Graph-based point-of-interest recommendation on location-based social networks

Guo, Qing

2019

Guo, Q. (2019). Graph-based point-of-interest recommendation on location-based social networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/84117>

<https://doi.org/10.32657/10356/84117>

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

**GRAPH-BASED POINT-OF-INTEREST
RECOMMENDATION ON
LOCATION-BASED SOCIAL
NETWORKS**

GUO QING

**WEE KIM WEE SCHOOL OF COMMUNICATION
AND INFORMATION**

2019

GRAPH-BASED POINT-OF-INTEREST RECOMMENDATION ON LOCATION-BASED SOCIAL NETWORKS

GUO QING

Supervisors: Prof. Theng Yin-Leng and Assoc Prof. Zhang Jie

Wee Kim Wee School of Communication and Information

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

2019

Statement of Originality

I certify that all work submitted for this thesis is my original work. I declare that no other person's work has been used without due acknowledgement. Except where it is clearly stated that I have used some of this material elsewhere, this work has not been presented by me for assessment in any other institution or University. I certify that the data collected for this project are authentic and the investigations were conducted in accordance with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

..... 22-Jan-2019
22-Jan-2019

..... Guo Qing
Guo Qing

Supervisor Declaration Statement

I have reviewed the content of this thesis and to the best of my knowledge, it does not contain plagiarised materials. The presentation style is also consistent with what is expected of the degree awarded. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accordance with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

22-Jan-2019
22-Jan-2019

Theng
Prof. Theng Yin-Leng

Authorship Attribution Statement

This thesis contains material from 2 papers published in the following peer-reviewed conferences where I was the first and corresponding author.

Chapter 3 is published as Guo, Q., Huang, Y., & Theng, Y.-L. (2015). Topic-sensitive location recommendation with spatial awareness. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 1, pages 237243. IEEE. DOI: 10.1109/WI-IAT.2015.203.

The contributions of the co-authors are as follows:

- I provided the initial project idea and direction.
- I performed the study including data collection, algorithm design, experiment conduction and result analysis.
- I prepared the manuscript drafts. The manuscript was revised by Prof. Yin-Leng Theng and Dr. Yi Huang.

Chapter 4 is published as Guo, Q., Sun, Z., Zhang, J., Chen, Q., & Theng, Y.-L. (2017). Aspect-aware point-of-interest recommendation with geo-social influence. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 1722. ACM. DOI: 10.1145/3099023.3099066.

The contributions of the co-authors are as follows:

- Prof. Zhang provided the initial project direction.
- I performed the study including data collection, algorithm design, experiment conduction and result analysis.
- Dr. Sun Zhu assisted in the experiment conduction.
- I prepared the manuscript drafts. The manuscript was revised by Prof. Jie Zhang, Prof. Yin-Leng Theng, Dr. Zhu Sun and Dr. Qi Chen.

22-Jan-2019
.....
22-Jan-2019

Guo Qing
.....
Guo Qing

Acknowledgments

First of all, I would like to express my sincere gratitude towards my supervisors Prof. Yin Leng Theng and Prof. Jie Zhang. Their guidance and encouragement do not only help me with the research, but also develop a critical thinking mindset, and most importantly how to persevere with my research when things were not going expectedly. Without their help and support, I would not be able to have this chance of writing these words, not to mention finishing my Ph.D journey. I also feel grateful to my mentors in SAP, Dr. Yi Huang, Dr. Qi Chen and Dr. Wenzhuo Yang, for their immerse insights, invaluable guidance and continuous help with my research study and career.

I also would like to express my deep appreciation to my senior, Dr. Zhu Sun, a role model for me, for her constant support and help. From her, I have received selfless guidance and sharing not only on the research studies, but also on how to be a qualified researcher. I really appreciate the help from all my friends: Dr. Aravind Sesagiri Raamkumar, Ranjan Satapathy, Sathik Basha, Peng Ni, Dr. Wenya Wang, Jianshu Li, Jin Wang, Bingshui Da, Xin Zheng, Ray Han, Haodan Yang, Wei Cui and Ruidan He. Many thanks to my managers, Steve Lee, Dr. Daniel Dahlmeier and Dr. Chen Wang, Tuan Choon Tang and all other colleagues in SAP, for their industrial knowledge and advices about my career path.

Last but not least, I would like to express my deepest gratitude to my parents for their encouragement, support and love. I feel truly grateful to my girlfriend from the bottom of my heart, for her tremendous support and company. Their immerse love helps me become a qualified researcher and evolve to be a better person in my life.

Contents

Acknowledgements	vii
Summary	xiii
List of Figures	xv
List of Tables	xvii
1 Introduction	1
1.1 Background	1
1.2 Motivation	7
1.3 Research Objectives	11
1.4 Research Contributions	14
1.5 Thesis Organization	18
2 Literature Review	19
2.1 General Recommender Systems	19
2.1.1 Content-based Recommender Systems	20
2.1.2 Collaborative Filtering Recommender Systems	21
2.1.3 Hybrid Recommender Systems	27
2.2 POI Recommendation	28
2.2.1 General POI Recommendation	29
2.2.2 Next POI Recommendation	34
2.2.3 Graph-based POI Recommender Systems	36
2.3 Summary	39

3	Topic-Sensitive POI Recommendation with Spatial Awareness	41
3.1	Topic-Sensitive POI Recommendation Algorithm with Spatial Awareness	42
3.1.1	Topic-Sensitive Location Recommendation Algorithm	42
3.1.2	Topic-Sensitive Location Recommendation with Spatial Awareness Algorithm	49
3.2	Experiments	51
3.2.1	Experimental Setup	52
3.2.2	Experimental Results	54
3.3	Conclusion	58
4	Aspect-aware Geo-Social Random Walk	59
4.1	Aspect-aware Geo-Social Influence Graph	60
4.1.1	Dataset Description	60
4.1.2	Problem Formalization	60
4.1.3	Intuitions for Constructing AGS-IG	61
4.1.4	Weight Assignment of Edges in AGS-IG	62
4.2	Graph-based POI Recommendation	66
4.2.1	Personalized PageRank with Meta-paths	66
4.3	Experiments	70
4.3.1	Experimental Setup	70
4.3.2	Results and Analysis	74
4.4	Conclusion	85
5	Aspect-aware Geo-Social Matrix Factorization	87
5.1	Meta-path based Random Walk	88
5.2	Aspect-aware Geo-Social Matrix Factorization	90
5.3	Experiments	93
5.3.1	Experiment Setup	93
5.3.2	Results and Analysis	95
5.4	Conclusion	105

6	An Attentional Recurrent Neural Network for Personalized Next POI Recommendation	107
6.1	Problem Analysis and Formulation	109
6.1.1	Problem Analysis	109
6.1.2	Problem Formulation	111
6.2	The Proposed Framework	112
6.2.1	Neighbor Discovery	113
6.2.2	Embedding Layer	115
6.2.3	Attention Layer	116
6.2.4	Recurrent Layer	117
6.2.5	Output Layer	118
6.2.6	Model Optimization	118
6.3	Experiments	120
6.3.1	Experimental Setup	120
6.3.2	Analysis of Meta-paths	122
6.3.3	Analysis of Time Threshold	124
6.3.4	Analysis of Hyper-parameters	125
6.3.5	Comparative Results	126
6.4	Conclusion	129
7	Conclusions and Future Work	131
7.1	Summary of Contributions	131
7.2	Future Work	134
	List of Publications	139
	List of Submissions	140
	References	141

Summary

Recent decades have witnessed a high-speed development of urban area with a large amount of POIs (point-of-interests) being built. A POI is a place with some functionalities in a location-based social network (LBSN), such as restaurants and movie theaters. In a LBSN, users can report their geographical locations and experiences explicitly via *check-ins*. However, the huge amount of heterogeneous information in LBSNs brings tremendous challenges to develop an effective POI recommender system.

Due to the extreme heterogeneity, user check-in decision exhibits two critical properties: (i) diversity; and (ii) imbalance. The diversity property is that the choice of visiting a POI is often jointly influenced by multiple factors, such as geographical and social factors. Meanwhile, the imbalance property represents that various influential factors carry different levels of importance for user check-in decisions. To capture both properties, this thesis proposes more advanced POI recommender systems by considering two perspectives: (i) representation: model the heterogeneous information in a unified data structure; and (ii) methodology: develop more effective algorithms to exploit the data structure to facilitate the POI recommendation generation. For the representation, this thesis explores the graph-based techniques from homogeneous graph to heterogeneous graph with the aim to embed various types of data in a unified space; for the methodology, this thesis proposes a series of approaches from random walk based method, to latent factor model, and deep learning model to effectively employ the graph structures for both general and next POI recommendation tasks.

The first study proposes a topic-sensitive POI recommendation model with a spatial awareness model (TSLRS) to exploit the geographical and content factors. Specifically, a homogeneous graph consisting of users is built with user topic preferences based on the textual information of POIs. The neighbors are discovered for each user by a topic-sensitive random walk over the graph. The opinions of neighbors are further aggregated

to infer the preference score for a POI. Finally, the geographical factor is also used in the neighbor discovery process to find nearby neighbors.

However, the homogeneous graph in first study projects the heterogeneous information into a homogeneous representation, leading to information loss. Hence, the second study designs a novel heterogeneous graph (AGS-IG) by fusing various relations among users, POIs and aspects from user reviews. Then, a novel graph-based ranking algorithm (AGS-RW) is proposed based on personalized PageRank (PPR) and meta-paths to model the diversity property by a full exploitation of both the heterogeneous graph structure and the semantic relations of AGS-IG.

Despite the exploitation of multiple factors, AGS-RW fails to model the imbalance of user check-in decisions. Thus, the third study develops a matrix factorization framework (AGS-MF) to effectively model both properties. First, AGS-IG is also adopted to represent the heterogeneous information in a common space. Then, an efficient meta-path based random walk over AGS-IG is developed to find relevant neighbors of each user and POI based on multiple factors, which are further incorporated into AGS-MF. By doing so, AGS-MF not only models multiple factors, but also learns the personalized weights for each individual user and POI. Thus, both the diversity and imbalance properties can be captured in a unified manner.

For next POI recommendation task, existing studies mainly model the sequential regularity of check-in sequences, but suffer from the severe data sparsity issue, making it extremely difficult to capture the transitional patterns between POIs. Thus, the fourth study proposes an recurrent model (ARNN) to jointly model both the sequential regularity and transition regularities of neighbors. Specifically, a meta-path based random walk over a novel knowledge graph is proposed to discover POI neighbors based on the heterogeneous factors. The transition regularities of various neighbors are integrated via the attention mechanism, which seamlessly cooperates with the sequential regularity as a unified recurrent framework.

In summary, a series of recommendation approaches have been proposed for both the general and next POI recommendation tasks, which exploit graph-based techniques to represent the heterogeneous information in a unified space for more effective POI recommendation. The extensive experiments demonstrate the superiority of the proposed approaches over state-of-the-art techniques.

List of Figures

1.1	Information services in LBS	3
1.2	A Location-based Social Network	5
1.3	The number of mobile unique visitor to Yelp	6
1.4	Research Framework	12
2.1	General recommendation process	20
2.2	Homogeneous graph construction	38
3.1	Graphical model for LDA	43
3.2	An example of POI sharing in LBSN	47
3.3	Geographical distributions of two users	49
3.4	Analysis of the number of topics (K)	55
3.5	Performance of Different Methods	56
4.1	AGS-IG Structure	61
4.2	Similarity between aspects by using categories	64
4.3	An example of meta-path: $U \rightarrow L \rightarrow A \rightarrow L$	68
4.4	Performance of our different variants on the three real-world datasets. . .	76
4.5	Analysis of w_{UUL}	77
4.6	Analysis of w_{ULAL} and w_{UAAAL} ($w_{UAAAL} = 0.8 - w_{ULAL}$)	78
4.7	Analysis of γ and η	80
4.8	Analysis of Distance Threshold ($dist0$)	81
5.1	A running example for AGS-IG structure	88
5.2	The effect of meta-paths on three datasets when meta-paths are gradually incorporated into AGSMF. Here, the arrows are omitted to better show the figure.	95

5.3	Analysis of α_u and α_l	98
5.4	The effect of meta-path length for AGS-MF.	99
5.5	The effect of dimensionality of AGS-MF.	100
5.6	Analysis of Convergence	101
6.1	A running example	108
6.2	The distribution of POIs with respect to # following POIs	110
6.3	The mean common ratio of neighbors vs. non-neighbors based on the three factors, including geographical, semantic and user preference factors. . . .	111
6.4	The architecture of ARNN	112
6.5	An illustrative example of the heterogeneous information network	114
6.6	Performance of variants of ARNN on the three datasets when various meta-path are cumulatively incorporated.	123
6.7	Analysis of Dimensionality	126
6.8	Analysis of the Number of Iterations	127
7.1	An overview of real-time POI recommendation	136

List of Tables

1.1	Categories of Location-based Services	2
1.2	Heterogeneous Information in LBSNs	9
1.3	Datasets used in the research	16
1.4	The overview of proposed approaches and state-of-the-art methods. Note that, the abbreviations of all methods are presented and the proposed approach is highlighted in bold. For techniques, ‘PPR’ denotes ‘Personalized PageRank’; ‘TopicPPR’ denotes ‘Topic Personalized PageRank’; ‘KDE’ denotes ‘Kernel Density Estimation’; ‘MetaPath’ denotes ‘Meta path’; ‘TF’ represents ‘Tensor Factorization’; ‘MF’ denotes ‘Matrix Factorization’; ‘LSTM’ denotes ‘Long Short-term Memory’; ‘Attention’ denotes ‘Attention Mechanism’; ‘RNN’ means ‘Recurrent Neural Network’; ‘CNN’ denotes ‘Convolutional Neural Network’. More details of these models can be found in Chapter 2 and corresponding studies.	17
3.1	Textual Information of a POI	44
3.2	Statistical of Selected Cities	52
4.1	Examples of semantically related aspects	65
4.2	Statistics of three cities	70
4.3	Precision and recall of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-RW achieves relative to the best performance of other methods.	83

4.4	F-measure (F1 score) of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-RW achieves relative to the best performance of other methods.	84
5.1	Derivation of $\mathcal{U}_i$, $\mathcal{V}_j$, $\Omega_{i,p}$ and $\Theta_{j,p}$	92
5.2	Average number of user-aspect and POI-aspect (per user and POI respectively)	97
5.3	Runtime Performance Comparison	102
5.4	Precision and recall of all methods on the three real-world datasets. For each metric, the best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that the proposed method AGS-MF achieved relative to the best performance of other comparison methods.	103
5.5	F-measure (F1 score) of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-MF achieves relative to the best performance of other methods.	104
6.1	Statistics of datasets ($\Delta t = 12h$)	120
6.2	Performance variation of ARNN with different time threshold (Δt) on three datasets evaluated by Acc@N. The best performance is highlighted in bold.	124
6.3	Performance of all the comparison methods on the three real-world datasets measured Acc@N ($N = 1, 5, 10, 20$). The best performance is highlighted in bold while the second best performance is marked by ‘†’. ‘Improve’ indicates the relative improvements that ARNN achieves relative to the best performance of other methods.	128

Chapter 1

Introduction

Recent decades have witnessed the rapid development of cities where points of interest (POIs) such as hotels, movie theaters and restaurants are expanding. The recent advancement of location-based social network services (LBSNs) such as Yelp¹ and Foursquare² enables users to find attractive POIs in a city. Although people are enthusiastic about exploring those POIs around, they are overwhelmed by a large amount of data available in LBSNs. The volume of heterogeneous data causes a heavy burden on users to find interesting places to visit. Hence, POI recommendation is an important component that is activated to enhance user experience in LBSNs. Compared with traditional recommendation tasks, POI recommendation is more complex due to the unique properties of LBSNs. Therefore, POI recommender systems recently gain increasingly more attention in the research community.

1.1 Background

The rapid advancement of web technology has enabled the development of a tremendous amount of mobile applications, such as Google Maps³, being widely adopted by people

¹<https://www.yelp.com/>

²<https://foursquare.com/>

³<https://www.google.com/maps>

Table 1.1: Categories of Location-based Services

Category of Services	Description
Emergency	Public or private emergency service can locate people in an emergent situation.
Navigation	Direct users to the specified destination
Information	Send users traffic news, weather information, interesting places to visit, etc.
Tracking & Management	Tracking the location of an object or user, for example, track postal packages to know the location of the goods.
Billing	Dynamically charge users using a particular service based on their location, for example, delivery fee is normally calculated based on the distance.

for navigation. Moreover, according to Global Digital 2019 report⁴ released in 30th Jan 2019, nearly 67% of users worldwide use mobile devices to browse the Internet. The high-speed development of mobile technology drives the huge opportunities for the adoption of Location-based Services (LBS) which connect users, devices and places to bring value for users, business owners (for example, a movie theater) and LBS operators (for example, Google Maps). LBS has already attracted significant attention of researchers from different disciplines including wireless networking, geographical information systems (GISs), and recommender systems to develop more advanced services for users (Schiller & Voisard, 2004; Jiang & Yao, 2006; Bao et al., 2015; Zhao et al., 2016; Hu et al., 2018).

Location-based Services. Location-based services use real-time data including geographical and user interaction data to provide information, entertainment and security services (Zhao et al., 2016; Arain et al., 2017; Ding et al., 2018). The rapid advancement of LBS is driven by various technologies including GIS, spatial positioning technology and web technology. This thesis addressed several significant research problems of POI recommendation, a critical component of LBS that is designed to enhance user experience for exploring more interesting POIs. Before discussing POI recommendation, it

⁴<https://wearesocial.com/blog/2019/01/digital-2019-global-internet-use-accelerates>

Figure 1.1: Information services in LBS

is necessary to have an overview of the various types of LBS. Table 1.1 classifies the commonly adopted location-based services including emergency, navigation, information services, tracking & management and billing (Steiniger et al., 2008).

As for information services⁵, there are basically two ways for users to obtain information: pull and push (Steiniger et al., 2008). Pull services deliver information only if users send requests to the service (Virrantaus et al., 2001). For example, like a search engine, users can input keywords as a query to search for a nearby Sushi bar or call a taxi (for example, Uber). Push services send information that is not directly requested by the user (Virrantaus et al., 2001). The information pushing is normally triggered when a new event is happening, for example, the user travels to a new area or simply a log-in action happens. For instance, if the user walks into a shopping mall, he/she might receive advertisements about the promotions that are available in this mall. By such definition, POI recommendation belongs to push services since users receive information about interesting POIs to visit, as shown in Figure 1.1. In a LBS, users can report their locations and experiences through *check-ins*. Each check-in represents a user's visit to a POI at a specific visiting time. A POI recommender system aims to learn users'

⁵Information services are summarized in Figure 1.1 based on <https://www.geoawesomeness.com/knowledge-base/location-based-services/location-based-services-applications>.

preference by their historical check-in data. And it is extremely necessary and timely to delicately develop advanced recommender systems for this particular domain.

There are basically two kinds of POI recommendation tasks: (i) the general POI recommendation; and (ii) the next POI recommendation. In the general POI recommendation task, based on historical check-in records of the user, the POI recommender system creates a list of POIs that meets personal preferences of the user (Ye et al., 2011; Zhang et al., 2014b; Wang et al., 2013). Since the research focus is to model users' general interests for POIs, the POIs that users have visited before could be also recommended as long as they meet the user's interests. With the recommended list of attractive POIs, users can easily choose their favorite places to visit in the future. In the next POI recommendation task, a list of POIs that the user is interested in visiting at next timestamp is generated given the sequence of his/her visited POIs, which provides valuable information for users to better plan their trips (Liu et al., 2016; He et al., 2016; Zhang et al., 2017b). For example, given a check-in sequence before 19:00 in a day, $home(8:00) \rightarrow bus\ stop(8:30) \rightarrow coffee\ shop(9:00) \rightarrow office(9:30) \rightarrow \dots \rightarrow ?(19:00)$, the next POI recommendation task is to predict where the user will visit at 19:00 after work. As a natural extension of the general POI recommendation, the next POI recommendation does not only estimate the general preference of users, but also considers the sequential effects of previous check-in records. In this sense, the next POI recommendation is a more complex task than the general one. Note that the POIs that have been visited before could also be recommended to users. This thesis identified several research gaps of both the general and next POI recommendation tasks, and proposed novel approaches to improve the recommendation performance for both tasks.

POI Recommendation. Prior to the Web 2.0 era, only the geographical location represented by the longitude and latitude tracked by GPS. However, point-of-interest information cannot be inferred from the pure geographical positioning information. For

Figure 1.2: A Location-based Social Network

instance, although the user’s spatial location can be recorded precisely, it is not easy to figure out whether the user stays at home, enjoys a meal in a restaurant or visits a museum since there is no mapping between the pairs of longitude and latitude and these point-of-interests.

Due to the advancements of mobile technology in recent years, several location-based mobile applications are becoming widely adopted by users, such as Foursquare and Yelp. Until Nov 2018, Foursquare has attracted more than 50 million users and surpassed 12 billion check-in records⁶. As of the second quarter of 2019, Yelp reported having a monthly average of 61.8 million unique visitors via desktop computer and 76.7 million unique visitors via its mobile website⁷. Usually, a check-in is associated with a pair of longitude and latitude corresponding to the POI. Moreover, users are also allowed to post reviews about POIs, make friends with other users and share their experience with them. Different from other types of social media like Facebook and Twitter, users can interact with physical places and share real experience via the location-based applications. This divergence drives a new concept of online social media — *Location-based Social Networks* (LBSN). Although having been founded in 2008, Foursquare has not introduced

⁶<https://www.foursquare.com/about>

⁷<https://en.wikipedia.org/wiki/Yelp>

Figure 1.3: The number of mobile unique visitor to Yelp

the social network layer until July 2014⁸. That is to say, Foursquare has evolved to be a LBSN in 2014. Unlike Foursquare, Yelp target itself as a crowd-sourced local business review and social networking site where the social feature has been always there since its launch. However, it was not until 2010 that Yelp started to add the check-in feature⁹. Figure 1.3 shows that increasingly more users are using Yelp¹⁰, for example, the number of App unique visitors has almost doubled in the past three years. With such booming LBSN applications in people’s life, it is now timely and necessary to delicately develop recommender systems for this particular domain. In a LBSN, there is a variety of information generated, as shown in Figure 1.2: (i) the check-in records that include several types of information such as POIs, ratings¹¹, POI geographical location, timestamp and reviews; (ii) the POI description information (categories and tags); (iii) the social network (or friendships) among users; and (iv) the relations among POIs, e.g., two locations are linked if they are geographically-close to each other.

⁸https://en.wikipedia.org/wiki/Foursquare_City_Guide

⁹<https://en.wikipedia.org/wiki/Yelp>

¹⁰<https://www.statista.com/statistics/385440/unique-mobile-visitors-yelp/>

¹¹Note that some datasets may contain rating information (e.g., Yelp) while others may not have ratings (e.g., Foursquare and Gowalla).

Users often are overwhelmed by the huge amount of POIs in a LBSN, it thus is hard for users to find interesting POIs to visit. Therefore, in order to enhance user experience in LBSNs, POI recommendation is developed to suggest places that users are interested in visiting by mining a variety of relevant information in a LBSN. However, recommendation techniques applied in the traditional item recommendation may not be suitable in LBSN context since a LBSN represents a space where the virtual and real-world activities of users intersect while only virtual behaviors are available in many other domains such as movie recommendation in Netflix or item recommendation in Amazon (Bell & Koren, 2007a; Sarwar et al., 2001; Koren et al., 2009). In other words, the check-in behaviors are geographically constrained in LBSNs, which significantly differentiates POI recommendation from the traditional item recommendation. Except the spatial constraint, various kinds of information are generated in a LBSN, making it difficult to model the user behaviors in LBSNs (Bao et al., 2012; Zhang & Chow, 2015; Zhao et al., 2016; Liao et al., 2018b). Because of the extreme challenges of developing an effective POI recommender system, POI recommendation has attracted significant research attention to improve the recommendation quality for both the general and next POI recommendation tasks.

1.2 Motivation

POI recommendation is a critical component of location-based services for improving user experience. As discussed in prior section, the characteristics of LBSNs are significantly different from other domains such as e-Commerce. Hence, POI recommendation has become an active research area in recommender system study, and has attracted many researchers to take efforts to propose more advanced POI recommendation techniques. The following presents the unique properties of LBSNs which create the great challenges to develop more efficient POI recommender systems.

Geographical Constraints. Geographical constraints distinguish POI recommendation from traditional item recommendation because the check-in behaviors largely depend on the geographical features of POIs (Ye et al., 2011; Zhang & Chow, 2013; Liu et al., 2014b; Yuan et al., 2013; Zhang & Chow, 2015; Lian et al., 2014; Zhang et al., 2017b). In Amazon or Netflix, the users can have the access to all items or movies if there are no geo-blocking restrictions as their interaction with those items are virtual or online, which is not limited by geographical boundaries (Bhargava et al., 2015; Cheng et al., 2011; Gao et al., 2013). Previous studies show that user check-in behaviors exhibit three kinds of representative phenomena regarding the geographical constraints:

- Users tend to visit places that are near to their previously visited places (Ye et al., 2011; Yuan et al., 2014, 2013). In the next POI recommendation scenario, the POI that the user visits next is usually close to the recent visited places (Cheng et al., 2013; Zhang et al., 2017b);
- User check-in records are usually around several activity centers (Cheng et al., 2012; Cho et al., 2011; Zhao et al., 2013), for instance, the user check-in records are often around home and office areas;
- Each user check-in records exhibits a personalized geographical distribution (Zhang & Chow, 2013; Zhang et al., 2015a), for example, some users might prefer the restaurants that are near to their homes while other users enjoy the food in the further restaurants.

Therefore, the geographical constraints need to be considered to capture the user check-in behaviors effectively. To this aim, various models are proposed to model the geographical features of user check-in records (Ye et al., 2011; Zhang & Chow, 2013; Cheng et al., 2012; Lian et al., 2014; Yuan et al., 2014). The representative studies will be further detailed in Chapter 2.

