} | \boldsymbol{\rho}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{UI} d\boldsymbol{\phi}^{ZI} \quad (4.2)
\end{aligned}$$

We employ collapsed Gibbs sampling to obtain samples of the hidden variable assignment, and to estimate the unknown parameters $\{\boldsymbol{\lambda}, \boldsymbol{\phi}^{ZU}, \boldsymbol{\phi}^{UI}, \boldsymbol{\phi}^{ZI}\}$. For ease of presentation, we define a user u together with the POI i selected by u as a user-POI pair $j = (u, i)$, where the user of j is $u \in U$, and the POI of j is $i \in I$.

Since there are two latent variables in the model, namely, z and c , we employ two-step Gibbs sampling method. We first sample topics z_j for all user-POI pairs j , and then sample switches c_j for all j . For each latent variable (*e.g.*, z_j), a Gibbs sampling method computes the full conditional probability for the assignment of the variable conditioned on all the other assignments (*e.g.*, \mathbf{z}_{-j}). However, it is challenging to get the full conditional probability because of the complex interdependencies between user u , topic z , switch c and POI i : u is sampled based on z , which influences the sampling of c , while i is sampled based on either z or u depending on c .

To solve this problem, we separate the POIs generated based on topics, and the POIs generated based on users' personal considerations of content factors. Then, the last part of Equation 4.2 becomes:

$$\begin{aligned}
& \int \int P(\mathbf{i} | \mathbf{u}, \mathbf{z}, \mathbf{c}, \boldsymbol{\phi}^{UI}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{UI} | \boldsymbol{\rho}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{UI} d\boldsymbol{\phi}^{ZI} \\
= & \int P(\mathbf{i}^{(0)} | \mathbf{u}, \mathbf{c}, \boldsymbol{\phi}^{UI}) P(\boldsymbol{\phi}^{UI} | \boldsymbol{\rho}) d\boldsymbol{\phi}^{UI} \cdot \\
& \int P(\mathbf{i}^{(1)} | \mathbf{z}, \mathbf{c}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{ZI} \quad (4.3)
\end{aligned}$$

where $\mathbf{i}^{(0)}$ is the set of POIs that are sampled based on users' personal considerations of content factors (*i.e.*, $c = 0$), and $\mathbf{i}^{(1)}$ is the set of POIs that are sampled based on topics (*i.e.*, $c = 1$).

Based on the new equation of total likelihood, we can derive the full conditional distribution of topic z_j and switch c_j assignments for each user-POI pair j . If the POI

of j is drawn based on topics, *i.e.*, $c_j = 1$, we sample z_j according to the following probability:

$$\begin{aligned}
 & P(z_j = k | \mathbf{z}_{-j}, \mathbf{u}, \mathbf{i}^{(1)}) \\
 = & \frac{\int P(\mathbf{z} | \boldsymbol{\theta}) P(\boldsymbol{\theta} | \boldsymbol{\alpha}) d\boldsymbol{\theta}}{\int P(\mathbf{z}_{-j} | \boldsymbol{\theta}) P(\boldsymbol{\theta} | \boldsymbol{\alpha}) d\boldsymbol{\theta}} \cdot \frac{\int P(\mathbf{u} | \mathbf{z}, \boldsymbol{\phi}^{ZU}) P(\boldsymbol{\phi}^{ZU} | \boldsymbol{\beta}) d\boldsymbol{\phi}^{ZU}}{\int P(\mathbf{u} | \mathbf{z}_{-j}, \boldsymbol{\phi}^{ZU}) P(\boldsymbol{\phi}^{ZU} | \boldsymbol{\beta}) d\boldsymbol{\phi}^{ZU}} \cdot \\
 & \frac{\int P(\mathbf{i}^{(1)} | \mathbf{z}, \mathbf{c}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{ZI}}{\int P(\mathbf{i}^{(1)} | \mathbf{z}_{-j}, \mathbf{c}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{ZI}} \\
 \propto & \frac{n_{g_j, k, -j}^{GZ} + \alpha_k}{\sum_{k' \in Z} (n_{g_j, k', -j}^{GZ} + \alpha_{k'})} \cdot \frac{n_{k, u, -j}^{ZU} + \beta_u}{\sum_{u' \in U} (n_{k, u', -j}^{ZU} + \beta_{u'})} \cdot \frac{n_{k, i, -j}^{ZI} + \eta_i}{\sum_{i' \in I} (n_{k, i', -j}^{ZI} + \eta_{i'})} \quad (4.4)
 \end{aligned}$$

where g_j is the group of j . If the POI of j is drawn based on user's personal considerations of content factors, *i.e.*, $c_j = 0$, we have:

$$\begin{aligned}
 & P(z_j = k | \mathbf{z}_{-j}, \mathbf{u}, \mathbf{i}^{(0)}) \\
 \propto & \frac{n_{g_j, k, -j}^{GZ} + \alpha_k}{\sum_{k' \in Z} (n_{g_j, k', -j}^{GZ} + \alpha_{k'})} \cdot \frac{n_{k, u, -j}^{ZU} + \beta_u}{\sum_{u' \in U} (n_{k, u', -j}^{ZU} + \beta_{u'})} \quad (4.5)
 \end{aligned}$$

After sampling topics for all user-POI pairs, we draw a switch c_j for each j according to the following posterior probability. When $c_j = 1$, we have:

$$\begin{aligned}
 & P(c_j = 1 | \mathbf{c}_{-j}, \mathbf{z}, \mathbf{u}, \mathbf{i}) \\
 = & \frac{\int P(\mathbf{c} | \boldsymbol{\lambda}) P(\boldsymbol{\lambda} | \gamma, \gamma_t) d\boldsymbol{\lambda}}{\int P(\mathbf{c}_{-j} | \boldsymbol{\lambda}) P(\boldsymbol{\lambda} | \gamma, \gamma_t) d\boldsymbol{\lambda}} \cdot \frac{\int P(\mathbf{i}^{(0)} | \mathbf{u}, \mathbf{c}, \boldsymbol{\phi}^{UI}) P(\boldsymbol{\phi}^{UI} | \boldsymbol{\rho}) d\boldsymbol{\phi}^{UI}}{\int P(\mathbf{i}^{(0)} | \mathbf{u}, \mathbf{c}_{-j}, \boldsymbol{\phi}^{UI}) P(\boldsymbol{\phi}^{UI} | \boldsymbol{\rho}) d\boldsymbol{\phi}^{UI}} \cdot \\
 & \frac{\int P(\mathbf{i}^{(1)} | \mathbf{z}, \mathbf{c}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{ZI}}{\int P(\mathbf{i}^{(1)} | \mathbf{z}, \mathbf{c}_{-j}, \boldsymbol{\phi}^{ZI}) P(\boldsymbol{\phi}^{ZI} | \boldsymbol{\eta}) d\boldsymbol{\phi}^{ZI}} \quad (4.6)
 \end{aligned}$$

Note that since $c_j = 1$, the second term in the right hand side of Equation 4.6 is 1. Thus, we cancel this part, and get:

$$\begin{aligned}
 & P(c_j = 1 | \mathbf{c}_{-j}, \mathbf{z}, \mathbf{u}, \mathbf{i}) \\
 \propto & \frac{n_{u, (1), -j}^{UC} + \gamma}{n_{u, (0), -j}^{UC} + n_{u, (1), -j}^{UC} + \gamma + \gamma_t} \cdot \frac{n_{z_j, i, -j}^{ZI} + \eta_i}{\sum_{i' \in I} (n_{z_j, i', -j}^{ZI} + \eta_{i'})} \quad (4.7)
 \end{aligned}$$

Similarly, we calculate the sampling probability for $c_j = 0$:

$$P(c_j = 0 | \mathbf{c}_{-j}, \mathbf{z}, \mathbf{u}, \mathbf{i}) \propto \frac{n_{u,(0),-j}^{UC} + \gamma_t}{n_{u,(0),-j}^{UC} + n_{u,(1),-j}^{UC} + \gamma + \gamma_t} \cdot \frac{n_{u,i,-j}^{UI} + \rho_i}{\sum_{i' \in I} (n_{u,i',-j}^{UI} + \rho_{i'})} \quad (4.8)$$

After sampling a sufficient number of iterations, we calculate the parameters ϕ^{ZU} , ϕ^{UI} , ϕ^{ZI} and λ as follows:

$$\hat{\phi}_{z,u}^{ZU} = \hat{P}(u|z) = \frac{n_{z,u}^{ZU} + \beta_u}{\sum_{u' \in U} (n_{z,u'}^{ZU} + \beta_{u'})} \quad (4.9)$$

$$\hat{\phi}_{u,i}^{UI} = \hat{P}(i|u) = \frac{n_{u,i}^{UI} + \rho_i}{\sum_{i' \in I} (n_{u,i'}^{UI} + \rho_{i'})} \quad (4.10)$$

$$\hat{\phi}_{z,i}^{ZI} = \hat{P}(i|z) = \frac{n_{z,i}^{ZI} + \eta_i}{\sum_{i' \in I} (n_{z,i'}^{ZI} + \eta_{i'})} \quad (4.11)$$

$$\hat{\lambda}_u = \hat{P}(c = 1|u) = \frac{n_{u,(1)}^{UC} + \gamma}{n_{u,(0)}^{UC} + n_{u,(1)}^{UC} + \gamma + \gamma_t} \quad (4.12)$$

4.1.4 Recommendation

When making recommendations for a target group g_t , we first discover its topic distribution based on the group members \mathbf{u}_{g_t} . The distribution, denoted by θ_{g_t} , can be learnt by performing Gibbs sampling on \mathbf{u}_{g_t} according to the following equation:

$$P(z_j = k | \mathbf{z}_{-j}, u_j = v, \mathbf{u}_{-j}) \propto \hat{\phi}_{k,v}^{ZU} (n_{g_t,k,-j}^{GZ} + \alpha_k) \quad (4.13)$$

Since the recommendations should match the topic distribution θ_{g_t} , based on the generative model, we define the recommendation score for candidate POI i as follows:

$$P(i | \mathbf{u}_{g_t}, \theta_{g_t}) \triangleq \prod_{u \in \mathbf{u}_{g_t}} \sum_{z \in Z} \theta_{g_t,z} \cdot \hat{\phi}_{z,u}^{ZU} (\hat{\lambda}_u \cdot \hat{\phi}_{z,i}^{ZI} + (1 - \hat{\lambda}_u) \cdot \hat{\phi}_{u,i}^{UI}) \quad (4.14)$$

Equation 4.14 embeds **Intuition 4** (when selecting POIs, different users in a group have different influence scores, and the influence scores are topic dependent) as follows: if the topic z is more relevant to group g_t , and a user u is an expert in z , then u will be more influential in POI selection. Recall that the expertise of a user u on topic z is

modeled by $\phi_{z,u}^{ZU}$. In Equation 4.14, $\theta_{g_t,z} \cdot \phi_{z,u}^{ZU}$ is the influence score of user u in group g_t for a given topic z , and $\lambda_u \cdot \phi_{z,i}^{ZI} + (1 - \lambda_u) \cdot \phi_{u,i}^{UI}$ is u 's preference to a candidate POI i given topic z . We margin out the topics, and get the overall preference of u to i . Then the preferences of all members to i are multiplied as the group preference to i . The rationale is three-fold: 1) the preference of a group to a POI depends on the preferences of all individuals; 2) ranking a POI based on the product of preferences is equal to the geometric mean of these individuals' preferences. Compared with the traditional strategies that calculate the arithmetic mean of preferences (averaging) or concentrate on the smallest preference (least-misery), the aggregated preference score by geometric mean is less sensitive to extreme values; 3) this definition matches the proposed model well.

4.1.5 Adapting COM for Movie Recommendation

Our proposed model COM can be adapted for other tasks by incorporating different content features. In this section, we take group movie recommendation as an example.

We believe when selecting a movie to watch, a user may consider several factors, such as genre, cast, etc. We take the cast to illustrate how to exploit content information. Intuitively, users tend to watch the movies starred by their favorite actors or actresses. We incorporate user u 's cast-based considerations to a movie i by modifying the prior $\rho_{u,i}$ as follows:

$$\rho_{u,i} \propto \sum_{s \in S_i} P(s|u) \quad (4.15)$$

where s is a movie star, and S_i is the cast list of movie i . $P(s|u)$ is estimated based on the occurrences of s in u 's watching history.

Note that the $|U| \times |I|$ dimensional matrix ϕ^{UI} requires a large amount of space, in which each value of $\phi_{u,i}^{UI}$ is determined by both the count $n_{u,i}^{UI}$ and the prior ρ_i (Equation 4.10). Since for each user u , the values of most POIs in ϕ^{UI} are 0 ($n_{u,i}^{UI} = 0$ and $\rho_{u,i} = 0$), we can use sparse matrix to store ϕ^{UI} to reduce the space complexity.

4.2 Experiments

We first introduce the setup of the experiments in Section 4.2.1, and then present the experimental results in Section 4.2.2, in which we compare the recommendation accuracy of our model with five baselines on four datasets. After that, we analyze which factor influences group members' choices more significantly in Section 4.2.3. In the end, we show some sample topics discovered by the proposed model to examine their semantics in Section 4.2.4.

4.2.1 Experimental Setup

4.2.1.1 Datasets

Four real-world datasets are used in our experiments. The first dataset is used in previous work [LHT⁺12], which is collected from Plancast [Pla], an event-based social network (EBSN). In Plancast, a user can follow others' calendars, and join different events. An event involves a group of members, and is held at a POI. A POI is associated with a geographical coordinate. We treat an event as a group, where the users involved in the event are the group members, and the POI of the event is the POI selected by them.

Second, we collect 45 million check-ins from Jiapang [Jie], a location-based social network (LBSN). As shown in Figure 1.1.b, LBSNs allow users to share their geographic information by check-ins, where a check-in has a user, time and POI, indicating the user visited the POI at that time. Each a POI in Jiapang is associated with its geographical coordinate. However, Jiapang does not contain explicit group information, and we extract implicit group check-ins as follows: we assume if a set of friends visit the same POI at the same time, they are the members of a group. Specifically, the set of individual check-ins made by friends within 0.5 hour is regarded as a group check-in. For both Jiapang and Plancast datasets, we aim to recommend POIs for given groups.

The last two datasets are extracted from 1M MovieLens dataset [Mov] by following the approach in [BMR10]. MovieLens allows users to rate the movies they have watched by stars ranging from 0 to 5. Two kinds of groups are considered in the experiments: similar and random, denoted by MovieLens-Simi and MovieLens-Rand, respectively. Groups in MovieLens-Simi have larger inner similarities between members, while groups in MovieLens-Rand are randomly formed. The two datasets simulate two kinds of groups in real-life: the groups formed by people who have similar preferences, and the groups that happen to be formed by a set of people. For each dataset, we randomly select 3000 groups with 5-members. We also evaluated groups of size 3 and 8, and obtained similar results. The details for generating the datasets can be found in [BMR10]. Given a group, if every member gives 4 stars or above to a movie, we assume that the movie is adopted by the group. We also collect the cast list of each movie from IMDB [IMD] as content information.

The information of the four datasets is shown in Table 4.2. For each dataset, we randomly mark off 20% of group events as the test set to evaluate the recommendation accuracy of different methods.

4.2.1.2 Evaluation Metrics

Following previous work [AYRC⁺09, BMR10, LTYL12, GLRW13], we evaluate the accuracy of different methods with three metrics, namely, average *precision@N* (*Pre@N*),

Table 4.2: Statistics of the Datasets

| Dataset | Plancast | Jiebang | MovieLens -Simi | MovieLens -Rand |
|---------------------------|----------|---------|--------------------|--------------------|
| #Users | 41,705 | 28,888 | 891 | 3,689 |
| #Group Events | 13,885 | 23,621 | 3,000 | 3,000 |
| #Items | 8,016 | 9,746 | 441 | 1,518 |
| Avg. Group Size | 20.30 | 4.68 | 5 | 5 |
| Avg. #Items for a Group | 1.00 | 1.01 | 14.97 | 3.73 |
| Avg. #Records for a User | 6.76 | 3.83 | 16.83 | 4.06 |
| Avg. #Friends for a User | 43.29 | 13.00 | N/A | N/A |
| Avg. #Records for an Item | 1.73 | 2.42 | 100.06 | 7.37 |

average $recall@N$ ($Rec@N$) and normalized discounted cumulative gain ($nDCG$), where N is the number of recommendations. We consider three values of N (*i.e.*, 5, 10, 20), where 5 is the default value. For all metrics, larger value indicates better recommendation performance.

4.2.1.3 Recommendation Methods

We evaluate 7 methods in our experiments, namely, CF-AVG, CF-LM, CF-RD [AYRC⁺09], SIG [YLL12], PIT [LTYL12], and the proposed methods COMP and COM. To the best of our knowledge, these state-of-the-art group recommendation methods have not been compared with each other in previous work, and our evaluation is the first experimental studies on them.

User-based CF with averaging strategy (CF-AVG): Given a candidate item i , CF-AVG first estimates the recommendation score of each user in the target group by user-based CF, and then uses the average of these scores as the recommendation score for the group.

User-based CF with least-misery strategy (CF-LM): Given a candidate item i , CF-LM first estimates the recommendation score of each user in the target group by user-based CF, and then uses the smallest score as the recommendation score for the group.

User-based CF with relevance and disagreement (CF-RD) [AYRC⁺09]: This model calculates the recommendation score for a candidate item i based on the relevance and disagreement of the group, where the relevance is calculated based on either CF-AVG or CF-LM, and the disagreement can be either the average difference of recommendation scores of pair-wise group members, or the variance of members' recommendation scores.

Social influence based group recommendation (SIG) [YLL12]: SIG is a topic model based approach, which has been introduced in Section 2.2.3. Since the MovieLens-Simi and MovieLens-Rand datasets have no friendship information, we do not report the results of SIG for them.

Personal impact topic model (PIT) [LTYL12]: PIT model assumes that different users have different impact scores, and in a group, the user who has a larger impact score is more likely to be selected as the representative. Given a group of users \mathbf{u}_{g_t} , PIT model first samples a representative user r from \mathbf{u}_{g_t} based on users’ impact scores, and then r selects a topic based on her topic preference, and finally the topic generates an item for the group.

Consensus Model Plain (COMP): To make a fair comparison with these baselines which do not exploit users’ considerations of content factors, we use a symmetric Dirichlet prior for ϕ^{UI} to disregard the effect of content information.

Consensus Model (COM): The proposed model incorporated with users’ considerations of content factors.

All baselines are evaluated under the optimal settings. For the hyperparameters in COMP and COM, we take fixed values ($\alpha = 50/K$, $\beta = \eta = 0.01$, $\gamma = \gamma_t = 0.5$ and $\rho = 0.01$ for COMP). The prior ρ in COM encodes the content-based knowledge, and needs to be set empirically. Previous work fixes its value as 0.01 [GS04], and thus the sum of the prior is $0.01 \times |I|$. In this chapter, we normalize the value of ρ of each user to a fix value $0.01 \times p \times |I|$, where the parameter p is used to tune the confidence in the prior knowledge.

The information exploited by each methods is summarized in Table 4.3.

Table 4.3: Comparison of Methods

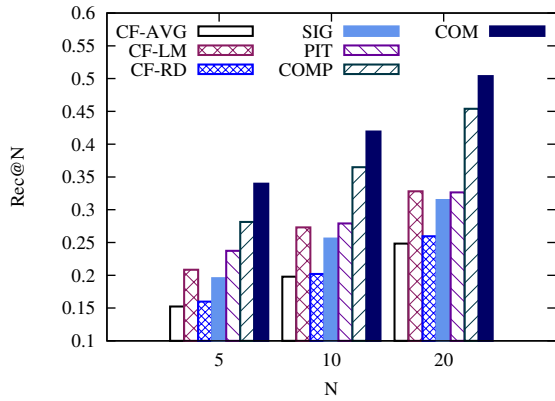
| Method | CF-AVG/ LM/RD | SIG | PIT | COMP | COM |
|------------------|------------------|-----|-----|------|-----|
| Adoption Records | ✓ | ✓ | ✓ | ✓ | ✓ |
| Item Tags | × | ✓ | × | × | × |
| Social Relations | × | ✓ | × | × | × |
| Content Features | × | × | × | × | ✓ |

4.2.2 Experimental Results

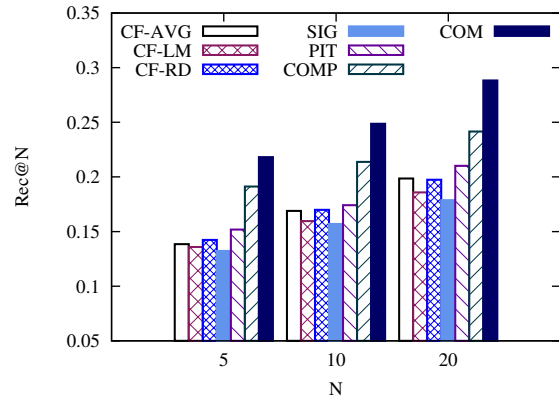
Precision and Recall Under Different N

We first fix the number of topics K at 250, and vary the number of recommendations N . The $Pre@N$ and $Rec@N$ values on the four datasets are plotted in Figure 4.2. Please note that the MovieLens-Simi/Rand datasets do not contain social relations, and thus the baseline SIG cannot be applied to them.