Table 1.2: Heterogeneous Information in LBSNs

Information Source	Information	Influential Factor
POI	Categories	Content
	Tags	Content
	Latitude, Longitude	Geographical
User-POI	Rating	Content
	Reviews	Content
	Timestamp	Sequential & Temporal
User-User	Social Network	Social

Heterogeneous Information. A LBSN consists of various kind of information, including not only check-in records, the geographical information and descriptions of POIs, but also users’ social relations and review information. Such heterogeneous information can be used to model the user check-in behaviors from different perspectives (Wang et al., 2015a,b; Zhang et al., 2014a), which inspires researchers to propose POI recommender systems of different kinds (Noulas et al., 2012; Liu et al., 2013; Lian et al., 2014; Gao et al., 2015; Guo et al., 2017; Feng et al., 2018). Table 1.2 summarizes the heterogeneous information in LBSNs. Due to the high heterogeneity of LBSN, there is a variety of influential factors that affect user check-in behaviors, which makes POI recommendation a much more challenging task than traditional item recommendation task. In light of such challenges, existing studies attempt to exploit different influential factors for POI recommendation (Ye et al., 2011; Zhang & Chow, 2013; Lian et al., 2015; He et al., 2015; Zhang & Chow, 2015; Tang & Wang, 2018). For instance, some studies leverage the social factor for capturing the preference sharing behaviors among friends, (Ye et al., 2010; Cheng et al., 2012; Wang et al., 2013; Li et al., 2016; Gao et al., 2018; Qiao et al., 2018). Moreover, various influential factors exhibit two critical properties: (i) diversity; and (ii) imbalance. To explain, “diversity” means that multiple influential factors impact a user’s check-in decision while “imbalance” is that various factors often have different levels of importance. Hence, unifying the effects of those factors can facilitate a deeper

understanding of user check-in behaviors. However, it is difficult to integrate all influential factors in such a way that they can be fully exploited. As depicted in Table 1.2, the influential factors are embedded in different information sources including POIs, social network, check-in records and check-in associated content. And some of the information are represented in free text (for instance, user reviews) while others are represented in a more structural way (for example, the check-in matrix). Therefore, the variety of data sources create huge challenges for a unified representation of LBSN.

Complex Relations. POIs are not only a new type of entities in a LBSN, but also serve as new information sources that have an important impact on other entities and relations as well. For instance, the relation between two users does not only depend on their social relation, but also the user-POI and POI-POI relations. To explain, location sharing activities alter the relations among users since users are apt to make new friends with geographically-close neighbors (Scellato et al., 2011; Cho et al., 2011; Hu & Ester, 2014). Traditional social network analysis mainly focuses on network properties without considering the geographical information. More recent studies identify the distinctions between online and offline social network (Cho et al., 2011; Wang et al., 2013; Li et al., 2016). Compared with online social network where the connections are not limited by their physical distance, geographical properties significantly influence the construction of social relationships between users in a LBSN. Specifically, the average geographical distance between non-friends is consistently larger than the average distance between friends. This empirical analysis implies that the probability of forming a social connection between two users decreases with the distance. Therefore, geographical properties play a crucial role in capturing the connection between two users in the context of LBSN.

Apart from users and POIs, temporal information and contents associated with the check-in records are also important information sources that might moderate other relations. Normally, the motivation of visiting a POI cannot usually be inferred from the

check-in record. For example, user reviews can provide additional information to understand the reason why the user checked in a POI as users tend to give comments on some aspects of the check-in experience such as the food and environment in a restaurant (Yang et al., 2013; Gao et al., 2015; Zhang et al., 2015b, 2017b). Thus, although the ratings or check-in frequency can explicitly reflect the preference of the user for a certain POI, review information facilitates a deeper understanding of user-POI relations. Moreover, the timestamps can serve as an indispensable information to understand the successive relations between POIs or model the sequential effect of a check-in sequence for the next POI recommendation (Cheng et al., 2013; Xie et al., 2016; Tang & Wang, 2018). Therefore, the relations among the various types of entities can be driven by multiple influential factors, which increase the complexity of modeling the them.

In summary, the variety of entities and relations involved in a LBSN create tremendous challenges for POI recommendation. In other words, to improve the POI recommendation quality, the impacts of those influential factors need to be modeled in a unified way with considering both the diversity and imbalance properties. There are basically two challenges to achieve a full exploitation of multiple factors: (i) how to represent the heterogeneous information and the complex relations among the entities in a unified space; and (ii) how to model the impacts of various influential factors for POI recommendation. To overcome such challenges, several research objectives of this thesis are presented in the following section.

1.3 Research Objectives

This thesis aims to achieve several critical research objectives by developing a series of approaches with the purpose of unifying the effects of various influential factors for both general and next POI recommendation tasks. As depicted in Figure 1.4, the first three studies propose novel approaches for the general POI recommendation task while the

Figure 1.4: Research Framework

fourth study proposes a novel POI recommender system for the next POI recommendation task. Overall, this thesis gradually goes deeper both in both representation and methodology perspectives. For the representation, the initial attempt is the exploration of homogeneous graph in the first study. Then, the subsequent studies go further by adopting heterogeneous graph to fully capture the variety of data available. For the methodology, it proceeds from the graph-based ranking algorithm, towards latent factor model and deep learning model in order to capture the highly heterogeneous information in a LBSN in a more effective way. The research objectives of this thesis are summarized as follows:

- The first objective aims to capture the diversity of user check-in behavior by modeling the geographical and content factors for the general POI recommender task in user-based collaborative filtering (CF) framework. However, CF-based methods suffer from the sparse check-in data in LBSNs, which increases the hardness to find effective neighbors (Bao et al., 2012; Wang et al., 2013; Yuan et al., 2013). Thus, instead of only using check-in records to find neighbors, the rich POI descriptions and geographical information are also exploited for neighbor discovery. Hence, the

first study aims to: (i) represent various kinds of data in a unified space by mapping the heterogeneous information into a homogeneous graph; and (ii) develop a graph-based ranking algorithm to find relevant neighbors for each user with the homogeneous graph.

- Existing studies mostly only have achieved a partial exploration of those influential factors for POI recommendation (Cheng et al., 2012; Zhang & Chow, 2013; He et al., 2015; Li et al., 2016). Moreover, although the first study exploits two significant factors (geographical and content factors), it projects the heterogeneous information into a homogeneous graph model, which could possibly cause information loss. Hence, the second study aims to: (i) propose a heterogeneous graph that could capture multiple types of entities and relations among them in a unified manner such that the diversity property could be modeled; and (ii) develop a random walk based algorithm to employ both graph structure and rich semantic relations in the heterogeneous graph for creating POI recommendations.
- Although the second study has achieved a full exploitation of multiple influential factors, it fails to model the imbalance property, that is, multiple influential factors may jointly affect the user’s check-in decision. As a user’s check-in decision is determined by both his/her preferences and the POI’s characteristics, so each user and POI are characterized by multiple influential factors at various levels of importance. Hence, the third objective aims to model both the diversity and imbalance properties for better POI recommendation performance. Specifically, the third study takes the advantage of the heterogeneous graph built in the second study to accommodate various kinds of information. Based on the heterogeneous graph, this study attempts to develop a latent factor model which could learn the personalized weights of each user and POI regarding to different influential factors to capture both the diversity and imbalance properties of user check-in behavior.

- The fourth objective turns the focus to improve the performance of the next POI recommendation task. Existing studies mostly focus on the sequential regularities for modeling user mobility (Zhang et al., 2014b; Liu et al., 2016; Tang & Wang, 2018), suffering severe data sparsity issue, which makes it hard to learn the transition patterns between POIs. To resolve the sparsity issue, the fourth study aims to: (i) conduct a series of insightful analysis experiments to unveil the effectiveness of transitional relations among multiple kinds of neighbors based on various factors for easing the sparsity issue; and (ii) develop a novel recurrent framework that could jointly model the transition regularities of various kind of neighbors of each POI based on multiple influential factors, to resolve the sparsity issue for better next POI recommendation performance.

1.4 Research Contributions

To achieve the four research objectives, a series of novel approaches was proposed. The contributions of the proposed approaches are summarized as follows:

- The first study proposed a novel approach, *Topic-Sensitive Location Recommendation with Spatial Awareness* Algorithm (TSLRS), to find more effective neighbors with the consideration of POI description and geographical information in the user-based collaborative filtering framework for the general POI recommendation task. Then a homogeneous graph-based ranking algorithm was developed based on users' preferences for various topics of POIs to find neighbors with similar topic interests. Next, a novel approach to model the spatial pattern of users was developed and combined with their topic interest to find nearby neighbors who also share common topic interest. Recommendations were generated by aggregating the opinions from the neighbors. The first study could capture the diversity property with respect to

geographical and content factors. Experimental results showed that the proposed approach significantly outperformed state-of-the-art graph-based methods.

- The second study explored more influential factors which were modeled to better capture the diversity property of user check-in decision for the general POI recommendation task. Specifically, a novel heterogeneous graph, *Aspect-aware Geo-Social Influence Graph* (AGS-IG) was constructed to incorporate content, geographical and social factors in a unified manner. Next, since there were different types of relations involved in the graph, a novel heterogeneous graph-based ranking algorithm, *Aspect-aware Geo-Social Random Walk* (AGS-RW) was developed by considering various semantic meanings of these relations. Hence, the proposed algorithm not only exploited the graph structure but also the semantics of relations to further enhance the recommendation quality. The extensive experiments indicated the proposed approach outperformed state-of-the-art methods.
- The third study first exploited AGS-IG to design a meta-path based random walk process to efficiently generate neighbors of users or POIs based on different influential factors. An *Aspect-aware Geo-Social Matrix Factorization* (AGS-MF) was developed which integrated regularizers of meta-path based neighbors into matrix factorization, and learned the personalized weights of meta-paths for each individual user and POI. This enabled AGS-MF to learn better user and POI latent representations that preserve user preference, POI characteristics, as well as the heterogeneous properties of AGS-IG. Extensive experiments were conducted to evaluate the proposed approach on multiple real-world LBSN datasets, and empirical results demonstrated that AGS-MF significantly outperformed state-of-the-art POI recommendation algorithms.

Table 1.3: Datasets used in the research

Dataset	Information
Gowalla	check-ins, POI categories and tags
Yelp	check-ins (without timestamps), social network, reviews
Foursquare	check-ins, social network, POI categories

- The fourth study proposed a unified recurrent framework, *Attentional Recurrent Neural Network* framework (ARNN), for the next POI recommendation task. Specifically, ARNN augmented the recurrent neural network (RNN) by jointly modeling both the sequential regularity and transition regularities of various kind of neighbors based on different influential factors, to resolve the sparsity. To find those relevant neighbors, a *meta-path* based random walk process over a heterogeneous graph was then proposed to efficiently discover the neighbors based on geographical, semantic and user preference factors. The neighbors were incorporated into ARNN through the attention layer, which could automatically determine the saliences of each neighbor. Extensive experiments on multiple LBSN datasets showed that ARNN outperformed state-of-the-art algorithms by a great margin.

To evaluate the proposed approaches, extensive experiments have been conducted in each study. However, different datasets and baselines were exploited due to the different motivation of each study (see Table 1.3) Specifically, the four studies utilized the following datasets: 1) Study 1 used the textual information of POIs (POI categories and tags) is required thus Gowalla¹² dataset was used. The reasons why Yelp is not suitable for Study 1 are: (i) there is no enough textual information of POIs, and (ii) user reviews contain much noise to learn user topic preference; 2) Study 2 and 3 exploited the aspects from user reviews, so the Yelp¹³ dataset was selected; 3) Study 4 aimed for next POI rec-

¹²<http://snap.stanford.edu/index.html>

¹³<https://engineeringblog.yelp.com/2017/08/yelp-open-dataset-and-dataset-challenge-round-10.html>

Table 1.4: The overview of proposed approaches and state-of-the-art methods. Note that, the abbreviations of all methods are presented and the proposed approach is highlighted in bold. For techniques, ‘PPR’ denotes ‘Personalized PageRank’; ‘TopicPPR’ denotes ‘Topic Personalized PageRank’; ‘KDE’ denotes ‘Kernel Density Estimation’; ‘MetaPath’ denotes ‘Meta path’; ‘TF’ represents ‘Tensor Factorization’; ‘MF’ denotes ‘Matrix Factorization’; ‘LSTM’ denotes ‘Long Short-term Memory’; ‘Attention’ denotes ‘Attention Mechanism’; ‘RNN’ means ‘Recurrent Neural Network’; ‘CNN’ denotes ‘Convolutional Neural Network’. More details of these models can be found in Chapter 2 and corresponding studies.

Study	Method	Influential Factors				Technique	Dataset
		Geo	Soc	Con	Seq		
Study 1	TSLRS	✓		✓		TopicPPR+KDE	Gowalla
	TSLR			✓		TopicPPR+KDE	
	TPR	✓		✓		TopicPPR	
	PPR	✓				PPR	
	GCF	✓				KDE	
Study 2	AGS-RW	✓	✓	✓		KG+PPR+MetaPath	Yelp
	TriRank	✓	✓	✓		KG+PPR	
	TR	✓	✓	✓		TF	
	GeoSoCa	✓	✓	✓		KDE	
	LFBCA	✓	✓	✓		PPR	
	PPR	✓		✓		PPR	
Study 3	AGS-MF	✓	✓	✓		KG+MF+MetaPath	Yelp
	AGS-RW	✓	✓	✓		KG+PPR+MetaPath	
	TriRank	✓	✓	✓		KG+PPR	
	TR	✓	✓	✓		TF	
	GeoSoCa	✓	✓	✓		KDE	
	GeoMF	✓				MF	
	SRMF		✓			MF	
	LFBCA	✓	✓	✓		PPR	
Study 4	ARNN	✓		✓	✓	LSTM+Attention	Foursquare Gowalla
	Caser				✓	CNN	
	DeepMove	✓		✓	✓	LSTM+Attention	
	ST-RNN	✓		✓	✓	RNN	
	SERM	✓		✓	✓	LSTM	

ommendation which requires check-ins with timestamps, thus Foursquare¹⁴ and Gowalla dataset were used for experiment. As different studies focus on different techniques and influential factors, the baseline and state-of-the-art methods vary for each studies. For a particular study, recent works are selected as state-of-the-art methods in the comparison experiment if they exploit similar heterogeneous information and similar technique as the proposed approach. Table 1.4 presents the details of the methods in the comparison experiment for each study.

1.5 Thesis Organization

This thesis is organized into seven chapters followed by references. Chapter 2 presents a critical review of literature related to the research objectives. Chapter 3-6 describe the four research studies where algorithms, experiments and results are included. Chapter 3 demonstrates TSLRS that exploited the POI description and geographical information to find more effective neighbors in user-based CF framework. Chapter 4 presents AGS-RW that incorporated multiple influential factors in a unified heterogeneous graph to further enhance POI recommendation. Chapter 5 introduces a unified latent model, AGS-MF, which can incorporated various influential factors in a unified space with the ability to model both diversity and imbalance properties. Chapter 6 demonstrates ARNN that modeled both the sequential regularity and transition regularities of various neighbors for the next POI recommendation. Chapter 7 comprises of the conclusion of this thesis, followed by some promising directions of the future work.

¹⁴<https://github.com/yaodi833/serm>

Chapter 2

Literature Review

This chapter presents a critical review of literature related to the four research objectives identified in Chapter 1. This chapter demonstrates an overview of general recommender systems and then dive deep into state-of-the-art POI recommender systems that are related to the research objectives in this thesis.

2.1 General Recommender Systems

There have been multiple definitions putting forth on recommender systems. In the context of personalization, a recommender system is defined as “any system that produces individualized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible objects” (Burke, 2002). Recommender systems are defined as “software agents that elicit the interests or preferences of individual user for products, either explicitly or implicitly and make recommendations accordingly” (Xiao & Benbasat, 2007). According to this definition, recommender systems can leverage multiple sources of data including both explicit and implicit feedback from users with contextual information to make personalized recommendations, as depicted in Figure 2.1.

In this section, instead of categorizing recommender system by application context, they are classified based on the techniques since the aim is to give a more general catego-

Figure 2.1: General recommendation process

rization. Basically, content-based (CB), collaborative filtering (CF), hybrid recommender systems are mostly adopted techniques depending on the underlying type of algorithm to estimate the relevance of the items for the target user. To better illustrate the technical details of each approach, the notations are given as follows. Let U and L represent the user set and POI¹ set, respectively; $|U|$ and $|L|$ are the total number of U and L , respectively; u, l denote a user and a POI, respectively; the POI preference of user u_i is defined as a preference vector $\mathbf{p}_{u_i} = (r_{u_i, l_1}, r_{u_i, l_2}, \dots, r_{u_i, l_j}, \dots, r_{u_i, l_{|L|}})$, where r_{u_i, l_j} is the rating or check-in frequency² given by user u_i on POI l_j ; the preference vector of POI l_j is $\mathbf{p}_{l_j} = (r_{u_1, l_j}, r_{u_2, l_j}, \dots, r_{u_i, l_j}, \dots, r_{u_{|U|}, l_j})$. Therefore, the general recommendation problem can be formulated as: given the historical check-in records of user u_i (denoted as $\mathbf{p}_{u_i}$), recommend a list of POIs that u_i would be interested in visiting in the future.

2.1.1 Content-based Recommender Systems

Content-based recommender systems are applicable in scenarios involving textual data, where users are provided with recommendations based on their interests. Content-based recommender systems utilize the preferences of users and content features of items to create recommendations. More specifically, items that are similar to what users have

¹To be consistent with the section about POI recommender systems, the notations for POI recommendation are also used for presenting general recommendation approaches.

²In different datasets, $r_{i,j}$ represents different meanings. In the Yelp dataset, $r_{i,j}$ is rating while it is check-in frequency in Foursquare and Gowalla datasets as rating information is not available.

shown positive feedback are ranked higher in the list. Letizia (Lieberman et al., 1995) is among the first CB techniques based on user profiles and it assists users to browse the web content. CB techniques are often used for recommending textual documents and TF-IDF model is widely applied to represent features of both documents and users (McNee et al., 2006). However, CB recommender systems usually encounter over-specialization challenge as they only recommend similar items to the user’s history, so they fail to provide any new and serendipitous items to users (Lops et al., 2011).

2.1.2 Collaborative Filtering Recommender Systems

Collaborative filtering techniques have widely been applied in e-Commerce context (Sarwar et al., 2001; Sun et al., 2019; Guo et al., 2019a). CF approaches do not utilize the content of items or external sources of knowledge but rather rely on user-item interaction information such as ratings, user reviews, tags and click records etc. Most of CF methods exploit the user-item matrix where each element is a rating representing the preference of a user for an item. There are two categories of CF approaches: memory-based and model-based.

2.1.2.1 Memory-based CF

The basic idea of memory-based CF is that similar users may share similar preferences for items (user-based) or users may prefer the items that are similar to previously purchased items (item-based) (Breese et al., 1998; Sarwar et al., 2001). So the memory-based methods are also called neighborhood-based methods, as the key point is to discover a set of reliable neighbors for the target users or items for generating recommendations. The advantages of memory-based approaches are their intuitiveness and simplicity to implement. Moreover, they can be generalized to any domains. To efficiently find the nearest neighbors of users or items, various types of similarity measurements have been designed. The widely adopted approaches include cosine similarity (Salton & McGill,

1986), Pearson correlation coefficient (Breese et al., 1998), Jaccard coefficient (Markines et al., 2009), and Bayesian Similarity (Guo et al., 2013), etc. The similarity computation thus has remarkable influence on the performance of memory-based recommender systems. Memory-based approaches can be further classified into user-based, item-based and graph-based approaches.

User-based CF. There are basically two phrases involved in this method: (i) find Top-K most similar users (called *neighbors*) of the given target user based on their historical records, for example, ratings; and (ii) aggregate the preference of these neighbors to predict the rating of the target user for an item. As mentioned, there are multiple methods for computing the similarity between two users. For instance, the cosine similarity is defined as follows:

$$\cos(\mathbf{p}_{u_i}, \mathbf{p}_{u_k}) = \frac{\mathbf{p}_{u_i} \cdot \mathbf{p}_{u_k}}{\|\mathbf{p}_{u_i}\| \|\mathbf{p}_{u_k}\|} \quad (\text{Eq. 2.1})$$

Normally, the preferences of Top-K neighbors of u_i are aggregated to predict the score of user u_i on POI l_j :

$$\hat{r}_{u_i, l_j} = \frac{\sum_{u_k \in N(u_i)} \cos(\mathbf{p}_{u_i}, \mathbf{p}_{u_k}) r_{u_k, l_j}}{\sum_{u_k \in N(u_i)} \cos(\mathbf{p}_{u_i}, \mathbf{p}_{u_k})} \quad (\text{Eq. 2.2})$$

where $\hat{r}_{u_i, l_j}$ is the predicted preference score of u_i on l_j and $N(u_i)$ is a set of neighbors of u_i . A recommendation list can be generated by selecting Top-N POIs with highest scores.

Item-based CF. There are also two steps in this method: (i) find Top-K most similar items of the given item; and (ii) aggregate the user preferences for these neighbors to predict the rating for a target item. The method of computing the similarity between two POIs are similar to user similarity. Finally, the Top-K most similar locations of l_j are aggregated to predict the score of user u_i on location l_j :

$$\hat{r}_{u_i, l_j} = \frac{\sum_{l_m \in N(l_j)} \cos(\mathbf{p}_{l_j}, \mathbf{p}_{l_m}) r_{u_i, l_m}}{\sum_{l_m \in N(l_j)} \cos(\mathbf{p}_{l_j}, \mathbf{p}_{l_m})} \quad (\text{Eq. 2.3})$$

where $N(l_j)$ is a set of neighbors of l_j . A recommendation list can be created by selecting Top-N POIs with highest scores.

Graph-based CF. Instead of directly using the user-item rating matrix, graph-based CF approaches first transfer the user-item interactions into a graph and apply graph-based ranking methods to find relevant nodes for each user or POI. Graph-based approaches can find neighbours that are not directly connected to the target users. Thus, this method ease the data sparsity issue to some extent as there are a huge amount of missing value in the user-item matrix. Path-based and random walk are two methods to estimate the similarity between two nodes. In path-based methods, the similarity (or distance) is measured by the number of paths and the length of these paths between the two nodes. However, this method can only consider the local information of a node.

In random walk methods, the relevance of other nodes to the target node is estimated based on the global link structure. A baseline method to learn the relevance of nodes is PageRank (PR) (Eirinaki et al., 2005). However, the basic PageRank algorithm cannot be used for personalized recommendation since it exploits the global information of a graph to compute the PR value of each node. Personalized PageRank (PPR) is further developed to find the relevant nodes for the given node via a personalized random walk process (Haveliwala, 2002). As the second proposed approach in Chapter 4 is based on PageRank, it is necessary to present the basics about PageRank which is followed by Personalized PageRank algorithm.

PageRank. The basic idea of PageRank is that if many other important nodes are linked to a node, this node is more important. PageRank is originally exploited for information retrieval in the web context (Page et al., 1999), and then is extended to recommender systems with demonstrating convincing performance in different domains. It is basically a recursive random walk process. Eventually, each node in the graph has a PR value representing its importance. Let's take an example to better illustrate

the mechanism of PageRank. Assume we have a graph G consisting of users and their relations. Usually, the edge connecting two users can be determined by their preference similarity or social friendship (Wang et al., 2013; Noulas et al., 2012). In PR, the edge weight is called *transition probability* between two users. Normally, the goal of PR is to find important users in G globally. The formulation of PR can thus be defined as: given the graph G , the PR value of an entity is defined as:

$$\mathbf{R}(t+1) = \mu \cdot \mathbf{M} \cdot \mathbf{R}(t) + (1 - \mu) \cdot \mathbf{d} \quad (\text{Eq. 2.4})$$

where μ is a damping factor that is often set as 0.85; $\mathbf{R}(t)$ denotes the ranking value of the entity at t -th iteration; $\mathbf{d}$ is a vector defined as $d_{u_i} = \frac{1}{|U|}$, which is a uniformly distributed vector; $\mathbf{M}$ is the transition probability matrix where each element is the edge weight between two nodes. The k -th value of $\mathbf{R}(t)$ can be used for measuring u_k 's importance in G at t -th iteration. PageRank is a Markov process with restart (Haveliwala, 2002), that is, the random surfer jumps to a neighbor with probability μ or to an arbitrary entity with probability $1 - \mu$.

Personalized PageRank. Personalized PageRank (PPR) extends the idea of PageRank by replacing uniformly distributed teleportation vector $\mathbf{d}$ with a personalized vector $\mathbf{d}_{u_i}$ where only the i -th element is set as 1 while the others are 0. Then, the random walker would return to the target user u_i with a certain probability instead of randomly jumping to entities that are irrelevant to u_i . The above Markov process can be re-formulated as:

$$\mathbf{R}(t+1) = \mu \cdot \mathbf{M} \cdot \mathbf{R}(t) + (1 - \mu) \cdot \mathbf{d}_{u_i} \quad (\text{Eq. 2.5})$$

The k -th value of $\mathbf{R}(t)$ can be considered as the relevance score (or similarity) of u_k for u_i at t -th iteration. Therefore, after the random walk process ending with convergence, the preferences of Top-K similar users are combined to determine u_i 's preference for l_j which is the same process as user-based CF (see Eq. 2.3).

On the other hand, Personalized PageRank can also be exploited on an item graph which compose a set of items and their relationships to find Top-K similar items. Above approaches of using PPR are similar to user-based or item-based CF. However, either methods only employ the homogeneity of user-item interaction with side information of users and items, that is, they only use user-user or item-item relations. Recent studies tend to take advantage of the flexibility of graph to incorporate more heterogeneous information in a unified graph model (Xie et al., 2016; Sun et al., 2011; Ostapuk et al., 2019). More research studies about heterogeneous graph techniques on POI recommendation would be presented in Section 2.2.3.

Summary of Memory-based CF. Although memory-based approaches are applied in several real applications such as CiteULike³, Amazon⁴ and Last.fm⁵, they might suffer from the data sparsity of the user-item rating matrix since the historical records of many users are not sufficient enough to learn their preferences. To ease such data sparsity issue, some studies exploit either user or item side information to find more effective neighbors (Hwang et al., 2012; Davidson et al., 2010; Ye et al., 2010; Bao et al., 2012). For instance, Hwang et al. (2012) introduce a new concept of *category expert*, and the user’s preference is predicted by aggregating the ratings of those similar category experts instead of similar users. There are also some studies utilizing social network for more effective neighbor discovery. For example, Ye et al. (2010) predict the user’s preference by aggregating his/her friends’ interests.

2.1.2.2 Model-based CF

Model-based CF techniques aim to create a predictive model based on the user-item matrix. Bayesian based models (Miyahara & Pazzani, 2000, 2002), regression based

³<http://www.citeulike.org/>

⁴<https://www.amazon.com/>

⁵<https://www.last.fm/>

models (Pennock et al., 2000; Vucetic & Obradovic, 2005; Lemire & Maclachlan, 2005) and Matrix Factorization (MF) (Shi et al., 2014; Koren et al., 2009; Jamali & Ester, 2010) are several examples of model-based CF techniques. MF models are the most famous approaches that have been investigated since they were introduced in the context of the Netflix prize competition (Bell & Koren, 2007a).

Specifically, MF methods perform dimension reduction to the highly sparse user-item matrix to infer the implicit properties of users and items. These implicit properties are represented as vectors with much fewer dimensions in the latent space. It factorizes the user-item rating matrix $R \in R^{|U| \times |L|}$ into low-rank user-latent matrix $\mathcal{U} \in R^{|U| \times d}$ and POI-latent matrix $\mathcal{V} \in R^{|L| \times d}$ (d is the dimension size of latent vectors where $d \ll |U|$ and $d \ll |L|$). Accordingly, user u_i is associated with a vector $\mathcal{U}_i$ and POI l_j is associated with a vector $\mathcal{V}_j$. For POI l_j , the elements of $\mathcal{V}_j$ represent the extent to which the item possesses those factors, positive or negative. For user u_i , the elements of $\mathcal{U}_i$ measures the extent of interest u_i has in POIs which are high on the corresponding factors. The rating u_i in l_j is can be measured by the dot product of the two vectors as:

$$\hat{r}_{i,j} = \mathcal{U}_i \mathcal{V}_j^\top \quad (\text{Eq. 2.6})$$

The main challenge is to compute the latent vectors of each user and POI. Earlier studies apply Singular Value Decomposition (SVD) which requires factoring the user-item rating matrix (Koren et al., 2009). However, it raises difficulties to learn the latent vectors due to the high sparsity of the user-item matrix with the high portion of missing values. Conventional SVD based methods rely on imputation to fill in missing ratings to make the rating matrix dense (Sarwar et al., 2000). But, the imputation might distort the data to some extent such that the learned latent vectors cannot represent users' preferences or POI characteristics precisely. Hence, more recent studies tend to directly model the observed ratings, that is, they minimize the regularized squared error on known

ratings:

$$\mathcal{L} = \frac{1}{2} \sum_{r_{i,j} \in R} I_{i,j} (r_{i,j} - \mathcal{U}_i \mathcal{V}_j^\top)^2 + \frac{\lambda_u}{2} \|\mathcal{U}\|_F^2 + \frac{\lambda_l}{2} \|\mathcal{V}\|_F^2 \quad (\text{Eq. 2.7})$$

where $I_{i,j}$ is an indicator function that equals 1 if user u_i rated POI l_j and equals 0 otherwise; λ_u and λ_l are parameters to control the importance of user and POI regularization respectively; $\|\cdot\|_F$ denotes the Frobenius norm.

Since the goal is to train a model with high generalizability, that is, the model should avoid overfitting the known ratings by penalizing learned parameters. λ_u and λ_l controls the extent of such regularization. Two methods, *stochastic gradient descent* (SDG) (Kingma & Ba, 2014) and *alternating least squares* (ALS) (Bell & Koren, 2007b) are often used to learn the latent vectors in Eq. 2.7. Recently, though some studies attempt to incorporate information beyond ratings such as social network into MF, it would be hard to embed other information as they can be represented in a less structural way, for example, user reviews are basically free text.

2.1.3 Hybrid Recommender Systems

As the name suggests, hybrid systems combine different recommendation techniques to achieve better performance. One advantage of hybrid recommender systems is that they can exploit various types of information. For instance, the neighbors of a user are restricted to his/her social network or the most preferred items among friends are recommended to the user (Liu & Lee, 2010; Bellogín et al., 2013). Another significant advantage of hybrid recommender system is that they can combine the strength of different recommendation techniques to overcome some common issues of recommender systems, such as the sparsity (Adomavicius & Tuzhilin, 2005) and cold start problem (Schein et al., 2002) associated with CF techniques. Generally, weighted, cascade and switching methods are most commonly used methods for combining different techniques (Burke, 2002):

- **Weighted methods.** The final list of recommendations is created by combining results from two or more recommender systems. The score of an item is a linear combination of scores generated from different recommendation techniques (Burke, 2002; Melville et al., 2002; Ye et al., 2011);
- **Cascade methods.** Several recommender systems are connected in a cascade manner (Lampropoulos et al., 2012; Burke, 2007). An initial recommendation list is generated by the first recommender and then sent to the next recommender to be re-rated. The combined score from all recommender systems are finally used to deliver the recommendations to users.
- **Switching methods.** Recommender systems can be changed such that the most suitable one is selected based on their advantages (Burke, 2002; Lekakos & Caravelas, 2008; Ghazanfar & Prugel-Bennett, 2010). The switching decision is achieved by analyzing current usage scenario and criteria.

2.2 POI Recommendation

The related work of POI recommendation consists of three parts: (i) general POI recommendation; (ii) next POI recommendation; and (iii) graph-based POI recommender systems. As this thesis focuses on both the general and next POI recommendation tasks, the cutting-edge recommendation algorithms related to both tasks are presented first. Then, graph-based POI recommender systems are presented, because graph technique is the foundation of the proposed methods in this thesis. Specifically, the general recommender systems are first categorized according to several influential factors. As described in Chapter 1, the user check-in behavior is influenced by various heterogeneous factors. These influential factors can be exploited in POI recommender systems for better performance. Therefore, it is reasonable to categorize POI recommender systems according

to these influential factors upon the user check-in activity. Then, recent studies are presented, which attempt to integrate those influential factors as a whole for the general POI recommendation task. Since there are fewer studies on the next POI recommender system with a focus on modeling the sequential factor by different methods, the related work is then categorized based on methodologies. Finally, the application of the graph-based techniques for POI recommendation is presented.

2.2.1 General POI Recommendation

2.2.1.1 Modeling Individual Influential Factors

Geographical Factor. User check-in behavior is geographically constrained. Previous studies show that people tend to visit places nearby what they have visited before (Cheng et al., 2012; Ye et al., 2011; Zhao et al., 2013), and the exploitation of geographical information can improve POI recommender systems. Ye et al. (2011) discover that the check-in probability of a POI and the distance between this POI and visited POIs follow a power-law distribution. Multi-center Gaussian distribution (Ye et al., 2011) and Gaussian mixture model (Zhao et al., 2013) are also utilized for modeling user geographical distribution since the footprints of users are often around certain centers such as home and office. Instead of applying the same distribution, Zhang & Chow (2013) personalize geographical influence for each user since they argue that each user has her own geographical pattern. GeoMF, a MF-based model proposed by Lian et al. (2014), incorporates geographical information into a weighted regularized matrix factorization. In GeoMF, each rating is determined by user and POI factors including user features, POI features, user activity area and POI influence area. Liu et al. (2014a) claim that user’s geographical preference and interest are mutually affected and user’s preference is determined by both of them. Thus, their model, GeoPFM jointly learns both geographical preference and interest for users. For each user, LORE models a check-in probability

distribution over a two-dimensional space using KDE (Kernel Density Estimation). The geographical probability of visiting a new POI is then estimated based on its location on the check-in probability distribution (Zhang et al., 2014b). Liu et al. (2014b) exploit weighted matrix factorization (WMF) based method (IRenMF) to capture the relationships between users. The intuitions behind IRenMF are: user has similar preferences on neighboring POIs (location-level influence) and POIs in the same geographical region may share similar user preferences (region-level influence). RankGeoFM is an ranking-based MF model that (i) learns users' preference rankings for POIs, and (ii) includes the geographical influence of neighboring POIs (Li et al., 2015). Yin et al. (2017) exploit the hierarchical spatial regions to learn the POI representations in a deep learning framework, called SH-CDL. To explain, the collective preference of the target region and user preference on adjacent regions are used in the form of social regularization and spatial smoothing. More recent studies often combine geographical factor with content factor via probabilistic graph model by arguing that each spatial region has a few specific topics such as food, office or school (Yin et al., 2013; Yuan et al., 2015; Zhao et al., 2015; Zhang & Chow, 2016). Feng et al. (2017) explore geographical factor for a new prediction tasks that is to predict the users who would visit the POI in the future period. The proposed method, POI2Vec, can jointly learn the transition pattern between two POIs as well as user personal preferences.

Social Factor. Inspired by the exploitation of social influence in traditional recommender systems, some studies also explore this idea in POI recommendation (Ye et al., 2010; Cheng et al., 2012; Wang et al., 2013; Li et al., 2016; Yu et al., 2017; Qiao et al., 2018). The motivation is that friends share more common interests than non-friends. Ye et al. (2010) propose a friend-based CF (FCF) which only considers the relation between friends. They find that the performance of FCF is comparable to the basic user-based CF and FCF improves the computation efficiency. In the framework of CF, Ye et al. (2011)

integrate social relations with geographical information to make POI recommendations. Cheng et al. (2012) propose a probabilistic matrix factorization (PMF) with social regularization for POI recommendation. By bringing the social constraints into PMF, the learned latent features of friends are more likely to be close in the latent subspace. Recent studies extend the concept of social connection to model various kind of relationships among users. Wang et al. (2013) propose a link-based model that constructs a graph to model user relations based on both their interest and social friendship. Personalized random walk is then applied to find similar users for each user. By integrating the familiarity and preference similarities, the hybrid similarity measurement can incorporate multiple features determining users' relationships, including the number of mutual friends, the Jaccard index and cosine similarity (Qiao et al., 2018). Li et al. (2016) define three types of friends (that is, social friends, location friends and neighboring friends) and develop a two-step framework to utilize the information of those friends for addressing the data sparsity and cold-start problems.

However, despite the substantial improvement by leveraging on social factor in traditional recommender systems, experimental results from related work suggest that the social information on POI recommendation demonstrate a limited influence (Ye et al., 2010; Cheng et al., 2012; Gao et al., 2013). For example, purely relying on social influence in CF deteriorates the performance to some extent (Ye et al., 2010). The opposite findings in traditional and POI recommendation can be explained by geographical constraints of the user check-in behavior. In a LBSN, although users with similar interests can form friendships without geographical limitation, their physical interaction (for example, eating in the same restaurant) requires they visit the same location. Therefore, only few studies exploit social factor alone to improve recommendation quality (Ye et al., 2010; Jin et al., 2012). Instead, social factor is often combined with other influential factors in a unified model for POI recommendation.

Content Factor. Extra information provided by content information can be useful to understand the motivation behind user check-in behaviors and is adopted by some studies to improve POI recommendation (Hong et al., 2012; Sizov, 2010; Yang et al., 2013). Bao et al. (2012) explore the categorical hierarchy to estimate users' preference at different category levels for recommending POIs for users who visit a new city. By so doing, their model can understand users' preference even without users' check-in history in the new city. Yang et al. (2013) apply sentiment analysis to extract aspects from user reviews with sentiments to measure users' preferences for POIs. Some studies (Hong et al., 2012; Sizov, 2010; Zhao et al., 2015) propose several geographical topic models that analyze the relation between the topics discovered from user reviews and locations. Hong et al. (2012) attempt to analyze the regional topics of geo-tagged tweets. In their method, each latent region modeled by a Gaussian distribution has a topical distribution that represents its spatial characteristics. By assuming that user's behavior varies with time, LRT models each user by different latent vectors for different time slots, and the final recommendation score is aggregated from all the latent vectors (Gao et al., 2013). Zhang et al. (2015b) develop a supervised aspect-dependent approach, called ORec, which can exploit the rich information of reviews to infer user preference for various aspects at finer levels. More recent studies tend to capture the complex relationships between content and other kind of information. For instance, SAR embeds the sentiments, aspects and region into a probabilistic graphical model (Zhao et al., 2015). Zhang & Chow (2016) propose CRATS based on Latent Dirichlet Allocation (LDA), which jointly mines the latent communities, regions, activities, topics, and sentiments based on the important dependencies among these latent variable (Zhang & Chow, 2016). A context-aware model, CAPRF, explores the relationships between users' check-in behavior and content information in a MF-based approach (Gao et al., 2015). However, CAPRF extracts keywords from user reviews, which could contain some noises without strong relations with user check-in behavior.

2.2.1.2 Unifying Multiple Influential Factors

Since the user check-in decision is affected by multiple factors (Yin et al., 2013; Guo et al., 2017; Zhang & Chow, 2015; Ogundele et al., 2017; Gao et al., 2018), recent studies propose different techniques to integrate the impacts of various influential factors in a unified framework. There are mainly two kinds of methods: (i) combine the effects of influential factors in a linear fashion (Ye et al., 2011; Cheng et al., 2012; Wang et al., 2013; Zhang et al., 2015b); and (ii) model the joint effects of multiple influential factors (Zhao et al., 2015; Yin et al., 2013; Guo et al., 2017; Yin et al., 2017). Specifically, earlier studies model each of the influential factors separately and then integrate their impacts in a linear manner. For instance, USG models user personal preference, social influence and geographical influence simultaneously for POI recommendations (Ye et al., 2011). Cheng et al. (2012) exploit MF to model social factor while the geographical factor is captured by Gaussian distribution. Ye et al. (2011) exploit geographical and social factors to determine the similarity between two users in the collaborative filtering framework. Zhang & Chow (2015) attempt to model the correlation between each influential factor and user preference, which are further combined linearly. However, by simply combining the effects of various factors via a linear way, the complex interrelationships between those influential factors cannot be captured effectively. ASMF is a two-step POI recommendation framework that (i) learns potential locations from users' friends and (ii) incorporates potential locations into WMF to overcome cold-start problem (Li et al., 2016).

More recent studies tend to propose a unified model, which can jointly capture the impacts of various factors. For example, Yang et al. (2013) propose a unified MF to integrate both reviews and social information. Liu et al. (2013) propose a geographical probabilistic factor analysis framework, which takes multiple influential factors into consideration. The proposed approach models the user check-in decision process to learn geographical user preferences for effective POI recommendations. Zhao et al. (2015) propose a model,

called SAR, which embeds the sentiment, aspects and categorical information into a probabilistic graphical model. The authors argue that the check-in record is determined by various factors including POI categories (content factor), spatial location (geographical factor) as well as sentiment towards the POI (content factor). Some studies adopt latent factor models to learn the latent representations of users and POIs with various influential factors (Xie et al., 2016; Gao et al., 2015; Lian et al., 2014; Cai et al., 2018). However, existing models fail to model the heterogeneous information in a unified space, thus they cannot exploit multiple factors in an integrated way. Furthermore, although some studies explore unified approaches, they fail to inherently capture the diversity and imbalance characteristics due to the absence of a unified representation for the various influential factors. In this thesis, the graph-based techniques have been employed to achieve the seamless integration of various influential factors with the consideration of the diversity and imbalance properties for more effective POI recommendation.

2.2.2 Next POI Recommendation

As discussed, the next POI recommendation is a more challenging task, so recent research efforts are taken for this problem. Note that the recommended POIs could be visited by the target user before. In other words, the next POI recommendation task studied in this thesis may recommend both new POIs and visited POIs to the user. Basically, there are two kinds of existing methods for user next POI prediction: (i) pattern-based methods; and (ii) model-based methods. Pattern-based methods can only mine explicitly predefined patterns on dense trajectories without capturing all mobility regularities about user movement (Monreale et al., 2009; Li et al., 2010). Thus, they are not suitable for LBSN scenario with extremely sparse trajectory data. In contrast, model-based methods are favored in LBSNs, due to their ability to model complex movement regularities and fuse heterogeneous data (He et al., 2016; Tang & Wang, 2018). Hence, this section mainly

discusses state-of-the-art model-based methods can be further classified into Markov-based and NN-based methods.

2.2.2.1 Markov-based Models

Generally, Markov-based models (MMs) predict the probability of future visit by constructing a location transition matrix. Due to the sparse check-in data, latent factor models are always adopted to help learn dense representations (Koren et al., 2009). By combining MMs with matrix factorization (MF), Rendle et al. (2010) propose FPMC to model user sequential behavior at personalized level. FPMC is extended to learn transition regularities with localized spatial constraint (Cheng et al., 2013). He et al. (2016) incorporate temporal and categorical information via a weighting scheme based on first-order Markov chain property.

However, MMs aim to learn transition probability between successive locations, thus failing to capture high-order sequential regularity. Besides, existing MMs fuse different contexts in a linear fashion such that impacts of those contexts cannot be well captured. Instead, the recurrent neural network has been adopted to model long-term sequences and flexibly fuse the diverse contextual information at each time step via the multi-modal embedding layer.

2.2.2.2 NN-based Models

The successful application of neural networks (NNs) in various domains (for example, image classification, speech recognition and natural language processing) have shown the great potential of deep learning models. Recent studies apply Neural Network (NN) in next location recommendation, owing to its strong capability of capturing sequential information (Hochreiter & Schmidhuber, 1997; Mikolov et al., 2010; Zhang et al., 2014c). To better capture the multi-level periodicity, DeepMove is developed by fusing an attention module into Recurrent Neural Network (RNN), to automatically select highly

correlated historical records for current status (Feng et al., 2018). Caser adopts Convolutional Neural Network (CNN) to capture the joint effects of previous check-ins on the current check-in (Tang & Wang, 2018). This enables Caser to advance over DeepMove, as DeepMove only models the impacts of each check-in independently.

Another research line explores new methods of using contextual information. Some studies model human mobility by using temporal and spatial contexts via time and distance transition matrices (Liu et al., 2016; Zhang et al., 2017b). However, the model training is complicated due to a large set of parameters. SERM jointly learns the embeddings of multiple contexts (for instance, temporal and semantic information) with user preference (Yao et al., 2017). Its architecture is highly flexible to incorporate various contexts with fewer parameters involved. However, only limited improvements are achieved by modeling both sequential regularity and multiple contexts, as these methods still cannot generate decent recommendations for the locations with only few following records. Therefore, a novel recurrent framework (ARNN) is proposed to exploit the transition regularities of neighbors based on the heterogeneous factors to overcome the sparsity challenge, introduced in Chapter 6.

2.2.3 Graph-based POI Recommender Systems

As graph-based techniques provide the foundation for our proposed approaches, it is necessary to have an overview of the graph-based POI recommender systems. Graph-based approaches have the edge over other counterparts in terms of their ability to model the diverse data. With the highly heterogeneous information available in LBSNs, graph-based approaches which have high generality are favored by research community for POI recommendation. Moreover, graph-based approaches can resolve the data sparsity issue. For instance, in a user-user graph, performing random walk can find two similar users even if they are not connected directly. Graph-based approaches can be categorized into

homogeneous graph-based and heterogeneous graph-based approaches based on the way of constructing the graph.

2.2.3.1 Homogeneous Graph-based Approaches

In a homogeneous graph, the graph only consists of the same type of entities and links. Information of other types of entities are embedded in the links (or edges) in the homogeneous graph as described in Figure 2.2. For instance, in order to build a user-user graph, both user check-in records and social network could be considered in the computation of edge weight. And some earlier studies mainly adopt the homogeneous graph-based techniques, as they focus more on exploiting one or two particular influential factors.

Jin et al. (2012) propose a user-user graph and dynamic graph ranking algorithm based on personalized PageRank (PPR). Their proposed framework can find popular locations for a user sending a query and top relevant users for a given POI. In the framework of PPR, Wang et al. (2013) propose a homogeneous graph ranking algorithm over the user-user graph to find top similar users and then apply user-based CF to create POI recommendations. The connection of two users incorporates both the preferences of users and social information. However, the recommendation performance still suffers from the data sparsity problem because users only share a limited portion of his/her check-ins with friends (Cheng et al., 2012). Other information can be further exploited in user-user graph to identify more effective neighbors. This idea corresponds to the first research objective.

There are fewer studies that explore homogeneous graph-based methods as they have lower generality to deal with the high heterogeneity of LBSN datasets and the information of other entities and relations are ignored through projecting heterogeneous information to a homogeneous graph. The complex relations between various types of data in LBSNs require approaches with the advantages of embedding various information, such as heterogeneous graph-based models.

Figure 2.2: Homogeneous graph construction

2.2.3.2 Heterogeneous Graph-based Approaches

In a heterogeneous graph, there are multiple types of entities and relations (Sun et al., 2011). Due to their strong ability to mine the complex relations, the heterogeneous graph models are also leveraged in POI recommendation task.

The incorporation of POIs into a graph is commonly adopted by some studies as user-item relations are the most important relation that express users' check-in preferences (He et al., 2015). Personalized PageRank (Noulas et al., 2012) is applied in user-item graph where both social network and user check-ins are exploited in determining the user-user similarity by taking a unified view of users and POIs. Noulas et al. (2012) argue that not only direct friends but also the distant friends have an influence on the user check-in activity. Bao et al. (2012) propose a novel framework based on CF where recommendations are created by aggregating the opinions from local experts, the users who frequently visit POIs in a certain area. In their framework, to find these local experts, HITS is applied to a user-item graph to learn the expertise of each user in certain geographical regions.

The heterogeneous graphs created in the above examples are all bipartite graphs with two types of entities, users and POIs. More recent research tends to bring more information to build more complex graph with various kinds of entities. He et al. (2015)

propose a tripartite graph that contains users, POIs and aspects. They argue that aspects extracted from user reviews can model users' preference in a finer level. The aspects represent the reason why the user visit the POI while a check-in or rating only convey limited information. Their proposed graph ranking model, TriRank, outperforms baselines without considering aspect information. In TriRank, a tripartite graph is constructed to model the relations of user-item, user-aspect and POI-aspect. However, TriRank does not consider the geographical and social factors. Hence, a heterogeneous graph that incorporates various influential factors in LBSNs is in need to create more relevant POI recommendations for users. This idea corresponds to the second research objective.

Some studies attempt to learn the embeddings of entities. For example, Xie et al. (2016) embed several bipartite graphs (for example, POI-POI, POI-Region, etc.) into a shared low-dimensional space and the user's preference is computed based on the embedding of her checked-in POIs learned in the latent space. Yang et al. (2017) propose a general framework, called PACE, for a slightly different task, that is, to create POI recommendations for different contexts. More specifically, the user preference can be regularized over a context graph so as to capture the relationships between them. Then, a deep neural architecture jointly learns the embeddings of users and POIs to predict both user preference over POIs and various context associated with users and POIs.

2.3 Summary

Recommender systems have been extensively studied for plenty of years. To be more structural, the literature review starts with the introduction of general recommender systems and move to state-of-the-art POI recommender systems. The first section presents an overview and motivation of developing recommender systems. Next, the main categories of techniques are presented. Specifically, the widely-used methods can be classified into content-based, collaborative filtering and hybrid recommender systems. For

collaborative filtering approaches, they can further be categorized into memory-based and model-based CF. For each category, representative methods are discussed with the strength and weakness of each method being summarized. Finally, the reasons why traditional recommendation techniques are not suitable in the context of LBSN are also discussed.

Next, the recommender system techniques related to POI recommendation are reviewed comprehensively. First, this section provides a general overview of POI recommender systems which are categorized by several influential factors upon users' check-in behaviors. Representative studies exploiting each influential factor are then discussed. These studies prove that the incorporation of influential factors can enhance recommendation quality. However, existing studies fail to model the diversity property by seamlessly integrating various factors together, not to mention capture different level of importance of those factors, that is, the imbalance property of user check-in behaviors. Therefore, the research studies gradually explored more advanced approaches in terms of both the representation and methodology perspectives. To alleviate the limitation of existing methods, a series of POI recommendation approaches have been proposed in this thesis: (1) The first study proposed a topic-sensitive with spatial awareness framework (TSLRS) to discover more effective neighbors based on a homogeneous graph by considering geographical and content factors in Chapter 3; (2) The second study developed a meta-path based random walk (AGS-RW) to fully capture the semantic information of a heterogeneous graph by integrating multiple factors in Chapter 4; (3) The third study proposed a matrix factorization based framework (AGS-MF) to capture both diversity and imbalance properties of user check-in decision based on the heterogeneous graph designed by the second study in Chapter 5; (4) The fourth study developed a recurrent framework (ARNN) to jointly model both the sequential regularity and transition regularities of neighbors based on various influential factors, so as to resolve the data sparsity issue for the next POI recommendation task in Chapter 6.

Chapter 3

Topic-Sensitive POI Recommendation with Spatial Awareness

As emphasized in Chapter 1, a user’s decision of visiting a place is determined by several influential factors. This implies the need to incorporate various influential factors to make more effective POI recommendation. This chapter presents the first study with the aim to explore the geographical and content factors for the general POI recommendation task in user-based collaborative filtering (CF) framework. More specifically, the spatial patterns of user check-in behavior and textual information associated with POIs can contribute to a more precise recommender system. As the check-in data is extremely sparse, the geographical and textual information such as POI categories are useful data to find more effective neighbors for the target user. In light of this challenge, the first study proposed a topic-sensitive recommendation model with spatial awareness to find more effective neighbors in the user-based CF. Specifically, latent Dirichlet allocation model (LDA) is applied to learn the user preference on different topics by mining the latent textual information of POIs. Then, a homogeneous graph is designed by considering user preferences for various topics. Next, a topic-sensitive probabilistic random walk over the homogeneous graph is developed to infer user expertise on each topic. Based

on the estimated expertise, neighbors are identified based on the topic preferences of the target user and topic expertise of neighbors. Then, the opinions of neighbors are aggregated to recommend POIs for the target user. Finally, the geographical factor is further incorporated to enhance the recommendation quality. The proposed approaches thus can model the diversity property of the geographical and content factors in a unified manner.