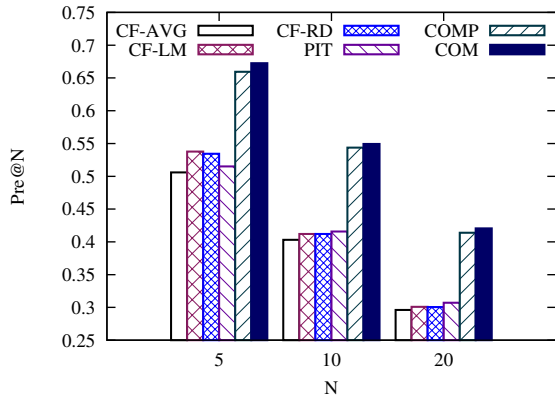
From Figure 4.2.a and 4.2.b, we can see that the CF-based approaches, namely, CF-AVG, CF-LM and CF-RD, do not perform well on the Plancast and Jiebang datasets. This is because the three methods exploit neither the difference of individuals, nor the



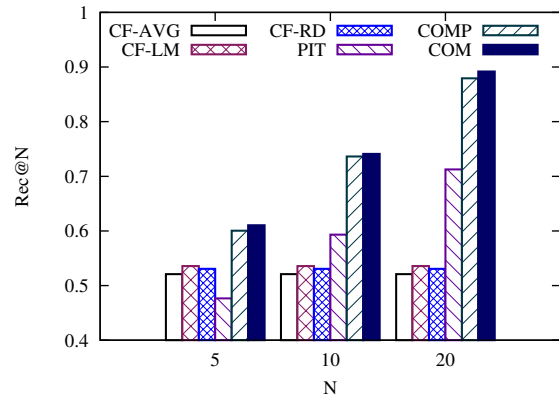
4.2.a: $Rec@N$ - Plancast



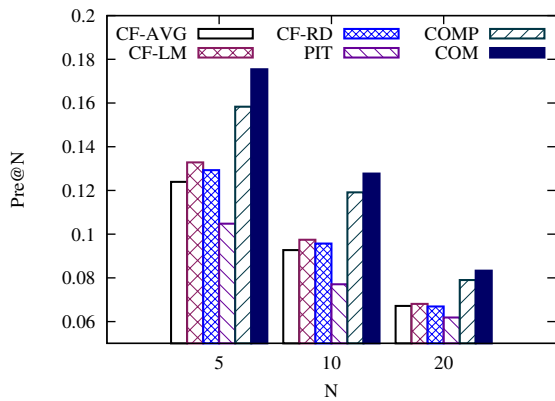
4.2.b: $Rec@N$ - Jiebang



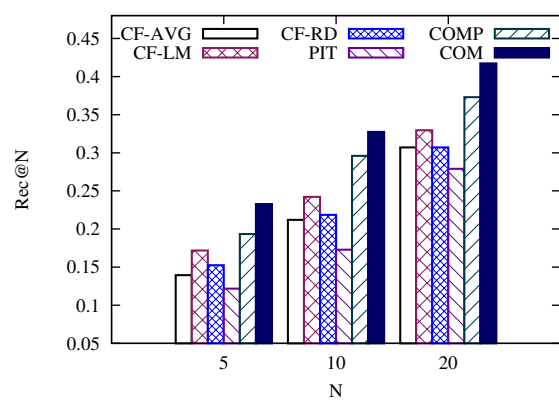
4.2.c: $Pre@N$ - MovieLens-Simi



4.2.d: $Rec@N$ - MovieLens-Simi



4.2.e: $Pre@N$ - MovieLens-Rand



4.2.f: $Rec@N$ - MovieLens-Rand

Figure 4.2: $Pre&Rec$ under #Recommendations Made (N)

interactions among group members. They assume that users in a group make choices independently, and aggregate their choices for recommendations. The performance of SIG is not satisfactory, either. The reason is that SIG makes group recommendations based on the social relations between users in a group, and it requires tags of candidate items. However, in the groups of Plancast, only several or even none of the members are friends to each other, and neither of the two datasets has tag information. The lack of social relations and tags brings down its recommendation accuracy.

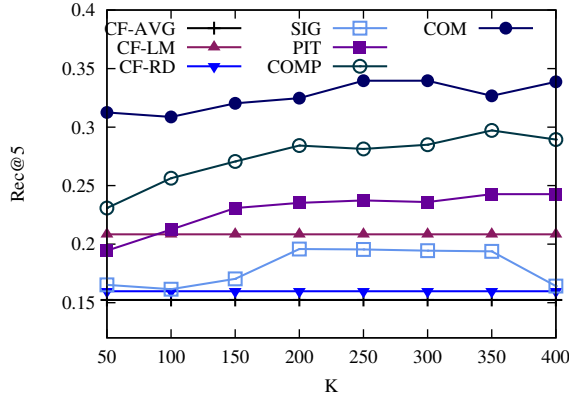
In contrast, PIT performs better than the CF based approaches on the Plancast and Jiebang datasets because PIT utilizes the interactions in a group by differentiating influences of users, and assumes that a user with a larger impact score will be influential in every group of the user. However, PIT ignores the fact that the influence of a user will be different across different topics. The performance of PIT on the two MovieLens datasets is not as good as on the Plancast and Jiebang datasets. This is because groups in the two datasets are loosely organized, and users select movie independently. Since no representative member exists to make item selections for a group, the basic assumption of PIT does not hold any more, which results in its bad recommendation accuracy.

Compared with the five baselines, our proposed method COMP always archives superior recommendation accuracy. For example, it outperforms CF-AVG, CF-LM, CF-RD, SIG and PIT by 84%, 34%, 76%, 43% and 19%, respectively, for $Rec@5$ on Plancast. The reasons are two-fold: on the one hand, COMP considers the behavior changes of users in a group (**Intuition 3**); on the other hand, it estimates the topic-dependent influences of users in a group (**Intuition 4**). Compared with COMP, COM further improves $Rec@5$ by more than 15% on Plancast, Jiebang and MovieLens-Rand, showing that content information is an important consideration when users selecting items (**Intuition 2**), and COM is effective in incorporating the content information (geographical distance for Plancast and Jiebang, and cast list for MovieLens-Rand). The improvement on MovieLens-Simi is marginal, since its user-item selection matrix has a high density (about 4%). As a result, COMP, which only utilizes users' selection history, already achieves very good accuracy (*e.g.*, $Rec@5$ is 67.3%), and thus the value of relative improvement is small.

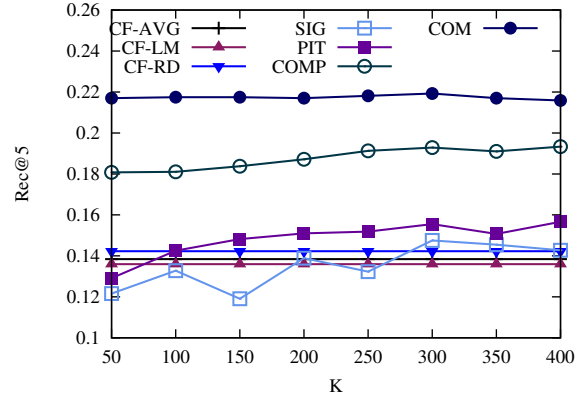
Precision and Recall Under Different K

We fix the number of recommendations at 5, and vary the number of topics K from 50 to 400. The $Pre@5$ and $Rec@5$ values on the four datasets are plotted in Figure 4.3. Since CF-AVG, CF-LM and CF-RD do not involve topics, their $Pre@5$ and $Rec@5$ values do not vary with K .

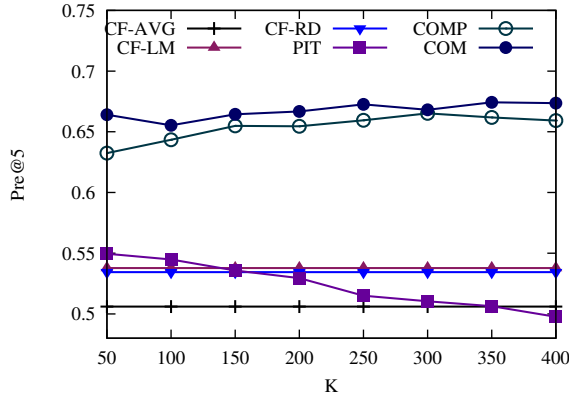
For the topic model based approaches, namely, SIG, PIT, COMP and COM, their $Pre@5$ and $Rec@5$ values do not change much with varying the number of topics. In addition, we notice that SIG performs worse than CF-AVG, CF-LM and CF-RD on Plancast, but better than these CF-based approaches on Jiebang when $K \geq 250$. This is because the group events of Jiebang are extracted based on friendships, and thus they fit



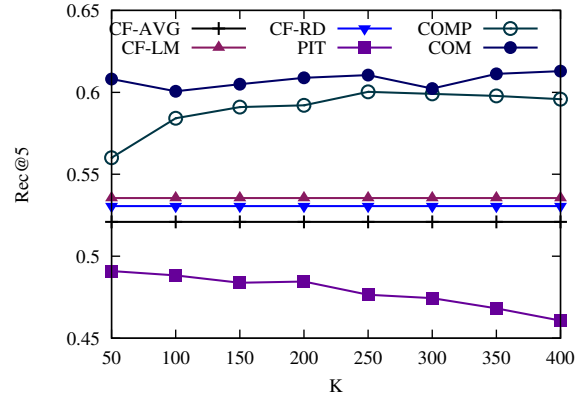
4.3.a: $Rec@5$ - Plancast



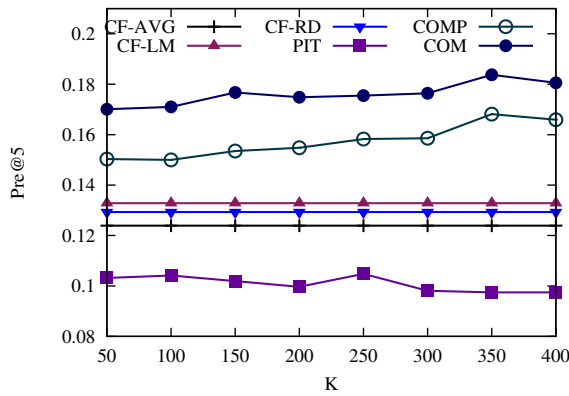
4.3.b: $Rec@5$ - Jiepang



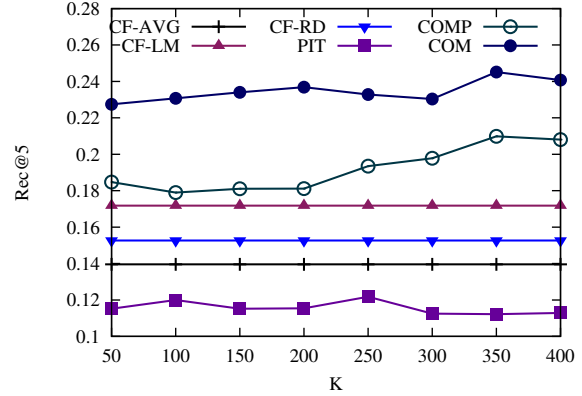
4.3.c: $Pre@5$ - MovieLens-Simi



4.3.d: $Rec@5$ - MovieLens-Simi



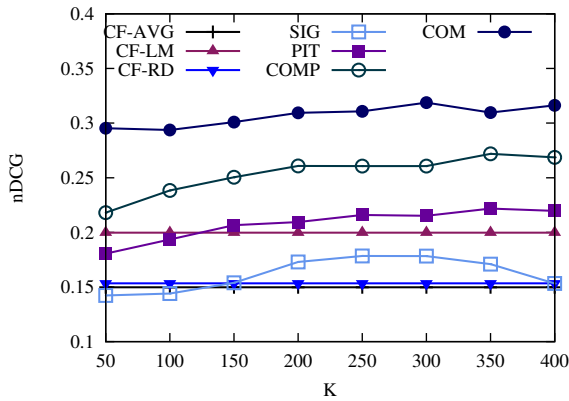
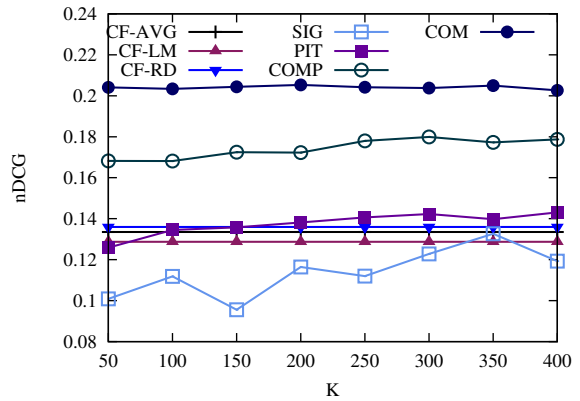
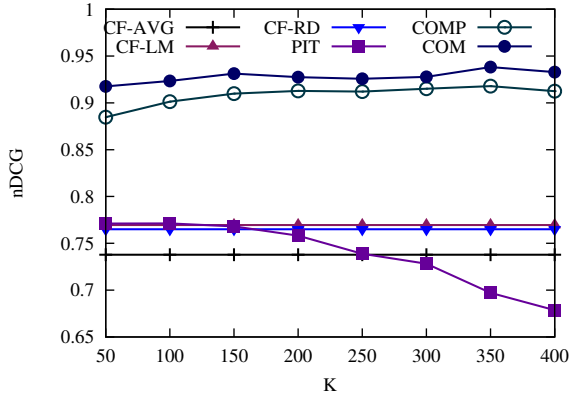
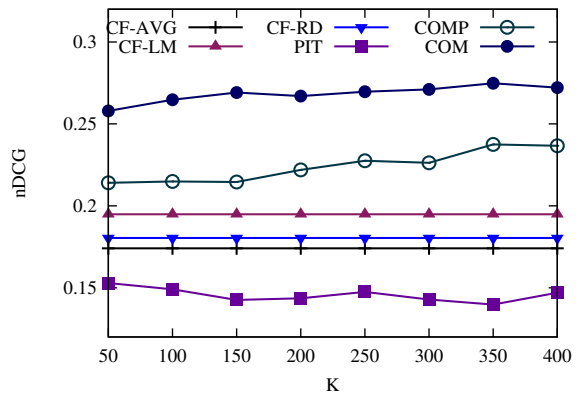
4.3.e: $Pre@5$ - MovieLens-Rand



4.3.f: $Rec@5$ - MovieLens-Rand

Figure 4.3: $Pre@5$ & $Rec@5$ under #Topics (K)

well with the assumption of the SIG. PIT’s $Rec@5$ value is the best among the baselines on the Plancast and Jiebang datasets, but is worse than that of CF-AVG, CF-LM and CF-RD on MovieLens-Simi and MovieLens-Rand. Potential reason is the generative process of groups in the MovieLens datasets is different from that of PIT model. Our proposed method COMP outperforms the best baselines by about 20% on the four datasets. After incorporating users’ personal considerations of content factors, COM further improve the $Rec@5$ values by more than 15% on the Plancast, Jiebang and MovieLens-Rand datasets, demonstrating the effectiveness of the proposed model.

4.4.a: $nDCG$ - Plancast4.4.b: $nDCG$ - Jiebang4.4.c: $nDCG$ - MovieLens - Simi4.4.d: $nDCG$ MovieLens - RandFigure 4.4: $nDCG$ under #Topics (K)

$nDCG$ Under Different K

Next, we vary the number of topics K , and examine the $nDCG$ results of different approaches to see how well they can rank the true items higher. The results are plotted

in Figure 4.4. We can see that the results display a similar trend with the previous experimental results based on $Pre@5$ and $Rec@5$: PIT performs the best among the baseline methods on Plancast and Jiebang, but the worst on MovieLens-Simi and MovieLens-Rand. However, our method COMP consistently outperforms the best baseline under different number of topics by more than 15% on the four datasets. COM achieves the best results, which are at least 16% greater than that of COMP on Plancast, Jiebang and MovieLens-Rand.

Effect of p

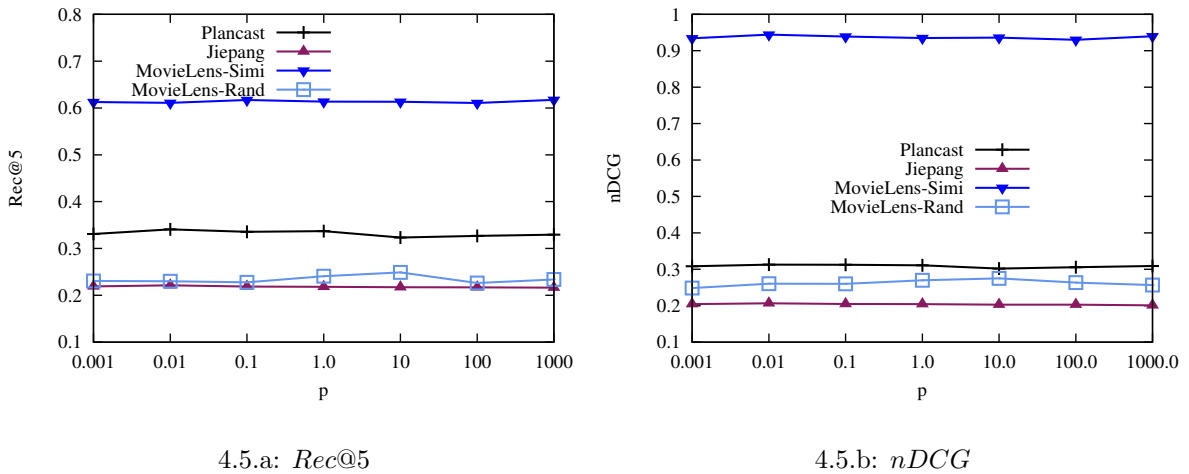


Figure 4.5: $Rec@5$ and $nDCG$ under Different p

We next examine the effect of p on the recommendation accuracy of COM. Recall that after incorporating the users' personal considerations of content factors into the prior, we normalize the prior of each user to $p \cdot 0.01 \cdot |I|$. Parameter p is set to adjust the effect of the prior, *i.e.*, larger p implies that the distribution ϕ_u^{UI} is more influenced by the content information. We examine the recommendation accuracy of COM under different value of p ranging from 0.001 to 1000. The $Rec@5$ and $nDCG$ on four datasets are plotted in Figure 4.5. We can see that the recommendation performance remains relatively stable when varying the value of p . The $Pre@5$ follows a similar trend with $Rec@5$.

Performance for Different Size of Groups

This set of experiments is to study the performance of each recommendation method for groups of different sizes. We group the Plancast groups into bins based on group size, and plot the $Rec@5$ and $nDCG$ curves of each method in Figure 4.6. The number of topics is fixed at 250. The results on the other datasets are not given here. Figure 4.6 shows that the proposed methods COMP and COM outperform the baselines for groups of different sizes. Among the baselines, CF-AVG, CF-LM and CF-RD perform the worst, followed

by SIG and PIT. Compared with that of other methods, the performance of CF-based approaches is better for groups of small size, because their group organizations are simple, and thus these simple aggregating strategies are good for making recommendations.

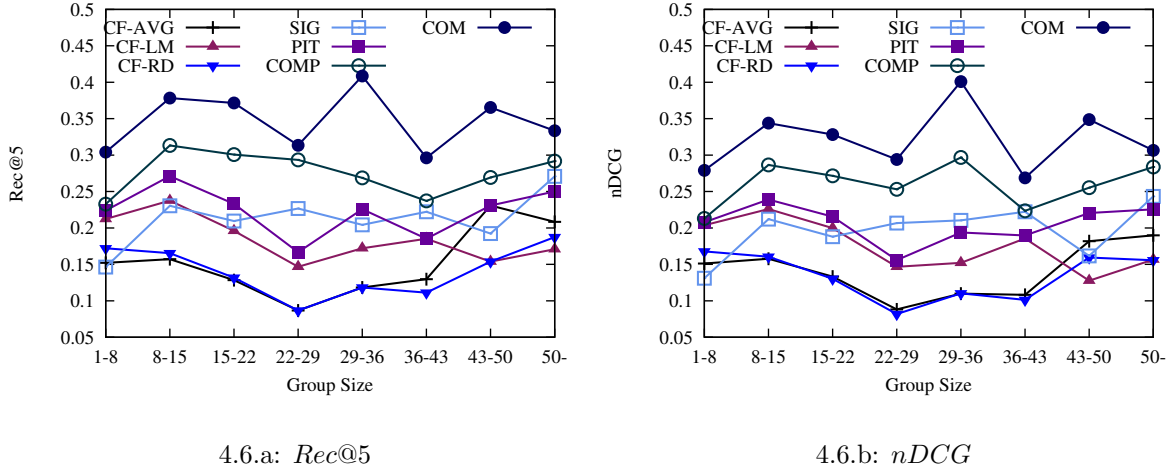


Figure 4.6: $Rec@5$ and $nDCG$ for Different Size of Groups

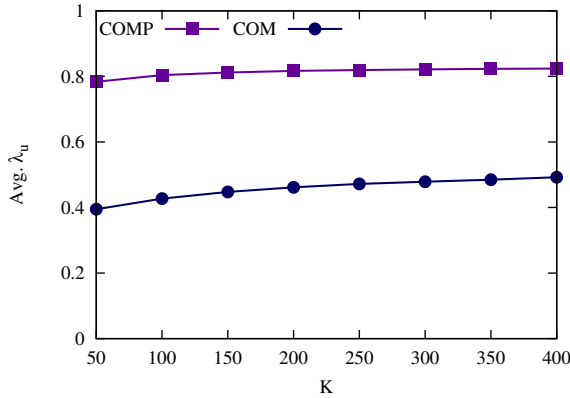
4.2.3 Weight of Topics in Item Selection

In this section, we study the weight of topics in users' item selections by investigating parameter λ_u , the probability that a user selects an item according to group topics.

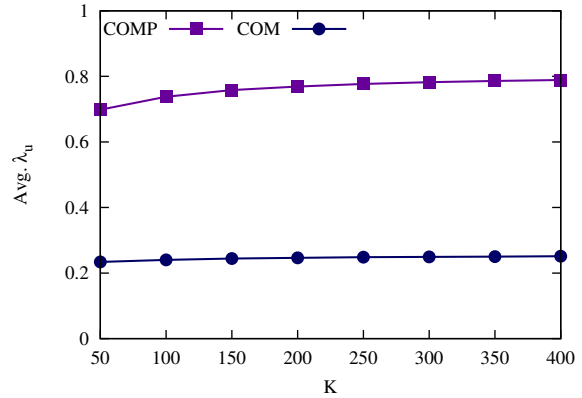
We first study the effect of the number of topics K on the value of λ_u . Specifically, for each dataset, we plot the average λ_u of all users as a function of K . The curves of COMP and COM on the four datasets are shown in Figure 4.7. We see that for COMP which does not exploit the content information, the average λ_u is almost not affected by the value of K , and its value is between 0.75 to 0.9. The results reveal that most of items are selected according to topics, but there is still a set of items that are selected based on users' personal considerations of content factors. Compared with the average λ_u of COMP, that of COM is much smaller, since additional content information is incorporated. The value of λ_u on Plancast (around 0.5) is larger than that on Jiebang (around 0.2) for COM, indicating that topics have larger weight in POI selections of Plancast users. In addition, the λ_u values of both COMP and COM on MovieLens-Simi are larger than those on MovieLens-Rand, because groups in MovieLens-Simi consist of people with high similarities, and thus topic is a very important consideration.

Then, we fix the number of topics at 250, and plot the distribution of λ_u of COM on the four datasets in Figure 4.8. First, we observe that the weights of topics are different

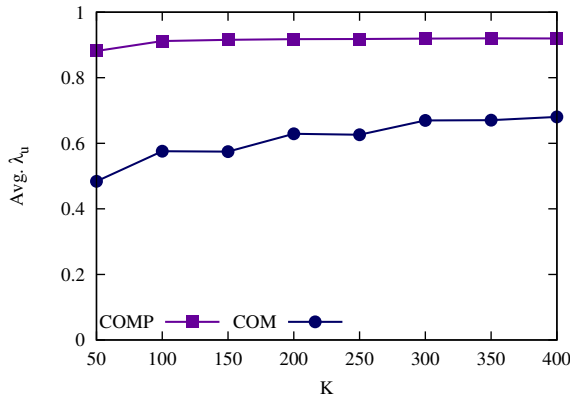
for different users, which is in accordance with **Intuition 2**. We also notice that on all datasets, the λ_u value of the majority of users is smaller than 0.4, showing that the personal considerations of content information is important for most of people. In addition, we see that the curve of Plancast has a long tail, indicating that a considerable portion of users treat topics important. In addition, λ_u on MovieLens-Simi reaches another peak at $\lambda_u = 1$, showing that in a group with high inner-similarity, a considerable portion of people select items based on topics.