3.1 Topic-Sensitive POI Recommendation Algorithm with Spatial Awareness

For the recommendation method, the basic recommendation model, *Topic-Sensitive Location Recommendation* Algorithm (TSLR)¹, is first presented in Section 3.1.1. First, the visiting records of a user are profiled as a document to learn his/her over various topics via latent Dirichlet allocation (LDA). Furthermore, topic-sensitive PageRank is modified to learn user expertise on specific topics. Recommendations are created by considering both the topic interest distribution of the target user and the expertise of his/her neighbors. Section 3.1.2 describes the *Topic-Sensitive Location Recommendation with Spatial Awareness* Algorithm (TSLRS) for the further exploitation of geographical factor with two-dimensional (2D) kernel density function.

3.1.1 Topic-Sensitive Location Recommendation Algorithm

Topic Distillation. Topic distillation is the process of automatically learning the topics that users are interested in based on user footprints and associated textual information. LDA is implemented to identify the latent topics from real-life collections (Blei et al., 2003). Specifically, the document-topic and word-topic distributions are learned from the textual description of POIs.

¹Note that the terms *location* and *POI* are sometimes used exchangeably in this thesis.

Figure 3.1: Graphical model for LDA

The intuitive method to formulate a user document is to include all the previous records of a user. However, the problem is these documents are usually too short to reflect user interest. Similar to Liu & Xiong (2013), the terms associated with each POI that a user has checked in, including categories and tags, are thus aggregated into the corresponding document. Table 3.1 shows an example of the textual information associated with a POI. Figure 3.1 is a graphical representation of the LDA model, where the shaded and unshaded variables indicate the observed and latent variables respectively. An arrow indicates a conditional dependency between variables, and the plate (rectangles) denotes a replication of sampling. K represents the number of latent topics, and M is the number of users (or documents). d_u denotes a document for user u , and N_d refers to the number of terms in a particular user profile. Each user is associated with a multinomial distribution over topics, represented by θ . Each topic is associated with a multinomial distribution over words, represented by ϕ . Notice that the multinomial distributions θ and ϕ have symmetric Dirichlet priors α and β respectively. In summary, the latent topics are generated through the following process:

Table 3.1: Textual Information of a POI

Name	Battery Park
Categories	Park, Plaza
Tags	art, artists, bike lane, concerts, gallery, greenway, near the water, outdoor theater, park, performing arts, south ferry, Staten Island ferry, tourists, trendy, view, Wall Street bankers
Address	Battery Pl, State St, New York, NY, 10004

1. For each document d_u , choose $\theta_{d_u} \sim Dir(\alpha)$.
2. For each topic $z \in Z = \{z_1, \dots, z_K\}$, choose $\varphi_z \sim Dir(\beta)$.
3. For i th word position (d_u, i) in document d_u :
 - (i) Choose a topic $z_{d_u, i} \sim Multinomial(\theta_{d_u})$.
 - (ii) Choose a word $w_{d_u, i} \sim Multinomial(\varphi_{z_{d_u, i}})$.

Model Parameter Learning. The LDA model includes two sets of hidden parameters: the M document-topic distribution θ and the K topic-word distribution φ . Moreover, the latent variable z corresponds to the assignments of individual words to topics. Although Expectation Maximization is a standard technique for estimating parameters, it is computationally inefficient. Note that during parameter estimation, it is only required to keep track of the two matrices, $\Theta_{M \times K}$ (user by topic) and $\Phi_{|V| \times K}$ (word by topic). Notice that V is the set of words. Thus, Gibbs sampling (Griffiths & Steyvers, 2004) is implemented to estimate the document-topic and word-topic distributions. First, for each word, the topic and user assignment are sampled as follows:

$$p(z_i = z, u_i = u | w_i = w, \mathbf{z}_{-i}, \mathbf{u}_{-i}) \propto \frac{c_w^z + \beta}{\sum_{w' \in V} c_w^{z'} + \beta |V|} \cdot \frac{c_u^z + \alpha}{\sum_{z' \in Z} c_u^{z'} + \alpha K} \quad (\text{Eq. 3.1})$$

where c_w^z is the number of times word w assigned to topic z and c_u^z is the number of times user u assigned to topic z . $z_i = z$ and $u_i = u$ represent the assignment of i th word

in the document to topic z and user u respectively. $w_i = w$ represents that i th word of the document is w . $\mathbf{z}_{-i}$ and $\mathbf{u}_{-i}$ represent the exclusion of i th word from the topic and user assignments. After yielding the sampling results based on Eq. 3.1, the word-topic distribution and document-topic (or user-topic) distribution are obtained, that is, φ and θ are estimated as follows:

$$\varphi_{w,z} = \frac{c_w^z + \beta}{\sum_{w' \in V} c_{w'}^z + \beta|V|} \quad (\text{Eq. 3.2})$$

$$\theta_{u,z} = \frac{c_u^z + \alpha}{\sum_{z' \in Z} c_u^{z'} + \alpha K} \quad (\text{Eq. 3.3})$$

where $\varphi_{w,z}$ is the probability that word w is assigned to topic z , and $\theta_{u,z}$ is the probability that user u is assigned to topic z .

Therefore, two matrices are generated: $\Theta_{M \times K}$ and $\Phi_{|V| \times K}$. Row normalization is then carried out to the document-topic matrix $\Theta_{M \times K}$, where each row represents the probability distribution of topics a user is interested in. After normalization, $\Theta'_{M \times K}$ is obtained, where θ'_m is the m -th row and $\|\theta'_m\|_1 = 1$. In $\Theta'_{M \times K}$, for instance, $p(z|u) = \theta'_{u,z}$ indicates the probability that user u is interested in topic z .

Topic-Sensitive PageRank for Inferring Expertise. The latent topics that users are interested in are learned, that is, user topic distribution. Topical PageRank (TPR) (Haveliwala, 2002) is then extended to measure user expertise on different topics. Normally, users may have different expertise in different topics, for example, a user likes “Chinese food” does not necessary have much knowledge about “Italian food” there. High expertise in a certain topic means that the user (or expert) have more interests in that topic than other users, that is, he/she should also gain much knowledge about the topic. Thus he/she can be selected a high-quality neighbor for the target user. By doing so, we are able to ignore some random users who have little data (and knowledge) in a certain topic. In order to infer users’ expertise level for various topics, a *Topic-Sensitive PageRank* Algorithm has been developed.

First, a directed graph $G = (U, E_U)$ is constructed. U is a set of entities that represent users; directed edges $e_{u,u'} \in E_U$ and $e_{u',u} \in E_U$ are created, indicating that users u and u' share at least one POI or they are called “place-friends” (Scellato et al., 2011). L is the set of all POIs. For each user u , his/her visiting records is considered as a vector defined as follows:

$$\mathbf{p}_u = (\omega_{u,l_1}, \omega_{u,l_2}, \dots, \omega_{u,l_{|L|}}) \quad (\text{Eq. 3.4})$$

where ω_{u,l_i} is the normalized frequency of POI $l_i \in L$ visited by user u defined as:

$$\omega_{u,l_i} = \frac{\text{count}(u, l_i)}{\sum_j \text{count}(u, l_j)} \quad (\text{Eq. 3.5})$$

Intuitively, each edge $e_{u,u'} \in E_U$ can be associated a similarity based on POI information from Eq. 3.4 and Eq. 3.5 between u and u' , which is denoted as cosine similarity:

$$\text{sim}(u, u') = \cos(\mathbf{p}_u, \mathbf{p}_{u'}) \quad (\text{Eq. 3.6})$$

A straightforward way of learning user expertise for different topics is to consider $\text{sim}(u, u')$ as the transition probability between two users, that is, the edge weights in E_U are defined as Eq. 3.6. And PageRank is performed for each topic separately over the same user-user graph, G . In particular, users with higher interest in the topic will be assigned a higher PageRank value under this topic. Besides, Haveliwala (2002) argue that the random jump probability should be non-uniformed. That is to say, the topic-specific preference $\theta''_{u',z} (\sum_{u' \in U} \theta''_{u',z} = 1)$ is assigned to user u' as his/her random jump probability. Thus, given a topic z , the saliency score is defined as:

$$\mathbf{R}_z(u') = \lambda \sum_{u: u \rightarrow u'} \mathbf{R}_z(u) \cdot \mathbf{M}_{u,u'} + (1 - \lambda) \cdot \theta''_{u',z} \quad (\text{Eq. 3.7})$$

where transition probability $\mathbf{M}_{u',u}$ is defined by cosine similarity as Eq. 3.6; $\lambda \in [0, 1]$ is a damping factor to control the probability of the random jump to another vertex in G ; $\mathbf{R}_z(u') = \frac{1}{|U|}$ for each $u' \in U$ initially; $\sum_{u' \in U} \mathbf{R}_z(u') = 1$ in each iteration; the

Figure 3.2: An example of POI sharing in LBSN

users who are more interested in topic z will be assigned a larger jump probability, $\theta''_{u',z}$. Finally, a higher $\mathbf{R}_z(u')$ represents that u' has more expertise in topic z . This approach for learning user expertise for various topics is referred as *Topic PageRank*, which is one of the comparison methods in the experiments.

However, the above weight calculating method ignores the influence of topics. Two users may be more similar in one topic while completely divergent in another one. In Figure 3.2, u_1 and u_2 have 10 common POIs while u_1 and u_3 share 5 POIs. In this example, the transitional probability from u_1 to u_2 is larger than the transitional probability from u_1 to u_3 without considering topics. Nevertheless, in term of “Chinese Restaurant”, u_1 shares more POIs with u_3 than u_2 . This indicates that considering the topics can deliver more precise estimation of the similarity among users. To utilize the various topics, topic PageRank (TPR) is thus modified by integrating of both POI and topic similarity between users. Specifically, the topic-sensitive random surfer model on G visits each vertex with a certain probability (that is, the transition probability is topic-sensitive). According to this idea, a topic-sensitive user relationship graph can be constructed. Given a topic z , the transition probability between user u and u' is defined as the follows:

$$p_z(u \rightarrow u') = \begin{cases} \frac{\text{sim}(u,u') \times \text{sim}_z(u \rightarrow u')}{\sum_{v \in U} \text{sim}(u,v) \times \text{sim}_z(u \rightarrow v)} & \text{if } \sum_{v \in U} \text{sim}(u,v) \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (\text{Eq. 3.8})$$

Algorithm 1 TSLR for Location Recommendation

Input: $u \in U$ is a target user; $G = (U, E_U)$; λ is the damping factor; N is the recommendation size; K is the topic number; α and β are the Dirichlet hyper-parameters.

Output: A set of N POIs represented as Q .

```

1: Compute the preference of  $u$  and the expertise of others in each topic.
2: Initialize  $Q$  as an empty min priority queue,  $e_u(u') := 0$  for each  $u' \in U - \{u\}$  and
    $r_{u,l} := 0$  for each  $l \in L$ .
3: for  $u' \in U - \{u\}$  do
4:   for  $z \in Z$  do
5:      $e_u(u') := e_u(u') + p(z|u) \cdot \mathbf{R}_z(u')$ 
6: for  $l \in L$  do
7:   for  $u' \in U - \{u\}$  do
8:      $r_{u,l} := r_{u,l} + e_u(u') \cdot \omega_{u',l}$ 
9:   if  $|Q| < N$  then
10:     $Q.push((l, r_{u,l}))$ 
11:   else if  $r_{u,l} \leq \min(Q)$  then
12:     continue
13:   else
14:     $Q.pop()$ 
15:     $Q.push((l, r_{u,l}))$ 
16: return  $Q$  as a Top- $N$  recommendation list
    
```

where $\sum_{v \in U} p_z(u \rightarrow v) = 1$. $sim_z(u \rightarrow u')$ is the topic similarity from u to u' defined as:

$$sim_z(u \rightarrow u') = (1 - |\theta'_{u,z} - \theta'_{u',z}|) \cdot \theta'_{u',z} \quad (\text{Eq. 3.9})$$

Notice that $p(z|u) = \theta'_{u,z}$ represents the probability that user u is interested in topic z . The larger $(1 - |\theta'_{u,z} - \theta'_{u',z}|) \in [0, 1]$ indicates that u and u' are more similar in topic z . As defined, if u and u' are more similar, and u' is interested in the topic with a higher probability, then their transition probability increases. The transition matrix M is then defined as:

$$\mathbf{M}_{u,u'} = p_z(u \rightarrow u') \quad (\text{Eq. 3.10})$$

Thus, by replacing $\mathbf{M}_{u,u'}$ in Eq. 3.7 with Eq. 3.10, the expertise score of user u' on topic z ($\mathbf{R}_z(u')$) is computed after the recursive process.

Figure 3.3: Geographical distributions of two users

POI Rating Inference. For POI rating inference, both user expertise and topic distribution of the target user u are taken into account. That is, u' is considered as an excellent expert for u if u' has high expertise in the topics that u is interested in. This process can be considered as a personalization of expertise denoted as $e_u(u')$ with consideration of topic preference of target user u and expertise of user u' in various topics. Finally, the POI rating of a target POI l for user u , $r_{u,l}$ is computed by combining all the opinions from other users. The pseudo-code for location recommendation based on TSLR is summarized in Algorithm 1.

3.1.2 Topic-Sensitive Location Recommendation with Spatial Awareness Algorithm

The geographical factor plays an important role in user check-in behaviors (Ye et al., 2011; Wang et al., 2013; Yuan et al., 2013). This section presents TSLRS, which exploits geographical factor based on TSLR, so as to further achieve more effective POI recommendation performance.

For simplicity, the geographical coordinates are transferred to Cartesian coordinates. The center of the city is set as (0,0), the y-axis is parallel to the longitudinal line and the x-axis is parallel to the latitudinal line. A 2D Kernel Density Estimation (KDE) is

Algorithm 2 TSLRS for POI Recommendation

Input: $u \in U$ is a target user; $G = (U, E_U)$ is the user relationship graph; λ is the damping factor; N is the recommendation size; K is the topic number; α and β are the Dirichlet hyper-parameters.

Output: A set of N POIs represented as Q .

- 1: Compute the preference of u and the expertise of others in each topic.
 - 2: Initialize Q as an empty min priority queue, $e_u(u') := 0$ for each $u' \in U - \{u\}$ and $r_{u,l} := 0$ for each $l \in L$.
 - 3: **for** $u' \in U - \{u\}$ **do**
 - 4: **for** each $z \in Z$ **do**
 - 5: $e_u(u') := e_u(u') + p(z|u) \cdot \mathbf{R}_z(u')$
 - 6: $e_u(u') := e_u(u') \cdot \text{sim}_g(u, u')$
 - 7: $\triangleright$ Same as lines 8–20 of Algorithm 1
 - 8: **return** Q as a Top-N recommendation list
-

applied to explore user geographical distribution. Let $L_u = \{(x_i, y_i), i = 1, \dots, s\}$, a subset of L , all visited POIs of user u drawn from an unknown distribution with density function f , where s is the number of previous visited POIs of user u . Multivariate extensions of the kernel approach generally rely upon the product kernel (Scott, 2015). Therefore, estimator $\hat{f}$ of a POI (x_l, y_l) is defined as:

$$\hat{f}_u(x_l, y_l) = \frac{1}{s} \frac{1}{h_x h_y} \sum_{i=1}^s K\left(\frac{x_l - x_i}{h_x}\right) K\left(\frac{y_l - y_i}{h_y}\right) \quad (\text{Eq. 3.11})$$

where $K(\cdot)$ is the kernel function. The normal kernel is applied in this study:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \quad (\text{Eq. 3.12})$$

h is the bandwidth for smoothing that is defined based on Scott's rule (Scott, 2015):

$$h_{\cdot} = s^{\frac{-1}{dim+4}} \sigma. \quad (\text{Eq. 3.13})$$

where σ is the standard deviation in a certain dimension, and $dim = 2$ is the number of dimensions.

After applying 2D kernel function estimation, each user's unique spatial distribution is obtained. Figure 3.3 depicts the spatial distributions of two users. Thus, their geographical similarity can be captured based on the spatial distributions of the two users.

First, the probability that u and u' visit a certain POI $l = (x_l, y_l)$ is estimated as:

$$p(l|u, u') = \hat{f}_u(x_l, y_l) \cdot \hat{f}_{u'}(x_l, y_l) \quad (\text{Eq. 3.14})$$

Thus, their geographical similarity can be measured as:

$$sim_g(u, u') = \iint_{x,y} p(l|u, u') \quad (\text{Eq. 3.15})$$

The geographical similarity is incorporated into TSLR in the personalizing expertise process, that is, users with both high expertise in the topics which the target user prefers and similar spatial pattern are considered as excellent experts. Algorithm 2 summarizes the above method.

Complexity Analysis. There are four components of TSLRS including the topic modeling, inferring user expertise for various topics, geographical similarity calculation and preference prediction. According to Sontag & Roy (2011), the computation time of LDA is $\mathcal{O}(|U| \times (K + N_d)^3)$ where K is the topic number and N_d is the number of words for document d . The computation time of PageRank per topic (Eq. 3.7) is $Iter \times \mathcal{O}(|U|^2)$ where $Iter$ is the iteration number, so it would be $Iter \times \mathcal{O}(K \times |U|^2)$ to infer the user expertise for the K topics. For geographical similarity calculation, the computation time is $\mathcal{O}(\frac{|U| \times (|U| - 1)}{2})$. The computation time of preference prediction is $\mathcal{O}(|U| \times (|U| - 1))$. As $K, N_d \ll |U|$, the time complexity of TSLRS is $Iter \times \mathcal{O}(|U|^2)$ which is quite efficient as the time complexity is the same as some baselines such as user-based collaborative filtering and personalized PageRank (Sarwar et al., 2001; Wang et al., 2013).

3.2 Experiments

This section describes the experimental evaluation of the proposed algorithms TSLR and TSLRS against the baseline methods.

Table 3.2: Statistical of Selected Cities

Cities	Users	Check-ins	POIs
NY	2171	91163	17879
LA	1279	56882	12081
SF	2531	139664	14824

3.2.1 Experimental Setup

Datasets. A large-scale real dataset, Gowalla (Cho et al., 2011), was utilized for experiments. It is publicly available from the Stanford Network Analysis Project². Gowalla was launched in 2007 as a LBSN and closed in 2012. The dataset is a collection of 6,264,203 check-in records made by 96,591 users over 1,280,969 POIs from Feb 2009 to Oct 2010. The Foursquare venue API³ was used to collect the textual data because Gowalla no longer offers such API services after it being closed, which hinders the direct collection of the associated information of POIs from Gowalla.

Owing to the nature of the recommendation task, identifying the topics all over the world is almost trivial work because of the data sparsity. Therefore, the proposed algorithms are restricted to three metropolises, Los Angeles (LA), New York (NY), and San Francisco (SF), where users actively use LBSN services. Specifically, given the latitude and longitude of the center of a city⁴, the check-in records with a certain range are included. Table 3.2 shows the statistics of each city. In this study, the users who checked in at least 10 POIs were considered for experiments. 80% of each user’s check-in records were marked off as the training set while the remaining 20% as the test set. The source code⁵ is given online.

Comparison Methods. The proposed algorithms were compared with the following baseline recommendation methods:

²<http://snap.stanford.edu/index.html>

³<https://developer.foursquare.com/docs/api>

⁴<http://www.census.gov/geo/maps-data/data/gazetter.html>

⁵<https://github.com/jeffreyguo24/topicPPR>

- **UCF**. User-based Collaborative Filtering (UCF) first retrieves user and POIs, and then formulates a user-POI matrix (Sarwar et al., 2001). This approach considers the opinions from similar users. Cosine similarity is applied to assign weight between two users. Librec⁶ has been used for implementing UCF (Guo et al., 2015a);
- **PPR**. Personalized PageRank (PPR) (Page et al., 1999) is implemented to measure user expertise. In PPR, the transition probability is calculated by cosine similarity of two users as defined in Eq. 3.6 and random jump probability is uniform as $1/|U|$, indicating that there are equal probabilities of random jump to all entities. Recommendations are generated based on the expertise measured by the PageRank value. iGraph⁷ has been used for implementation and PPR is one of its built-in function;
- **TPR**. Topic PageRank (TPR) (Haveliwala, 2002) is also applied to measure user expertise. In TPR, the transition probability is calculated by cosine similarity of two users as defined in Eq. 3.6. Note that the random jump probability is non-uniform, which is defined as $\theta''_{u',z}$. Recommendations are generated by aggregating the preferences from the topic experts. The implementation is similar to PPR;
- **GCF**. Geographical Collaborative Filtering (GCF) is similar to UCF except that this method considers geographical similarity between two users defined by Eq. 3.15. Recommendations are generated by aggregating the opinions from geographically similar users. Scikit learn⁸ was used to implement the Kernel Density Estimation.

Evaluation Metrics. To evaluate the quality of our recommendation, three standard metrics were applied: *precision*, *recall* and *F-measure* (also known as F1 score). Precision

⁶<https://github.com/guoguibing/librec>

⁷<https://igraph.org/redirect.html>

⁸<https://scikit-learn.org/stable/modules/density.html>

indicates the ratio of retrieved POIs to the N recommendations. Recall defines the ratio of retrieved POIs to the POIs in a user testing set. F-measure considers both the precision and the recall metrics to compute a score. For each method, Top- N recommendations (5, 10, and 20) were generated. $Precision@N$ ($Recall@N$) is measured by averaging the precision (recall) values of all users to evaluate the overall performance defined as follows:

$$Precision@N = \sum_{u \in U} \frac{|TopN(u) \cap GT(u)|}{|TopN(u)|} \quad (\text{Eq. 3.16})$$

$$Recall@N = \sum_{u \in U} \frac{|TopN(u) \cap GT(u)|}{|GT(u)|} \quad (\text{Eq. 3.17})$$

where $TopN(u)$ and $GT(u)$ are the set of recommended POIs and the ground truth (or the test set) of user u , respectively. The F-measure ($F1@N$) is computed as:

$$F1@N = 2 \times \frac{Precision@N \times Recall@N}{Precision@N + Recall@N} \quad (\text{Eq. 3.18})$$

Parameter Setting. Several parameters should be set: Dirichlet hyper-parameters α , β , topic number K , and damping factor λ (for PPR, TPR, TSLR, and TSLRS). The Dirichlet priors were set as $\alpha = 50/K$, $\beta = 0.05$ recommended by Griffiths & Steyvers (2004), and LDA was run for 500 iterations for Gibbs sampling. We applied a grid search in $\{5, 10, 15, \dots, 50\}$ to find the optimal number number of topics, $K = 15$. For the damping factor, λ was set as 0.85, the commonly used setting for PageRank algorithms (Haveliwala, 2002). The tolerance threshold was set as 0.001 and the iteration number ($Iter$) was normally 100.

3.2.2 Experimental Results

The Effect of the Topic Number for TSLRS. The number of topics is an important hyper-parameter that have significant influence on recommendation performance.

 Figure 3.4: Analysis of the number of topics (K)

According to Figure 3.4, the performance (i.e., Precision@N, Recall@N and F1@N) gradually became better when K (the number of topics) was increased to 15. The recommendation performance kept at a stable level when K increased. When K was too small, the topics learned were too broad to be meaningful. As a result, the random walker cannot reach a good neighbor who shares similar preference with the target user. In fact, a proper amount of topics typically leads to fairly general topics that are mixtures of more specific subtopics. Such general topics have a higher chance to be evoked by one of the descriptive words of each user. The main contribution of topic modeling is to reduce the data sparsity of both item and word space so as to discover more effective neighbors.

Comparative Results. The recommendation approaches were evaluated by conducting several experiments. Figure 3.5 depicts the overall performance of 6 techniques over three cities with $N = 5, 10, 20$. The three cities shared the similar trends for both

3.5.a: Precision@N - LA

3.5.b: Precision@N - NY

3.5.c: Precision@N - SF

3.5.d: Recall@N - LA

3.5.e: Recall@N - NY

3.5.f: Recall@N - SF

3.5.g: F1@N - LA

3.5.h: F1@N - NY

3.5.i: F1@N - SF

Figure 3.5: Performance of Different Methods

precision, recall and F-measure. Precision@ N decreased while Recall@ N increased as more recommendations were generated. The relatively low results of the three metrics were reasonable (Ye et al., 2011; Wang et al., 2013) because of the low density of the datasets.

Generally, UCF, PPR and TPR performed worse than other methods. Both methods only consider user preference for POIs to retrieve similar users without incorporating any other information. As depicted, GCF showed comparable results and significantly outperformed UCF. This reveals the existence of a strong physical interaction between user check-in activities and POIs, that is, users' check-in decisions are highly influenced by geographical factor. In TPR, although the random jump probability is defined by a user's preference for a topic, the transition probability between two users are purely determined by their preferences for POIs, which is the same for different topics. Thus, it is difficult to find relevant neighbors regarding a certain topic. In contrast, by incorporating the topic preference in the transitional probability, TSLR can find useful neighbors who are more relevant to the topic. According to the experimental results, TSLR enhanced Precision by 15%, 10% and 6%, Recall by 9%, 8% and 4%, F1 score by 12%, 9% and 5% averagely across different settings of N for LA, NY and SF, respectively. The superior performance of TSLR shows that exploring users' preference for different topics can help find more valuable neighbors (or experts) for generating high-quality recommendations. Meanwhile, it also demonstrates the usefulness of content factor (in this study, POI categories and tags).

According to Figure 3.5, TSLRS demonstrated the best results in terms of precision and recall. Specifically, TSLRS significantly improved Precision@5 by 36%, 22%, and 29% for LA, NY, and SF respectively, while promoted Recall@5 by 33%, 23%, and 22% compared to the baseline method, UCF. Moreover, in comparison with TSLR, TSLRS increased Precision by 13%, 12% and 8%, Recall by 12%, 5% and 8%, and F1 score by

12%, 9% and 8% averagely across different settings of N for LA, NY and SF, respectively. The significance test (t-test) over the results of TSLR and TSLRS with p -value < 0.01 suggests that the results of them are significantly different from each other, that is, TSLRS improved the recommendation performance in a large margin. The results also unveils the effectiveness of geographical factor to improve POI recommendation quality, which in turn, explained why TSLRS performed better than TSLR. Therefore, incorporating users' geographical preference with topic preference can help find out more useful experts who hold high expertise on the topics that the user is interested as well as share similar spatial patterns. In summary, combining both content and geographical factors could deliver more effective general POI recommendations.

3.3 Conclusion

This study exploited the geographical and content factors for general POI recommendation. A topic-sensitive PageRank (TSLR) was proposed to learn expertise of users over various topics. Recommendations were created by combining opinions from neighbors based on their expertise and topic preferences of the target users. Next, TSLR was extended by incorporating geographical factor as TSLRS, so as to model the diversity property regarding to the two factors. The experimental results based on the real-world check-in records showed that the proposed methods outperformed the baselines.

Chapter 4

Aspect-aware Geo-Social Random Walk

Although the first study (Guo et al., 2015b) exploited graphical and content factors, it projected heterogeneous information into a homogeneous model, that is, the various types of data were represented in a homogeneous graph structure. However, such homogeneous representation broke the natural structure of LBSNs, causing the information loss. Thus, the second study introduced in this chapter focused on leveraging such rich information in an integrated manner to better capture the diversity property of user check-in behavior. Specifically, the proposed model employs geographical distance, social connection and user reviews to encode geographical, social and content influences, respectively. In particular, aspects from user reviews are considered, because they can model users' preferences in a finer granularity (Yang et al., 2013; He et al., 2015; Zhao et al., 2015). In order to fully utilize various types of information, a novel heterogeneous graph, *Aspect-aware Geo-Social Influence Graph* (AGS-IG) is constructed by fusing various relations among the three types of entities including users, POIs and aspects. The personalized POI recommendation task is thus transformed as a graph entity ranking problem in AGS-IG. A graph-based recommendation algorithm is developed based on both personalized PageRank (PPR) and meta-paths, named as *Aspect-aware Geo-Social Random Walk* (AGS-RW). It can fully exploit the heterogeneous graph structure as well

as capture the semantic relations among the various kinds of entities for more effective POI recommendation.

4.1 Aspect-aware Geo-Social Influence Graph

To model the heterogeneous information in a LBSN, a graph-based approach is adopted because it holds the advantage of embedding various types of data. This section presents the construction of the proposed novel heterogeneous graph, *Aspect-aware Geo-Social Influence Graph* that exploits geographical factor, social factor and aspects.

4.1.1 Dataset Description

The Yelp¹ dataset was utilized for experimental purpose. It contains various information which is suitable for this study, that is, user check-in records, geographical coordinates, social network, reviews as well as categories of POIs. Reviews are posted by users to evaluate their experience at a POI, for example, “The *staff* was super friendly and *food* was nicely cooked! will visit again.” in which “*staff*” and “*food*” are aspects. To extract aspects from user reviews, the toolkit in (Zhang et al., 2014d) was applied in this study. Further details are introduced in Section 4.3.1. Note that the aspects are domain specific, that is, aspects that are used for restaurants may not be suitable for describing parks. Hence, this study restricted the POIs to restaurants. Besides, a POI is described by categories, such as “*Italian restaurant*”, “*Pasta*”, etc.

4.1.2 Problem Formalization

The notations are introduced as follows. Let u, l, c, a denote user, POI, category and aspect, respectively; the POIs visited by user u_i are defined as a preference vector $\mathbf{p}_{u_i} = (r_{u_i, l_1}, r_{u_i, l_2}, \dots, r_{u_i, l_m}, \dots, r_{u_i, l_{|L|}})$, where r_{u_i, l_m} is the rating given by user u_i on POI l_m ,

¹<https://engineeringblog.yelp.com/2017/08/yelp-open-dataset-and-dataset-challenge-round-10.html>

and $|L|$ is the number of total POIs; a POI l_m is associated with a set of categories $C_{l_m} = (c_1, c_2, \dots, c_h)$; each triple $\langle u_i, l_m, a_p \rangle$ represents that user u_i has rated POI l_m and posted a review associated with aspect a_p .

Let U, L, A denote the sets of entities in AGS-IG representing users, POIs and aspects, respectively; E denotes the set of edges linking two arbitrary entities. Specifically, E_{UU} denotes the set of edges representing friendships. That is, if u_i is a friend of u_j , so $u_i \in F_{u_j}$ (F_{u_j} is the set of u_j 's friends) and $e_{u_i, u_j} \in E_{UU}$. Similarly, E_{AA} and E_{LL} represent the sets of edges for aspect-aspect and POI-POI relations, respectively; E_{UL} , E_{UA} and E_{LA} are the sets of edges representing user-POI, user-aspect and POI-aspect relations, respectively. With these entities and edges as input, AGS-IG = $(U \cup L \cup A, E_{UU} \cup E_{UL} \cup E_{UA} \cup E_{LA} \cup E_{LL} \cup E_{AA})$. Figure 4.1 is a running example to depict the graph structure where all the edges are directed. The recommendation task thus could be modeled as a graph entity ranking problem, that is, given AGS-IG, for a target user u_i , a ranked list of N recommendations is generated.

Figure 4.1: AGS-IG Structure

4.1.3 Intuitions for Constructing AGS-IG

Some intuitions provide directions to incorporate the three factors, that is, geographical, social and content factors (aspects) into AGS-IG. The following intuitions have been proved to be effective according to previous studies as described below:

Intuition 1: Due to the geographical constraints, users prefer to visit places that are near to their usual activity areas (Ye et al., 2011; Yuan et al., 2014).

Intuition 2: Friends often have more shared preferences than non-friends (Ye et al., 2010; Yang et al., 2013). The reputation of locations are spreading within the social network. Also, users would usually consult their friends about the quality of a POI before making decisions.

Intuition 3: A user visits a POI because certain aspects of this place attract this user (He et al., 2015), for example, the food quality and the atmosphere of a Sushi restaurant. With consideration of aspects, user preferences can be represented in a finer granularity instead of a coarse level by merely utilizing overall ratings. Likewise, a POI is preferred by users as it is operated well in certain aspects. Therefore, the extracted aspects can also reflect some properties of a POI.

Intuition 4: If user u_i expresses concerns about aspect a_p which is related to aspect a_q semantically, u_i may also be concerned about a_q . For instance, if “flavor” is important for u_i , he/she may also prefer POIs that are positive regarding “taste”, since “flavor” is semantically similar to “taste”.

4.1.4 Weight Assignment of Edges in AGS-IG

One challenge of applying a ranking algorithm in a heterogeneous graph is to assign weights to various types of edges. Different strategies need to be developed for weight computation of different types of edges in the heterogeneous graph. This section details the weight computation and normalization in AGS-IG.

Weights of E_{LL} . In total, there would be $|L||L - 1|$ edges in E_{LL} which would make the random walk process inefficient over AGS-IG. Following the common practice (Yuan et al., 2014), a threshold $dist_0$ is thus set to address this issue. That is, only if

$dist(l_m, l_n) < dist_0$, then l_m and l_n are connected, where $dist(l_m, l_n)$ is the geographical distance between l_m and l_n . After filtering out POIs that are too far away from l_m , the weight of edge e_{l_m, l_n} is computed. Ye et al. (2011) find that the probability of visiting a POI decays as the distance between this POI and the user's active area increases, which basically follows a power-law distribution (**Intuition 1**). In this study, the power-law function is also adopted to model the willingness of a user moving from l_m to l_n , that is, the edge weight of e_{l_m, l_n} is defined as:

$$w_{l_m, l_n} = \alpha \cdot dist(l_m, l_n)^\beta \quad (\text{Eq. 4.1})$$

where α, β are parameters of the power-law function. The parameter learning follows a linear curving fitting method (Anzai, 2012).

Weights of E_{UU} . To assign weights between user u_i and his/her friends (F_{u_i}), following the previous work (Wang et al., 2013), e_{u_i, u_j} is augmented by combining both friendship and similarity relations instead of treating each $u_j \in F_{u_i}$ equally. By incorporating similarity, more weights could be assigned to friends who share more interests with the target user u_i (**Intuition 2**). Hence, the weight of e_{u_i, u_j} is defined as:

$$w_{u_i, u_j} = \frac{(1 - \lambda)}{|F_{u_i}|} + \frac{\lambda}{|F_{u_i}|} \cdot sim(u_i, u_j) \quad (\text{Eq. 4.2})$$

where λ is the parameter to adjust the importance of friendship and similarity. $sim(u_i, u_j)$ is calculated by cosine similarity, $cos(\mathbf{p}_{u_i}, \mathbf{p}_{u_j})$.

Weights of E_{AA} . Users may also be concerned about aspects that are semantically related to the aspects that appear in their previous reviews (**Intuition 4**). Thus, an effective measurement to define the semantic relation between two aspects is critical to model users' preferences. Suppose aspect a_p has been used for commenting on a set of POIs, L_{a_p} . A straightforward way to estimate the edge weight of e_{a_p, a_q} is to apply the Jaccard Coefficient to L_{a_p} and L_{a_q} . However, there are many more POIs than aspects (in

all datasets, one aspect is used for describing less than two POIs on average). This data sparsity problem makes this method fails to capture the relation between two aspects since $w_{a_p, a_q} = 0$ in most cases. Hence, the associated categories are used to help define this measurement since the data is much denser (one aspect can be used for over ten categories of POIs). Moreover, it is reasonable to infer that if two aspects can be used for the same categories of POIs, they might be semantically related. More specifically, an aspect a_p is mentioned in the reviews of a POI l_m which is associated with a set of categories C_{l_m} . Here, the categories associated with a_p are defined as $C_{a_p} = C_{l_1} \cup C_{l_2} \cup \dots \cup C_{l_n}$ where $l_1, l_2, \dots, l_n$ are the POIs where a_p has been mentioned. The edge weight of a_p and a_q is computed by the Jaccard Coefficient:

$$w_{a_p, a_q} = \frac{|C_{a_p} \cap C_{a_q}|}{|C_{a_p} \cup C_{a_q}|} \quad (\text{Eq. 4.3})$$

Here an example is provided to further illustrate the benefits brought by using category when calculating the weights of E_{AA} . As shown in Figure 4.2, if only POIs are used, the similarity between “burger” and “hamburger” is 0 while it is $\frac{2}{3}$ by using categories which can help capture the semantic relation of these two aspects. Table 4.1 presents the Top-5 semantically related aspects of five selected example aspects. These examples show that this new measurement could capture the subtlety of various relations between

Figure 4.2: Similarity between aspects by using categories

Table 4.1: Examples of semantically related aspects

Aspect	Top-5 semantically related aspects
atmosphere	seating, cool, staff, service, ambiance
burger	chips, fries, bacon, potato, hamburger
flavor	taste, eating, menu, lunch, drink
appetizer	entrees, menus, tasting, shrimp, soup
service	meal, wait, price, staff, food

two aspects. Taking “burger” as an example, this measurement can not only detect the literally similar aspect “hamburger”, but also find “chips” and “fries” that are often eaten with “burger”.

Weights of E_{UL} , E_{UA} , E_{LA} . The edge weight between user u_i and POI l_m , that is, w_{u_i, l_m} , is defined by the rating that u_i provides to l_m . w_{u_i, a_p} indicates how much u_i is concerned about a_p and w_{l_m, a_p} represents what are the aspects of l_m concerning users (**Intuition 3**). They are defined as the frequency that aspect a_p has been mentioned in the reviews of u_i and l_m respectively.

Weight Normalization. The scales of the six types of weights are different. To be consistent, it is necessary to map all the weights into the same scale, that is, $(0, 1)$. Note that in AGS-IG, each entity is connected to three types of entities (user, POI, aspect). Hence, for each entity, the normalization is conducted in two steps: first we normalized the adjacent entities in same type and then all the adjacent entities. Suppose a user entity u_i is connected with user entity u_j , POI entity l_m and aspect entity a_p . First the weights of all edges, that is, $w_{u_i, u_j}, w_{u_i, l_m}, w_{u_i, a_p}$, are mapped into the range of $(0, 1)$. Take w_{u_i, u_j} as an example, and the first step normalization is formulated as:

$$w_{u_i, u_j} = \frac{w_{u_i, u_j}}{\sum_{u_k \in F_{u_i}} w_{u_i, u_k}} \quad (\text{Eq. 4.4})$$

After that, the second step normalization is performed, formulated as follows:

$$w_{u_i, v} = \frac{w_{u_i, v}}{\sum_{v \in N_{u_i}} w_{u_i, v}}, \quad (\text{Eq. 4.5})$$

where $N_{u_i} = \{u_j, l_m, a_p\}$ is the set of all neighbors for u_i in AGS-IG. By doing this, we ensure $\sum_{v \in N_{u_i}} w_{u_i, v} = 1$.

4.2 Graph-based POI Recommendation

This section describes a unified graph-based POI recommendation approach by leveraging the constructed AGS-IG. It is built upon personalized PageRank (PPR) and meta-paths, which enables the full exploitation of the heterogeneous graph structure as well as to capture the semantic relations among various types of entities.

4.2.1 Personalized PageRank with Meta-paths

As illustrated, the POI recommendation is converted to a graph entity ranking task on a heterogeneous graph which has high generality and expressiveness of modeling various types of information. However, algorithms like PPR that are only based on graph structure, fail to capture the semantic relevance of different types of entities and links. By solely relying on the graph structure, the entities linked to many neighbors are more likely to have higher PR values. These highly ranked POIs are mostly popular places that users might have already visited before. To address this issue, PPR is elaborated by incorporating meta-paths (Sun et al., 2011; Yao et al., 2015).

A meta-path is a sequence of relations connecting different entity types (Sun et al., 2011). By incorporating meta-paths, the ideas of other recommendation models such as user/item-based CF can be easily modeled in a generic way. Consider an example in AGS-IG where the random walker starts from a target user u_i and follows a meta-path $U \xrightarrow{\text{isFriendOf}} U \xrightarrow{\text{visited}} L$. Then it would reach POIs that are visited by the friends of u_i . This meta-path underpins the idea of user-based CF. Item-based CF also can be reflected by meta-path $U \xrightarrow{\text{visited}} L \xrightarrow{\text{isCloseTo}} L$, implying that the random walker would reach POIs that are close to the places u_i has visited. Thus, meta-path is also known as a path

guide since the random walker is somehow induced to follow the prescribed path (Lee et al., 2013). In terms of the meta-path lengths, they are generally within four steps as suggested in (Sun et al., 2011; Yao et al., 2015), since long meta-paths might introduce much noise.

Meta-path selection. There are many possible meta-paths between the three types of entities. Emulating all of them is computationally costly and also may hurt the recommendation quality since noise could be introduced. Since the goal is to recommend POIs to a target user, all the meta-paths should start with U and reach L eventually. Based on the proposed intuitions, three meta-paths² are selected:

- $U \xrightarrow{\text{isFriendOf}} U \xrightarrow{\text{visited}} L$
- $U \xrightarrow{\text{visited}} L \xrightarrow{\text{mentioned}} A \xrightarrow{\text{isMentionedBy}} L$
- $U \xrightarrow{\text{mentioned}} A \xrightarrow{\text{co-occuredWith}} A \xrightarrow{\text{isMentionedBy}} L$

Meta-path $U \rightarrow U \rightarrow L$ (for simplicity, the semantic above the arrow is omitted) is selected based on **Intuition 2**, implying that a user's check-in behavior can be influenced by his/her friends. The basic idea of $U \rightarrow L \rightarrow A \rightarrow L$ (**Intuition 3**) is similar to the item-based CF. That is, if l_n is similar to l_m visited by u_i , then l_n might be a good recommendation for u_i . Different from traditional item-based CF that estimates the similarity between two POIs by using rating information at a coarse level, the aspects are adopted to measure the similarity in a finer granularity, that is, if l_n shares more aspects with l_m , l_n is more likely to be on the recommendation list. Figure 4.3 depicts an example of this kind of meta-path. The meta-path $U \rightarrow A \rightarrow A \rightarrow L$ suggests that if aspect a_q is semantically related to aspect a_p that has been mentioned by user u_i , a_q might also influence the decision making of u_i (**Intuitions 3, 4**).

² $U \xrightarrow{\text{mentioned}} A$ means that a user has mentioned about an aspect in his/her reviews;
 $A \xrightarrow{\text{isMentionedBy}} L$ means that an aspect has been mentioned in the reviews of a POI.

 Figure 4.3: An example of meta-path: $U \rightarrow L \rightarrow A \rightarrow L$

One potentially useful meta-path is $U \rightarrow L \rightarrow L$ (**Intuition 1**), meaning that if u_i has visited l_m , he/she would probably also prefer l_n that is near l_m . However, the empirical study shows that the incorporation of this meta-path could not help improve recommendation accuracy, but deteriorated the performance. This might be caused by the over-emphasis of geographical factor which could demote the other factors.

Incorporating meta-paths into PPR. To fully exploit AGS-IG for effective POI recommendation, a unified ranking approach – AGS-RW (Aspect-aware Geo-Social Influence Random Walk) was proposed by extending PPR with the integration of selected meta-paths. Therefore, the new PR value can be reformulated as follows:

$$\mathbf{R}(t+1) = \mu \cdot \mathbf{M} \cdot \mathbf{R}(t) + (1 - \mu - \nu) \cdot \mathbf{d}_{u_i} + \nu \cdot \sum_{k=1}^{|P|} w_{p_k} \cdot \mathbf{M}_{p_k} \cdot \mathbf{R}(t) \quad (\text{Eq. 4.6})$$

where μ and ν are tunable parameters to balance the importance of three terms in the above equation; p_k denotes the k -th meta-path; w_{p_k} is the weight of meta-path p_k ; $\mathbf{M}_{p_k}$ is the transition matrix of p_k , defined as follows:

$$\mathbf{M}_{p_k} = \mathbf{M}_{T_1 T_2} \cdot \mathbf{M}_{T_2 T_3} \cdot \dots \cdot \mathbf{M}_{T_{s-1} T_s} \cdot \dots \quad (\text{Eq. 4.7})$$

where T_s is the type of the s -th entity in p_k and $\mathbf{M}_{T_{s-1} T_s}$ is a $|V| \times |V|$ adjacency matrix for entities in T_{s-1} to entities in T_s ($V = U \cup L \cup A$). For instance, the $\mathbf{M}_{p_k}$ for $U \rightarrow U \rightarrow L$ is calculated as: $\mathbf{M}_{p_k} = \mathbf{M}_{UU} \cdot \mathbf{M}_{UL}$. And $\mathbf{M}_{T_{s-1} T_s}$ is:

$$\mathbf{M}_{T_{s-1} T_s}(v, v') = \begin{cases} \frac{w_{v, v'}}{\sum_{v'' \in N_v} w_{v, v''}} & \text{if } T(v) = T_{s-1} \text{ and } T(v') = T_s \\ 0 & \text{otherwise} \end{cases} \quad (\text{Eq. 4.8})$$

Algorithm 3 AGS-RW for POI Recommendation

Input: check-in records, geographical information, social network, user reviews, selected meta-paths

Output: Top-N POIs

- 1: Initialize model parameters with the optimal settings.
// AGS-IG Construction
 - 2: Extract aspects from reviews.
 - 3: Assign weights with normalization to edges of AGS-IG.
// Recommendation for target user u_i
 - 4: Construct $\mathbf{M}$ based on the edge weights of AGS-IG.
 - 5: Construct $\mathbf{M}_{p_k}$ for each meta-path p_k based on Eq. 4.8.
 - 6: Set the personalized vector $\mathbf{d}_{u_i}$ based on Eq. 4.9.
 - 7: Run Eq. 4.6 iteratively to update $\mathbf{R}(t)$ until convergence.
 - 8: **return** Top-N POIs.
-

where $T(v)$ returns the type of entity v .

Moreover, the personalized vector $\mathbf{d}_{u_i}$ should be set initially. Different from homogeneous graph models, the value for the three types of entities need to be set in AGS-IG. For POIs and aspects, their values in $\mathbf{d}_{u_i}$ are set based on the preference of u_i . As for users, the similar method is adopted as PPR, that is, other users except u_i are set as 0. Thus, the value of a entity can be defined as:

$$\mathbf{d}_{u_i}(v) = \begin{cases} \gamma \cdot w_{u_i,v} & \text{if } v \in L \\ \eta \cdot w_{u_i,v} & \text{if } v \in A \\ 1 - \gamma - \eta & \text{if } v = u_i \\ 0 & \text{otherwise} \end{cases} \quad (\text{Eq. 4.9})$$

where $\mathbf{d}_{u_i}(v)$ is the value of v in $\mathbf{d}_{u_i}$, γ and η are tunable parameters that control the likelihood that the random walker teleports to an arbitrary entity. Algorithm 3 summarizes AGS-RW for POI recommendation, which mainly consists of two steps: AGS-IG construction (line 2-3) and recommendation generation (line 4-8).

Complexity Analysis. There are mainly two components in the random walk process of AGS-RW, which are the PPR part and meta-path part in Eq. 4.6. In one iteration, the computation time of the PPR part is $\mathcal{O}((|U| + |L| + |A|)^2)$ while it is $\mathcal{O}(|P| \times (|U| +$

Table 4.2: Statistics of three cities

	Cities	CH	PH	LV
Dataset	Users	1,106	2,148	4,794
	POIs	2,724	2,603	4,274
	Aspects	1,355	1,952	4,180
	Reviews	33,966	61,596	150,080
	Categories	148	155	175
AGS-IG	User-POI	25,847	46,206	113,239
	User-Aspect	141,691	284,946	784,550
	User-User	5,803	23,766	46,407
	POI-Aspect	130,364	210,953	549,527
	POI-POI	79,530	112,248	405,310
	Aspect-Aspect	915,981	1,898,326	87,257,531
	Density	14.8%	11.5%	12.1%

$|L| + |A|)^2$) for the meta-path part, where $|U| + |L| + |A|$ is the total number of entities in AGS-IG including users, POIs and aspects, and $|P|$ is the number of meta-paths. As $|P| \ll (|U| + |L| + |A|)$, the time complexity of AGS-RW is $Iter \times \mathcal{O}((|U| + |L| + |A|)^2)$, where $Iter$ is the number of iterations. Therefore, the computation time for AGS-RW scales with the size of AGS-IG. Hence, AGS-RW is scalable to large datasets.

4.3 Experiments

This section presents the evaluation of the proposed method compared with state-of-the-art methods on three real-world datasets. First, the detailed experimental setup is introduced followed by the results and analysis.

4.3.1 Experimental Setup

Datasets. The Yelp dataset³ (Round 10) was used for the experiment. Three cities were further chosen for experiments: Phoenix, Las Vegas and Charlotte where people

³<https://engineeringblog.yelp.com/2017/08/yelp-open-dataset-and-dataset-challenge-round-10.html>

actively used Yelp services. The statistics of the datasets of three cities are presented in Table 4.2. Following the preprocessing practice for evaluating recommender systems (He et al., 2015; Zhang & Chow, 2015), users who have visited more than 4 POIs were selected for evaluation. For the three datasets, the earlier 80% check-ins of each user were selected as training set and the remaining 20% check-ins were testing set. AGS-RW⁴ was implemented based on iGraph.

Aspect Extraction. Aspect extraction has been widely studied in previous works. Early studies (Hu & Liu, 2004) apply language rules on this task, and some recent advanced techniques integrate Recursive Neural Network (RNN) and Conditional Random Field (CRF) to extract opinion from reviews (Wang et al., 2016). This study utilized a state-of-art aspect extraction toolkit in Zhang et al. (2014d) to extract aspects from user reviews. The toolkit would generate aspect-opinion with sentiment polarity, for instance, (*food, delicious, positive*). The default settings of the toolkit were used for experiments.

Comparison Methods. The performance of the proposed algorithm was compared with the following baselines:

- **POP:** The top POIs are generated by randomly for each user. Librec has been used for implementing POP (Guo et al., 2015a);
- **UCF** (Herlocker et al., 1999): UCF is the user-based collaborative filtering which generates recommendations by aggregating ratings of the K most similar users of the target user. Cosine similarity is applied to measure user-user similarity. It was implemented by Librec (Guo et al., 2015a);
- **ICF** (Sarwar et al., 2001): ICF is the item-based collaborative filtering by aggregating ratings of the K most similar items of the target item. It has been widely used

⁴<https://github.com/jeffreyguo24/AGS>

in e-Commerce recommender systems (Sarwar et al., 2001). Cosine similarity is also adopted to calculate item-item similarity. Librec has been used for implementing ICF (Guo et al., 2015a);

- **PMF** (Salakhutdinov & Mnih, 2007): This is a matrix factorization based collaborative filtering method, which decomposes the user-item rating matrix into two low-rank user and item matrices. Librec has been used for implementing PMF (Guo et al., 2015a);
- **ItemRank** (Gori et al., 2007): This is a graph-based ranking algorithm which is performed on item-item graph where two items are connected if they have been visited by at least one common user. It has been implemented with Numpy by Gori et al. (2007)⁵;
- **Personalized PageRank (PPR)** (Haveliwala, 2002): PPR is a widely used graph-based method for personalized recommendation, which extends the idea of PageRank by replacing uniformly distributed teleportation vector with a personalized one. Here, PPR is performed on user-POI graph. PPR is a built-in function in iGraph library⁶;
- **Tensor Reduction (TR)** (Symeonidis et al., 2009): This is a state-of-the-art algorithm which models the three types of entities that exist in a social tagging system (users, items, and tags). The tags are similar to aspects as they can all be used for annotating the content of items. Therefore, it is necessary to compare AGS-RW with the advanced method in social-tagging area. TR was implemented with TensorFlow by Symeonidis et al. (2009)⁷;

⁵<https://github.com/arashkhoeini/itemrank/blob/master>

⁶<https://igraph.org/redirect.html>

⁷<https://github.com/Large-Scale-Tensor-Decomposition/tensorD>

- **LFBCA** (Wang et al., 2013): Location-friendship Bookmark-coloring Algorithm (LFBCA) performs PPR over a user-user graph, where the relations among users are determined by both their social links and check-ins similarity. Then, a top- N recommendations within a fixed geographical distance ($dist_0$) from K nearest neighbors found by PPR is created for the target user. It has been implemented by Liu et al. (2017)⁸;
- **GeoSoCa** (Zhang & Chow, 2015): This is a state-of-the-art POI recommendation approach by integrating social, geographical and category information into a unified linear framework. It has been implemented by Liu et al. (2017);
- **TriRank** (He et al., 2015): This is another recent state-of-art graph-based method. It first constructs a tripartite graph composed of users, items and aspects. Then, a regularization-based approach is developed for POI recommendation. The source code⁹ was given by the authors (He et al., 2015).