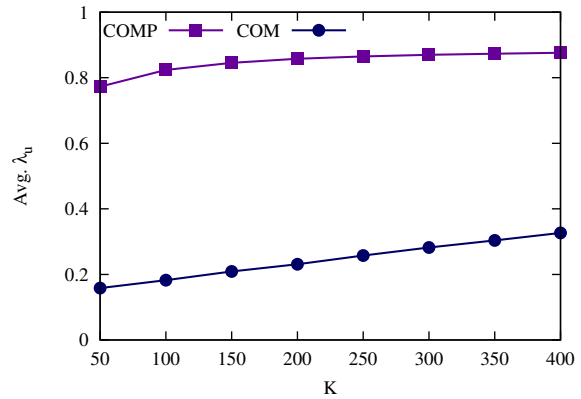
4.7.a: λ_u vs K - Plancast



4.7.b: λ_u vs K - Jiebang

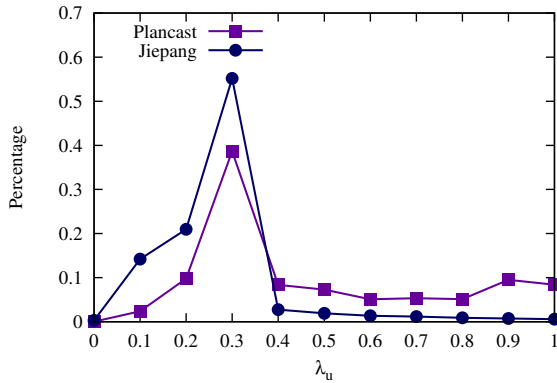


4.7.c: λ_u vs K - MovieLens-Simi

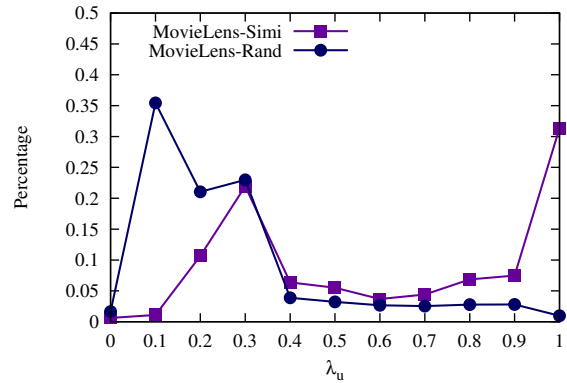


4.7.d: λ_u vs K - MovieLens-Rand

Figure 4.7: Avg. λ_u under #Topics (K)



4.8.a: Plancast/Jiebang



4.8.b: MovieLens Simi/Rand

Figure 4.8: Distribution of λ_u

4.2.4 Topic Analysis

We first investigate the POI distribution of each topic generated by COM on Plancast and Jiebang datasets, where the number of topics is set to 250. For each topic z , we rank the POIs i based on $\phi_{z,i}^{ZI}$. The top 5 POIs of 5 randomly selected topics on the two datasets are plotted in Figure 4.9. We observe that for each topic, the top-ranked POIs are close to each other. This is because topics are estimated based on users' group participation history. Since users tend to join groups held at their nearby POIs due to the spatial constraint, the POIs visited by each user fall in a small geographical region, and thus the top-ranked POIs of each topic are close to each other.

Table 4.4: Representative Movies for COM Topics

| Topic | Movies |
|-----------|--|
| Comedy | Big (1988), Romancing the Stone (1984), Four Weddings and a Funeral (1994) |
| Action | Raiders of the Lost Ark (1981) Star Wars: Episode V - The Empire Strikes Back (1980), Full Metal Jacket (1987) |
| Drama | Fight Club (1999), Trainspotting (1996), Magnolia (1999) |
| Thriller | Reservoir Dogs (1992), Gattaca (1997), Heavenly Creatures (1994) |
| Animation | Toy Story 2 (1999), Mulan (1998), Peter Pan (1953) |

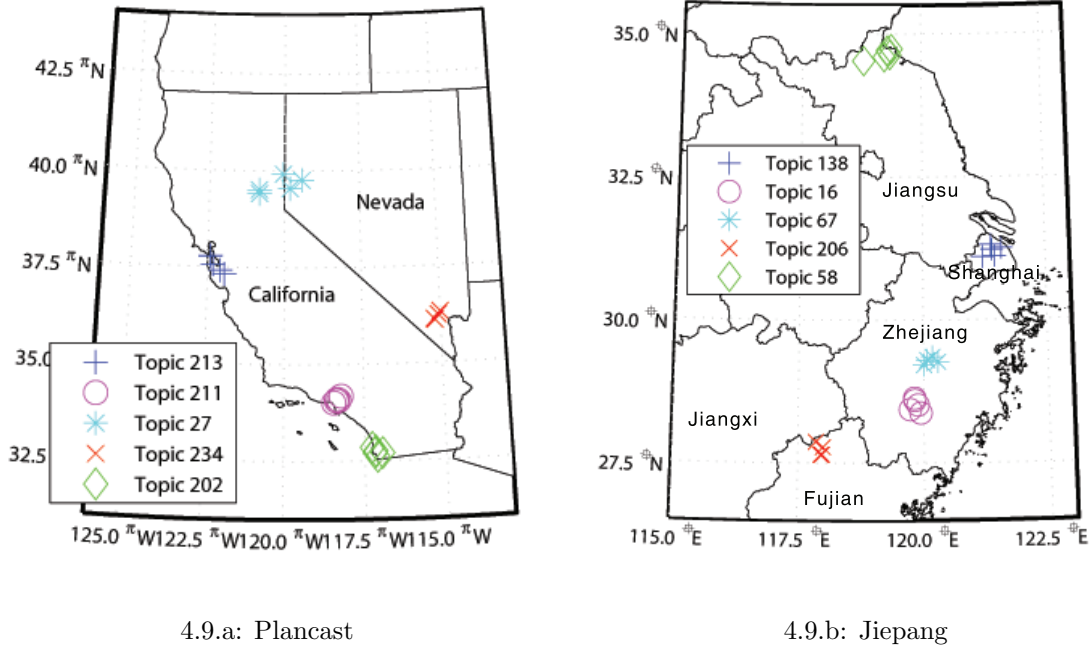


Figure 4.9: POI Distribution of Topics

Then, we examine the movie distributions of topics of COM on the MovieLens-Rand dataset. Specifically, we set the number of topics at 50 for COM, and randomly select 5 topics. For each topic z , we rank the movies i based on the learnt $\phi_{z,i}^{ZI}$. The top 3 movies of the 5 topics are listed in Table 4.4. The name of each topic is generated from the top 10 movies' genres in IMDB by majority vote. We can find that the discovered topics are semantically meaningful.

4.3 Summary

A number of POI recommendation methods have been proposed, but most of them are designed for individuals. How to make accurate POI recommendations for groups is still an open problem. In this chapter, we propose a new probabilistic model COM to simulate the generative process of group events and make POI recommendations for a group of users. COM is designed based on the novel intuitions that group members' influences are topic-dependent, and a user may behave differently as a group member from that as individual. Since users' POI selections are not only influenced by topics, but also by users' personal considerations of content factors like distance, we incorporate the content information into the model. Experimental results on four real-world datasets show that

the proposed method outperforms five baselines significantly in terms of recommendation accuracy.

Chapter 5

Requirement-aware POI Recommendation

Users may have clear requirements before submitting the recommendation requests, *e.g.*, shopping or having dinner. Obviously, the requirements explicitly reveal users' interests, and thus can be utilized for recommendation.

In this chapter, we define a new task, namely, requirement-aware POI recommendation, which aims at predicting POIs for a target user based on her specific requirement, where the requirement can be formulated as short text, *e.g.*, "spots for birthday party". In addition, as time is an important consideration for recommendation, the requirement-aware results could be also time-specific when the target time is available.

Obviously, the recommended POIs of time and requirement aware POI recommendation should meet two points: 1) the recommended POIs should be close to the position of the target user at the given time, because users tend to visit their nearby places; 2) the recommended POI should be relevant to the given requirements. The first point requires the recommender system should be able to model the mobility patterns of individual users, so that it can predict the most likely mobility area of the target user at the target time, while the second point requires the recommender system can model the activity of the user at different places, and model the semantic topics of different POIs. Based on the given requirement (word) and time, we can infer the current location of the target user, and recommend nearby POIs that match the given requirement. Thus, in order to return accurate requirement-aware results, we need to design a model that can model the spatial, temporal and semantic information of each individual. Thanks to the development of location-aware social networks such as Twitter and Foursquare, we have a sheer amount of data to infer users' mobility patterns (time and locations which can be extracted from check-ins and geo-annotated tweets) and topic interests (shout text, such as tweets content and check-in shouts). However, developing a model that can capture the four factors simultaneously is still a challenging task, since they together make

the modeling and parameter estimation complicated. Moreover, the interdependencies among them and role played by each is unclear.

In this chapter, we propose a pLSA-based generative model W^4 (short for **Who**, **Where**, **When**, and **What**) to discover individual users' mobility behaviors from user, spatial, temporal and semantic aspects, and to make requirement-aware POI recommendation. Then, we enhance it and propose a non-parametric bayesian model EW^4 (short for **Enhanced W^4**), which does not require any input parameters. Experimental results on two real-world datasets show that the proposed models are effective in not only requirement-aware POI recommendation, but also a variety of other applications.

The rest of this chapter is organized as follows: We study the characteristics of users' mobility behaviors in Section 5.1. Based on the intuitions described in Section 5.2, we build our models W^4 and EW^4 in Sections 5.3 and 5.4, respectively. The parameter estimation algorithms are introduced in Section 5.5. We discuss some applications of our model in Section 5.6, and present the experimental results in Section 5.7. Section 5.8 concludes our work.

5.1 Characteristics of Individual Mobility

In this section, we study the characteristics of individual user mobility pattern on two datasets, namely, world-wide tweets collection (WW) and microblogs collection from USA (USA. see Section 5.7 for more details on the datasets).

DataAnalysis 1: We first examine the effect of spatial distance to users' mobility. Specifically, for each user, we calculate the distance between every pair of her visited POIs. Then, we aggregate the results of all users and plot the number of check-ins as a function of distance in Figure 5.1. From the figures, we observe that users are more likely to visit POIs close to their visited POIs, and thus the POIs visited by a user will form several spatial clusters. The observations are in accordance with that in Section 3.2.3.

DataAnalysis 2: Next, we randomly select a user and plot her visited POIs in a map. The POIs visited in weekdays and weekends are plotted in different colors. From Figure 5.2, we observe that the POIs visited by the user can be clustered into several geographical regions, and the user visited a region with different probabilities in weekdays and weekends.

DataAnalysis 3: Then, we analyze the temporal pattern of users' mobility. Specifically, we divide the check-ins of the selected user into two sets based on the check-in day (*i.e.*, weekday and weekend). For each set, we plot the visited POIs on the map and distinguish three visiting time slots by different colors in Figure 5.3. From the figures, we make an observation that the visiting time of POIs in a region is different in weekdays and weekends.

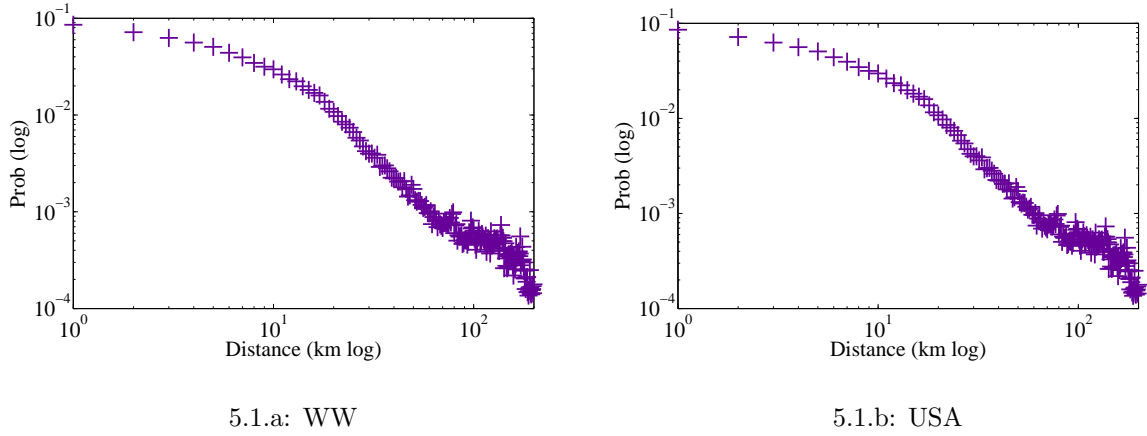


Figure 5.1: Distribution of Distance Between Individuals' Check-ins

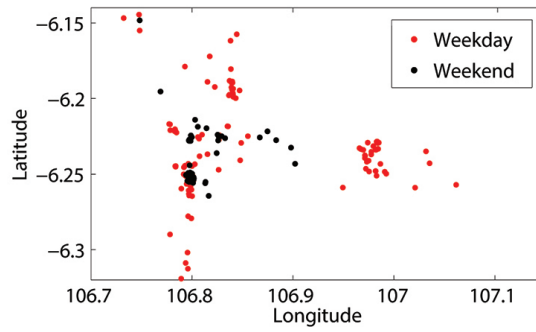


Figure 5.2: Visited POIs of a User

5.2 Intuitions and Notation

We develop our model based on the following intuitions, which enable us to model the mobility patterns and topic interests of users, and the topics of POIs.

Intuition 1 An individual's mobility usually centers at several personal geographical regions, *e.g.*, home region and work region (**DataAnalysis 1 and 2**). The number of personal regions is user-dependent, *e.g.*, users who are interested in outdoor activities may have hiking regions, and users who are interested in shopping may have shopping regions.

Intuition 2 A user may visit a region with different probabilities on weekdays and weekends (**DataAnalysis 2**), *e.g.*, she may go to work region on weekdays rather than on weekends. In addition, the visiting time of a region is influenced by the day (**DataAnalysis 3**), *e.g.*, a user is more probable to stay at home region in

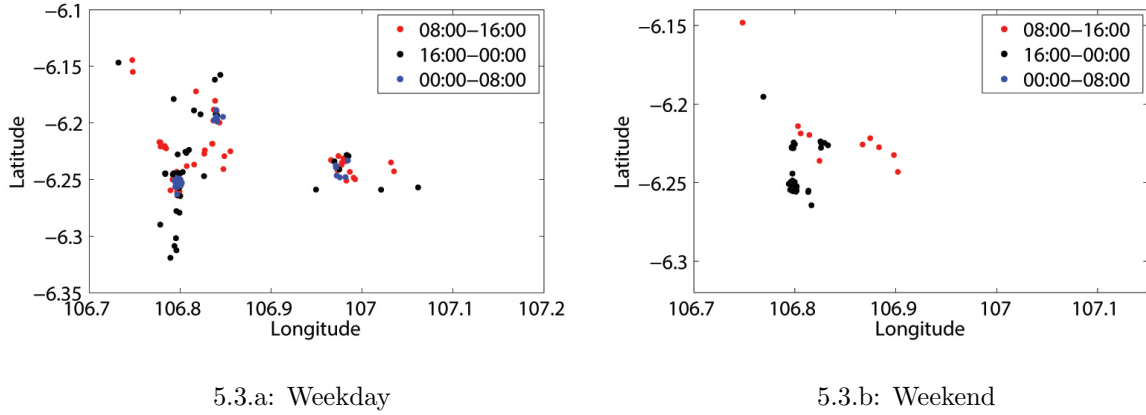


Figure 5.3: Visiting Time of POIs in Different Days

the evening of weekdays, but it is also likely to stay at home in the daytime on weekends.

Intuition 3 The topics of a user at a place are influenced by both the user’s personal topic preference and the region where the user stays. For example, suppose a user who is interested in both eating and hiking comes to a place full of restaurants, the user is more likely to be interested in the eating topic. In addition, the topics of a user at her home region (*e.g.*, entertainment and shopping) are expected to be different from the work-related topics at her work region.

Intuition 4 When a user chooses a POI to visit, both the topic requirement and the region where the user stays are considered. Intuitively, a user tends to visit nearby POIs within her current region of stay (**DataAnalysis 1**) that meet her requirement (*e.g.*, for meal). In addition, different users may make different trade-offs between the topic and region factors, *e.g.*, comparing to those without cars, the users who have cars may treat the region with less importance, because it is much easier for them to drive to the POIs they want to visit.

Intuition 5 Different regions and different topics lead to different language variations [HAG⁺12], which in turn reflect the user’s activity. Therefore, the words in user’s tweets are affected by both the topic and the region. For example, if a user is shopping at her home region, the words she would use are related to both the shopping topic and home region, such as “grocery”, “family”, etc.

We consider each user u has several personal regions, denoted by $\{r_{u,0}, r_{u,1}, \dots, r_{u,|R_u|}\}$, where $|R_u|$ is the number of regions of user u . The personal regions are estimated based

Table 5.1: Symbols for Requirement-aware POI Recommendation

| Symbol | Description |
|-------------------------------|--|
| U, L, S, W, D | user set, POI set, day set {weekday, weekend}, vocabulary set, tweets set |
| Z, R, R_u | topic set, region set of all users, region set of user u , where $R = \bigcup_{u \in U} R_u$ |
| $ \cdot $ | the number of elements in set (\cdot) |
| u, ℓ, w, s, t | user $u \in U$, POI $\ell \in L$, word $w \in W$, day of a week $s \in S$, time of a day $< hh : mm : ss >$ |
| $d_i, z_i, r_i, \mathbf{w}_i$ | the i^{th} tweet in D , the topic, region and the text of of d_i |
| $r_{u,j}$ | the j^{th} region of user u , where $1 \leq j \leq R_u $ |
| c_i^L, c_i^W | the POI and word switches of tweet i |
| $c_{w,\mathbf{w}}$ | the occurrence times of word w in \mathbf{w} |
| $\{t\}_r$ | the collection of time of the tweets that are assigned to region r |
| $td(t_1, t_2)$ | the difference between time t_1 and t_2 in a day |
| G_0 | global probability measure over topic space with mixing proportion τ |
| G_r | region-specific measure over topic space with mixing proportion θ_r |
| τ | global multinomial distribution of topics |
| θ_r | multinomial distribution of topics specific to region r |
| $\psi_{u,s}$ | multinomial distribution of regions specific to user u on day s |
| ϕ_z^{ZL} | multinomial distribution of POIs specific to topic z |
| ϕ_z^{ZW} | multinomial distribution of words specific to topic z |
| ϕ_r^{RW} | multinomial distribution of words specific to region r |
| ξ_u^L | Bernoulli distribution specific to user u for sampling the binary switch c^L |
| ξ_u^W | Bernoulli distribution specific to user u for sampling the binary switch c^W |
| $\mu_r, \Lambda_r, \Sigma_r$ | mean, precision matrix and covariance matrix of Gaussian distribution over geographic coordinates specific to region r , where $\Lambda_r = \Sigma_r^{-1}$ |
| $\nu_{r,s}, \lambda_{r,s}$ | mean and precision of Gaussian distribution over time specific to region r in day s |

on the POIs of all geo-tagged tweets from a user. We model a POI ℓ as a two-tuple $\ell = \{id_\ell, \mathbf{cd}_\ell\}$, where id_ℓ is the identifier of the POI, and \mathbf{cd}_ℓ is the latitude and longitude coordinates of the POI. A region r is modeled by a bi-variant Gaussian over the latitude and longitude, parameterized by the mean vector $\boldsymbol{\mu}_r$ and covariance matrix $\boldsymbol{\Lambda}_r^{-1}$. Note that we use r to represent a region (*i.e.*, any one of the personal regions) when the semantic is clear.

Table 5.2: Symbols for Requirement-aware POI Recommendation (continue)

| Symbol | Description |
|--|--|
| γ | parameter of the prior of $\boldsymbol{\tau}$ |
| α | concentration parameter for G_r |
| β | parameter for Chinese Restaurant Process for $\boldsymbol{\psi}$ |
| η, χ, ζ | Dirichlet prior vector for $\boldsymbol{\phi}^{ZL}$, $\boldsymbol{\phi}^{ZW}$ and $\boldsymbol{\phi}^{RW}$ |
| $\boldsymbol{o}, \boldsymbol{\delta}$ | Beta priors for $\boldsymbol{\xi}^L$ and $\boldsymbol{\xi}^W$, where $\boldsymbol{o} = \{o_0, o_1\}$ and $\boldsymbol{\delta} = \{\delta_0, \delta_1\}$ |
| $\boldsymbol{\mu}_0, \kappa_0, \nu_0, \boldsymbol{\epsilon}_0$ | Normal-Wishart prior for $\boldsymbol{\mu}$, and $\boldsymbol{\Lambda}$ |
| $\nu_0, \iota_0, \rho_0, \lambda_0$ | Normal-Gamma prior for ν , and λ |
| $n_{s,r,-i}^{SR}$ | number of times region r is assigned to day s , excluding tweet i |
| $n_{r,z,-i}^{RZ}$ | number of times topic z is assigned to region r , excluding tweet i |
| $n_{z,\ell,-i}^{ZL}$ | number of times POI ℓ is assigned to topic z , excluding tweet i |
| $n_{z,w,-i}^{ZW}$ | number of times word w is assigned to topic z , excluding tweet i |
| $n_{r,w,-i}^{RW}$ | number of times word w is assigned to region r , excluding tweet i |
| $n_{u,(.),-i}^{UCL}$ | number of times switch $c^L = (\cdot)$ is assigned to user u , excluding tweet i |
| $n_{u,(.),-i}^{UCW}$ | number of times switch $c^W = (\cdot)$ is assigned to user u , excluding tweet i |
| m_z | number of tweets that are assigned to topic z |
| σ | a parameter to balance the importance between the region and the topic for word generation in W^4 |
| κ | a parameter to balance the importance between the region and the topic for POI generation in W^4 |

We model time t in a day as a continuous variable in $\langle hh : mm : ss \rangle$ format, and categorize days into two classes, namely, weekdays and weekends. Specifically, we use $s \in S = \{0, 1\}$ to denote a day of a week, *i.e.*, $s = 0$ for a weekday and $s = 1$ for

a weekend day. Note that t is cyclic on a daily basis. For instance, the time difference between 23:00:00 and 1:00:00 is the same as the difference between 1:00:00 and 3:00:00. All notions used in this chapter is shown in Tables 5.1 and 5.2.

We consider a tweet d is a five-tuple $d_i = \{u_i, \ell_i, \mathbf{w}_i, t_i, s_i\}$, where u_i denotes the user or the author of the tweet; ℓ_i , t_i , and s_i denote the POI, the time in a day, and the day of a week, as described earlier; \mathbf{w}_i are the words in tweet d_i . For easy presentation, we use D , U , and L to denote the collection of tweets, users, and POIs respectively. The word vocabulary is denoted by V . That is, $d \in D$, $u \in U$, $\ell \in L$, and each word in \mathbf{w}_i belongs to V . The topics of a user are reflected by the words in the user’s tweets.