Parameter Settings. The optimal parameter settings were empirically found out for all the methods. Specifically, we applied a grid search in $\{10, 20, 30, \dots, 100\}$ to find the optimal $K = 50$ for UCF and ICF; we empirically searched for the optimal settings for the learning rate $\gamma = 0.0001$ and regularization coefficient $\lambda = 0.0001$ of PMF in $\{0.5, 0.1, 0.005, 0.01, 0.0005, 0.0001\}$; for ItemRank and PPR $\mu = 0.85$ based on (Haveli-wala, 2002); for LFBCA, we set $dist_0 = 3km$ and $\mu = 0.85$ based on (Wang et al., 2013); for TriRank, two sets of parameters were set according to the original work (He et al., 2015), $(\alpha = \eta_U = \eta_I = 1.0)$ for the traditional collaborative filtering effect and $(\beta = \lambda = \eta_A = 1.0)$ for the aspect filtering effect.

For AGS-RW, since there are quite many hyper-parameters, we relied on relevant papers and experiments to find the optimal values for them. As existing studies indicate

⁸<http://spatialkeyword.sce.ntu.edu.sg/eval-vldb17/>

⁹<https://github.com/jeffreyguo24/TriRank>

that the transition part and meta-path part play more important role (Lee et al., 2013), the default values of μ and ν were set as 0.4 while the weight of restart part $((1-\mu-\nu) \cdot \mathbf{d}_{u_i})$ was set as 0.2 in Eq. 4.6. The default value for λ was 0.7 based on (Wang et al., 2013; Yuan et al., 2014). We further followed the below process to find the optimal values of other hyper-parameters: 1) To find the proper weights for the three meta-paths, we first conducted a grid search for the weight of $U \rightarrow U \rightarrow L$ (w_{UUL}) while the weights of another two meta-paths (w_{ULAL}) and (w_{UAAL}) were the same (note $w_{UUL} + w_{ULAL} + w_{UAAL} = 1.0$), for instance, if w_{UUL} is 0.6, then w_{ULAL} and w_{UAAL} are both 0.2. The optimal weight of $U \rightarrow U \rightarrow L$ was set as 0.2, and then applied a grid search for each meta-path weight in $\{0.0, 0.1, 0.2, 0.3, \dots, 0.8\}$ for $U \rightarrow L \rightarrow A \rightarrow L$ and $U \rightarrow A \rightarrow A \rightarrow L$ (note $w_{ULAL} + w_{UAAL} = 0.8$); 2) We conducted a grid search for γ and η in $\{0, 0.1, 0.2, \dots, 1.0\}$. For Phoenix, $\gamma = 0.7$ and $\eta = 0.1$; for Las Vegas, $\gamma = 0.1$ and $\eta = 0.8$; for Charlotte, $\gamma = 0$ and $\eta = 0.3$; 3) With all other parameters fixed, we also applied a grid search in $\{1, 2, 3, \dots, 8\}$ to find the optimal value of $dist_0$, that is, $dist_0 = 3km$. After tuning $dist_0$, the proposed AGS-RW delivered the best recommendation results. The tolerance threshold of the random process was set as 0.001 and the iteration number ($Iter$) was normally 300.

Evaluation Metrics. To evaluate the performance of all the methods, three widely used metrics applied in Chapter 3 were also adopted: Precision@ N , Recall@ N and F-measure@ N (denoted as Pre@ N , Rec@ N and F1@ N) where N is the size of the Top- N recommended POI ranking list. In this study, $N = 5, 10, 20$ and each metric for each user was calculated.

4.3.2 Results and Analysis

Results of the Variants. First, the impacts of the three factors were investigated, that is, geographical factor (geographical distance between POIs), social factor (friendships)

and content factor (aspect-aspect relations). Five different variants of the proposed methods were compared:

- Rec is the basic recommendation method without any of the three factors above. Meta-path $U \rightarrow L \rightarrow A \rightarrow L$ is available in Rec;
- GRec is the upgraded version of Rec by incorporating geographical factor. Meta-path $U \rightarrow L \rightarrow A \rightarrow L$ is also used in GRec;
- SRec considers social factor based on Rec. Meta-paths $U \rightarrow U \rightarrow L$ and $U \rightarrow L \rightarrow A \rightarrow L$ are applied in SRec;
- ARec considers aspect-aspect relations based on Rec. Meta-paths $U \rightarrow A \rightarrow A \rightarrow L$ and $U \rightarrow L \rightarrow A \rightarrow L$ are used in ARec;
- AGS-RW is the proposed method by simultaneously fusing the three factors with all three meta-paths.

For Rec, since none of the three factors was considered mentioned above, the edges E_{LL} , E_{UU} and E_{AA} are removed from AGS-IG, and only other three types of links, that is, user-POI (E_{UL}), user-aspect (E_{UA}), POI-aspect (E_{LA}) are kept. Therefore, the graph is degraded as $G_0 = (U \cup L \cup A, E_{UL} \cup E_{UA} \cup E_{LA})$ in Rec. A previous study (He et al., 2015) has already shown that user-item, user-aspect and item-aspect relations can improve the recommendation performance. Therefore, Rec is regarded as the baseline method focusing on the impacts of E_{LL} , E_{UU} , E_{AA} . Figure 4.4 records the performance of all variants in terms of precision and recall with $N = 5, 10, 20$. It demonstrates that the performance of Rec was worse than that of the other four methods in most cases, implying the effectiveness of the three influential factors.

For GRec, the geographical factor is incorporated into the basic Rec, that is, $G_{+G} = G_0 \cup E_{LL}$. According to the results, GRec generally outperformed Rec, which reinforced

Figure 4.4: Performance of our different variants on the three real-world datasets.

the fact that geographical factor is a useful factor for POI recommendation. In addition, the meta-path $U \rightarrow L \rightarrow L$ is also incorporated. However, the performance significantly dropped, because the geographical factor might be over-emphasized. This meta-path might introduce much noise into the random walk process, that is, the random surfer might reach nearby POIs, but they might be against the user's interests.

By fusing social factor, SRec was proposed with $G_{+S} = G_0 \cup E_{UU}$. It could be observed that SRec performed better than Rec, with improvements of 3.21%, 1.38% and 1.93% regarding precision, recall and F1 score on average across different settings of $N = 5, 10, 20$ on the three datasets, indicating that social factor can help improve recommendation accuracy to some extent. The results verifies the finding in (Ye et al., 2011; Cheng et al., 2012) that social factor in LBSN context has limited influence since

Figure 4.5: Analysis of w_{UUL}

friends may share common interest but may not always visit common POIs since they might stay far away from each other. Then, by adding aspect-aspect relations, ARec was developed with $G_{+A} = G_0 \cup E_{AA}$. The results indicates ARec performed better than Rec in all cases. On average, ARec outperformed Rec by 2.97%, 2.52% and 2.72% in terms of precision, recall and F1 score, respectively. This demonstrates that the effect of aspect-aspect relations is more stable compared with the other two factors. Finally, by fusing the three factors into Rec, AGS-RW exhibited the best performance in comparison with the other four methods. Averagely, AGS-RW improved the performance by 7.33%, 5.64% and 6.37% in precision, recall and F1 score respectively compared with Rec, proving that the strength of exploitation of all three factors. In summary, there are various factors affecting the check-in decision by the user, and properly combining these factors into a general model could promote the recommendation quality.

Analysis of Hyper-parameters. Since there are quite many hyper-parameters in-

 Figure 4.6: Analysis of w_{ULAL} and w_{UAAL} ($w_{UAAL} = 0.8 - w_{ULAL}$)

involved in AGS-RW, it is necessary to analyze their impacts on recommendation performance. As mentioned in parameter setting, μ , ν and λ were set as the default values according to related papers. Extensive experiments have been conducted to analyze the effects of various settings for 1) meta-path weights in Eq. 4.6, that is, w_{UUL} , w_{ULAL} and w_{UAAL} ; 2) the parameters which control the random walker teleportation, that is, γ and η in Eq. 4.9; and 3) the distance threshold ($dist_0$) which filters distant POI pairs in AGS-IG as described in Section 4.1.4.

The Effect of w_{UUL} , w_{ULAL} and w_{UAAL} . As described in parameter setting, two steps were performed to find the optimal values of the three meta-path weights: 1) find the optimal value of w_{UUL} ; and 2) while fixing w_{UUL} , find the optimal values of w_{ULAL} and w_{UAAL} . Note, tuning the meta-path weights is the second step so that even the best results in Figure 4.5 and Figure 4.6 are worse than the results of AGS-RW in Table 6.3. Figure 4.5 plots the precision and recall metrics under different settings of w_{UUL} . Although, AGS-RW

generated better results (Pre@10 (LV), Pre20@10 (LV) and Rec@5 (CH)) with $w_{UUL} = 0.3$ or 0.4 , we can observe that when $w_{UUL} = 0.2$, the recommendation performance reached the best for the remaining cases. By fixing $w_{UUL} = 0.2$, we further conducted experiments to tune w_{ULAL} and w_{UAAL} as shown in Figure 4.6. Obviously, the best recommendation results were achieved for most cases when $w_{ULAL} = 0.4$ and $w_{UAAL} = 0.4$. Either w_{ULAL} (or w_{UAAL}) was set too small or too large, the performance became worse. Overall, $w_{UUL} = 0.2$ and $w_{ULAL} = 0.4$ and $w_{UAAL} = 0.4$ are optimal settings for the three meta-path weights. Such results indicate that the social factor carries less important role for user check-in decision, because the preference sharing is constrained by geographical distance between friends. To explain, two friends could share very few POIs as they might live far away from each other (Wang et al., 2013; Cheng et al., 2012; Zhang & Chow, 2015). Moreover, $ULAL$ and $UAAL$ are both related to aspects, which could explain why their weights are close.

The Effect of γ and η . According to Eq. 4.9, γ and η control the probability that the random walker jumps to a POI, an aspect or a user. Figure 4.7 illustrates the recommendation performance with various settings of γ and η . As $\gamma + \eta = 1.0$, the figure is a heatmap in a triangle shape. Figure 4.7 shows that the results with $N = 5$ over the three cities as similar trends can be observed when $N = 10, 20$. Based on the experimental results, the random walker tends to teleport to different types of entities for obtaining better recommendation quality. Specifically, it jumps to POIs, aspects and users with a larger probability in PH, LV and CH, respectively.

The Effect of $dist_0$. Two POIs are connected if their geographical distance is within the defined distance threshold in AGS-IG. Thus, the distance threshold ($dist_0$) may have a huge impact on recommendation quality. Experiments were conducted to investigate the effect of $dist_0$. Figure 4.8 demonstrates the performance for different settings of $dist_0$ with other optimal hyper-parameters fixed. It can be observed that the performance

 Figure 4.7: Analysis of γ and η .

became poor when $dist_0$ was set either too small or too large. While $dist_0$ was set below $2km$, then many of the interesting POIs about a POI could not be discovered by the random walker. On the other hand, the random walker might reach some irrelevant POIs when $dist_0$ was beyond $4km$. The experimental results align with previous studies showing that a POI has significant influences on other POIs within a $3km^2$ region (Wang et al., 2013; Lian et al., 2014).

Figure 4.8: Analysis of Distance Threshold ($dist0$)

Comparative Results. Table 5.4 and 5.5 detail the performance of all comparison methods on the three datasets. The main findings from this table can be summarized as follows.

Unsurprisingly, by generating recommendations based on POI popularity without considering personalization, POP performed worst. UCF and ICF are all memory-based collaborative filtering (CF) methods, whereas PMF is a model-based CF approach. The performance of PMF was generally better than that of UCF and ICF, verifying that model-based CF methods are more effective than memory-based CF methods.

The four state-of-the-art graph-based ranking algorithms – ItemRank, PPR, LFBCA and TriRank – they all outperformed PMF. One explanation might be that recommendation could be actually regarded as a ranking problem (Rendle et al., 2009). PMF is a rating prediction based method while the others are graph ranking methods. The empirical results further confirm that ranking based methods are more effective than

rating prediction based approaches for recommendation tasks. Furthermore, among the four graph ranking methods, LFBCA and TriRank outperformed PPR and ItemRank in most cases since these two approaches exploit auxiliary information in LBSNs while the other two only use on the user-item interaction information. TriRank outperformed LFBCA, demonstrating the effectiveness of aspects. TriRank and TR both exploit aspects. However, TR puts users, POIs and aspects in a tensor which is extremely sparse, the learned embeddings thus cannot precisely represent users' preference or POIs' characteristics. In contrast, TriRank leverages a unified graph structure to accommodate users, POIs and aspects in a common space. Hence, TriRank can easily take advantage of their relationships for POI recommendation. GeoSoCa exploits geographical, social and category information in a linear way, thus it cannot well capture the complex relationship of various factors.

Compared with all the other methods, the AGS-RW approach consistently performed best. Averagely, the improvements with respect to precision, recall, F1 score were 5.35%, 4.82% and 4.97%, respectively (both with p -value < 0.01) across different settings of N on the three datasets, respectively. This implies that recommendation performance can be further enhanced by appropriately considering the three influential factors. Furthermore, it could also be observed that in Phoenix dataset, there was no significant performance difference among the four baseline graph-based methods, even ItemRank performed the best in terms of Rec@5. By conducting analysis on Phoenix dataset, the number of reviews per user is the least among the three datasets, that is, one user has 28 reviews in Phoenix while the user in Las Vegas has more than 31 reviews. This restrains the strength of using reviews to enhance the recommendation. As indicated, there was also a less significant improvement by AGS-RW compared with other two cities. However, AGS-RW could deliver better results more consistently while TriRank performed worse than other graph ranking algorithms regarding Pre@20(PH). This demonstrates that the

Table 4.3: Precision and recall of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-RW achieves relative to the best performance of other methods.

City	Metric(%)	POP	UCF	ICF	PMF	ItemRank	PPR	LFBCA	GeoSoCa	TR	TriRank	AGS-RW	Improve(%)
Phoenix	Pre@5	0.493	0.894	0.708	1.248	1.390	1.415	1.467	1.623	1.650	1.890*	2.060	8.99
	Pre@10	0.393	0.987	0.796	1.173	1.350	1.396	1.174	1.588	1.602	1.830*	1.939	5.96
	Pre@20	0.382	1.034	0.784	1.101	1.544	1.621	1.014	1.500	1.527	1.734*	1.869	7.79
	Rec@5	0.370	0.693	0.771	1.028	1.211	1.269	1.536	1.600	1.652	1.751*	1.821	4.00
	Rec@10	0.718	1.604	1.742	1.880	2.573	2.617	3.092	2.901	2.914	3.300*	3.531	7.00
	Rec@20	1.137	3.370	3.489	3.603	5.742	6.111	6.130*	5.700	5.713	6.126	6.502	6.07
Las Vegas	Pre@5	0.622	0.626	0.588	0.948	1.176	1.255	1.120	1.470	1.452	1.589*	1.653	4.03
	Pre@10	0.407	0.542	0.501	0.878	0.976	0.960	1.060	1.280	1.258	1.331*	1.371	3.01
	Pre@20	0.357	0.485	0.433	0.754	1.139	1.190	0.975	1.370	1.362	1.416*	1.455	2.75
	Rec@5	0.411	0.552	0.508	0.865	1.171	1.244	0.672	1.420	1.428	1.431*	1.474	3.00
	Rec@10	0.530	0.939	0.882	1.478	2.074	2.047	1.406	2.260	2.260	2.459*	2.557	3.99
	Rec@20	1.076	1.647	1.610	2.436	4.768	5.303	2.823	5.180	5.163	5.345*	5.502	2.94
Charlotte	Pre@5	0.691	1.483	0.976	1.817	2.724	2.762	2.007	2.586	2.602	2.821*	3.018	6.98
	Pre@10	0.566	1.456	0.922	1.798	2.383	2.535	1.881	2.134	2.128	2.532*	2.633	3.86
	Pre@20	0.488	1.433	0.945	1.581	2.280	2.356	1.939	1.917	2.054	2.360*	2.474	4.83
	Rec@5	0.417	1.117	0.977	1.485	2.421	2.367	1.964	1.875	1.898	2.548*	2.625	3.02
	Rec@10	0.773	2.229	1.940	2.805	4.431	4.457	3.669	3.070	3.145	4.407*	4.760	6.80
	Rec@20	1.241	4.137	4.143	4.849	7.974	7.971	7.562	5.833	5.987	8.034*	8.558	6.52

Table 4.4: F-measure (F1 score) of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-RW achieves relative to the best performance of other methods.

City	Metric(%)	POP	UCF	ICF	PMF	ItemRank	PPR	LFBCA	GeoSoCa	TR	TriRank	AGS-RW	Improve(%)
Phoenix	F1@5	0.423	0.781	0.738	1.127	1.294	1.338	1.501	1.611	1.651	1.818*	1.933	6.34
	F1@10	0.508	1.222	1.093	1.445	1.771	1.820	1.702	2.052	2.067	2.354*	2.503	6.33
	F1@20	0.572	1.582	1.280	1.687	2.433	2.562	1.740	2.375	2.410	2.703*	2.903	7.42
Las Vegas	F1@5	0.495	0.587	0.545	0.905	1.173	1.249	0.840	1.445	1.440	1.506*	1.558	3.49
	F1@10	0.460	0.687	0.639	1.102	1.327	1.307	1.209	1.634	1.616	1.727*	1.785	3.35
	F1@20	0.536	0.749	0.682	1.152	1.839	1.943	1.449	2.167	2.155	2.239*	2.301	2.79
Charlotte	F1@5	0.520	1.274	0.976	1.634	2.564	2.549	1.985	2.174	2.195	2.678*	2.808	4.86
	F1@10	0.653	1.761	1.250	2.191	3.099	3.232	2.487	2.518	2.538	3.216*	3.391	4.91
	F1@20	0.700	2.129	1.539	2.385	3.547	3.637	3.087	2.886	3.059	3.648*	3.838	5.21

exploitation of the three factors could lead to a more robust approach, thus achieving more stable improvements.

In the other two datasets (Las Vegas and Charlotte), the performance was enhanced more significantly. For Las Vegas, the improvements with respect to precision, recall, F1 score were 3.26%, 3.31% and 3.28%, respectively (both with p -value < 0.01) across different settings of N on the three datasets, respectively. In Charlotte, the data density is the largest among the three datasets according to Table 4.2. This rich information in Charlotte enabled the proposed approach to generate better results than other methods by a large margin. The high heterogeneity in these datasets further reinforced the effectiveness of modeling heterogeneous information with a unified graph approach.

4.4 Conclusion

This study focused on exploiting heterogeneous information of LBSNs in an integrated way to improve POI recommendation quality. An Aspect-aware Geo-Social Influence recommendation approach – AGS-RW was proposed to enhance POI recommendation quality. First, a heterogeneous graph, AGS-IG was constructed to model geographical factor, social factor and aspects in a unified way. Then a graph-based ranking method was designed by combining personalized PageRank with meta-paths for full exploitation of both graph structure and various semantic relations in AGS-IG. More importantly, AGS-RW could inherently model the diversity characteristic of multiple influential factors, which could facilitate a more deeper understanding of user check-in decision. Empirical evaluation on real-world LBSN datasets demonstrates the superiority of AGS-RW against state-of-the-art algorithms.

Chapter 5

Aspect-aware Geo-Social Matrix Factorization

As mentioned in Chapter 1, user check-in behavior exhibits two properties: diversity and imbalance, that is, multiple factors influence user check-in decision and carry different levels of importance for the decision. Although, the second study has explored the heterogeneous representation to organize various types of information in a unified space, it fails to model the imbalance property. In fact, a user’s check-in decision is determined by both his/her preferences and the POI’s characteristics. To explain, each user and POI are characterized by multiple influential factors at various levels of importance. Thus, to better model user check-in decision, it is necessary to capture subtle differences about the effects of various influential factors on both users and POIs.

Hence, to jointly capture both two unique characteristics of user check-in behavior, an *Aspect-aware Geo-Social Matrix Factorization* (AGS-MF) approach is proposed to leverage the capability of heterogeneous graph and matrix factorization techniques. AGS-MF is capable of unifying various influential factors as well as learning the saliences of them at the personalized level. Specifically, this study adopts the heterogeneous graph, *Aspect-aware Geo-Social Influence Graph* (AGS-IG) proposed in Chapter 4 to accommodate various types of information in a unified representation space including users, POIs and aspects in user reviews, as well as their relations. Then, a meta-path based random walk

Figure 5.1: A running example for AGS-IG structure

process is designed to efficiently discover reliable neighbors of each user and POI based on the diverse influential factors. By assuming that neighbors should be closer to the given entity (either a user or POI) in the latent space, the regularizers are incorporated into AGS-MF to constrain the distance of latent representations between them. To further capture the imbalance of various influential factors, then personalized weights are added to those regularizers, which represent the strength of neighbor relations regarding the corresponding meta-path, that is, the importance of the influential factor. In this way, the saliency of each influential factor can be jointly learned with user and POI latent feature vectors. As a result, the learned representations of users and POIs can not only reflect user preference and POI characteristic respectively, but also preserve the heterogeneous information encoded in AGS-IG.

5.1 Meta-path based Random Walk

Due to the large amount of entities involved in AGS-IG, it is not practical to iterate the whole set of entities to find their neighbors. Therefore, this study developed an effective walking strategy on AGS-IG, a meta-path based random walk process that can retrieve semantically related neighbors efficiently.

The connection between two entities might be driven by multiple influential factors, for example, in Figure 5.1, u_1 and u_2 are not only social friends, but also share aspect a_2 .

Algorithm 4 Meta-path based Random Walk

Input: AGS-IG, entity v , meta-path p , walk length wl , walk number wn
Output: A set of neighbors, $\mathcal{N}_p(v)$

```

1:  $Paths = []$ 
2: for  $n \leftarrow 1$  to  $wn$  do
3:    $path = []$ 
4:    $s \leftarrow 1$ 
5:   while  $s \leq wl$  do
6:     Walk to  $v_s$  based on  $\text{Prob}(v_s|v_{s-1}, p)$ 
7:     if  $T(v_s) == T_s$  then
8:       Add  $v_s$  to  $path$ 
9:        $s \leftarrow s + 1$  //move one step forward
10:  Add  $path$  to  $Paths$ 
11: Extract neighbors from  $Paths$  as  $\mathcal{N}_p(v)$ 
12: return  $\mathcal{N}_p(v)$ 

```

Hence, u_2 can be considered as a relevant neighbor for u_1 as their relation is determined by both their social connection and similar opinions towards POIs. As meta-path (Sun et al., 2011) is a very useful concept to characterize the semantic patterns for a KG, it could capture the semantic relations in AGS-IG. First, the concept of *neighbors* are extended as the entities connected to the given entity (a user or POI) via different meta-paths, instead of directly linked ones. For instance, in Figure 5.1, although they are not directly linked, u_3 can be considered as a neighbor of u_1 based on meta-path $U \rightarrow L \rightarrow U$ as they share a common POI l_1 .

Given a meta-path $p = T_1 \rightarrow T_2 \cdots \rightarrow T_m \cdots$, where T_m is the type of the m -th entity, the transition probability between two linked entities is determined by the neighborhood size with constraint based by p , which is defined as follow:

$$\text{Prob}(v_m|v_{m-1}, p) = \begin{cases} \frac{1}{|\mathcal{N}_{T_m}(v_{m-1})|} & \text{if } T(v_m) = T_m \text{ and } T(v_{m-1}) = T_{m-1} \\ 0 & \text{otherwise} \end{cases} \quad (\text{Eq. 5.1})$$

where v_m is the m -th entity in p , $T(v_m)$ returns the type of v_m and $\mathcal{N}_{T_m}(v_{m-1})$ is the neighbor set of v_{m-1} in type T_m . By following p with the transition probability, the random walker can generate a path until it reaches the walk length. The process terminates

if enough paths are created. Finally, the neighbors from those paths are extracted for a given user or POI.

Note that as the goal is to find reliable neighbors for users and POIs, all meta-paths should start with U (L) and reach U (L) eventually. The following meta-paths are selected¹: $U \rightarrow L \rightarrow U$, $U \rightarrow U$, $U \rightarrow L \rightarrow A \rightarrow L \rightarrow U$ and $U \rightarrow A \rightarrow U$ are selected for users; $L \rightarrow U \rightarrow L$, $L \rightarrow L$ and $L \rightarrow A \rightarrow L$ are selected for POIs. These meta-paths represent various influential factors that encode different semantic relations, for example, $U \rightarrow A \rightarrow U$ ($L \rightarrow A \rightarrow L$) can help discover neighbors sharing the same aspects while $L \rightarrow L$ can find nearby POIs. Algorithm 4 describes the details about the meta-path based random walk process. After the paths are generated, neighbors for each user and POI are discovered. Suppose for meta-path $p = U \rightarrow A \rightarrow U$, two paths are obtained for u_1 : $u_1 \rightarrow a_3 \rightarrow u_8$ and $u_1 \rightarrow a_7 \rightarrow u_2$, u_8 and u_2 are thus included in the neighbor set of u_1 regarding meta-path p , that is, $\{u_2, u_8\} \subset \mathcal{N}_p(u_1)$.

5.2 Aspect-aware Geo-Social Matrix Factorization

This section presents the proposed approach – Aspect-aware Geo-Social Matrix Factorization (AGS-MF). It incorporates the extracted neighbors with a model-based strategy to automatically learn the optimal weights for different meta-path based neighbor sets at personalized level. By this way, the impacts of various factors could be captured more precisely for both individual user and POI. Matrix factorization (MF) (Salakhutdinov & Mnih, 2007) is an efficient method widely applied in recommender systems. It factorizes the user-POI rating matrix $R \in R^{|U| \times |L|}$ into low-rank user-latent matrix $\mathcal{U} \in R^{|U| \times d}$ and POI-latent matrix $\mathcal{V} \in R^{|V| \times d}$ (d is the dimension of latent vectors). The rating prediction of user u_i on a POI l_j , that is, $\hat{r}_{i,j} = \mathcal{U}_i \mathcal{V}_j^\top$.

¹The semantics above the arrows are the same as Chapter 4. Here they are omitted for simplicity.

The objective is to incorporate the discovered neighbors that encode diverse influential factors into MF to better model users' preference and POIs' characteristics. By assuming that neighbors should be close to each other in the latent space, regularization terms are integrated into MF, so as to constrain the distance of latent feature vectors of the neighbors. Meanwhile, personalized meta-path weights for individual users are also incorporated to control the strength of regularization. Inspired by this idea, "personalized" weights of meta-paths are also assigned for each POI.

All the weights are jointly learned with user and POI latent feature vectors, which enabled AGS-MF to capture the saliences of various influential factors via a data-driven method with user historical check-in records. By doing so, both diversity and imbalance properties of user check-in behavior can be effectively modeled in a unified framework. The objective function is thus defined as follows:

$$\begin{aligned}
 \mathcal{L} = & \frac{1}{2} \sum_i I_{i,j} (r_{i,j} - g(\mathcal{U}_i \mathcal{V}_j^\top))^2 + \frac{\lambda_u}{2} \|\mathcal{U}\|_F^2 + \frac{\lambda_l}{2} \|\mathcal{V}\|_F^2 \\
 & + \frac{\alpha_u}{2} \sum_{p \in \mathcal{M}_u} \sum_i \|\Omega_{i,p} (\mathcal{U}_i - \sum_{u_k \in \mathcal{N}_p(u_i)} s_{i,k} \mathcal{U}_k)\|_F^2 \\
 & + \frac{\alpha_l}{2} \sum_{p \in \mathcal{M}_l} \sum_j \|\Theta_{j,p} (\mathcal{V}_j - \sum_{l_q \in \mathcal{N}_p(l_j)} s_{j,q} \mathcal{V}_q)\|_F^2 \\
 & + \frac{\lambda_\Omega}{2} \|\Omega\|_F^2 + \frac{\lambda_\Theta}{2} \|\Theta\|_F^2
 \end{aligned} \tag{Eq. 5.2}$$

where $I_{i,j}$ is an indicator function that equals 1 if user u_i rated POI l_j and equals 0 otherwise; $r_{i,j} \in [0, 1]$ is the rating of user u_i on POI l_j after min-max normalization; $g(x) = 1/(1 + \exp(-x))$ is the logistic function that bounds the range of prediction into $[0, 1]$; α_u and α_l are parameters to control the importance of user and item regularization respectively; $\mathcal{M}_u$ and $\mathcal{M}_l$ represent the sets of meta-paths for users and POIs respectively, that is, $\mathcal{M}_u = \{ULU, UU, ULALU, UAU\}$ ² and $\mathcal{M}_l = \{LUL, LL, LAL\}$; $s_{i,k}$ represents the personalized PageRank value of k after normalization (that is, $\sum_{k \in \mathcal{N}_p(i)} s_{i,k} = 1$); $\Omega_{i,p} \in \Omega$ and $\Theta_{j,p} \in \Theta$ represent the weights of meta-path p for u_i and l_j respectively, Ω

²The arrows are removed for better readability.

Table 5.1: Derivation of $\mathcal{U}_i$, $\mathcal{V}_j$, $\Omega_{i,p}$ and $\Theta_{j,p}$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathcal{U}_i} &= \sum_j I_{i,j} g'(\mathcal{U}_i \mathcal{V}_j^\top) (g(\mathcal{U}_i \mathcal{V}_j^\top) - r_{i,j}) \mathcal{V}_j + \lambda_u \mathcal{U}_i \\ &\quad + \alpha_u \sum_{p \in \mathcal{M}_u} (\Omega_{i,p} (\mathcal{U}_i - \sum_{u_k \in \mathcal{N}_p(u_i)} s_{i,k} \mathcal{U}_k)) \\ &\quad - \alpha_u \sum_{p \in \mathcal{M}_u} \Omega_{k,p} s_{k,i} \sum_{\{u_k | u_i \in \mathcal{N}_p(u_k)\}} (\mathcal{U}_k - \sum_{u_{k'} \in \mathcal{N}_p(u_k)} s_{k,k'} \mathcal{U}_{k'}) \end{aligned} \quad (\text{Eq. 5.3})$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathcal{V}_j} &= \sum_i I_{i,j} g'(\mathcal{U}_i \mathcal{V}_j^\top) (g(\mathcal{U}_i \mathcal{V}_j^\top) - r_{i,j}) \mathcal{U}_i + \lambda_v \mathcal{V}_j \\ &\quad + \alpha_v \sum_{p \in \mathcal{M}_l} (\Theta_{j,p} (\mathcal{V}_j - \sum_{l_q \in \mathcal{N}_p(l_j)} s_{j,q} \mathcal{V}_q)) \\ &\quad - \alpha_v \sum_{p \in \mathcal{M}_l} \Theta_{q,p} s_{q,j} \sum_{\{l_q | l_j \in \mathcal{N}_p(l_q)\}} (\mathcal{V}_q - \sum_{l_{q'} \in \mathcal{N}_p(l_q)} s_{q,q'} \mathcal{V}_{q'}) \end{aligned} \quad (\text{Eq. 5.4})$$

$$\frac{\partial \mathcal{L}}{\partial \Omega_{i,p}} = \alpha_u \Omega_{i,p} \|\mathcal{U}_i - \sum_{u_k \in \mathcal{N}_p(u_i)} s_{i,k} \mathcal{U}_k\|_F^2 + \lambda_\Omega \Omega_{i,p} \quad (\text{Eq. 5.5})$$

$$\frac{\partial \mathcal{L}}{\partial \Theta_{j,p}} = \alpha_l \Theta_{j,p} \|\mathcal{V}_j - \sum_{l_q \in \mathcal{N}_p(l_j)} s_{j,q} \mathcal{V}_q\|_F^2 + \lambda_\Theta \Theta_{j,p} \quad (\text{Eq. 5.6})$$

and Θ are weight matrices for users and POIs respectively; $\|\cdot\|_F$ denotes the Frobenius norm; and λ_u , λ_l , λ_Ω , λ_Θ are regularization coefficients for easing over-fitting.

Optimization. Stochastic gradient descent approach was adopted (Bottou, 2010) to optimize AGS-MF formulated by Eq. 5.2. Table 5.1 presents the derivatives of $\mathcal{U}_i$, $\mathcal{V}_j$, $\Omega_{i,p}$ and $\Theta_{j,p}$, where $g'(x) = e^{-x}/(1+e^{-x})^2$ is the derivative of $g(x)$. The learning process is summarized in Algorithm 5.

Complexity Analysis. The computational time is mainly taken by evaluating the objective function $\mathcal{L}$ and updating the relevant variables. The time to compute $\mathcal{L}$ is $\mathcal{O}(d \times |R| + d \times |U| + d \times |L| + d \times |U| \times |\mathcal{M}_u| \times |\bar{\mathcal{N}}| + d \times |L| \times |\mathcal{M}_l| \times |\bar{\mathcal{N}}| + d \times |U| \times |\mathcal{M}_u| + d \times |L| \times |\mathcal{M}_l|)$. Wherein $|R|$ is the number of non-zero observations in the rating matrix R ; $|U|$ is the number of users; $|L|$ is the number of POIs; d is the dimension size of the latent factors; $|\mathcal{M}_u|$ is the number meta-paths related to users; $|\mathcal{M}_l|$ is the number meta-paths related to POIs; and $|\bar{\mathcal{N}}|$ is the average number of neighbors for each meta-path. For the gradients $\frac{\partial \mathcal{L}}{\partial \Omega_{i,p}}$, $\frac{\partial \mathcal{L}}{\partial \Theta_{j,p}}$, $\frac{\partial \mathcal{L}}{\partial \mathcal{U}_i}$ and $\frac{\partial \mathcal{L}}{\partial \mathcal{V}_j}$, the computational time are $\mathcal{O}(d \times |U| \times |\mathcal{M}_u| \times |\bar{\mathcal{N}}|)$,

Algorithm 5 The Learning Algorithm of AGS-MF

Input: AGS-IG, $\mathcal{M}_u$, $\mathcal{M}_l$, learning rate γ
Output: $\mathcal{U}$, $\mathcal{V}$, Ω , Θ

```

1: Randomly initialize  $\mathcal{U}^{(0)}$ ,  $\mathcal{V}^{(0)}$ ,  $\Omega^{(0)}$ ,  $\Theta^{(0)}$  with uniform distribution on interval  $[0, 1]$ 
   and  $t \leftarrow 1$ 
2: while  $t \leq Iter$  do
3:   for  $u_i \in U$ ,  $l_j \in L$  do
4:     for  $m \in \mathcal{M}_u$  do
5:        $\Omega_{i,p}^{(t)} \leftarrow \Omega_{i,p}^{(t-1)} - \gamma \frac{\partial \mathcal{L}}{\partial \Omega_{i,p}}$ 
6:     for  $m \in \mathcal{M}_l$  do
7:        $\Theta_{j,p}^{(t)} \leftarrow \Theta_{j,p}^{(t-1)} - \gamma \frac{\partial \mathcal{L}}{\partial \Theta_{j,p}}$ 
8:   for  $u_i \in U$ ,  $l_j \in L$  do
9:      $\mathcal{U}_i^{(t)} \leftarrow \mathcal{U}_i^{(t-1)} - \gamma \frac{\partial \mathcal{L}}{\partial \mathcal{U}_i}$ 
10:     $\mathcal{V}_j^{(t)} \leftarrow \mathcal{V}_j^{(t-1)} - \gamma \frac{\partial \mathcal{L}}{\partial \mathcal{V}_j}$ 
11:   if  $\mathcal{L}$  is converged then
12:     break
13:    $t \leftarrow t + 1$ 
14: return  $\mathcal{U}^{(t)}$ ,  $\mathcal{V}^{(t)}$ ,  $\Omega^{(t)}$ ,  $\Theta^{(t)}$ 
    
```

$\mathcal{O}(d \times |L| \times |\mathcal{M}_l| \times |\overline{\mathcal{N}}|)$, $\mathcal{O}(d \times |R| + d \times |U| \times |\mathcal{M}_u| \times |\overline{\mathcal{N}}| + d \times |U| \times |\mathcal{M}_u| \times |\overline{\mathcal{N}}|^2)$,
 and $\mathcal{O}(d \times |R| + d \times |L| \times |\mathcal{M}_l| \times |\overline{\mathcal{N}}| + d \times |L| \times |\mathcal{M}_l| \times |\overline{\mathcal{N}}|^2)$, respectively. Since
 $|\mathcal{M}_l| < |\mathcal{M}_u| < |\overline{\mathcal{N}}| < d \ll |U|, |L|$ and $|R|$, the overall computational complexity of
 Algorithm 2 is $Iter \times \mathcal{O}(|R| + |U| + |L|)$, where $Iter$ is the iteration number. So the training
 time for AGS-MF scales linearly with the number of observations including ratings, users
 and POIs. That is, it generally scales with the dataset size. Hence, AGS-MF is scalable
 to large datasets.

5.3 Experiments

5.3.1 Experiment Setup

Datasets. The dataset used in this study is the same as Study 2. The statistics of the
 datasets of three cities are presented in Table 4.2.

Evaluation Metrics. Several widely used metrics were adopted (Zhang & Chow, 2015; Yuan et al., 2014; Cheng et al., 2012): Precision, Recall and Mean Average Precision (denoted as $Pre@N$, $Rec@N$ and $MAP@N$) where N is the size of the Top- N recommended POI ranking list.

Comparison Methods. The performance of the proposed algorithm was compared with both baselines (UCF, ICF and PMF) and state-of-the-arts (LFBCA, Trirank, GeoSoCa and AGS-RW (Guo et al., 2017)) which have been described in Chapter 4. Since AGS-MF is based on MF, it is thus necessary to compare it with other MF-based advances including: 1) SRMF (Ma et al., 2011), a method that integrates social factor into basic MF; and 2) GeoMF (Jamali & Ester, 2010), a state-of-the-art POI recommendation method which incorporates geographical factor into MF.

Parameter Settings. The parameters of all comparison methods were tuned to achieve the best results as described in Chapter 4. For AGS-MF, the default values of the learning rate γ was set as 0.002 and the walk number wn was 100. The iteration numbers were set as a relatively large number (200) to tune other parameters. In the convergence analysis, the proper number of iterations of each city was found empirically. Previous studies indicate that the regularization coefficients are normally very small (Yang et al., 2013; Cheng et al., 2012), we thus fixed λ_u , λ_l , λ_Ω , and λ_Θ as 0.005 for Charlotte, λ_u , λ_l , λ_Ω , and λ_Θ as 0.001 for Phoenix and Las Vegas after a few trials. We followed the below procedure to tune the various hyper-parameters: 1) For α_u and α_l , we first applied the first grid search in $\{0, 0.1, 0.2, \dots, 1.0\}$ where α_u and α_l were 0.1 and 0.8, respectively. Then we conducted the second round of grid search in a small range around 0.1 and 0.8, that is, $\{0.05, 0.06, \dots, 0.1, \dots, 0.15\}$ and $\{0.75, 0.76, \dots, 0.8, \dots, 0.85\}$, to further find the optimal values of α_u and α_l . Specifically, for Charlotte, $\alpha_u = 0.08$ and $\alpha_l = 0.8$, for Phoenix and Las Vegas, $\alpha_u = 0.02$ and $\alpha_l = 0.8$; 2) we conducted a grid search for the walk length wl . To generate meta-paths with various lengths, we repeated a meta-path

Figure 5.2: The effect of meta-paths on three datasets when meta-paths are gradually incorporated into AGSMF. Here, the arrows are omitted to better show the figure.

for z times. Hence, a grid search was conducted for different z in $\{1, 2, 3, 4, 5\}$; and 3) we applied a grid search for the dimension size (d) in $\{10, 20, \dots, 80\}$, and the optimal $d = 40, 30, 30$ for Charlotte, Phoenix, and Las Vegas, respectively.

5.3.2 Results and Analysis

Results of the Variants. Figure 5.2 presents the experimental results of different AGS-MF variants on the three datasets regarding the performance of all variants in terms of precision and recall with $N = 5, 10, 20$. The selected meta-paths into AGS-MF were cumulatively incorporated and the performance changes were recorded accordingly.

Generally, it can be observed that the performance became better with more meta-paths incorporated. This confirms the motivation that more relevant neighbors could be discovered by diverse semantic relations. Overall, the meta-paths starting with L delivered more significant enhancements than the ones starting with U . In particular, with the incorporation of meta-path $L \rightarrow L$ which encodes geographical factor, the performance was enhanced significantly by 42.70%, 51.76%, 22.33% and 46.8% averagely in terms of precision, recall, MAP and F1 score across different settings of N over three datasets. The great improvements by $L \rightarrow L$ reinforce the effectiveness of geographical factor on user check-in behavior.

Nevertheless, the performance was not always improved with more meta-paths integrated. Particularly, in Charlotte and Phoenix datasets, the performance fluctuated slightly with the incorporation of aspect-related meta-paths including $U \rightarrow A \rightarrow U$, $U \rightarrow L \rightarrow A \rightarrow L \rightarrow U$ and $L \rightarrow A \rightarrow L$. For example, Pre@10 and Rec@10 decreased a bit by adding $U \rightarrow L \rightarrow A \rightarrow L \rightarrow U$. Table 5.2 depicts the average number of user-aspect and POI-aspect relations per user and POI, respectively. It could be observed that the average amount of the aspect associated relations in Las Vegas was much larger than that in another two cities. In other words, users in Las Vegas were more willing to give valuable opinions, which helps explain why the incorporation of aspect-related meta-paths led to more remarkable and stable improvements in Las Vegas. The slight performance fluctuation confirms previous finding (Sun et al., 2011) that few high-quality meta-paths can contribute to great performance improvements. In other words, certain meta-paths might be ineffective for some users, even caused some noises. This issue was eased by exploiting personalized weights for different meta-paths, that is, AGS-MF could determine the saliency of each meta-path for each user and POI, implying it could well deal with noisy information. Therefore, AGS-MF still consistently generated decent results with all meta-paths incorporated.

Table 5.2: Average number of user-aspect and POI-aspect (per user and POI respectively)

City	Charlotte	Phoenix	Las Vegas
user-aspect	128	132	163
POI-aspect	96	108	131

Analysis of Hyper-parameters. This section presents the experimental results about the impacts of various settings of hyper-parameters.

The Effect of α_u and α_l . According to Eq. 5.2, α_u and α_l are two regularization coefficients that control the strength of the user and POI regularization terms, respectively. Figure 5.3 plots the recommendation performance with different settings of α_u and α_l . The results show that α_u is generally larger than α_l , which indicate that the POI regularization terms carry more importance than the user ones. Such conclusion also reinforces the above finding (see **Results of the Variants**) and Chapter 4 (see the SRec description in **Results of the Variants**), that is, the meta-paths related to POIs contribute more to the improvements than meta-paths associated with users. Previous studies have also implied that user-based relationships are not as effective as POI-based relationships. For instance, friendships have limited influence on user check-in decision due to the sharing behaviors among friends are constrained by the geographical distance between friends in location-based social networks (Cheng et al., 2012; Zhang & Chow, 2015; Ye et al., 2010).

The Effect of Meta-path Length. Previous research (Sun et al., 2011) indicates that meta-paths with long length might introduce noises in capturing the semantic relations between entities. Therefore, it is reasonable to infer that the random walker might walk to a distant entity with longer meta-paths, resulting in a set of irrelevant neighbors. Experiments were thus necessarily conducted to investigate the impact of meta-path length. To obtain meta-paths with different lengths, meta-paths were concatenated, for example, $U \rightarrow A \rightarrow U$ with different repeating times z , for example, $U \rightarrow A \rightarrow$

 Figure 5.3: Analysis of α_u and α_l .

Figure 5.4: The effect of meta-path length for AGS-MF.

$U \rightarrow A \dots A \rightarrow U = (U \rightarrow A \rightarrow U)^z$. Given $z = 2$, the extended meta-path would be $U \rightarrow A \rightarrow U \rightarrow A \rightarrow U$. Hence, the objective was to study the effect of parameter z on recommendation performance. Experiments were conducted with $z \in [1, 5]$. Figure 5.4 records the performance change with different meta-path lengths in terms of precision, recall, MAP and F1 score with $N = 5, 10, 20$. It could be observed that as z increased, the performance did not change heavily and was still better than all the other comparison methods. This suggests that AGS-MF could discriminate irrelevant neighbors even if the random walker reached distant neighbors, that is, it is not sensitive to the length of meta-paths, demonstrating the robustness of AGS-MF.

The Effect of Dimensionality. Figure 5.5 shows the recommendation performance for

Figure 5.5: The effect of dimensionality of AGS-MF.

various d (ranging from 10 to 80, scaled by 10) with other optimal hyper-parameters fixed. And the performance became stable when the dimension increased. It could be observed that the performance remained stable in a large range of dimension size on different datasets, where the optimal dimension sizes were set as 30, 40 and 40 for Charlotte, Phoenix and Las Vegas, respectively. Hence, a conclusion could be drawn that AGS-MF is not sensitive to the dimensionality. In summary, AGS-MF can deliver decent recommendation results even without best setting of dimension size, indicating it is a robust approach with strong ability in dealing with noises.

The Effect of the Number of Iterations. To understand the convergence of AGS-MF, the performance change was recorded with respect to the number of iterations. As

Figure 5.6: Analysis of Convergence

shown in Figure 5.6, it can be observed that AGS-MF could converge when *Iter* reached a certain point. Specifically, when the iteration number was increased to 700, 800 and 1100, the performance of Charlotte, Phoenix and Las Vegas became stable, respectively. In summary, with relatively fast convergence speed, AGS-MF is a robust approach that can be applied in practical applications.

Analysis of Efficiency. In this section, the runtime performance of AGS-MF is presented. Specifically, AGS-MF was compared with other matrix factorization based approaches including PMF, SRMF and GeoMF, because they all aim to learn the latent

Table 5.3: Runtime Performance Comparison

	Methods	PMF	SRMF	GeoMF	AGS-MF
CH	<i>Iter</i>	400	400	600	700
	Time per iteration	0.1s	0.4s	0.6s	0.9s
PH	<i>Iter</i>	600	500	700	800
	Time per iteration	0.2s	0.8s	1.1s	1.6s
LV	<i>Iter</i>	600	600	700	1000
	Time per iteration	0.8s	1.9s	2.4s	3.5s

factors of users and POIs. The running environment was a PC with Core4 2.6GHz with Windows 10 and 8GB DDR4 memory. The results are shown in Table 5.3. Unsurprisingly, PMF took the fewest iterations to converge and the shortest running time per iteration as it only uses user-venue matrix without considering other types of information. For example, it took only 480s to converge for Las Vegas. When considering social factor or geographical factor, the runtime of SRMF and GeoMF were longer than PMF. The proposed approach AGS-MF took longer time to converge as it considers both social, geographical and content factors. Particularly, SRMF and GeoMF only consider the user-user and POI-POI relations respectively while AGS-MF incorporates multiple kind of relations among users and POIs based on various meta-paths. Besides, AGS-MF also aims to learn the personalized weights of various factors for each user and POI. Therefore, comparing to other methods, the AGS-MF approach promotes the performance with more learning time at an acceptable level.

Comparative Results. Table 5.4 describes the performance of all comparison methods on the three real-world datasets with respect to precision, recall, MAP and F1 score with $N = 5, 10, 20$. The analysis of comparison methods can be found in Chapter 4. Compared with state-of-the-art methods, the proposed approach performed better. This implies that recommendation performance could be further enhanced by appropriately considering the three influences. AGS-MF incorporated various influential factors in a

Table 5.4: Precision and recall of all methods on the three real-world datasets. For each metric, the best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that the proposed method AGS-MF achieved relative to the best performance of other comparison methods.

City	Metric	UCF	ICF	PMF	SRMF	LFBCA	GeoMF	GeoSoCa	TR	TriRank	AGS-RW	AGS-MF	Improve(%)
Charlotte	Pre@5	1.483	0.976	1.817	2.586	2.007	2.901	2.586	2.602	2.821	3.018*	3.776	25.12
	Pre@10	1.456	0.922	1.798	2.134	1.881	2.562	2.134	2.128	2.532	2.633*	3.186	21.00
	Pre@20	1.433	0.945	1.581	1.917	1.939	2.372	1.917	2.054	2.360	2.474	2.411*	-2.54
	Rec@5	1.117	0.977	1.485	1.875	1.964	2.509	1.875	1.898	2.548	2.625*	3.103	18.21
	Rec@10	2.229	1.940	2.805	3.070	3.669	4.557	3.070	3.145	4.407	4.760*	5.250	9.87
	Rec@20	4.137	4.143	4.849	5.833	7.562	8.194	5.833	5.987	8.034	8.558	8.092*	-9.02
	MAP@5	3.086	2.039	3.659	5.328	4.188	5.702	5.328	5.409	5.586	6.033*	7.323	21.28
	MAP@10	3.788	2.566	4.369	5.922	4.939	6.807	5.922	5.901	6.747	6.949*	8.345	20.09
Phoenix	MAP@20	4.255	3.061	4.751	6.281	5.589	7.528	6.281	6.162	7.203	7.946*	8.441	6.25
	Pre@5	0.894	0.708	1.248	1.322	1.467	1.905	1.623	1.650	1.890	2.060*	2.196	6.60
	Pre@10	0.987	0.796	1.173	1.080	1.174	1.889	1.588	1.602	1.830	1.939*	2.154	11.09
	Pre@20	1.034	0.784	1.101	1.022	1.014	1.801	1.500	1.527	1.734	1.869	1.852*	-0.91
	Rec@5	0.693	0.771	1.028	1.022	1.536	1.723	1.600	1.652	1.751	1.821*	1.932	6.10
	Rec@10	1.604	1.742	1.880	1.628	3.592*	3.520	2.901	2.914	3.300	3.531	3.845	7.04
	Rec@20	3.370	3.489	3.603	3.092	6.130	5.850	5.700	5.713	6.126	6.502	6.163*	-5.21
	MAP@5	2.147	1.562	2.920	3.768	3.919	3.842	3.498	3.469	3.771	3.921*	4.464	13.85
Las Vegas	MAP@10	2.685	2.094	3.479	4.098	4.760	4.859	4.013	4.109	4.690	4.972*	5.416	8.93
	MAP@20	3.136	2.457	3.895	4.477	5.217	5.742	5.056	5.182	5.304	5.823*	5.864	0.70
	Pre@5	0.626	0.588	0.948	0.960	1.120	1.509	1.470	1.452	1.589	1.653*	2.069	25.17
	Pre@10	0.542	0.501	0.878	0.859	1.060	1.291	1.280	1.258	1.331	1.371*	1.821	32.82
	Pre@20	0.485	0.433	0.754	0.825	0.975	1.406	1.370	1.362	1.416	1.455*	1.495	2.75
	Rec@5	0.552	0.508	0.865	0.627	0.672	1.461	1.420	1.428	1.431	1.474*	1.806	22.52
	Rec@10	0.939	0.882	1.478	1.182	1.406	2.565	2.260	2.260	2.459	2.557*	3.132	22.49
	Rec@20	1.647	1.610	2.436	2.413	2.823	5.345	5.180	5.163	5.345	5.502	5.398*	-1.89
Las Vegas	MAP@5	1.340	1.310	2.191	2.222	2.222	3.428	3.250	3.246	3.306	3.571*	4.644	30.05
	MAP@10	1.612	1.586	2.530	2.074	2.685	3.860	3.820	3.809	3.882	4.154*	5.385	29.63
	MAP@20	1.819	1.613	2.251	2.393	3.063	4.282	4.100	4.116	4.349	4.883*	5.580	14.27

Table 5.5: F-measure (F1 score) of all methods on all datasets. The best performance is highlighted in bold; the second best performance is labeled with ‘*’; ‘Improve’ indicates the relative improvements that AGS-MF achieves relative to the best performance of other methods.