5.3 W^4 Model Description

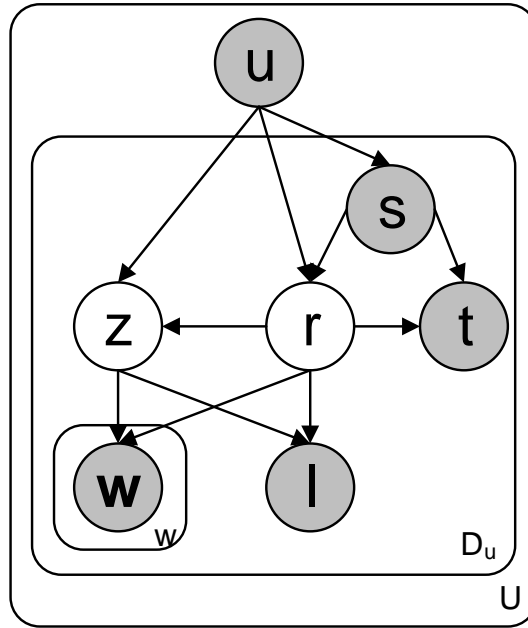
W^4 generates the day, time, words, and POI for each tweet posted by a user, shown in Figure 5.4. The generative process is briefly described below.

- (i) For each tweet d of a given user u , a day s is first selected based on a Bernoulli distribution $P(s|u)$, and then a personal region r is generated by drawing from the multinomial distribution $P(r|u, s)$, which further generates a time in that day based on Gaussian distribution $\mathcal{N}(\nu_{r,s}, \sigma_{r,s})$.
- (ii) Parameterized by the topic preference of the user u and the sampled region r , a topic z is generated using the multinomial distribution $P(z|u, r)$ (**Intuition 3**).
- (iii) After generating the region and the topic, the POI ℓ and each word w are sampled based on $P(\ell|r, z)$ and $P(w|r, z)$, respectively (**Intuition 4** and **5**).

The generative process of a a collection of tweets D is summarized in Algorithm 2.

Note that we do not sample time t based on its exact time point, *e.g.*, 11:00 pm. In stead, the time t is generated based on the time difference $td(t, \nu_{r,s})$ between it and the mean time of the time Gaussian of region r on day s , *i.e.*, $t \sim \mathcal{N}(td(t, \nu_{r,s})|\nu_{r,s}, \sigma_{r,s})$. In other words, the time that is close to the mean time is more likely to be sampled. Since the time in a day is cyclic, the time difference is always less than or equal to 12, *e.g.*, the time difference between 11:00 pm and 1:00 am is 2 hours.

To model $P(w|r, z)$, a parameter σ is introduced to balance the importance between the region and the topic, *i.e.*, $P(w|r, z) = \sigma P(w|z) + (1 - \sigma)P(w|r)$, where $P(w|z)$ and $P(w|r)$ are the word distribution of topic z and region r , respectively. Because a tweet is very short (limited within 140 characters), we assume all words in a tweet come from the same topic.

Figure 5.4: The Graphical Representation of W^4

Next, we generate a POI according to r and z . It is however challenging to model the generating process of POI. Previous studies treat a POI either as geographic coordinates or a POI identifier. Treating POIs as geographic coordinates makes it feasible to capture user’s mobility regions [YCH⁺11], or discover geo-regions that have specific topics [HAG⁺12, Siz10], but fails to capture the topic variations of different POIs. On

Algorithm 2: Generative Process of W^4

```

1 for each user  $u$ , ( $u = 1, \dots, |U|$ ) do
2   for the tweet  $d_i \in D_u$  do
3     Draw a day  $s \sim P(s|u)$ ;
4     Draw a region  $r \sim P(r|u, s)$ ;
5     Draw a time  $t \sim P(t|u, r, s)$ ;
6     Draw a topic  $z \sim P(z|u, r)$ ;
7     Draw a POI  $\ell \sim P(\ell|r, z)$ ;
8     for the  $k$ -th word ( $k = 1, \dots, |w_i|$ ) do
9       Draw  $w \sim P(w|r, z)$ ;
10    end
11  end
12 end

```

the other hand, treating POIs as POI identifiers enables us to differentiate the topics of POIs [MLSZ06, WWXM07, HCW⁺10] (because $P(\ell|z)$ is always modeled by a multinomial distribution, which calls for a limited POI set L), but the geographic coordinate information is ignored.

As stated in **Intuition 4**, a user tends to visit a nearby POI (*e.g.*, restaurant) that can fulfill her topical needs (*e.g.*, lunch). That is, when choosing a POI to visit, a user jointly considers both its geographic POI and its topic (*e.g.*, restaurant or bar). We use a parameter κ to balance $P(\ell|z)$ and $P(\ell|r)$, *i.e.*, $P(\ell|r, z) = \kappa P(\ell|z) + (1 - \kappa)P(\ell|r)$.

Note that the proposed model only considers the general situations that a user stays at her own cities. When user travels to new cities, the mobility behavior might be different from the propose model. For example, the user may visit some popular regions, where the regions are global. In addition, the influence of day to the mobility is not significant any longer. We leave the model as our future work.

5.4 EW⁴ Model Description

Note that W⁴ cannot fully exploit the intuitions given in Section 5.2. First, it assumes that all users have the same number of personal regions, which does not fully exploit Intuition 1. Second, it assumes that different users make the same trade-offs between the topic and region factors when choosing a POI, which does not fully exploit Intuition 4. In addition, these tuning parameters have to be set manually in W⁴. To solve these problems, we propose a new model named Enhanced W⁴.

Specifically, for each tweet d of a given user u with a day s , a personal region r is generated based on the day (**Intuition 1** and **2**). Here we employ Chinese Restaurant Process (CRP) to draw the region, and automatically discover personal regions for each user. CRP is a stochastic process that customers select seats at a restaurant with an infinite number of tables. The first customer randomly selects a table to sit, while the other customers can either sit at an occupied table i with probability of $\frac{n_i}{n+\beta}$, or sit at a new table with probability of $\frac{\beta}{n+\beta}$, where n_i is the number of customers at table i , and n is the total number of customers in the restaurant. As a Bayesian nonparametric approach, CRP is effective in clustering data (*i.e.*, customers) into clusters (*i.e.*, tables), and it can automatically estimate how many clusters are needed to model the data.

In our problem, given the day s , user u can either select an existing region, or create a new region and select it. The probability that u selects a region r is computed as follows:

$$CRP(r|u, s) = \begin{cases} \frac{\beta}{\sum_{r'=1}^{|R_u|} n_{s,r'}^{SR} + \beta} & r \notin R_u \\ \frac{n_{s,r}^{SR}}{\sum_{r'=1}^{|R_u|} n_{s,r'}^{SR} + \beta} & r \in R_u \end{cases} \quad (5.1)$$

After selecting the region r , the time t is drawn based on Gaussian distribution $\mathcal{N}(td(t, \nu_r) | \nu_{r,s}, \lambda_{r,s}^{-1})$ (**Intuition 2**). Parameterized by the topic preference of the user u and the sampled region r , a topic z is drawn from the region-specific probability measure G_r over topic space, where the multinomial mixing proportions of G_r is θ_r (**Intuition 3**). After selecting the region and topic, we draw the POI ℓ and each word w . Since ℓ can be either generated based on topic or region (**Intuition 4**), we use a switch c^L to decide which one accounts for the POI selection. If $c^L = 1$, the POI is sampled based on the topic-specific multinomial distribution over POIs ϕ_z^{ZL} , and if $c^L = 0$, the POI is sampled based on the Gaussian distribution $\mathcal{N}(\mu_r, \Lambda_r^{-1})$ of region r . Since the words selection is also influenced by both region and topic (**Intuition 5**), we introduce another switch c^W . If $c^W = 1$, the word is sampled based on the topic-specific multinomial distribution over words ϕ_z^{ZW} , and if $c^W = 0$, the word is sampled based on the region-specific multinomial distribution over words ϕ_r^{RW} . Based on the same assumption of W^4 , all words in a tweet with $c^W = 1$ come from the same topic. But unlike W^4 , we assume different users will make different trade-offs between topics and regions when selecting POIs and words. Thus, c^L and c^W in EW^4 are drawn from two user-specific Bernoulli distributions ξ_u^L and ξ_u^W , respectively.

How to set the number of topics $|Z|$ is an important issue. Most of previous studies are built on topic models such as pLSA and LDA, in which the number of topics $|Z|$ to be set empirically. In this article, we proposed a Hierarchical Dirichlet Process (HDP) based model, which can automatically learn $|Z|$ from the data. Specifically, we introduce a global probability measure G_0 over the region-specific measure G_r , where the mixing proportions of G_0 and G_r are τ and θ_r , respectively. In a finite model, the number of topics $|Z|$ is a positive integer, and τ is drawn from the Dirichlet distribution $Dir(\gamma/|Z|, \dots, \gamma/|Z|)$. After that, each θ_r is drawn from the Dirichlet distribution $Dir(\alpha\tau)$, where α is a concentration parameter that controls the variance of the draws around τ . Taking $|Z| \rightarrow \infty$, the global topic distribution $\tau \sim Dir(\gamma/|Z|)$, and we have $G_r \sim DP(\alpha, G_0)$, a Dirichlet process with base measure G_0 and concentration parameter α . Finally, the finite model becomes an HDP. More details can be found in [TJBB06].

Based on the aforementioned intuitions and notations, EW^4 generates the day, time, words, and POI for each tweet posted by a user in an integrated manner. The generative process is described in Algorithm 3, and the graphical model is shown in Figure 5.5.

5.5 Parameter Estimation

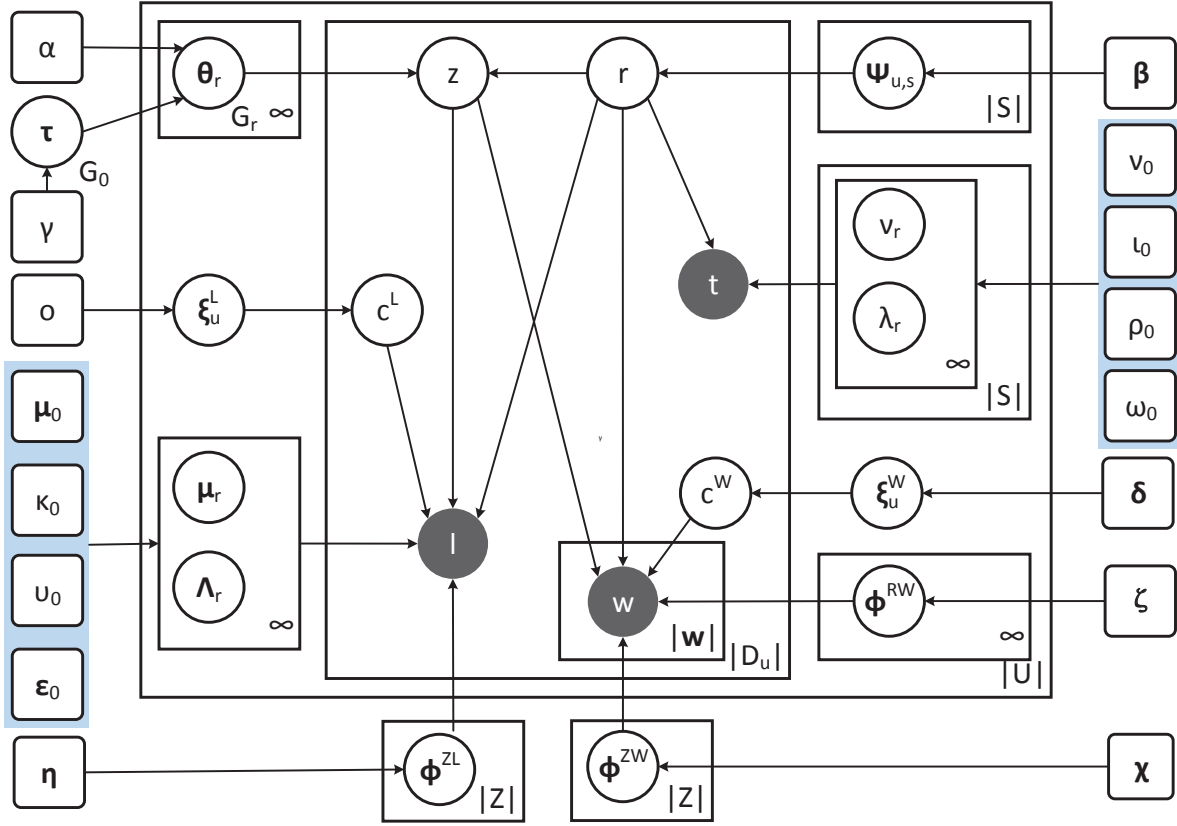
In this section, we introduce the parameter estimation approaches for W^4 and EW^4 , respectively.

Algorithm 3: Generative Process of EW⁴

```

1 for each user  $u$ , ( $u = 1, \dots, |U|$ ) do
2   Draw POI switch Bernoulli distribution  $\xi_u^L \sim \text{Beta}(o)$ ;
3   Draw word switch Bernoulli distribution  $\xi_u^W \sim \text{Beta}(\delta)$ ;
4 end
5 Draw global topic multinomial distribution  $\tau \sim \text{Dir}(\gamma/|Z|)$ ;
6 for each user  $u$ , ( $u = 1, \dots, |U|$ ) do
7   for the tweet  $d_i \in D_u$  do
8     Draw a region  $r$  based on  $\text{CRP}(r|u, s_i)$ , where  $s_i$  is the day of  $d_i$ ;
9     if  $r \notin R_u$  then
10      for each day  $s \in S$  do
11        Draw time distribution
12          $\mathcal{N}(\nu_{r,s}, \lambda_{r,s}^{-1}) \sim \text{Normal} - \text{Gamma}(\nu_0, \iota_0, \rho_0, \omega_0)$ ;
13      end
14      Draw geographical distribution
15        $\mathcal{N}(\mu_r, \Lambda_r^{-1}) \sim \text{Normal} - \text{Wishart}(\mu_0, \kappa_0, \mathbf{v}_0, \epsilon_0)$ ;
16      Draw region-specific topic multinomial distribution  $\theta_r \sim \text{Dir}(\alpha\tau)$ ;
17      Draw region-specific word distribution  $\phi_r^{RW} \sim \text{Dir}(\zeta)$ ;
18      Add  $r$  into  $R_u$ ;
19    end
20    Draw a topic  $z \sim \theta_r$ ;
21    if  $z \notin Z$  then
22      Draw topic-specific POI multinomial distribution  $\phi_z^{ZL} \sim \text{Dir}(\eta)$ ;
23      Draw topic-specific multinomial distribution  $\phi_z^{ZW} \sim \text{Dir}(\chi)$ ;
24      Add  $z$  into  $Z$ ;
25    end
26    Draw a time  $t \sim \mathcal{N}(td(t, \nu_{r,s_i})|\nu_{r,s_i}, \lambda_{r,s_i}^{-1})$ ;
27    Draw a POI switch  $c^L \sim \xi_u^L$ ;
28    if  $c^L = 0$  then
29      Draw a POI  $\ell \sim \mathcal{N}(\mu_r, \Lambda_r^{-1})$ ;
30    end
31    else
32      Draw a POI  $\ell \sim \phi_z^{ZL}$ ;
33    end
34    Draw a word switch  $c^W \sim \xi_u^W$ ;
35    if  $c^W = 0$  then
36      for the  $k$ -th word ( $k = 1, \dots, |\mathbf{w}_i|$ ) do
37        Draw a word  $w \sim \phi_r^{RW}$ ;
38      end
39    end
40    else
41      for the  $k$ -th word ( $l = 1, \dots, |\mathbf{w}_i|$ ) do
42        Draw a word  $w \sim \phi_z^{ZW}$ ;
43      end
44    end
45  end
46 end

```


 Figure 5.5: The Graphical Representation of EW^4

5.5.1 Parameter Estimation for W^4

As shown in Figure 5.4, there are two latent variables in W^4 , namely, region r and topic z . The joint probability over tweet $d = \{u_d, \ell_d, \mathbf{w}_d, t_d, s_d\}$, region r , and topic z , can be written as:

$$\begin{aligned}
 p(d, r, z) &= p(u_d, r, z, s_d, t_d, \ell_d, \mathbf{w}_d) \\
 &= p(u_d)p(s_d|u_d)p(t_d|u_d, r, s_d)p(r|u_d, s_d) \\
 &\quad p(z|u_d, r)p(\ell_d|r, z)p(\mathbf{w}_d|r, z),
 \end{aligned} \tag{5.2}$$

where

$$\begin{aligned}
 p(\ell_d|r, z) &= \kappa p(\ell_d|z) + (1 - \kappa)p(\ell_d|r), \\
 p(\mathbf{w}_d|r, z) &= \prod_{w \in \mathbf{w}_d} (\lambda p(w|z) + (1 - \lambda)p(w|r))^{c(w, \mathbf{w}_d)}.
 \end{aligned}$$

In the above equation, $c(w, \mathbf{w}_d)$ is the count of word w in \mathbf{w}_d .

This model has a set of parameters $p(r|u, s)$, $p(z|u, r)$, $\nu_{u,s,r}$, $\sigma_{u,s,r}$, $p(\ell|z)$, $\boldsymbol{\mu}_{u,s,r}$, $\boldsymbol{\Sigma}_{u,s,r}$, $p(w|z)$ and $p(w|r)$. Denoting them by Ψ , we have the log-likelihood of the historical data D :

$$\mathcal{L}(\Psi; D) = \log p(D|\Psi). \quad (5.3)$$

We use Expectation-Maximization (EM) to find parameters Ψ that can maximize the log-likelihood of the historical data.

In the **E-step**, since there are two latent variables r and z in W^4 , we update their joint expectation $p(r, z|d)$ according to Bayes rule as Equation 5.4.

$$p(r, z|d) = \frac{p(d, r, z)}{p(d)} = \frac{p(d, r, z)}{\sum_r \sum_z p(d, r, z)}. \quad (5.4)$$

In the **M-step**, we find the new Ψ that can maximize the log-likelihood as follows:

$$p(r|u, s) = \frac{\sum_{d \in D_{u,s}} \sum_z p(r, z|d)}{\sum_{d \in D_{u,s}} \sum_z \sum_{r'} p(r', z|d)}, \quad (5.5)$$

where $D_{u,s}$ is the collection of tweets written by user u on the day s . We will not explain $D_{(\cdot)}$ unless necessary.

$$p(z|u, r) = \frac{\sum_{d \in D_u} p(r, z|d)}{\sum_{d \in D_u} \sum_{z'} p(r, z'|d)}, \quad (5.6)$$

$$\nu_{u,s,r} = \frac{\sum_{d \in D_{u,s}} \sum_z p(r, z|d) \cdot t_d}{\sum_{d \in D_{u,s}} \sum_z p(r, z|d)}, \quad (5.7)$$

$$\sigma_{u,s,r}^2 = \frac{\sum_{d \in D_{u,s}} \sum_z p(r, z|d) \cdot td^2(t_d, \nu_{u,s,r})}{\sum_{d \in D_{u,s}} \sum_z p(r, z|d)}, \quad (5.8)$$

where $td(t_1, t_2)$ is the difference between time in a day t_1 and t_2 , because the time in a day is cyclical. Note that given a collection of time, we can get two Gaussian distributions with different ν_r and λ_r , and the ν_r of the two distributions are 12-hour apart from each other. For example, ν_r for 1:00 and 23:00 can be either 0:00 or 12:00. Obviously, 0:00 is a better choice for the mean, since it is closer to 1:00 and 23:00 than 12:00, which leads to a greater λ_r value. Thus, between the two sets of ν_r and λ_r , we choose the ν_r , λ_r pair with the greater λ_r value as the mean and precision for the temporal Gaussian distribution.

Estimating $p(w|r)$ and $p(w|z)$ is not straightforward, because they are coupled by the sum in logarithm in the log-likelihood, *i.e.*, $\log(\lambda p(w|z) + (1 - \lambda)p(w|r))$. We solve this problem by applying Jensen's inequality [Jen06]. Because logarithm is a concave function, we have:

$$\begin{aligned} \log(\lambda p(w|z) + (1 - \lambda)p(w|r)) &\geq \lambda \log(p(w|z)) + (1 - \lambda) \log(p(w|r)), \\ \log(\kappa p(\ell|z) + (1 - \kappa)p(\ell|r)) &\geq \kappa \log(p(\ell|z)) + (1 - \kappa) \log(p(\ell|r)). \end{aligned}$$

By substituting the above two Equations into Equation 5.3, we have a lower bound of the log-likelihood. By maximizing the lower bound, we have:

$$p(w|r) = \frac{\sum_{d \in D_w} \sum_z c(w, \mathbf{w}_d) p(r, z|d)}{\sum_{w'} \sum_{d \in D_w} \sum_z c(w', \mathbf{w}_d) p(r, z|d)}, \quad (5.9)$$

$$p(w|z) = \frac{\sum_{d \in D_w} \sum_r c(w, \mathbf{w}_d) p(r, z|d)}{\sum_{w'} \sum_{d \in D_w} \sum_r c(w', \mathbf{w}_d) p(r, z|d)}, \quad (5.10)$$

$$\boldsymbol{\mu}_{u,r} = \frac{\sum_{d \in D_u} \sum_z p(r, z|d) \cdot \mathbf{c}d_{\ell_d}}{\sum_{d \in D_u} \sum_z p(r, z|d)}, \quad (5.11)$$

$$\boldsymbol{\Sigma}_{u,r} = \frac{\sum_{d \in D_u} \sum_z p(r, z|d) \cdot (\mathbf{c}d_{\ell_d} - \boldsymbol{\mu}_{u,r})^T (\mathbf{c}d_{\ell_d} - \boldsymbol{\mu}_{u,r})}{\sum_{d \in D_u} \sum_z p(r, z|d)}, \quad (5.12)$$

$$p(\ell|z) = \frac{\sum_{d \in D_\ell} \sum_r p(r, z|d)}{\sum_{d \in D_\ell} \sum_{z'} \sum_r p(r, z'|d)}. \quad (5.13)$$

Please note that the lower bound technique has no impact on the inference of other parameters, since they are surrounded in different logarithms. As a result, the derivative with respect to a parameter does not involve with the others.

5.5.2 Parameter Estimation for EW⁴

We employ collapsed Gibbs sampling to obtain samples of the hidden variable assignment, and to estimate the unknown parameters $\{\boldsymbol{\theta}, \boldsymbol{\psi}, \phi^{ZL}, \phi^{ZW}, \phi^{RW}, \boldsymbol{\xi}^L, \boldsymbol{\xi}^W, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\nu}, \boldsymbol{\lambda}\}$. There are four latent variables in the model, namely, region r , topic z , the switch for POI sampling c^L , and the switch for word sampling c^W . We initialize z , c^L and c^W for each tweet by random values. Since the personal regions of each user is generated based on CRP, we create a region for each user at the initialization step, and assign all the user's tweets to the region. Then, we use two-step Gibbs sampling to obtain the samples: region r_i and topic z_i of each tweet d_i are sampled in the first step, and the two switches c_i^L and c_i^W of each tweet d_i are sampled in the second step. For each set of variables, (*e.g.*, r_i and z_i), a Gibbs sampler computes the full conditional probability for their assignments conditioned on all the other assignments (*e.g.*, \mathbf{r}_{-i} , \mathbf{z}_{-i}), while the assignments of the other set of variables (*e.g.*, \mathbf{c}^L , \mathbf{c}^W) are fixed.

For the first-step sampling, we derive the update equation for region r_i and topic z_i for tweet d_i based on the following equation:

$$P(r_i = r, z_i = z | \mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot) \propto \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} \quad (5.14)$$

However, with different c^L and c^W assignments, the generative processes of POI and words of a tweet are different, which makes it difficult to get an update equation applicable to all tweets. To solve this problem, we divide the tweet collections D into four subsets based on the assignments of c^L and c^W , namely, $D_{1,1}$, $D_{1,0}$, $D_{0,1}$ and $D_{0,0}$, where D_{c_1, c_2} denotes the collection of tweets with $c^L = c_1$ and $c^W = c_2$. Comparing to that for D , it is much easier to obtain the update equation for tweets in each subset, since given a subset, the generative processes of POIs and words of its tweets are fixed. Then, we compute the conditional probability for each set.

We first focus on $D_{1,1}$, in which tweets' POIs and words are sampled according to the topic-specific distributions over POIs and words, respectively. We can derive the sampling equation 5.14 for r_i and z_i of tweet $d_i \in D_{1,1}$ as follows:

- If $r \notin R_{u_i}$, then

$$\begin{aligned} \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{\beta}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{\alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \frac{n_{z, \ell_i, -i}^{ZL} + \eta}{\sum_{\ell'=1}^{|L|} n_{z, \ell', -i}^{ZL} + |L|\eta} \cdot \\ &\mathcal{N}(td(t_i, \nu_0) | \nu_0, \lambda_0^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_{w, \mathbf{w}_i}} (n_{z, w, -i}^{ZW} + \chi + y)}{\prod_{y=0}^{c_{(\cdot), \mathbf{w}_i}} \sum_{w=1}^{|V|} (n_{z, w, -i}^{ZW} + |V|\chi + y)} \end{aligned} \quad (5.15)$$

- If $r \in R_{u_i}$, then

$$\begin{aligned} \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{n_{s_i, r, -i}^{SR}}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{n_{r, z, -i}^{RZ} + \alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \frac{n_{z, \ell_i, -i}^{ZL} + \eta}{\sum_{\ell'=1}^{|L|} n_{z, \ell', -i}^{ZL} + |L|\eta} \cdot \\ &\mathcal{N}(td(t_i, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_{w, \mathbf{w}_i}} (n_{z, w, -i}^{ZW} + \chi + y)}{\prod_{y=0}^{c_{(\cdot), \mathbf{w}_i}} \sum_{w=1}^{|V|} (n_{z, w, -i}^{ZW} + |V|\chi + y)} \end{aligned} \quad (5.16)$$

where ℓ_i , t_i and \mathbf{w}_i are the POI, time and words of tweet d_i ; c_{w, \mathbf{w}_i} is the occurrence times of word w in \mathbf{w}_i , and $c_{(\cdot), \mathbf{w}_i}$ is the length of \mathbf{w}_i . Other parameters involved in sampling are omitted in Equation 5.14. If $z \notin Z$, $\forall \ell n_{z, \ell}^{ZL} = 0$, $\forall w n_{z, w}^{ZW} = 0$, and $\forall r n_{r, z}^{RZ} = 0$. $\mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1})$ is the likelihood that the temporal Gaussian distribution of r generates time t .

We estimate the parameters ν_r , λ_r for the temporal Gaussian distribution based on the time of the tweets assigned to region r , where the time collection is denoted by $\{t\}_r$. The posterior of ν_r , λ_r can be derived as follows:

$$\begin{aligned} P(\nu_r, \lambda_r | \{t\}_r, \cdot) &\propto P(\{t\}_r | \nu_r, \lambda_r) \cdot \mathcal{NG}(\nu_r, \lambda_r | \nu_0, \rho_0, \lambda_0) \\ &= \prod_{t \in \{t\}_r} \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \mathcal{NG}(\nu_r, \lambda_r | \nu_0, \rho_0, \lambda_0) \\ &= \mathcal{NG}(\nu_r, \lambda_r | \nu'_r, \lambda'_r, \rho'_r, \lambda'_r), \end{aligned} \quad (5.17)$$

where $\mathcal{NG}(\cdot)$ is Normal-Gamma function, and the parameters $\nu'_r, \iota'_r, \rho'_r, \lambda'_r$ are estimated as follows:

$$\begin{aligned}
 \nu'_r &= \frac{\iota_0 \nu_0 + |\{t\}_r| \bar{t}_r}{\iota_0 + |\{t\}_r|} \\
 \iota'_r &= \iota_0 + |\{t\}_r| \\
 \rho'_r &= \rho_0 + \frac{|\{t\}_r|}{2} \\
 \omega'_r &= \omega_0 + \frac{1}{2} \sum_{t_k \in T_r} td(t_k - \bar{t}_r)^2 + \frac{\iota_0 |\{t\}_r| \cdot td(\bar{t}_r - \nu_0)^2}{2(\iota_0 + |\{t\}_r|)}
 \end{aligned} \tag{5.18}$$

In the above equation, \bar{t}_r is the average time of tweets in region r . Given Equations 5.17 and 5.18, we can update ν_r, λ_r as follows:

$$\begin{aligned}
 \nu_r &= \nu'_r \\
 \lambda_r &= \frac{\rho'_r}{\omega'_r}
 \end{aligned} \tag{5.19}$$

The details of the equations about Gaussian parameters can be found in [Ras99, Mur07].

Note that given a collection of time, we can get two Gaussian distributions with different ν_r and λ_r , and the ν_r of the two distributions are 12-hour apart from each other. For example, ν_r for 1:00 and 23:00 can be either 0:00 or 12:00. Obviously, 0:00 is a better choice for the mean, since it is closer to 1:00 and 23:00 than 12:00, which leads to a greater λ_r value. Thus, between the two sets of ν_r and λ_r , we choose the ν_r, λ_r pair with the greater λ_r value as the mean and precision for the temporal Gaussian distribution.

For tweet d_i in $D_{1,0}$, the Equation 5.14 for sampling the region r_i and topic z_i is as follows:

- If $r \notin R_{u_i}$, then

$$\begin{aligned}
 \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{\beta}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{\alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \frac{n_{z, \ell_i, -i}^{ZL} + \eta}{\sum_{\ell'=1}^{|L|} n_{z, \ell', -i}^{ZL} + |L| \eta} \cdot \\
 &\mathcal{N}(td(t_i, \nu_0) | \nu_0, \lambda_0^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_w, w_i} (\zeta + y)}{\prod_{y=0}^{c_{(\cdot), w_i}} \sum_{w=1}^{|V|} (|V| \zeta + y)}
 \end{aligned} \tag{5.20}$$

- If $r \in R_{u_i}$, then

$$\begin{aligned} \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{n_{s_i, r, -i}^{SR}}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{n_{r, z, -i}^{RZ} + \alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \frac{n_{z, \ell_i, -i}^{ZL} + \eta}{\sum_{\ell'=1}^{|L|} n_{z, \ell', -i}^{ZL} + |L| \eta} \cdot \\ &\mathcal{N}(td(t_i, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_{w, \mathbf{w}_i}} (n_{r, w, -i}^{RW} + \zeta + y)}{\prod_{y=0}^{c_{(\cdot), \mathbf{w}_i}} \sum_{w=1}^{|V|} (n_{r, w, -i}^{RW} + |V| \zeta + y)} \end{aligned} \quad (5.21)$$

For tweet d_i in $D_{0,1}$, the Equation 5.14 for sampling the region r_i and topic z_i is as follows:

- If $r \notin R_{u_i}$, then

$$\begin{aligned} \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{\beta}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{\alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \mathcal{N}(\ell_i | \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0^{-1}) \cdot \\ &\mathcal{N}(td(t_i, \nu_0) | \nu_0, \lambda_0^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_{w, \mathbf{w}_i}} (n_{z, w, -i}^{ZW} + \chi + y)}{\prod_{y=0}^{c_{(\cdot), \mathbf{w}_i}} \sum_{w=1}^{|V|} (n_{z, w, -i}^{ZW} + |V| \chi + y)} \end{aligned} \quad (5.22)$$

- If $r \in R_{u_i}$, then

$$\begin{aligned} \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{n_{s_i, r, -i}^{SR}}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{n_{r, z, -i}^{RZ} + \alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \mathcal{N}(\ell_i | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) \cdot \\ &\mathcal{N}(td(t_i, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_{w, \mathbf{w}_i}} (n_{z, w, -i}^{ZW} + \chi + y)}{\prod_{y=0}^{c_{(\cdot), \mathbf{w}_i}} \sum_{w=1}^{|V|} (n_{z, w, -i}^{ZW} + |V| \chi + y)} \end{aligned} \quad (5.23)$$

The parameters $\boldsymbol{\mu}_r$ and $\boldsymbol{\Lambda}_r$ for the spatial Gaussian distribution of region r can be estimated based on the coordinates of tweet POIs assigned to r with $c^L = 0$. We use $\{\mathbf{cd}\}_r$ to denote the collection of such coordinates, and obtain the posterior of $\boldsymbol{\mu}_r$ and $\boldsymbol{\Lambda}_r$ as follows:

$$\begin{aligned} P(\boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r | \{\mathbf{cd}\}_r, \cdot) &\propto P(\{\mathbf{cd}\}_r | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r) \cdot \mathcal{NW}(\boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r | \boldsymbol{\mu}_0, \kappa_0, \nu_0, \boldsymbol{\epsilon}_0) \\ &= \prod_{\mathbf{cd} \in \{\mathbf{cd}\}_r} \mathcal{N}(\mathbf{cd} | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) \cdot \mathcal{NW}(\boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r | \boldsymbol{\mu}_0, \kappa_0, \nu_0, \boldsymbol{\epsilon}_0) \\ &= \mathcal{NW}(\boldsymbol{\mu}_r', \kappa_r', \nu_r', \boldsymbol{\epsilon}_r'), \end{aligned} \quad (5.24)$$

where $\mathcal{NW}(\cdot)$ is Normal-Wishart function, and $\boldsymbol{\mu}_r', \kappa_r', \nu_r', \boldsymbol{\epsilon}_r'$ are estimated as follows:

$$\begin{aligned}
 \boldsymbol{\mu}'_r &= \frac{\kappa_0 \boldsymbol{\mu}_0 + |\{\mathbf{cd}\}_r| \overline{\mathbf{cd}}_r}{\kappa_0 + |\{\mathbf{cd}\}_r|} \\
 \kappa'_r &= \kappa_0 + |\{\mathbf{cd}\}_r| \\
 \nu'_r &= \nu_0 + |\{\mathbf{cd}\}_r| \\
 \boldsymbol{\epsilon}'_r &= \boldsymbol{\epsilon}_0 + \sum_{\mathbf{cd} \in \{\mathbf{cd}\}_r} (\mathbf{cd} - \overline{\mathbf{cd}}_r)(\mathbf{cd} - \overline{\mathbf{cd}}_r)^T + \frac{\kappa_0 |\{\mathbf{cd}\}_r|}{\kappa_0 + |\{\mathbf{cd}\}_r|} (\boldsymbol{\mu}_0 - \overline{\mathbf{cd}}_r)(\boldsymbol{\mu}_0 - \overline{\mathbf{cd}}_r)^T
 \end{aligned} \tag{5.25}$$

In the above equation, $\overline{\mathbf{cd}}_r$ is the average coordinate of $\{\mathbf{cd}\}_r$. Given Equations 5.24 and 5.25, we can update $\boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r$ as follows:

$$\begin{aligned}
 \boldsymbol{\mu}_r &= \boldsymbol{\mu}'_r \\
 \boldsymbol{\Lambda}_r &= \nu'_r \cdot \boldsymbol{\epsilon}'_r{}^{-1}
 \end{aligned} \tag{5.26}$$

Last, for tweet d_i in $D_{0,0}$, the Equation 5.14 for sampling the region r_i and topic z_i is as follows:

- If $r \notin R_{u_i}$, then

$$\begin{aligned}
 \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{\beta}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{\alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \mathcal{N}(\ell_i | \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0^{-1}) \cdot \\
 &\mathcal{N}(td(t_i, \nu_0) | \nu_0, \lambda_0^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_w, w_i} (n_{r, w, -i}^{RW} + \zeta + y)}{\prod_{y=0}^{c(\cdot), w_i} \sum_{w=1}^{|V|} (n_{r, w, -i}^{RW} + |V| \zeta + y)} \tag{5.27}
 \end{aligned}$$

- If $r \in R_{u_i}$, then

$$\begin{aligned}
 \frac{P(\mathbf{r}, \mathbf{z}, \cdot)}{P(\mathbf{r}_{-i}, \mathbf{z}_{-i}, \cdot)} &= \frac{n_{s_i, r, -i}^{SR}}{\sum_{r'=1}^{|R_{u_i}|} n_{s_i, r', -i}^{SR} + \beta} \cdot \frac{n_{r, z, -i}^{RZ} + \alpha \tau_z}{\sum_{z'=1}^{|Z|} n_{r, z', -i}^{RZ} + \alpha} \cdot \mathcal{N}(\ell_i | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) \cdot \\
 &\mathcal{N}(td(t_i, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \frac{\prod_{w=1}^{|V|} \prod_{y=0}^{c_w, w_i} (n_{r, w, -i}^{RW} + \zeta + y)}{\prod_{y=0}^{c(\cdot), w_i} \sum_{w=1}^{|V|} (n_{r, w, -i}^{RW} + |V| \zeta + y)} \tag{5.28}
 \end{aligned}$$

When the number of topics $|Z|$ changes, and when an sampling iteration is finished, we sample new global topic distribution $\boldsymbol{\tau}$ based on the following equation:

$$\boldsymbol{\tau} \sim Dir(\{m_z\}_z, \gamma), \tag{5.29}$$

where m_z is the number of tweets that are assigned to topic z . More details of the equation can be found in [Hei11].

After sampling region r_i and topic z_i for all tweets $d_i \in D$, we sample c_i^L and c_i^W based on the following posterior probability distributions:

$$P(c_i^L = 0 | \mathbf{c}_{-i}^L, \cdot) \propto \frac{n_{u_i, (0), -i}^{UCL} + o_0}{n_{u_i, (0), -i}^{UCL} + n_{u_i, (1), -i}^{UCL} + o_0 + o_1} \cdot \mathcal{N}(\ell_i | \mu_{r_i}, \Lambda_{r_i}^{-1}) \quad (5.30)$$

$$P(c_i^L = 1 | \mathbf{c}_{-i}^L, \cdot) \propto \frac{n_{u_i, (1), -i}^{UCL} + o_1}{n_{u_i, (0), -i}^{UCL} + n_{u_i, (1), -i}^{UCL} + o_0 + o_1} \cdot \frac{n_{z_i, \ell_i, -i}^{ZL} + \eta}{\sum_{\ell'=1}^{|\mathcal{L}|} n_{z_i, \ell', -i}^{ZL} + |\mathcal{L}| \eta} \quad (5.31)$$

$$P(c_i^W = 0 | \mathbf{c}_{-i}^W, \cdot) \propto \frac{n_{u_i, (0), -i}^{UCW} + \delta_0}{n_{u_i, (0), -i}^{UCW} + n_{u_i, (1), -i}^{UCW} + \delta_0 + \delta_1} \cdot \frac{\prod_{w=1}^{|\mathcal{V}|} \prod_{y=0}^{c_{w, w}} (n_{r_i, w, -i}^{RW} + \zeta + y)}{\prod_{y=0}^{c_{(\cdot), w}} \sum_{w=1}^{|\mathcal{V}|} (n_{r_i, w, -i}^{RW} + V\zeta + y)} \quad (5.32)$$

$$P(c_i^W = 1 | \mathbf{c}_{-i}^W, \cdot) \propto \frac{n_{u_i, (1), -i}^{UCW} + \delta_1}{n_{u_i, (0), -i}^{UCW} + n_{u_i, (1), -i}^{UCW} + \delta_0 + \delta_1} \cdot \frac{\prod_{w=1}^{|\mathcal{V}|} \prod_{y=0}^{c_{w, w}} (n_{z_i, w, -i}^{ZW} + \chi + y)}{\prod_{y=0}^{c_{(\cdot), w}} \sum_{w=1}^{|\mathcal{V}|} (n_{z_i, w, -i}^{ZW} + V\chi + y)} \quad (5.33)$$

where u_i is the user of tweet d_i , and r_i and z_i are the region and topic assigned d_i in the first step.

After sampling a sufficient number of iterations, we calculate the parameters as follows:

$$\boldsymbol{\psi}_{u, s, r} = P(r|s) = \frac{n_{s, r}^{SR} + \frac{\beta}{|R_u|}}{\sum_{r'=1}^{|R_u|} n_{s, r'}^{SR} + \beta} \quad (5.34)$$

$$\boldsymbol{\theta}_r = P(z|r) = \frac{n_{r, z}^{RZ} + \alpha \tau_z}{\sum_{z'=1}^{|\mathcal{Z}|} n_{r, z'}^{RZ} + \alpha} \quad (5.35)$$

$$\boldsymbol{\xi}_u^L = P(c^L = 1|u) = \frac{n_{u, (1)}^{UCL} + o_1}{n_{u, (0)}^{UCL} + n_{u, (1)}^{UCL} + o_0 + o_1} \quad (5.36)$$

$$\xi_u^W = P(c^W = 1|u) = \frac{n_{u,(1)}^{UCW} + \delta_1}{n_{u,(0)}^{UCW} + n_{u,(1)}^{UCW} + \delta_0 + \delta_1} \quad (5.37)$$

$$\phi_z^{ZL} = P(\ell|z) = \frac{n_{z,l}^{ZL} + \eta}{\sum_{\ell'=1}^{|L|} n_{z,\ell'}^{ZL} + |L|\eta} \quad (5.38)$$

$$\phi_z^{ZW} = P(w|z) = \frac{n_{z,w}^{ZW} + \chi}{\sum_{w'=1}^{|V|} n_{z,w'}^{ZW} + |W|\chi} \quad (5.39)$$

$$\phi_r^{RW} = P(w|r) = \frac{n_{r,w}^{RW} + \zeta}{\sum_{w'=1}^{|V|} n_{r,w'}^{RW} + |W|\zeta} \quad (5.40)$$

5.5.2.1 About the Hyper-parameters

We can give hyper priors for the hyper-parameters of our model, and sampling these hyper-parameters during Gibbs sampling. For example, we can give vague Gamma priors for α , κ_0 , ν_0 , etc., and give Gaussian priors for μ_0 and ν_0 [Ras99, TJBB06]. However, these hyper priors will make the model more complicated and slow the parameter estimation. Thus, in this article, we empirically set these hyper-parameters at fix values. Specifically, we fix $\alpha = 200$, $\gamma = 0.25$, $o = \delta = 0.5$, $\beta = \nu_0 = \rho_0 = 0.1$, $\eta = \chi = \zeta = \kappa_0 = \iota_0 = 0.001$, $\epsilon = 0.0001 \cdot I_2$, where I_2 is 2×2 identity matrix. Since the mobility regions and active time of different users can be greatly diverse from each other, we set user-specific values for μ_0 , ν_0 and ω_0 : μ_0 is the mean coordinates of a user's visited POIs, ν_0 is the mean of a user's visiting time, and ω_0 is 0.01 times the variance of user's visiting time.

5.6 Applications

The proposed model W^4 and EW^4 involve with four aspects of user's mobility behavior, namely, who, where, when and what, which enables us to infer the information in some aspects given the information in other aspects. Thus, they have a variety of applications, and we name some of them as examples including the requirement-aware POI recommendation:

POI prediction for tweet. Given a tweet with its text content, user id, and posting time, the task of *POI prediction* is to predict the most likely POI at which this tweet is posted. It has been shown [CML11, CCLS11] that geographical POIs can be used to predict user's behavior, discover users' interest, and deliver POI-based advertisement

or content. However, it is reported that only 1%–2% of tweets have geographical POIs explicitly attached. Hence, POI prediction for tweets is a very important application.

A number of methods have been proposed for this task [LSdV⁺11, KMO11, EOSX10, WB11, HAG⁺12]. The studies [LSdV⁺11, KMO11] build language models for each candidate POI, and make prediction based on these language models. They are designed to predict POI identifier for a text. Instead of predicting a POI for a give text, the work [WB11] segments the world into grids, and employs supervised models, such as Naive Bayes, to predict a grid for a given text. The recent proposals [HAG⁺12, AHS13] present approaches for predicting geographic coordinates of a text from a user.

Since EW⁴ incorporates both POI identifiers and geographic coordinates, we can make both kinds of predictions for a text from a user, namely, predicting POI identifiers [LSdV⁺11, KMO11] and geographic coordinates [HAG⁺12]. Our method is also able to take the time factor into consideration.

Formally, given a user u , day s , time t , and words \mathbf{w}_d , a POI ℓ (represented with both POI identifier and geographic coordinates) is predicted by maximizing $P(\ell|u, s, t, \mathbf{w}_d)$. Specifically, we calculate $P(\ell|u, s, t, \mathbf{w}_d)$ for each candidate POI ℓ as follows:

$$\begin{aligned}
 P(\ell|u, s, t, \mathbf{w}_d) &= \frac{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(u, s, t, r, z, \mathbf{w}_d, \ell)}{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \sum_{\ell'=1}^{|\mathcal{L}|} P(u, s, t, r, z, \mathbf{w}_d, \ell')} \\
 &\propto \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(r|u, s) P(z|r) P(t|r) P(\ell|r, z) P(\mathbf{w}_d|r, z) \\
 &= \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \psi_{u,s,r} \cdot \theta_{r,z} \cdot \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \\
 &\quad (\xi_{u,0}^L \cdot \mathcal{N}(\ell | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) + \xi_{u,1}^L \cdot \phi_{z,\ell}^{ZL}) \cdot \\
 &\quad \prod_{w \in \mathbf{w}_d} (\xi_{u,0}^W \cdot \phi_{r,w}^{RW} + \xi_{u,1}^W \cdot \phi_{z,w}^{ZW})
 \end{aligned} \tag{5.41}$$

Requirement-aware POI recommendation. POI recommendation aims to recommend new POIs for users to visit at given time. Previous studies only rely on users' historical visiting information [YLL11, CYKL12], neglecting the specific needs at a given time. EW⁴ is able to utilize both the time and the needs (in the form of short text), to make more accurate recommendation. Given a user u , day s , time t and words \mathbf{w}_d that describe the need, the candidate POIs are ranked by $P(\ell|u, s, t, \mathbf{w}_d)$, defined by Equation 5.41, and the top-ranked ones are returned as results.

Activity prediction. EW^4 is able to predict the activity of a user at a given time. Given a user u and time s and t , the words describing the activity are ranked by:

$$\begin{aligned}
P(w|u, s, t) &= \frac{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(u, s, t, r, z, w)}{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \sum_{w'=1}^{|V|} P(u, s, t, r, z, w')} \\
&\propto \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(r|u, s) P(z|r) P(t|r) P(w|r, z) \\
&= \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \psi_{u,s,r} \cdot \theta_{r,z} \cdot \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \\
&\quad (\xi_{u,0}^W \cdot \phi_{r,w}^{RW} + \xi_{u,1}^W \cdot \phi_{z,w}^{ZW})
\end{aligned} \tag{5.42}$$

User prediction. User prediction aims to predict the likelihood of a user visiting a POI at a given time. This could be very useful for merchants for planning purpose, or for them to target on specific costumers. Specifically, given POI ℓ , day s , and time t , we rank candidate users by $P(u|\ell, s, t)$, which is calculated as follows:

$$\begin{aligned}
P(u|\ell, s, t) &= \frac{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(u, s, t, r, z, \ell)}{\sum_{u'=1}^{|U|} \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_{u'}|} P(u', s, t, r, z, \ell)} \\
&\propto \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(r|u, s) P(z|r) P(t|r) P(\ell|r, z) \\
&= \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \psi_{u,s,r} \cdot \theta_{r,z} \cdot \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \\
&\quad (\xi_{u,0}^L \cdot \mathcal{N}(\ell | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) + \xi_{u,1}^L \cdot \phi_{z,\ell}^{ZL})
\end{aligned} \tag{5.43}$$

Note that previous studies on user mobility modeling (*e.g.*, [CML11]) can also be used for user prediction, if we use POI and time as input, and find the user who can maximize the likelihood.

POI prediction for user. This task is to predict the place where a user stays at a given time. This would be useful for logistic planning, *e.g.*, to arrange a meeting with a user or a group of users, and POI-based advertisement delivery. Formally, given a user u and time t , we aim to rank all candidate POIs based on $P(\ell|u, s, t)$, which is calculated

by:

$$\begin{aligned}
 P(\ell|u, s, t) &= \frac{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(u, s, t, r, z, \ell)}{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \sum_{\ell'=1}^{|\mathcal{L}|} P(u, s, t, r, z, \ell')} \\
 &\propto \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(r|u, s) P(z|r) P(t|r) P(\ell|r, z) \\
 &= \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \psi_{u,s,r} \cdot \theta_{r,z} \cdot \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \\
 &\quad (\xi_{u,0}^L \cdot \mathcal{N}(\ell | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) + \xi_{u,1}^L \cdot \phi_{z,\ell}^{ZL})
 \end{aligned} \tag{5.44}$$

Tweets recommendation. This task is to recommend tweets that are interested to a user based on the user’s topic preferences, current POI and time. Specifically, given user u , day s , time t , and POI ℓ , we aim to rank tweets by considering $P(\mathbf{w}_d|u, s, t, \ell)$, where \mathbf{w}_d is the word vector of a candidate tweets, and

$$\begin{aligned}
 P(\mathbf{w}_d|u, s, t, \ell) &= \frac{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(u, s, t, r, z, \mathbf{w}_d, \ell)}{\sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \sum_{\mathbf{w}'_d} P(u, s, t, r, z, \mathbf{w}'_d, \ell)} \\
 &\propto \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} P(r|u, s) P(z|r) P(t|r) P(\ell|r, z) P(\mathbf{w}_d|r, z) \\
 &= \sum_{z=1}^{|Z|} \sum_{r=1}^{|R_u|} \psi_{u,s,r} \cdot \theta_{r,z} \cdot \mathcal{N}(td(t, \nu_r) | \nu_r, \lambda_r^{-1}) \cdot \\
 &\quad (\xi_{u,0}^L \cdot \mathcal{N}(\ell | \boldsymbol{\mu}_r, \boldsymbol{\Lambda}_r^{-1}) + \xi_{u,1}^L \cdot \phi_{z,\ell}^{ZL}) \cdot \\
 &\quad \prod_{w \in \mathbf{w}_d} (\xi_{u,0}^W \cdot \phi_{r,w}^{RW} + \xi_{u,1}^W \cdot \phi_{z,w}^{ZW})
 \end{aligned} \tag{5.45}$$

5.7 Experiments

We evaluate the proposed models in this section. We first evaluate the accuracy of EW⁴ for the application of *POI prediction for tweets* and *requirement-aware POI recommendation* in Section 5.7.2 and Section 5.7.3 by comparing with several state-of-the-art baseline methods. Then, we present the discovered mobility pattern of an example user in Section 5.7.4. Results of other example applications of EW⁴ are reported in Section 5.7.5.

5.7.1 Dataset

Two real-world datasets are used in the experiments, namely, *WW* dataset and *USA* dataset.

Table 5.3: Statistics of Datasets

| | <i>WW</i> | <i>USA</i> |
|---------------------------|-----------|------------|
| Number of users | 3,883 | 4,122 |
| Number of POIs | 60,962 | 35,989 |
| Number of tweets/messages | 89,007 | 171,768 |

WW Dataset. Foursquare users can associate their accounts to Twitter, so that when they make check-ins in Foursquare, corresponding tweets will be posted in Twitter. Using the streaming API provided by Twitter [Twib], we collect 3,478,394 Foursquare check-ins from November 1, 2012 to February 13, 2013, among which 1,322,437 contains shouts (short messages) in English. We examine the users who posted English shouts, and remove the inactive users who visited less than 5 different POIs. Since users may check in when traveling to new places, and incorporating such check-ins will make it hard to estimate personal regions. Thus, we filter out the outlier check-ins as follows: we train GMM for each user, and remove invalid Gaussian components whose weights are smaller than 0.1. Check-ins that are most close to these invalid components are deleted. We iterate this process for each user. In the end, 89,007 check-ins are left after pre-processing. We refer to this dataset as *WW* (World-wide) dataset as the tweets are from users in different countries.

USA dataset. This dataset is the GeoText [Geo] (Geo-tagged Microblog Corpus) published by researchers from Carnegie Mellon University [EOSX10], which comprises messages from geo-located microblog users approximately in the United States. Each message is associated with its geographic coordinates. To map the geographic coordinates of each message to a POI identifier, we crawl the geographic coordinates of POIs in United States from Foursquare, and map the coordinates of each message to its nearest POI. After that, we apply the same pre-processing with *WW* to this dataset.

We remove stop-words from the text of both datasets. The statistics of the datasets after pre-processing is shown in Table 5.3. For each dataset, we randomly split the documents (tweets or messages) into three collections in proportion of 8:1:2 as the training set, development set, and testing set, respectively.

5.7.2 POI Prediction for Tweets

Given a tweet with its text content, user id, and posting time, the task of *POI prediction* is to predict the most likely POI at which this tweet is posted.

5.7.2.1 Evaluation Metrics

To evaluate the prediction performance of different models, we use two metrics, namely, prediction accuracy (*Acc*) and average error distance (*Dis*).

Prediction accuracy (Acc) is the percentage of tweets for which the predicted locations are exactly the true location among all tweets in the test set.

Average error distance (Dis) is the average of the Euclidian distance between the predicted geographic coordinates and the true geographic coordinates for all tweets in the test set.

Note that Acc and Dis are different—it is possible that the number of correctly predicted tweets is similar, but the wrongly predicted locations are deviated from the true locations very differently for different methods. Apparently, larger Acc and smaller Dis indicate better prediction performance.

5.7.2.2 Evaluated methods

We compare our model with 6 baseline methods to evaluate the performance, including the state-of-the-art models for predicting locations for text.

KL-divergence based Model (KL) [LSdV⁺11, KMO11]. This method builds language models (LM) for each candidate location during training. Given a test text, it computes the KL-divergence between the LM of the test text and the LM of each candidate location. The location with smallest KL-divergence is returned as the prediction result, and its coordinates are used to calculate the error distance.

Mean Coordinates (Mean). This model estimates the mean coordinates of visited locations for each user. Given a tweet, it returns the location that is closest to its author’s mean coordinates as the prediction result.

Popular Location (Pop). This model first finds out the location for each user that she visited most frequently. Given a tweet, it returns the most frequently visited location of its author as the prediction result.

Topic+Region Model (TR) [HAG⁺12]. This model captures the user preference over latent regions and topics. The locations, which are treated as geographic coordinates, are generated from the Gaussian distributions of regions, and words are generated based on the topics and regions. In addition, the latent regions in this model are not personal. Given a tweet from a user, TR can predict the geographic coordinates of the tweet.

Hierarchical Geographical Model (HG) [AHS13]. This model organizes geographical regions in a hierarchy, where regions in lower level are more specific w.r.t geographical area and topics. Each tweet is generated by a path from the root region to a leaf region, while the text content is drawn based on topics and the language model of the selected leaf region. Similar to that of TR, regions in HG are also global.

Table 5.4: Comparison of baseline methods with $EW_{\setminus T}^4$ and EW^4

| Factors | KL | Mean | Pop | TR | HG | ST | W^4 | $EW_{\setminus T}^4$ | EW^4 |
|---------|----|------|-----|------|------|------|-------|----------------------|--------|
| User | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Geo | × | ✓ | × | GlbR | GlbR | GlbR | PsnR | PsnR | PsnR |
| Time | × | × | × | × | × | × | ✓ | × | ✓ |
| Words | ✓ | × | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Tuning | × | × | × | ✓ | × | ✓ | ✓ | × | × |

Spatial Topic Model (ST) [HE13]. In ST model, each user has a distribution over global regions and topics. Different from TR, ST considers the identifiers of locations, and each topic has a distribution over location identifiers. ST assumes each tweet has a region and a topic: the topic determines the text content of the tweet, and topic and region together influence the draw of a location, *i.e.*, the sampling probability of a location is proportional to the product of the likelihoods of the topic generating the location and the region’s Gaussian distribution generating the location.

Who+Where+When+What (W^4) [YCM⁺13b]. This model is built based on similar intuitions as in EW^4 . Specifically, for each tweet, a personal region is first drawn based on its user and time, and then a topic is drawn based on the user’s topic distribution at that region. Finally, the topic and region together generate the location and the words of the tweet.

Enhanced Who+Where+When+What (EW^4). The differences between EW^4 and other methods are summarized in Table 5.4, where “PsnR” and “GlbR” represent “using geographical information by estimating personal regions” and “using geographical information by estimating global regions for all users”, respectively.

Enhanced Who+Where+What ($EW_{\setminus T}^4$). Except our preliminary work [YCM⁺13b], none of existing studies makes use of the time factor in prediction. To study the performance of our model without time factor, we consider a simplified version of EW^4 , known as $EW_{\setminus T}^4$, which does not consider the time factor. Note that $EW_{\setminus T}^4$ exploits the similar set of aspects as the baseline approaches TR HG and ST do, but its modeling method is different from theirs.

Note that both TR and HG are designed to predict the geographical coordinates, and cannot return the location identifier. Thus we cannot compute Acc for the two baselines. In order to compare with those approaches in terms of Acc , we identify the location identifier for the predicted geographic coordinates by finding the nearest location to the coordinates.

All baselines have been optimized by the development set. Specifically, for TR, the numbers of regions for WW and USA data sets are 500 and 600, respectively. For HG,

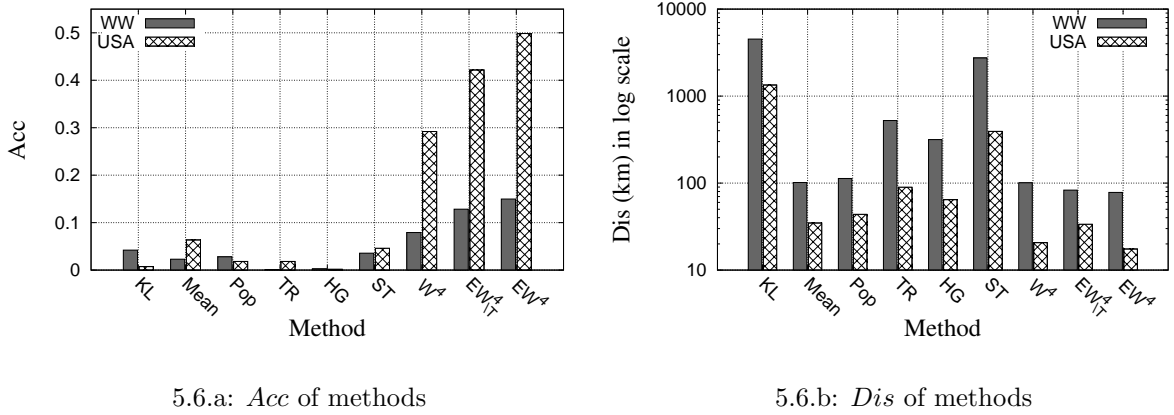
5.6.a: *Acc* of methods5.6.b: *Dis* of methods

Figure 5.6: Prediction Performance of all Methods

the priors over topics, topics mixing vectors, parameter λ and ω are all set to 0.1, and the numbers of regions for both data sets are 600. The number of topics for TR and HG are 50. For W^4 , the number of topics for WW and USA data sets are 10 and 20, parameters λ , κ for the two data sets are both 0.6 and 0.1, respectively.

We empirically set the hyper-parameters of our proposed models at fixed values. Specifically, we fix $\alpha = 200$, $\gamma = 0.25$, $o = \delta = 0.5$, $\beta = v_0 = \rho_0 = 0.1$, $\eta = \chi = \zeta = \kappa_0 = \nu_0 = 0.001$. ϵ is empirically set as $0.0001 \cdot I_2$ for WW data set and $0.00001 \cdot I_2$ for USA data set, respectively, where I_2 is a 2×2 identity matrix. Because the mobility regions and active time of different users can be greatly diverse from each other, we set user-specific values for μ_0 , ν_0 , and ω_0 : μ_0 is the mean coordinates of a user’s visited venues, ν_0 is the mean of the user’s visiting time, and ω_0 is 0.01 times the variance of the user’s visiting time.

5.7.2.3 Experimental results

We compare the prediction performance of the ten methods (KL, Mean, Pop, TR, ST, HG, W^3 , W^4 , EW^3 , and EW^4). The *Dis* and *Acc* of each method are reported in Figure 5.6. Note that only W^4 and EW^4 make use of the time information in prediction.

As shown in Figure 5.6, our preliminary model W^4 outperforms the state-of-the-art baseline methods KL, TR ST and HG significantly in terms of both *Acc* and *Dis*. The *Acc* of W^4 on WW and USA are 0.0792 and 0.2920, outperforming KL in terms of *Acc* by 88.50% and 3953.04% on the two data sets, respectively. The *Dis* of W^4 on WW is 100.93 km on USA is and 20.63 km. It reduces the *Dis* of TR by 80.73% and 77.02%, and reduce the *Dis* of HG by 68.09% and 68.02%, on the two data sets, respectively. The EW^4 proposed in this article achieves *Acc* of 0.1498 and 0.4986 on the two data

sets, which are 89.14% and 70.75% greater than that of W^4 on *WW* and *USA* data sets, respectively. The *Dis* of EW^4 are 78.16 km on *WW* and 17.47 km on *USA*, indicating that EW^4 reduces the average error distance against W^4 by 22.55% and 15.31% on the two data sets, respectively.

Mean and Pop are two model-free baselines which do not make use of time and word information. Interestingly, they achieve comparable *Acc* and much better *Dis* against other baselines, even include the complex models TR, ST and HG. Potential reason is a user’s mobility is constrained in a limited region which centers at a specific point, particularly when the user only visited very few locations. Thus, using their mean coordinates and mostly visited locations as predictions already achieves satisfactory results. However, compared with them, our proposed model EW^4 improves their *Acc* by more than 651.30% in *Acc*, and reduces their *Dis* by more than 30.27% on both data sets.

KL is designed to predict the location label for short text. Because it does not exploit geographic coordinates information, its prediction performance in terms of *Dis* is much worse than other methods, *i.e.*, the average error distance of KL is much greater than those of the other methods. In addition, KL builds language models for locations based on the words posted by all users without considering the individuals’ visiting history. In other words, it does not consider the preferences of individual users on locations. Moreover, the number of tweets posted at each location is small on average as observed from Table 5.3, and thus the language models of location are usually sparse, limiting the prediction performance of KL.

Different from KL, TR and HG are designed to predict the geographic coordinates for short text. They return the mean of the Gaussian distribution of the most likely latent region for a given tweet as the prediction result, but not the location identifier of the prediction. We observe that TR performs much better than KL in terms of *Dis* on both data sets. TR is based on topic models while KL adopts language models. Furthermore, TR incorporates the user preference information and the geographic coordinates information in its model. Comparing with TR, HG achieves a better performance, because it exploits the hierarchical relations between regions. However, *Acc* of TR and HG are approximate to 0, since the means of the global regions are less likely to be the exact locations of individuals’ tweets.

ST makes use of both the identifiers and geographic coordinates of locations, but its *Dis* is the worst among the topic-model based methods, and its *Acc* is better than TR and HG that do not use location identifiers. We checked the results, and found ST often returns the same location for the test tweets posted by the same user. After investigation, we found that many of the returned locations lie in the centers of regions with a quite large precision value. The large likelihood that the regions’ Gaussian distributions generate

the location makes the location always receive the greatest ranking score among the candidate locations.

Our model $EW_{\setminus T}^4$ utilizes the same types of information as do TR, HG and ST, but it outperforms the latter three baselines significantly. The reasons are two-fold. First, the latent geographic regions in $EW_{\setminus T}^4$ are personal while the latent geographic regions in the three baselines are global for all the users. Hence, the regions in $EW_{\setminus T}^4$ can describe individuals' mobility areas more precisely than the regions in TR, HG and ST. Second, both the location identifiers and the geographic information of locations are used by $EW_{\setminus T}^4$ to enhance the prediction, while TR and HG only exploit the geographic coordinates of locations.

EW^4 outperforms $EW_{\setminus T}^4$ in terms of both measures. This is because EW^4 incorporates the time factor in its model, which can further improve the prediction results. EW^4 is capable of capturing the user's mobility patterns in terms of geographic, temporal, and activity aspects.

Comparing with W^4 , the enhanced version EW^4 achieves better results in terms of both *Acc* and *Dis*. The reasons are three-fold: 1) EW^4 is designed under the framework of HDP, which is more robust to the overfitting problem; 2) in EW^4 , users can have different numbers of regions, which can be automatically learnt from the data by CRP. The user-specific region number can help better model users' mobility regions; 3) the weights of topic and region for the selections of locations and words are learnt from training data, which are user-specific.

5.7.3 Requirement-aware POI recommendation

Given a user and the user's specific requirements (represented by a set of words), requirement-aware location recommendation aims to recommend a ranked list of locations that the user has not visited but might be interested in. When the target time is available, we can also incorporate the time as additional contextual information. Although requirement-aware location recommendation uses the same ranking equation as location prediction for tweets, they are two different tasks: when predicting locations for tweets, the true locations may be the locations that the users have visited many times, while for location recommendation, the true locations are new to the users, *i.e.*, the users have not visited the true locations before.

However, it is hard to evaluate the accuracy of requirement-aware location recommendation. Consider a user who wants to have pizza at 7:00 PM. To get recommendations, the user can submit a requirement-aware location recommendation query with word "pizza" and time "7:00 PM" before the target time (*e.g.*, at 1:00 PM). The only way to verify the recommendation accuracy is to check whether the user indeed visited one of the recommended locations at 7:00 PM. However, it is very difficult to collect such

requirements and ground-truth recommendations for evaluating the requirement-aware location recommendation task.

In this chapter, we choose to use the information of a tweet (including user, time, words) in the test set as a requirement query, and return a ranked list of locations that the user has not been to (*i.e.*, has not visited in the training set). Here the time indicates the context, and the words describe the requirement of the user. In fact, we treat a location visit of a tweet query as a future event rather than a past event, and we use the location of the tweet as the ground-truth for evaluation. We admit that the tweet content may not always reflect the real user requirement.

To evaluate the recommendation performance, we use the same training and development sets that are used in the task of location prediction for tweets, but only keep the tweets in test set whose locations do not appear in their users' tweets in the training set. The number of such tweets is 1,221 and 1,178 in WW and USA data sets, respectively, and they are used as a group of test data, denoted as "full". However, many of the tweets do not represent specific requirements. Thus, we ask two annotators to create another group of data sets by removing non-English (but written in English alphabets) and requirement-irrelevant tweets. Only those tweets that are validated by both of the annotators are kept. After annotation, 193 and 219 tweets are left for WW and USA data sets, respectively, as another group of test data, denoted as "filtered".

5.7.3.1 Evaluation Metric

We evaluate the recommendation performance of different models by Hit ratio @ N ($Hit@N$), which measures the percentage of test instances whose true POIs are captured in the top N recommendations. Obviously, larger $Hit@N$ values indicate better performance. We set N as 1, 5, 10, and 20.

5.7.3.2 Methods to be evaluated

We compare the effectiveness of the methods that can utilize text for recommendation (KL, TR, ST, HG, W^4 , $EW_{\sqrt{T}}^4$, and EW^4). In order to examine the effectiveness of text, we remove the word factor from EW^4 as another baseline named $EW_{\sqrt{W}}^4$.

5.7.3.3 Experimental results

The $Hit@N$ of the 8 methods are reported in Figure 5.7.

Among these methods, the performance of KL is the worst, because it does not exploit user, time and geographical information in recommendation. The $Hit@N$ values of TR and HG are also very low. Because no location identifiers are used in the two models, they are ineffective in recommending the unvisited location identifiers for users. Compared

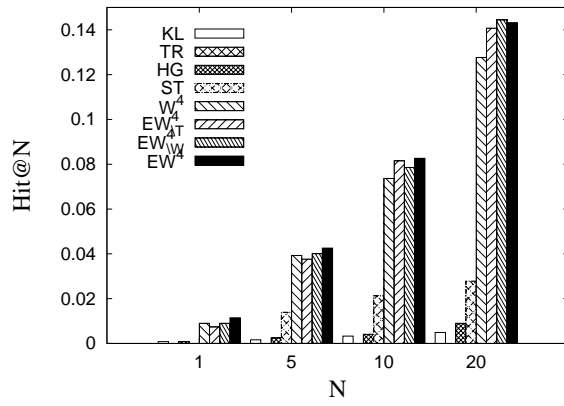
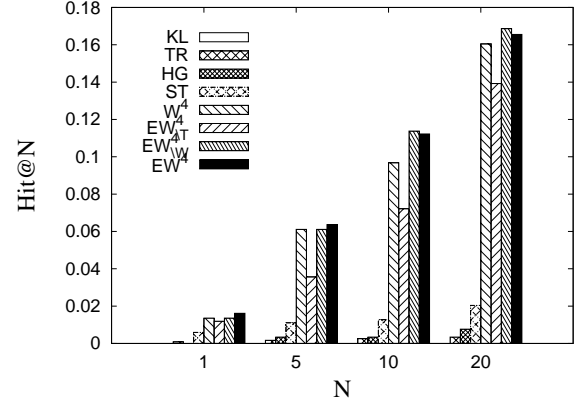
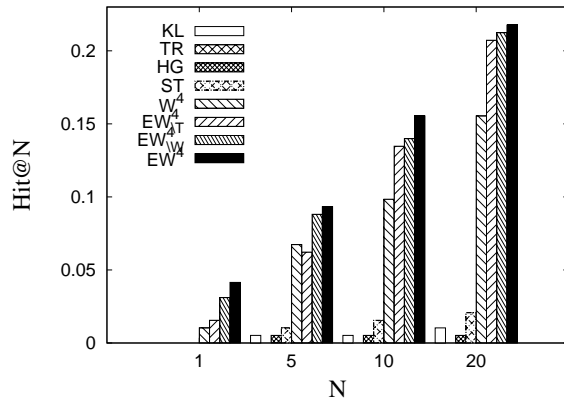
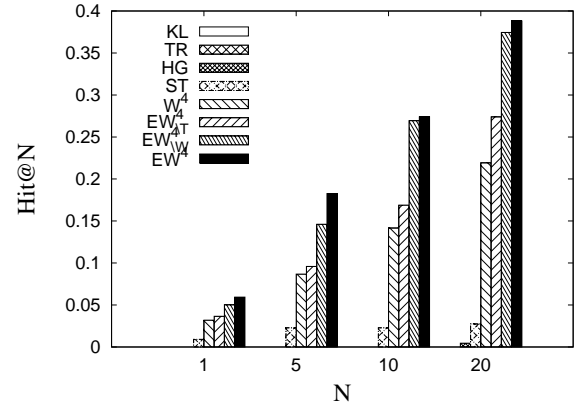
5.7.a: $Hit@N$ on WW (full)5.7.b: $Hit@N$ on USA (full)5.7.c: $Hit@N$ on WW (filtered)5.7.d: $Hit@N$ on USA (filtered)

Figure 5.7: Recommendation Performance of all methods on both data sets

with TR and HG, ST achieves much better $Hit@N$ values, because it makes use of both the geographical coordinates and identifiers of locations.

Compared with KL, TR, HG, and ST, our proposed method $EW^4_{\sqrt{T}}$ always achieves much better $Hit@N$ values on different N , even though it utilizes the same information with TR, HG, and ST. For example, before removing the requirement-irrelevant tweets, $EW^4_{\sqrt{T}}$ outperforms the best baseline ST by 283.4% and 466.7% in terms of $Hit@10$ on WW and USA, respectively, as shown in Figures 5.7.a and 5.7.b. After the filtering, the improvement becomes 769.0% and 640.0%, respectively, as shown in Figures 5.7.c and 5.7.d. The improvement may be attributed to two reasons: 1) personal regions that can describe user mobility more precisely, and 2) the consideration of both identifiers and coordinates of locations. After incorporating time factor, our full model EW^4 always

outperforms $EW_{\setminus T}^4$, demonstrating the importance of time for requirement-aware location recommendation.

$EW_{\setminus W}^4$ is another simplified version of EW^4 that does not consider text. However, the improvement of EW^4 over $EW_{\setminus W}^4$ is not very significant, especially when N is large. For example, before removing the requirement-irrelevant tweets (full), $EW_{\setminus W}^4$ even achieves slightly better $Hit@20$ than EW^4 , as shown in Figures 5.7.a and 5.7.b. The reason would be that the noisy tweet content deteriorates the recommendation accuracy. After removing such tweets (filtered), EW^4 performs slightly better than $EW_{\setminus W}^4$ in terms of $Hit@20$. However, when N is small, the improvement of EW^4 over $EW_{\setminus W}^4$ becomes significant. For example, either before or after removing the requirement-irrelevant tweets, EW^4 always outperforms $EW_{\setminus W}^4$ by more than 19% w.r.t. $Hit@1$. The results show that text requirement is important to generate accurate recommendations among the top several results.

In summary, our full model EW^4 achieves superior accuracy in recommending locations based on the target time and specific requirements. In addition, we note that EW^4 always outperforms its preliminary versions W^4 on the data sets either before or after removing the requirement-irrelevant testing tweets. Three reasons contribute to the improvement: (1) the number of personal regions is user-specific in EW^4 , which help better understand user mobility; (2) the weights of topics and regions for selecting words and locations in EW^4 are also user-specific; and (3) EW^4 is a non-parametric Bayesian model, which is more robust to overfitting.

5.7.4 Example Mobility Pattern

We take the model trained on the WW data set as an example to demonstrate the mobility pattern discovered by EW^4 .

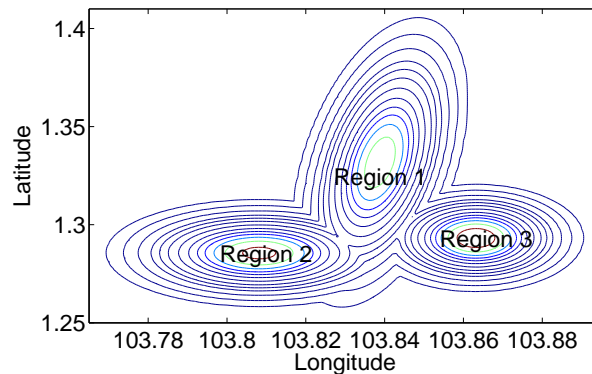


Figure 5.8: Personal Regions

We randomly select a user, and plot her personal regions in Figure 5.8, and the time patterns of each region in Figure 5.9. Figure 5.8 shows that the user has three personal regions centering at different locations in the city. In addition, the contour lines of the region 3 are more concentrated than that of region 1, showing that the user usually stays in a small region at the center of region 3, but visits a relatively larger range of places around region 1.

From Figure 5.9, we observe that the user has different time patterns over the personal regions on weekday and weekend: *e.g.*, the user is more likely to stay at region 2 on weekday afternoon, but to stay at region 2 on weekend evening. In addition, the user is more likely to spend more time in region 1 in the daytime of weekends, but only visit region 1 at dinner time of weekdays.

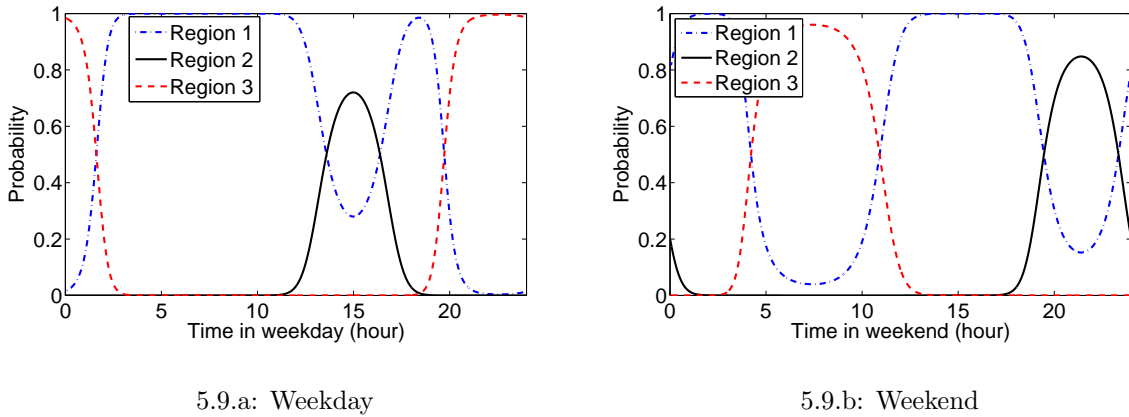


Figure 5.9: Region Distribution over Time

5.7.5 Results of Example Applications

In addition to location prediction for tweets and requirement-aware location recommendation, we implement another two applications, namely, user prediction and user’s location prediction. We now present their evaluation results in this subsection.

Location prediction for user. This task aims to predict the location at which a given user is most likely to stay at a given time. For each tweet in the test set, its time and user are used as input; if the predicted location is the true location of the tweet, it is a correct prediction. We employ prediction accuracy (Acc) as the evaluation metric, which shows the percentage of correct predictions.

We compare the performance of W^4 and EW^4 with a user mobility model PMM [CML11], on both data sets. Note that here we do not use the text of tweets. PMM is therefore

Table 5.5: Location prediction *Acc* of PMM, W^4 and EW^4

| <i>Acc</i> | <i>WW</i> | <i>USA</i> |
|------------|---------------|---------------|
| PMM | 0.0423 | 0.1102 |
| W^4 | 0.0776 | 0.2953 |
| EW^4 | 0.1423 | 0.5054 |

Table 5.6: User prediction *Acc* of PMM, W^4 and EW^4

| <i>Acc</i> | <i>WW</i> | <i>USA</i> |
|------------|---------------|---------------|
| PMM | 0.4163 | 0.4021 |
| W^4 | 0.5063 | 0.5863 |
| EW^4 | 0.5351 | 0.7679 |

applicable but not the other baselines for predicting locations of tweets using text as input [LSdV⁺11, KMO11, HAG⁺12].

The results are reported in Table 5.6. In location prediction, W^4 outperforms PMM by 83.45% and 167.97% on the two data sets, respectively. Potential reasons are two-fold. First, we use a new way to calculate the probability of latent regions at a given time, which is different from the way used in PMM. Second, W^4 exploits both the functional and geographical information of locations, while PMM only utilizes the latter. Comparing to W^4 , the proposed model EW^4 improves the accuracy by 83.38% and 71.15%, respectively. The improvement may come from: (1) the HDP model, which is more robust to overfitting problem; (2) user-specific number of personal regions, which enables us to model users' mobility more precisely; (3) user-specific weight between topics and regions for location and word generation, which can discover the different preferences between users.

User prediction. User prediction aims to predict the user who is most likely to visit a given location at a given time. For each tweet in the test set, its time and location are used as input; if the predicted user is the true user of the tweet, it is a correct prediction. We evaluate the performance using prediction accuracy. Note that the PMM model proposed in [CML11] can also be used for user prediction, if we use location and time as input, and find the user who can maximize the likelihood. The experimental results are reported in Table 5.6, which show that our method outperforms the baseline method significantly for similar reasons discussed earlier.

5.8 Summary

Requirement, which directly reflects a user's interest and intention, is an important consideration for POI recommendation. To make requirement-aware POI recommendations,

we need to model user mobility behavior from four factors, namely, user, POI, time, and words. To the best of our knowledge, none of the previous studies considers all of them. In this chapter, we present two models W^4 and EW^4 , which are capable of jointly modeling the four factors. The proposed model can discover the personal geographical regions, and spatiotemporal topic interests of users. We evaluate the performance of the proposed models for several applications including requirement-aware recommendation on two real-world datasets, and the experimental results show that the proposed method outperforms state-of-the-art baselines significantly.

Chapter 6

Conclusions and Future Work

In this chapter, we first conclude this dissertation, and then point out several promising directions for future work.

6.1 Conclusions

Online social networks, such as Facebook, Twitter, Foursquare, and Meetup, have accumulated a tremendous amount of UGC that is associated with location, time stamp and text content. The spatial information distinguishes such UGC from the traditional one which contains only text content and timestamp, such as question-answer pairs in community-based question answering websites (CQA) and threads in forums. Such geo-annotated UGC enables a number of new appealing applications, such as POI recommendation, which aims at recommending users POIs that they have not visited before. In this dissertation, we exploit the spatial, temporal, and semantic knowledge from geo-annotated UGC to discover users' interests, and study three specific POI recommendation problems.

In Chapter 3, we conduct the first analysis about the influence of time to the check-in behavior of users, and observe that users tend to visit different POIs at different time, and the check-in behavior of a user is daily periodic. To exploit the time information, we define a novel recommendation task, namely, time-aware POI recommendation, which aims to recommend POIs for individuals to visit at a given time. Although time has been exploited as important context information in many recommender systems, to the best of our knowledge, there is no existing work that can recommend POIs for users to visit at the given time. We can directly apply conventional time-aware recommendation techniques (*e.g.*, [DL05, XYZ⁺10]) to recommend POIs, but these approaches neglect the correlations between time and user mobility. As a result, their recommendation accuracy is limited. To solve this problem, we devise two algorithms, namely, UTE+SE and GTAG-BPP. UTE extends user-based CF by introducing time dimension into the

user-POI matrix. To address sparsity problem brought by the new dimension, we employ several techniques to smooth the data to better estimate the similarities between users. SE is a Bayesian model that exploits the spatial influence based on the observation that users tend to visit POIs close to them. Finally, UTE and SE are combined by linear interpolation to produce the recommendations. Different from UTE+SE, GTAG-BPP is a graph-based method. In GTAG, users' temporal preferences are modeled by session nodes, and the spatial influence is embedded by the edges between POI nodes. Given a target user and target time, we first adjust the weights of edges in GTAG, and inject preference into the target user node. The preference is then propagated to POIs by an effective and efficient algorithm BPP. We evaluate the proposed methods on two real-world datasets, and the experimental results show that both temporal and spatial influences make great contributions to the POI recommendation accuracy, and our methods outperform a number of baselines significantly.

Chapter 4 focuses on recommending POIs for a group of users. Group recommendation has been studied for years, and several approaches have been proposed. However, these approaches suffer from several limitations, *e.g.*, some models are built based on some assumptions that may not hold in real world [YLL12, LTYL12], and some model has a very high time complexity and cannot return results in real time [GLRW13]. In addition, existing solutions cannot exploit the geographical coordinates of POIs. To solve these problem, we develop an LDA-based generative model "COM" that is able to exploit both users' historical check-ins and the spatial influence. COM is designed based on several novel intuitions that the influence of a user in a group is topic-dependent, and a user may behave differently as a group member and as an individual. COM also incorporates the spatial influence as priors to boost the recommendation accuracy. Experimental results on four datasets reveal that the users' personal considerations of content factors, such as traveling distance, have a significant impact in users' choices. In addition, our proposed model COM achieves superior recommendation accuracy against the state-of-the-art baselines.

In Chapter 5, we concentrate on recommending POIs for users based on their specific requirements and time. This is a new problem that has not been studied before. However, this problem is difficult to solve, because it calls for a model that can model users' spatio-temporal mobility behaviors from user, location, time, and words aspects. We observe that a user's visited POIs always form spatial clusters, which can be viewed as personal regions. In addition, the visiting time in a region is different on weekdays and weekends. Based on the observations, we develop two novel generative models, namely, the pLSA-based W^4 and HDP-based EW^4 , to model the four aspects of user behaviors in an integrated manner. We estimate the parameters of the two models by EM and Gibbs sampling approaches, respectively. Besides POI recommendation, the proposed models have a variety of applications, *e.g.*, it can help business owners to target their potential users. We evaluate the two models on two datasets, and the results demonstrate their outstanding effectiveness in various applications against the state-of-the-art baselines.

6.2 Future Work

There are multiple potential directions for future work. Here, we present three directions:

- Besides time, companion, and words, there are many other kinds of contextual information that would be important for POI recommendation, such as weather and traffic pattern. It would be interesting to exploit such information to improve the accuracy of POI recommendation.
- Some researchers find that social relations have great influence in users mobility behavior, *e.g.*, users tend to visit places that are close to the places recently visited by their friends [CML11]. However, according to the experimental results in [YYLL11, CYKL12, YLL12], the contribution of social relations is limited to the task of POI recommendation. We are interested in the intrinsic reasons, and we plan to develop new models to exploit the social influence for POI recommendations, for both individual users and groups.
- Although user mobility typically follows a daily or weekly based routine (in-town), sometimes users may also travel to new cities (out-of-town). Our proposed models for POI recommendations are not applicable to the out-of-town case. Although several approaches have been proposed to address this problem [KIH⁺13, YSC⁺13, FYL13], none of them exploited the time, group and requirement information. Thus, it would be interesting to develop models that can recommend POIs when the target users are traveling in new cities.
- There are several issues that need to be considered when applying our methods to real applications, such as sparsity problem, cold-start problem, scalability and efficiency. The methods described in the dissertation are only prototypes. We will try to address these problems in our future work.

Appendix A

List of Publications

- **Quan Yuan**, Gao Cong, Kaiqi Zhao, Zongyang Ma, and Aixin Sun: Who, Where, When and What: a Non-parametric Bayesian Approach to Context-aware Recommendation and Search for Twitter Users. *ACM Trans. Inf. Syst.*, Volume 33 Issue 1, Article No. 2, March 2015.
- Kaiqi Zhao, Gao Cong, **Quan Yuan**, Kenny Q. Zhu: SAR: A Sentiment-Aspect-Region Model for User Preference Analysis in Geo-tagged Reviews. In *ICDE*, to appear, 2015.
- Zongyang Ma, Aixin Sun, **Quan Yuan**, Gao Cong: A Tri-Role Topic Model for Domain-Specific Question Answering. In *AAAI*, to appear, 2015.
- **Quan Yuan**, Gao Cong, Aixin Sun: Graph-based Point-of-interest Recommendation with Geographical and Temporal Influences. In *CIKM*, 659-668, 2014.
- Zongyang Ma, Aixin Sun, **Quan Yuan**, Gao Cong: Tagging Your Tweets: A Probabilistic Modeling of Hashtag Annotation in Twitter. In *CIKM*, 999-1008, 2014.
- **Quan Yuan**, Gao Cong, Chin-Yew Lin: COM: a Generative Model for Group Recommendation. In *KDD*, 163-172, 2014.
- Bo Liu, **Quan Yuan**, Gao Cong, Dong Xu: Where Your Photo is Taken: Geolocation Prediction for Social Images. *JASIST*, Volume 65 Issue 6, 1232-1243, June 2014.
- **Quan Yuan**, Gao Cong, Zongyang Ma, Aixin Sun, Nadia Magnenat-Thalmann: Who, Where, When and What: Discover Spatio-Temporal Topics for Twitter Users. In *KDD*, 605-613, 2013.

- **Quan Yuan**, Gao Cong, Zongyang Ma, Aixin Sun, Nadia Magnenat-Thalmann: Time-aware Point-of-interest Recommendation. In ***SIGIR***, 363-372, 2013.
- Zongyang Ma, Aixin Sun, **Quan Yuan**, Gao Cong: Topic-Driven Reader Comments Summarization. In ***CIKM***, 265-274, 2012.
- **Quan Yuan**, Gao Cong, Aixin Sun, Chin-Yew Lin, Nadia Magnenat-Thalmann: Category Hierarchy Maintenance: A Data-Driven Approach. In ***SIGIR***, 791-800, 2012.
- **Quan Yuan**, Gao Cong, Nadia Magnenat-Thalmann: Enhancing Naive Bayes with Various Smoothing Methods for Short Text Classification. In ***WWW (Companion Volume)***, 645-646, 2012.
- Xin Cao, Gao Cong, Bin Cui, Christian S. Jensen, **Quan Yuan**: Approaches to Exploring Category Information for Question Retrieval in Community Question-Answer Archives. ***ACM Trans. Inf. Syst.***, Volume 30 Issue 2, Article No. 7, May 2012.

References

- [AEK00] Asim Ansari, Skander Essegaier, and Rajeev Kohli. Internet recommendation systems. *Journal of Marketing research*, 37(3):363–375, 2000.
- [Ahn08] Hyung Jun Ahn. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Inf. Sci.*, 178(1):37–51, 2008.
- [AHS13] Amr Ahmed, Liangjie Hong, and Alexander J. Smola. Hierarchical geographical modeling of user locations from social media posts. In *WWW*, pages 25–36, 2013.
- [Arn83] B.C. Arnold. *Pareto Distribution*. International Cooperative Publishing House, Fairland, Maryland, 1983.
- [ASST05] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, and Alexander Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.*, 23(1):103–145, 2005.
- [AYRC⁺09] Sihem Amer-Yahia, Senjuti Basu Roy, Ashish Chawla, Gautam Das, and Cong Yu. Group recommendation: Semantics and efficiency. *PVLDB*, 2(1):754–765, 2009.
- [AZSD07] Silvana Aciar, Debbie Zhang, Simeon J. Simoff, and John K. Debenham. Informed recommender: Basing recommendations on consumer product reviews. *IEEE Intelligent Systems*, 22(3):39–47, 2007.
- [BA09] Linas Baltrunas and Xavier Amatriain. Towards time-dependant recommendation based on implicit feedback. In *Workshop on context-aware recommender systems (CARS09)*, 2009.
- [BFZ14] Yang Bao, Hui Fang, and Jie Zhang. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 2–8, 2014.

REFERENCES

- [BHG06] D. Brockmann, L. Hufnagel, and T. Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–5, 2006.
- [BHK98] John S. Breese, David Heckerman, and Carl Myers Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *UAI '98: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, University of Wisconsin Business School, Madison, Wisconsin, USA, July 24-26, 1998*, pages 43–52, 1998.
- [BMR10] Linas Baltrunas, Tadas Makcinskas, and Francesco Ricci. Group recommendations with rank aggregation and collaborative filtering. In *RecSys*, pages 119–126, 2010.
- [BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [BNS⁺12] Sandro Bauer, Anastasios Noulas, Diarmuid O. Seaghdha, Stephen Clark, and Cecilia Mascolo. Talking places: Modelling and analysing linguistic content in foursquare. In *SocialCom/PASSAT*, pages 348–357, 2012.
- [BOH12] Jesús Bobadilla, Fernando Ortega, and Antonio Hernando. A collaborative filtering similarity measure based on singularities. *Inf. Process. Manage.*, 48(2):204–217, 2012.
- [BSS⁺08] Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In *WWW*, pages 895–904, 2008.
- [Bur02] Robin D. Burke. Hybrid recommender systems: Survey and experiments. *User Model. User-Adapt. Interact.*, 12(4):331–370, 2002.
- [BZM12] Jie Bao, Yu Zheng, and Mohamed F. Mokbel. Location-based and preference-aware recommendation using sparse geo-social networking data. In *SIGSPATIAL/GIS*, pages 199–208, 2012.
- [BZWM13] Jie Bao, Yu Zheng, David Wilkie, and Mohamed F. Mokbel. A survey on recommendations in location-based social networks. *submitted to Geoinformatica*, August 2013.
- [CBDC10] Pedro G Campos, Alejandro Bellogín, Fernando Díez, and J Enrique Chavarriaga. Simple time-biased knn-based recommendations. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, pages 20–23. ACM, 2010.

REFERENCES

- [CBH02] Andrew Crossen, Jay Budzik, and Kristian J. Hammond. Flytrap: intelligent group music recommendation. In *IUI*, pages 184–185, 2002.
- [CCJ10] Xin Cao, Gao Cong, and Christian S. Jensen. Mining significant semantic locations from GPS data. *PVLDB*, 3(1):1009–1020, 2010.
- [CCLS11] Zhiyuan Cheng, James Caverlee, Kyumin Lee, and Daniel Z. Sui. Exploring millions of footprints in location sharing services. In *ICWSM*, 2011.
- [CDC14] Pedro G. Campos, Fernando Díez, and Iván Cantador. Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *User Model. User-Adapt. Interact.*, 24(1-2):67–119, 2014.
- [CHW01] Sonny Han Seng Chee, Jiawei Han, and Ke Wang. Rectree: An efficient collaborative filtering method. In *DaWaK*, pages 141–151, 2001.
- [CKPC09] Ye Chen, Michael Kapralov, Dmitry Pavlov, and John Canny. Factor modeling for advertisement targeting. In *NIPS*, pages 324–332, 2009.
- [CM13] Lucas Augusto Montalvão Costa Carvalho and Hendrik Teixeira Macedo. Users’ satisfaction in recommendation systems for groups: an approach based on noncooperative games. In *WWW (Companion Volume)*, pages 951–958, 2013.
- [CML11] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD*, pages 1082–1090, 2011.
- [CS08] Òscar Celma and Xavier Serra. Foafing the music: Bridging the semantic gap in music recommendation. *J. Web Sem.*, 6(4):250–256, 2008.
- [CYKL12] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.
- [CYLK13] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, 2013.
- [DL05] Yi Ding and Xue Li. Time weight collaborative filtering. In *CIKM*, pages 485–492, 2005.
- [DLS07] Marco Degemmis, Pasquale Lops, and Giovanni Semeraro. A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation. *User Model. User-Adapt. Interact.*, 17(3):217–255, 2007.

REFERENCES

- [EFL04] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5220–5227, 2004.
- [EOSX10] Jacob Eisenstein, Brendan O’Connor, Noah A. Smith, and Eric P. Xing. A latent variable model for geographic lexical variation. In *EMNLP*, pages 1277–1287, 2010.
- [Faca] Facebook. <http://www.facebook.com/>.
- [Facb] Facebook places. <http://www.facebook.com/directory/places/>.
- [Fis53] Ronald Fisher. Dispersion on a sphere. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 217(1130):295–305, 1953.
- [Foua] Foursquare. <http://foursquare.com/>.
- [Foub] Foursquare about. <https://foursquare.com/about/>. Accessed: 2014-02-18.
- [FYL13] Gregory Ferenca, Mao Ye, and Wang-Chien Lee. Location recommendation for out-of-town users in location-based social networks. In *CIKM*, pages 721–726, 2013.
- [Geo] Geotext. <http://www.ark.cs.cmu.edu/GeoText/>.
- [GHB08] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.
- [GLRW13] Jagadeesh Gorla, Neal Lathia, Stephen Robertson, and Jun Wang. Probabilistic group recommendation via information matching. In *WWW*, pages 495–504, 2013.
- [GM07] Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, pages 1606–1611, 2007.
- [GS04] Thomas L Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, 2004.

- [GTHL13] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. Exploring temporal effects for location recommendation on location-based social networks. In *RecSys*, pages 93–100, 2013.
- [GZT12] Guibing Guo, Jie Zhang, and Daniel Thalmann. A simple but effective method to incorporate trusted neighbors in recommender systems. In *User Modeling, Adaptation, and Personalization - 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings*, pages 114–125, 2012.
- [GZY13] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. A novel bayesian similarity measure for recommender systems. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, 2013.
- [HAG⁺12] Liangjie Hong, Amr Ahmed, Siva Gurumurthy, Alexander J. Smola, and Kostas Tsioutsoulis. Discovering geographical topics in the twitter stream. In *WWW*, pages 769–778, 2012.
- [HCW⁺10] Qiang Hao, Rui Cai, Changhu Wang, Rong Xiao, Jiang-Ming Yang, Yanwei Pang, and Lei Zhang. Equip tourists with knowledge mined from travel-ogues. In *WWW*, pages 401–410, 2010.
- [HE13] Bo Hu and Martin Ester. Spatial topic modeling in online social media for location recommendation. In *RecSys*, pages 25–32, 2013.
- [Hei11] Gregor Heinrich. Infinite lda implementing the hdp with minimum code complexity. *Technical note, Feb*, 170, 2011.
- [HJE13] Bo Hu, Mohsen Jamali, and Martin Ester. Spatio-temporal topic modeling in mobile social media for location recommendation. In *ICDM*, pages 1073–1078, 2013.
- [HKV08] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, pages 263–272, 2008.
- [HNV06] Tzvetan Horozov, Nitya Narasimhan, and Venu Vasudevan. Using location for personalized poi recommendations in mobile environments. In *SAINT*, pages 124–129, 2006.
- [Hof99] Thomas Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57, 1999.

REFERENCES

- [Hof04] Thomas Hofmann. Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst.*, 22(1):89–115, 2004.
- [HP99] Thomas Hofmann and Jan Puzicha. Latent class models for collaborative filtering. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI 99, Stockholm, Sweden, July 31 - August 6, 1999. 2 Volumes, 1450 pages*, pages 688–693, 1999.
- [Hun] Hungryhowhere. <http://www.hungrygowhere.com/>.
- [IMD] Imdb. <http://www.imdb.com/>.
- [Jen06] J.L.W.V. Jensen. Sur les fonctions convexes et les inegalits entre les valeurs moyennes. *Acta Mathematica*, 30:175–193, 1906.
- [Jie] Jiepang. <http://www.jiepang.com/>.
- [JS07] Anthony Jameson and Barry Smyth. Recommendation to groups. In *The adaptive web*, pages 596–627. Springer, 2007.
- [JSZC03] Rong Jin, Luo Si, ChengXiang Zhai, and James P. Callan. Collaborative filtering with decoupled models for preferences and ratings. In *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA, November 2-8, 2003*, pages 309–316, 2003.
- [KBV09] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.
- [KDK11] Noam Koenigstein, Gideon Dror, and Yehuda Koren. Yahoo! music recommendations: modeling music ratings with temporal dynamics and item taxonomy. In *RecSys*, pages 165–172, 2011.
- [KFN09] Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. Latent dirichlet allocation for tag recommendation. In *RecSys*, pages 61–68, 2009.
- [KIGI09] Kazuki Kodama, Yuichi Iijima, Xi Guo, and Yoshiharu Ishikawa. Skyline queries based on user locations and preferences for making location-based recommendations. In *Proceedings of the 2009 International Workshop on Location Based Social Networks, LBSN 2009, November 3, 2009, Seattle, Washington, USA, Proceedings*, pages 9–16, 2009.

REFERENCES

- [KIH⁺13] Takeshi Kurashima, Tomoharu Iwata, Takahide Hoshide, Noriko Takaya, and Ko Fujimura. Geo topic model: joint modeling of user’s activity area and interests for location recommendation. In *WSDM*, pages 375–384, 2013.
- [KKSS14] Christoph Carl Kling, Jérôme Kunegis, Sergej Sizov, and Steffen Staab. Detecting non-gaussian geographical topics in tagged photo collections. In *WSDM*, pages 603–612, 2014.
- [KMO11] Sheila Kinsella, Vanessa Murdock, and Neil O’Hare. ”i’m eating a sandwich in glasgow”: modeling locations with tweets. In *SMUC*, pages 61–68, 2011.
- [Kor09] Yehuda Koren. Collaborative filtering with temporal dynamics. In *KDD*, pages 447–456, 2009.
- [KPS13] Younghoon Kim, Yoonjae Park, and Kyuseok Shim. Digtobi: a recommendation system for digg articles using probabilistic modeling. In *Proceedings of the 22nd international conference on World Wide Web*, pages 691–702. International World Wide Web Conferences Steering Committee, 2013.
- [KSJ09] Ioannis Konstas, Vassilios Stathopoulos, and Joemon M. Jose. On social networks and collaborative recommendation. In *SIGIR*, pages 195–202, 2009.
- [LCX⁺14] Qi Liu, Enhong Chen, Hui Xiong, Yong Ge, Zhongmou Li, and Xiang Wu. A cocktail approach for travel package recommendation. *Knowledge and Data Engineering, IEEE Transactions on*, 26(2):278–293, Feb 2014.
- [LDGS11] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*, pages 73–105. Springer, 2011.
- [LDH⁺10] Zhenhui Li, Bolin Ding, Jiawei Han, Roland Kays, and Peter Nye. Mining periodic behaviors for moving objects. In *KDD*, pages 1099–1108, 2010.
- [LFYX13] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. Learning geographical preferences for point-of-interest recommendation. In *KDD*, pages 1043–1051, 2013.
- [LHT⁺12] Xingjie Liu, Qi He, Yuanyuan Tian, Wang-Chien Lee, John McPherson, and Jiawei Han. Event-based social networks: linking the online and offline social worlds. In *KDD*, pages 1032–1040, 2012.
- [LHZ13] Nathan Nan Liu, Luheng He, and Min Zhao. Social temporal collaborative ranking for context aware movie recommendation. *ACM TIST*, 4(1):15, 2013.

REFERENCES

- [LHZC10] Yanen Li, Jia Hu, ChengXiang Zhai, and Ye Chen. Improving one-class collaborative filtering by incorporating rich user information. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010*, pages 959–968, 2010.
- [LLAM13] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. Personalized point-of-interest recommendation by mining users’ preference transition. In *CIKM*, pages 733–738, 2013.
- [LLL11] Kenneth Wai-Ting Leung, Dik Lun Lee, and Wang-Chien Lee. Clr: a collaborative location recommendation framework based on co-clustering. In *SIGIR*, pages 305–314, 2011.
- [LM07] Daniel Lemire and Anna Maclachlan. Slope one predictors for online rating-based collaborative filtering. *CoRR*, abs/cs/0702144, 2007.
- [LSdV⁺11] Wen Li, Pavel Serdyukov, Arjen P. de Vries, Carsten Eickhoff, and Martha Larson. The where in the tweet. In *CIKM*, pages 2473–2476, 2011.
- [LSEM12] Justin J. Levandoski, Mohamed Sarwat, Ahmed Eldawy, and Mohamed F. Mokbel. Lars: A location-aware recommender system. In *ICDE*, pages 450–461, 2012.
- [LTYL12] Xingjie Liu, Yuan Tian, Mao Ye, and Wang-Chien Lee. Exploring personal impact for group recommendation. In *CIKM*, pages 674–683, 2012.
- [LX11] Defu Lian and Xing Xie. Learning location naming from user check-in histories. In *19th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, ACM-GIS 2011, November 1-4, 2011, Chicago, IL, USA, Proceedings*, pages 112–121, 2011.
- [LX13] Bin Liu and Hui Xiong. Point-of-interest recommendation in location based social networks with topic and location awareness. In *SDM*, pages 396–404, 2013.
- [MA98] Joseph F. McCarthy and Theodore D. Anagnost. Musicfx: An arbiter of group preferences for computer supported collaborative workouts. In *CSCW*, pages 363–372, 1998.
- [MC07] Rada Mihalcea and Andras Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM*, pages 233–242, 2007.

REFERENCES

- [McC02] Joseph F. McCarthy. Pocket restaurant finder: A situated recommender systems for groups. In *Proceeding of Workshop on Mobile Ad-Hoc Communication at the 2002 ACM Conference on Human Factors in Computer Systems*, 2002.
- [MCG⁺99] Tim Miranda, Mark Claypool, Anuja Gokhale, Tim Mir, Pavel Murnikov, Dmitry Netes, and Matthew Sartin. Combining content-based and collaborative filters in an online newspaper. In *In Proceedings of ACM SIGIR Workshop on Recommender Systems*. Citeseer, 1999.
- [Mee] Meetup about. <http://www.meetup.com/about/>. Accessed: 2014-09-01.
- [MLSZ06] Qiaozhu Mei, Chao Liu, Hang Su, and ChengXiang Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *WWW*, pages 533–542, 2006.
- [MMN02] Prem Melville, Raymond J. Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. In *AAAI/IAAI*, pages 187–192, 2002.
- [MOC99] J. Herlocker M. O. Connor. Clustering items for collaborative filtering. In *SIGIR Workshop on Recommender System*, 1999.
- [Mov] Movielens. <http://grouplens.org/datasets/movielens/>.
- [MP00] Koji Miyahara and Michael J. Pazzani. Collaborative filtering with the simple bayesian classifier. In *PRICAI*, pages 679–689, 2000.
- [MP02] Koji Miyahara and Michael J. Pazzani. Improvement of collaborative filtering with the simple bayesian classifier 1, Information Processing Society of Japan, vol. 43, no.11, 2002.
- [MR00] Raymond J. Mooney and Loriene Roy. Content-based book recommending using learning for text categorization. In *ACM DL*, pages 195–204, 2000.
- [MRJ12] James McInerney, Alex Rogers, and Nicholas R. Jennings. Improving location prediction services for new users with probabilistic latent semantic analysis. In *UbiComp*, pages 906–910, 2012.
- [MRM12] Wesley Mathew, Ruben Raposo, and Bruno Martins. Predicting future locations with hidden markov models. In *UbiComp*, pages 911–918, 2012.

REFERENCES

- [MSC⁺06] Kevin McCarthy, Maria Salamó, Lorcan Coyle, Lorraine McGinty, Barry Smyth, and Paddy Nixon. Cats: A synchronous approach to collaborative group recommendation. In *FLAIRS Conference*, volume 2006, pages 86–91, 2006.
- [Mur07] Kevin P Murphy. Conjugate bayesian analysis of the gaussian distribution. *def*, 1(2 σ 2):16, 2007.
- [MZRJ13] James McInerney, Jiangchuan Zheng, Alex Rogers, and Nicholas R. Jennings. Modelling heterogeneous location habits in human populations for location prediction under data sparsity. In *UbiComp*, pages 469–478, 2013.
- [NSLM12] Anastasios Noulas, Salvatore Scellato, Neal Lathia, and Cecilia Mascolo. A random walk around the city: New venue recommendation in location-based social networks. In *SocialCom/PASSAT*, pages 144–153, 2012.
- [OCKR01] Mark O’Connor, Dan Cosley, Joseph A. Konstan, and John Riedl. Polylens: A recommender system for groups of user. In *ECSCW*, pages 199–218, 2001.
- [PB97] Michael J. Pazzani and Daniel Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.
- [PDCCA05] Sebastiano Pizzutilo, Berardina De Carolis, Giovanni Cozzolongo, and Francesco Ambruso. Group modeling in a public space: methods, techniques, experiences. In *Proceedings of the 5th WSEAS International Conference on Applied Informatics and Communications*, pages 175–180, 2005.
- [PHC07] Moon-Hee Park, Jin-Hyuk Hong, and Sung-Bae Cho. Location-based recommendation system using bayesian user’s preference model in mobile devices. In *Ubiquitous Intelligence and Computing, 4th International Conference, UIC 2007, Hong Kong, China, July 11-13, 2007, Proceedings*, pages 1130–1139, 2007.
- [PHLG00] David M. Pennock, Eric Horvitz, Steve Lawrence, and C. Lee Giles. Collaborative filtering by personality diagnosis: A hybrid memory and model-based approach. In *UAI*, pages 473–480, 2000.
- [Pla] Plancast. <http://plancast.com/>.
- [PLEV04] Stratos Paulakis, Charalampos Lampos, Magdalini Eirinaki, and Michalis Vazirgiannis. Sewep: A web mining system supporting semantic personalization. In *PKDD*, pages 552–554, 2004.

REFERENCES

- [Ras99] Carl Edward Rasmussen. The infinite gaussian mixture model. In *NIPS*, pages 554–560, 1999.
- [RDP⁺09] Lakshmith Ramaswamy, P. Deepak, Ramana Polavarapu, Kutula Gunasekera, Dinesh Garg, Karthik Visweswariah, and Shivkumar Kalyanaraman. CAESAR: A context-aware, social recommender system for low-end mobile devices. In *MDM 2009, Tenth International Conference on Mobile Data Management, Taipei, Taiwan, 18-20 May 2009*, pages 338–347, 2009.
- [RFGST12] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *CoRR*, abs/1205.2618, 2012.
- [RFST10] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, pages 811–820, 2010.
- [SC10] Wang P Barabasi AL Song C, Koren T. Modelling the scaling properties of human mobility. *Nature Physics*, Sep 2010.
- [SGMB12] Filippo Simini, Marta C González, Amos Maritan, and Albert-László Barabási. A universal model for mobility and migration patterns. *Nature*, 484(7392):96–100, 2012.
- [Sin01] Amit Singhal. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 24(4):35–43, 2001.
- [Siz10] Sergej Sizov. Geofolk: latent spatial semantics in web 2.0 social media. In *WSDM*, pages 281–290, 2010.
- [SJ03] Luo Si and Rong Jin. Flexible mixture model for collaborative filtering. In *ICML*, volume 3, pages 704–711, 2003.
- [SK09] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. Artificial Intelligence*, 2009, 2009.
- [SKKR00] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. Analysis of recommendation algorithms for e-commerce. In *EC*, pages 158–167, 2000.
- [SKKR01] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW*, pages 285–295, 2001.

REFERENCES

- [SKKR02] Badrul M Sarwar, George Karypis, Joseph Konstan, and John Riedl. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the fifth international conference on computer and information technology*, volume 1. Citeseer, 2002.
- [SLBdG09] Giovanni Semeraro, Pasquale Lops, Pierpaolo Basile, and Marco de Gemmis. Knowledge infusion into content-based recommender systems. In *RecSys*, pages 301–304, 2009.
- [SLH14] Yue Shi, Martha Larson, and Alan Hanjalic. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Comput. Surv.*, 47(1):3, 2014.
- [SM07] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *NIPS*, 2007.
- [SQBB10] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-Lszl Barabasi. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.
- [SYMMyM11] Shunichi Seko, Takashi Yagi, Manabu Motegi, and Shin yo Muto. Group recommendation using feature space representing behavioral tendency and power balance among members. In *RecSys*, pages 101–108, 2011.
- [TJBB06] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476), 2006.
- [TKP13] Alexey Tarasov, Felix Kling, and Alexei Pozdnoukhov. Prediction of user location using the radiation model and social check-ins. In *Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing*, page 8. ACM, 2013.
- [TS06] Yuichiro Takeuchi and Masanori Sugimoto. Cityvoyager: An outdoor recommendation system based on user location history. In *Ubiquitous Intelligence and Computing, Third International Conference, UIC 2006, Wuhan, China, September 3-6, 2006, Proceedings*, pages 625–636, 2006.
- [Twia] The geography of twitter: Mapping the global heartbeat. <http://irevolution.net/2013/06/09/mapping-global-twitter-heartbeat/>.
- [Twib] Movielens. <http://grouplens.org/datasets/movielens/>.
- [Twic] Twitter. <http://twitter.com/>.

REFERENCES

- [VO05] Slobodan Vucetic and Zoran Obradovic. Collaborative filtering using a regression-based approach. *Knowl. Inf. Syst.*, 7(1):1–22, 2005.
- [wABG⁺07] Jae wook Ahn, Peter Brusilovsky, Jonathan Grady, Daqing He, and Sue Yeon Syn. Open user profiles for adaptive news systems: help or harm? In *WWW*, pages 11–20, 2007.
- [WB11] Benjamin Wing and Jason Baldridge. Simple supervised document geolocation with geodesic grids. In *ACL*, pages 955–964, 2011.
- [WTM13] Hao Wang, Manolis Terrovitis, and Nikos Mamoulis. Location recommendation in location-based social networks using user check-in data. In *SIGSPATIAL/GIS*, pages 364–373, 2013.
- [WWXM07] Chong Wang, Jinggang Wang, Xing Xie, and Wei-Ying Ma. Mining geographic knowledge using location aware topic model. In *GIR*, pages 65–70, 2007.
- [XCH⁺10] Liang Xiong, Xi Chen, Tzu-Kuo Huang, Jeff G. Schneider, and Jaime G. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *SDM*, pages 211–222, 2010.
- [XLY⁺05] Gui-Rong Xue, Chenxi Lin, Qiang Yang, Wensi Xi, Hua-Jun Zeng, Yong Yu, and Zheng Chen. Scalable collaborative filtering using cluster-based smoothing. In *SIGIR*, pages 114–121, 2005.
- [XYZ⁺10] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal recommendation on graphs via long- and short-term preference fusion. In *KDD*, pages 723–732, 2010.
- [YCH⁺11] Zhijun Yin, Liangliang Cao, Jiawei Han, Chengxiang Zhai, and Thomas S. Huang. Geographical topic discovery and comparison. In *WWW*, pages 247–256, 2011.
- [YCL14] Quan Yuan, Gao Cong, and Chin-Yew Lin. COM: a generative model for group recommendation. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 163–172, 2014.
- [YCM⁺13a] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. Time-aware point-of-interest recommendation. In *SIGIR*, pages 363–372, 2013.

REFERENCES

- [YCM⁺13b] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. Who, where, when and what: discover spatio-temporal topics for twitter users. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, pages 605–613, 2013.
- [YCS14] Quan Yuan, Gao Cong, and Aixin Sun. Graph-based point-of-interest recommendation with geographical and temporal influences. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 659–668, 2014.
- [YCZ⁺15] Quan Yuan, Gao Cong, Kaiqi Zhao, Zongyang Ma, and Aixin Sun. Who, where, when and what: a non-parametric bayesian approach to context-aware recommendation and search for twitter users. *ACM Trans. Inf. Syst.*, to appear, 2015.
- [Yel] Yelp. <http://www.yelp.com/>.
- [YLL12] Mao Ye, Xingjie Liu, and Wang-Chien Lee. Exploring social influence for recommendation: a generative model approach. In *SIGIR*, pages 671–680, 2012.
- [YLT13] Josh Jia-Ching Ying, Wang-Chien Lee, and Vincent S. Tseng. Mining geographic-temporal-semantic patterns in trajectories for location prediction. *ACM TIST*, 5(1):2, 2013.
- [YLWT11] Josh Jia-Ching Ying, Wang-Chien Lee, Tz-Chiao Weng, and Vincent S. Tseng. Semantic trajectory mining for location prediction. In *GIS*, pages 34–43, 2011.
- [YSC⁺13] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. Lcars: a location-content-aware recommender system. In *KDD*, pages 221–229, 2013.
- [YST⁺04] Kai Yu, Anton Schwaighofer, Volker Tresp, Xiaowei Xu, and Hans-Peter Kriegel. Probabilistic memory-based collaborative filtering. *IEEE Trans. Knowl. Data Eng.*, 16(1):56–69, 2004.
- [YYL10] Mao Ye, Peifeng Yin, and Wang-Chien Lee. Location recommendation for location-based social networks. In *GIS*, pages 458–461, 2010.

REFERENCES

- [YYLL11] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik Lun Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*, pages 325–334, 2011.
- [YZHG06] Zhiwen Yu, Xingshe Zhou, Yanbin Hao, and Jianhua Gu. Tv program recommendation for multiple viewers based on user profile merging. *User Model. User-Adapt. Interact.*, 16(1):63–82, 2006.
- [ZCX⁺14] Hengshu Zhu, Enhong Chen, Hui Xiong, Kuifei Yu, Huanhuan Cao, and Jilei Tian. Mining mobile user preferences for personalized context-aware recommendation. *ACM TIST*, 5(4):58, 2014.
- [ZCZ⁺10] Vincent Wenchen Zheng, Bin Cao, Yu Zheng, Xing Xie, and Qiang Yang. Collaborative filtering meets mobile recommendation: A user-centered approach. In *AAAI*, volume 10, pages 236–241, 2010.
- [Zha]
- [ZXC03] Yi Zhang, Wei Xu, and Jamie Callan. Exploration and exploitation in adaptive filtering based on bayesian active learning. In *ICML*, volume 3, pages 896–904, 2003.
- [ZZXM09a] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining correlation between locations using human location history. In *17th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, ACM-GIS 2009, November 4-6, 2009, Seattle, Washington, USA, Proceedings*, pages 472–475, 2009.
- [ZZXM09b] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *WWW*, pages 791–800, 2009.