City	Metric	UCF	ICF	PMF	SRMF	LFBCA	GeoMF	GeoSoCa	TR	TriRank	AGS-RW	AGS-MF	Improve(%)
Charlotte	F1@5	1.274	0.976	1.634	2.174	1.985	2.691	2.174	2.195	2.678	2.808*	3.407	21.32
	F1@10	1.761	1.250	2.191	2.518	2.487	3.280	2.518	2.538	3.216	3.391*	3.966	16.96
	F1@20	2.129	1.539	2.385	2.886	3.087	3.679	2.886	3.059	3.648	3.838	3.715*	-3.21
Phoenix	F1@5	0.781	0.738	1.127	1.153	1.501	1.809	1.611	1.651	1.818	1.933*	2.056	6.33
	F1@10	1.222	1.093	1.445	1.299	1.770	2.459	2.052	2.067	2.354	2.503*	2.761	10.30
	F1@20	1.582	1.280	1.687	1.536	1.740	2.754	2.375	2.410	2.703	2.903	2.848*	-1.90
Las Vegas	F1@5	0.587	0.545	0.905	0.759	0.840	1.485	1.445	1.440	1.506	1.558*	1.929	23.76
	F1@10	0.687	0.639	1.102	0.995	1.209	1.718	1.634	1.616	1.727	1.785*	2.303	29.02
	F1@20	0.749	0.682	1.152	1.230	1.449	2.226	2.167	2.155	2.239*	2.301	2.342	1.74

non-trivial manner, that is, leveraged AGS-IG to unify them seamlessly, and learns the personalized weights of each user and POI. In particular, AGS-MF promoted the recommendation performance by a large margin, that is, precision, recall, MAP and F1 score were improved by 18.28%, 12.82%, 21.71% and 15.7% on average across different settings of N on the three datasets (with p -value < 0.01), compared with the best of other comparison approaches. Moreover, AGS-MF consistently outperformed other state-of-the-art methods with any setting of d . In particular, by learning personalized weights for different meta-paths, AGS-MF outperformed AGS-RW in most cases by significant percentages – the precision, recall, MAP and F1 score were averagely boosted by 13.36%, 8.94%, 16.89% and 10.86% across different settings of N on the three datasets, respectively (with p -value < 0.01). Especially, AGS-MF consistently outperformed AGS-RW in terms of MAP, indicating AGS-MF provided recommendations with better ranking quality, which was critical to improve user experience. In summary, the enhancements of the proposed approach reinforced the effectiveness of exploiting various influential factors via an integrated way.

5.4 Conclusion

This study focused on exploiting the heterogeneous information in LBSNs to model both diversity and imbalance properties of user check-in behavior in a unified way for more effective POI recommendation. A novel graph-based POI recommendation approach (AGS-MF) was proposed. It not only incorporated various influential factors, but also learned the personalized weights of each user and POI, resulting in a unified modeling of both the diversity and imbalance properties. Empirical study on multiple real-world datasets demonstrates that AGS-MF significantly outperformed state-of-the-art algorithms.

Chapter 6

An Attentional Recurrent Neural Network for Personalized Next POI Recommendation

The fourth study presented in this chapter focused on the next POI recommendation task, which is to predict the next POI a user will visit given his/her historical check-in data. Compared with general POI recommendation, the sequential effects of a user’s previous check-ins plays a crucial role in determining his/her next movement (Zhang et al., 2014b, 2017b). However, most existing studies on next POI recommendation propose to model the *sequential regularity* of check-in sequences, but suffer from the severe data sparsity issue where most POIs have fewer than five *following POIs*.

To solve the sparsity problem, we consider leveraging the *transition regularities* of similar locations (neighbors) by assuming they share similar transitional patterns. Fig. 6.1 presents a running example to illustrate our intuition. l_1 is an Italian restaurant followed by only a few locations, so it is challenging to predict the next location a user might visit after l_1 by only capturing the sequential regularity. In fact, the transitional information of l_1 ’s neighbors can help ease such sparsity issue. For instance, if users often watch a film at a cinema l_3 after dinner at l_2 (l_1 ’s neighbor which is a nearby Sushi restaurant), then l_3 can also be recommended as the next location for l_1 . In other words,

Figure 6.1: A running example

the transition regularities of neighbors serve as complementary information for locations with insufficient following check-in records. Due to the heterogeneity of LBSNs, multiple factors may influence the relationships among locations. Thus two locations are similar if they are: 1) geographically-close; 2) described by the same semantics; and 3) preferred by the same users. Thus, this idea can be considered as a neighborhood-based strategy – aggregating useful features of neighbors to ease the data insufficiency of the target user. To our best knowledge, we are the first to solve such data sparsity issue and coherently model the sequential regularity with transition regularities of neighbors for next location recommendation.

Existing studies either model the lengthy sequence or incorporate different contexts (Liu et al., 2016; Yao et al., 2017; Zhang et al., 2017b), and still fail to capture effective transitional patterns between POIs because of the sparsity issue. Therefore, this study aims to augment the recurrent neural network (RNN) by jointly modeling both the sequential regularity and transition regularities of various neighbors to resolve the sparsity for better next POI recommendation. To this end, an *Attentional Recurrent Neural Network* framework (ARNN) was proposed, which could seamlessly integrate the

RNN (Hochreiter & Schmidhuber, 1997) and attention mechanism (Bahdanau et al., 2014) to model the two kind of regularities in a unified way. Specifically, a multi-modal embedding layer is designed to transform the sparse features (that is, user, POI, time and semantics) that govern user movement into dense representations, which are further fed into the recurrent layer. To extract various types of neighbors, a novel heterogeneous information network (HIN) is first constructed to represent the heterogeneous information including users, POIs and semantics (categories and tags of POIs) into a unified space. A meta-path based random walk process is then designed to efficiently discover the neighbors based on geographical, semantic and user preference factors. After that, an attention layer is designed to capture the transition regularities of neighbors at each time step. The attention layer is capable of selecting highly salient neighbors which are positively correlated to the current POI. Finally, the attention layer is tailored to effectively cooperate with the recurrent layer as a unified recurrent framework.

6.1 Problem Analysis and Formulation

This section first analyzes the problem using real-world datasets, and then presents the problem formulation.

6.1.1 Problem Analysis

Several real-world datasets were adopted for experiments, which are from two famous LBSNs, Foursquare (Yang et al., 2015) and Gowalla¹. Previous studies show that most successive check-ins happen within $10km$ (Cheng et al., 2013; Feng et al., 2015). Hence, the check-in records of three cities were selected for the data analysis and experiments: New York (NY) and Tokyo (TK) from Foursquare and San Fransisco (SF) from Gowalla.

¹<https://snap.stanford.edu/data/loc-gowalla.html>

Since semantic information about POIs was not available in Gowalla, Foursquare venue APIs² were applied to collect categories and tags of POIs in SF.

Sparsity Issue. To assess the sparsity, for each POI l_k , the number of its following POIs is calculated as $L_f(l_k)$. A following POI of l_k is defined as the POI visited after l_k successively. Figure 6.2 shows that most POIs have a very limited number of following POIs. In particular, over 50% POIs have only one following POI in NY and TK while more than 80% have fewer than 5 following POIs in all three cities. Though the contextual information facilitates a deeper understanding of each check-in decision, the transitional patterns of them can not be well captured with such insufficient following POIs. Thus, various neighbors are leveraged to solve this sparsity issue.

Figure 6.2: The distribution of POIs with respect to # following POIs

Transition Regularities of Neighbors. To quantitatively validate the assumption on transition regularities of neighbors, the *common ratio* about the following POIs for a given POI pair (l_k and l_j) is calculated. Let $L_f(l_k)$ and $L_f(l_j)$ denote the sets of the following POIs of l_k and l_j , respectively. The common ratio measures the overlap degree of $L_f(l_k)$ and $L_f(l_j)$ defined as: $\alpha(l_k, l_j) = \frac{|L_f(l_k) \cap L_f(l_j)|}{|L_f(l_k) \cup L_f(l_j)|}$. The mean common ratios of

²<https://developer.foursquare.com/places-api>

Figure 6.3: The mean common ratio of neighbors vs. non-neighbors based on the three factors, including geographical, semantic and user preference factors.

the three types of neighbors with respect to geographical (Geo), semantic (Sem) and user preference (User) factors compared with non-neighbors. The non-neighbors are randomly sampled from POIs that are not similar to the given POI by any factor. Figure 6.3 shows that the mean common ratios of different types of neighbors are consistently much higher than non-neighbors on all the three cities. This verifies the assumption that similar POIs tend to share more similar transition patterns. Therefore, aggregating the transition regularities of neighbors can help ease the sparsity issue for high-quality recommendation.

6.1.2 Problem Formulation

To formulate the problem, let $U = \{u_1, u_2, \dots, u_{|U|}\}$, $L = \{l_1, l_2, \dots, l_{|L|}\}$ and $V = \{w_1, w_2, \dots, w_{|V|}\}$ denote a set of users, POIs and words³, respectively.

Definition 1 (Historical Sequence) *The historical sequence of user u_i is a temporally ordered sequential check-in records, i.e., $His(u_i) = \{r_1, r_2, \dots, r_H\}$. And each check-in $r_k \in His(u_i)$ is a tuple $(u_i, l_k, t_k, S(l_k))$ where l_k is the location and t_k is the timestamp, $S(l_k)$ is the semantics of l_k that include a bag of words (i.e., categories and tags).*

³A set of words of a POI represents the same meaning as the semantics mentioned above, which is essentially a bag of words containing the categories or tags of a POI.

Figure 6.4: The architecture of ARNN

Definition 2 (Trajectory) A trajectory of a user u_i is a subsequence of $His(u_i)$ where the time interval between two successive check-ins is smaller than the pre-defined time threshold, Δt , that is, $Tra(u_i) = \{r_1, r_2, \dots, r_K\}$ is a segment of $His(u_i)$, if $0 < t_{k+1} - t_k \leq \Delta t, \forall 1 \leq k < K$.

The next POI recommendation is formulated as: given a trajectory of user u_i , that is, $Tra(u_i) = \{r_1, r_2, \dots, r_{K-1}\}$, a list of POIs that u_i is interested in visiting at time step t_K would be generated for him/her. As the following POI of l_k is highly sparse, that is, $|L_f(l_k)|$ is always fewer than five, thus the this study leverages the transition regularities of neighbors with the sequential regularity, so as to ease such sparsity issue for better recommendation.

6.2 The Proposed Framework

To overcome the sparsity challenge, both sequential regularity and transition regularities of neighbors simultaneously are considered by devising an Attentional Recurrent Neural

Network framework. It consists of four layers: embedding layer, attention layer, recurrent layer and output layer as shown in Figure 6.4, which are further described as follows.

A multi-modal embedding layer learns the dense representations of POIs and various contexts. To find relevant neighbors, a novel heterogeneous information network proposed fuse the heterogeneous data into a unified space. Then, a meta-path based random walk over the HIN is developed to efficiently discover the neighbors based on multiple factors. To capture the transition regularities of relevant neighbors, an attention layer is incorporated into ARNN to generate a weighted embedding by distinguishing each neighbor of the current POI. By integrating the weighted embedding from attention layer and current status, the recurrent layer can generate the hidden state to encode the information observed until the current time step. Finally, the output layer jointly captures both the information of the trajectory and user general interest to recommend the next visit.

6.2.1 Neighbor Discovery

Before training ARNN, it is necessary to extract reliable neighbors for each POI. Due to the huge computation by enumerating all POIs for neighbor discovery, the heterogeneous information network and meta-path are leveraged to capture the heterogeneous factors that enable the connections among POIs. First, a novel heterogeneous information network is constructed, that is, $G = \{U \cup L \cup V, E_{UL} \cup E_{LL} \cup E_{LV}\}$, where E_{LL} represents POI-POI relations, that is, two POIs are linked if they are geographically-close, that is, $dist(l_k, l_j) \leq \Delta d$; E_{LV} denotes POI-word affiliations; E_{UL} denotes user historical visits. Each input tuple (u_i, l_k, t_k) denotes a check-in, r_k , while each POI is described by a set of words, $S(l_k)$. By aggregating the historical check-ins, the words of POIs, and geographical distance between POIs, G is capable of accommodating the heterogeneous data of historical records into a unified space as shown in Figure 6.5.

Figure 6.5: An illustrative example of the heterogeneous information network

Next, *meta-path* is utilized since it is a critical technique to capture the diverse semantic relations in a heterogeneous information network (Sun et al., 2011), to extract various kinds of neighbors from G . The meta-path selection follows two principles: (i) each meta-path only represents one factor to avoid noises, for example, following $L \rightarrow U \rightarrow L \rightarrow L$, the random walker might reach a POI that the user does not like though the two POIs are close; and (ii) the length of meta-paths is kept within 4 as longer meta-path may introduce noises (Sun et al., 2011). As the goal is to find relevant neighbors for POIs, the meta-paths should start with L and reach L eventually. Based on prior data analysis in section 6.1.1, a POI shares more following POIs with three kinds of neighbors based on geographical, semantic and user preference factors. In other words, the transition regularities of those neighbors can be exploited to predict the next POI after the user visiting a given POI. Therefore, three meta-paths, $L \rightarrow L$, $L \rightarrow V \rightarrow L$ and $L \rightarrow U \rightarrow L$, are selected to represent geographical, semantic and user preference factors, respectively. For each meta-path, a random walk process is conducted to generate paths from which neighbors are extracted.

Given a meta-path $p = T_1 T_2 \cdots T_m \cdots$, where T_m is the type of m -th entity, the transition probability between two linked entities is determined by the neighborhood size with constraint based by p , that is, $\text{Prob}(v_m | v_{m-1}, p) = \frac{1}{|N_{T_m}(v_{m-1})|}$, if $T(v_{m-1}) = T_{m-1}$ and $T(v_m) = T_m$, where $T(v)$ is the type of entity v and $N_{T_m}(v)$ is the first-order neighbor

Algorithm 6 Meta-Path based Random Walk

Input: G , POI l_k , meta-path p , walk length $walk_len$, walk number $walk_num$

Output: A set of neighbors, $N_p(l_k)$

```

1:  $Paths = []$ 
2: for  $i \leftarrow 1$  to  $walk\_num$  do
3:    $path = []$ 
4:    $m \leftarrow 1$ 
5:   while  $m \leq walk\_len$  do walk to entity  $v_m$  by  $\text{Prob}(v_m|v_{m-1}, p)$ 
6:     if  $T(v_m) == T_m$  then
7:       Add  $v_m$  to  $path$ 
8:        $m \leftarrow m + 1$  //move one step forward
9:   Add  $path$  to  $Paths$ 
10: Extract neighbors from  $Paths$  as  $N_p(l_k)$ 
11: return  $N_p(l_k)$ 

```

set of v in type T_m . By following p with the transition probability, the random walker could generate a path until it reaches the walk length. The process terminates if enough paths are created. Finally, the POIs from those paths are extracted to create the neighbor set of a given POI. For example, $l_1 \rightarrow w_2 \rightarrow l_3$ are available in Figure 6.5, so $l_3 \in N_p(l_1)$ where $p = L \rightarrow V \rightarrow L$ and $N_p(l_1)$ is the neighbor set of l_1 with respect to meta-path p . Algorithm 6 describes the details about the random walk process. The extracted neighbors are further incorporated into the attention layer.

6.2.2 Embedding Layer

To embed multiple features that govern the sequential regularity, a multi-modal embedding layer is designed to jointly learn the embeddings of the POI with its temporal context, semantic context, and user general interest.

Specifically, different information of each check-in is initially represented as one-hot vectors. One-hot representation can encode categorical integer features using a one-of-N schema, for instance, a user's preference for POIs in one-hot vector is $[0, 1, 1, \dots, 0]$, where the size of this vector is $|L|$ and the k -th element is 1 meaning the user has visited the

k -th POI in L otherwise it equals to 0. So, the above one-hot representation means that the user has visited l_2 and l_3 before. As the timestamp is a continuous value, then one day is mapped into 24 hours so that t_k could be transformed into 24-dimensional one-hot vector. Each POI is represented by a $|L|$ -dimensional one-hot vector. For the semantics of each POI, each word is transferred into a $|V|$ -dimensional one-hot vector, and sum them up as a new vector. Besides, each user has his/her general preference, represented as a $|U|$ -dimensional one-hot vector. These one-hot vectors are fed into the embedding layer to learn the low-dimensional dense representations of the timestamp, POI, semantics and user general interest, denoted as $\mathbf{e}_t \in \mathbb{R}^{D_t}$, $\mathbf{e}_l \in \mathbb{R}^{D_l}$, $\mathbf{e}_s \in \mathbb{R}^{D_s}$ and $\mathbf{e}_u \in \mathbb{R}^{D_u}$, respectively. Note, unlike other embeddings, user embedding, $\mathbf{e}_u$, is not taken as an input into the recurrent layer, but only used by the output layer at last time step of the trajectory, as it is considered to be stable through time. These dense representations can model the semantic and spatio-temporal features of each check-in more precisely as well as reduce the computation.

6.2.3 Attention Layer

To model the transition regularities of neighbors, it is necessary to design a selector to choose salient neighbors based on current POI automatically at each time step along the sequence. Thus, the attention mechanism is adopted, due to its ability to select relevant parts of the input, which can be considered as a soft-alignment process (Mnih et al., 2014). Specifically, the attention layer is designed to calculate the similarity (that is, attention weights) between current POI and each neighbor. If a neighbor is more similar with current POI with respect to transition patterns, it would be assigned larger attention weight. Besides, the attention layer is parameterized as a feed-forward neural network, which is jointly trained with other layers. Figure 6.4 also illustrates the structure of the

attention layer. The following describes the attention computation at time step t_k :

$$\mathbf{c}_k = \sum_n \alpha_k(n) \mathbf{e}_{l_n}, \quad (\text{Eq. 6.1})$$

$$\alpha_k(n) = \text{softmax}(f_a(\mathbf{e}_{l_k}, \mathbf{e}_{l_n})), \quad (\text{Eq. 6.2})$$

$$f_a(\mathbf{e}_{l_k}, \mathbf{e}_{l_n}) = \tanh(\mathbf{e}_{l_k}^\top \cdot \mathbf{W}_a \cdot \mathbf{e}_{l_n}) \quad (\text{Eq. 6.3})$$

where $\mathbf{e}_{l_k}$ is the embedding of current POI; $\mathbf{e}_{l_n}$ is the embedding of POI l_n , a neighbor of l_k ; $\mathbf{c}_k \in \mathbb{R}^{D_l}$ is computed as the weighted average embedding over all the neighbors; $\mathbf{W}_a$ is the weight matrix; $\alpha_k(n) \in [0, 1]$ is the attention weight of l_n and $\sum_n \alpha_k(n) = 1$; f_a is the score function, which measures the relevance between the neighbor and current POI.

6.2.4 Recurrent Layer

Due to the relatively long-range check-in sequences in LBSNs, *Long Short-Term Memory* (LSTM) is adopted as the recurrent unit, due to its strong capability of memorizing the historical information of long-range sequences (Hochreiter & Schmidhuber, 1997).

At time step t_k , the representations are generated by different components of ARNN, including: (i) $\mathbf{e}_{l_k}$, $\mathbf{e}_{t_k}$, $\mathbf{e}_{s_k}$ from the multi-modal embedding layer, which embed current POI, its temporal and semantic contexts; (ii) $\mathbf{c}_k$ from the attention layer, which represents the weighted embedding based on relevant neighbors; and (iii) $\mathbf{h}_{k-1}$ from previous recurrent unit, which encodes the information of the trajectory until t_{k-1} . The goal of the recurrent unit is to jointly integrate the three groups of representations to update the hidden state, which preserves the information observed until the current time step t_k . First, they are concatenated to generate a new representation, that is, $\mathbf{e}_k = [\mathbf{e}_{l_k}; \mathbf{e}_{t_k}; \mathbf{e}_{s_k}; \mathbf{c}_k]$, where $\mathbf{e}_k \in \mathbb{R}^{D_e}$, where $D_e = 2D_l + D_t + D_s$. Then, the k -th recurrent unit takes $\mathbf{e}_k$ and $\mathbf{h}_{k-1}$ to update the hidden state as follows:

$$\mathbf{h}_k = f(\mathbf{W}_{r_1} \cdot \mathbf{h}_{k-1} + \mathbf{W}_{r_2} \cdot \mathbf{e}_k + \mathbf{b}_r) \quad (\text{Eq. 6.4})$$

where $\mathbf{h}_k \in \mathbb{R}^{D_h}$ is a D_h -dimensional vector which represents the hidden state at t_k ; $\mathbf{W}_{r_1}$ and $\mathbf{W}_{r_2}$ are the weight matrices, and $\mathbf{b}_r$ is the bias term; $f(\cdot)$ denotes the forward function of LSTM⁴.

6.2.5 Output Layer

By performing Eq. 6.4 through time, the hidden state could be generated, $\mathbf{h}_{K-1}$, which coherently inherits the information of the trajectory until time step t_{K-1} . It is further combined with user general preference to predict where user u_i would visit at t_K . To this purpose, $\mathbf{h}_{K-1}$ is first decoded into a D_u -dimensional vector, that is, $\mathbf{o}_{K-1} \in \mathbb{R}^{D_u}$. Then, the user embedding, $\mathbf{e}_{u_i}$ is concatenated with $\mathbf{o}_{K-1}$ to calculate the distribution over $|L|$ POIs as follows:

$$\mathbf{o}_{K-1} = \mathbf{W}_{o_1} \cdot \mathbf{h}_{K-1} + \mathbf{b}_{o_1} \quad (\text{Eq. 6.5})$$

$$\mathbf{o}'_{K-1} = \mathbf{W}_{o_2} \begin{bmatrix} \mathbf{o}_{K-1} \\ \mathbf{e}_u \end{bmatrix} + \mathbf{b}_{o_2} \quad (\text{Eq. 6.6})$$

$$\mathbf{y}_K^{u_i} = \text{softmax}(\mathbf{o}'_{K-1}) \quad (\text{Eq. 6.7})$$

where $\mathbf{W}_{o_1} \in \mathbb{R}^{D_u \times D_h}$, $\mathbf{W}_{o_2} \in \mathbb{R}^{|L| \times 2D_u}$ are transformation matrices and $\mathbf{b}_{o_1}, \mathbf{b}_{o_2}$ are the bias terms; $\mathbf{y}_K^{u_i}$ represents the probability distribution over L via softmax function at time step t_K . Note that user embedding, $\mathbf{e}_{u_i}$, is only utilized in the output layer as it represents the general interest of u_i , which should not change with time.

6.2.6 Model Optimization

Given the training samples, the parameters are optimized by minimizing the following loss function:

$$J = \sum_{i=1}^{|U|} \sum_{k=1}^{|L|} \hat{\mathbf{y}}_K^{u_i}(l_k) \cdot \log \mathbf{y}_K^{u_i}(l_k) + \lambda \|\Theta\|_2 \quad (\text{Eq. 6.8})$$

⁴The activation functions of input, forget and output gates are all *sigmoid* function while it is the *tanh* function for cell gate.

where J is the Cross Entropy Loss between the predictions and ground truth; $\hat{\mathbf{y}}_K^{u_i}$ is a one-hot vector to describe the ground truth POI at t_K , that is, $\hat{\mathbf{y}}_K^{u_i}(l_k) = 1$ if u_i visited l_k at t_K ; $\|\Theta\|_2$ is the regularization term to avoid over-fitting; λ controls the strength of the regularization. Stochastic Gradient Descent (Bottou, 1991) and Back Propagation Through Time (Rumelhart et al., 1986) are utilized to learn the parameters with a batch size of 64.

Complexity Analysis. There are mainly two components in the ARNN, which are the neighbor discovery and the deep learning module (LSTM with attention mechanism) (see Figure 6.4). For a single user, the computation time of finding neighbors is $\mathcal{O}(1)$. So the computation time of neighbor discovery for all users is $\mathcal{O}(|U|)$. In the deep learning module, there are embedding layer, recurrent layer, attention layer, output layer and learning process. In one iteration, given an input sentence of size n , for the embedding layer, the computation time is $\mathcal{O}(n \times (D_l \times |L| + D_s \times |V| + D_h \times |H|) + D_u \times |U|)$ where H is the hours of a day and $|H|=24$. For the recurrent layer, the computation time is $\mathcal{O}(n \times (D_e \times D_h + D_h^2))$ where $D_e = 2D_l + D_t + D_s$. Note that $D_l = D_u = D_s = D_t$ and let D denotes the dimension size. For the attention layer, the computation time is $\mathcal{O}(n \times D_l^2)$. For the output layer, the computation time is $\mathcal{O}(D_u \times D_h + 2 \times |L| \times |D_u|)$. As $n, |H| \ll |U|, |V|, |L|, |Traj|$, the time complexity is $\mathcal{O}(D \times (|U| + |L| + |V| + D_h) + D_h^2)$. For the learning process, the computation time of calculating J is $\mathcal{O}(|U|)$; the computation time of SGD and BPTT at one time step is $\mathcal{O}(D_h^2)$ (Gruslys et al., 2016). So the time complexity of the learning process is $\mathcal{O}(D_h^2 + |U|)$. In summary, the time complexity of ARNN is $Iter \times |Traj| \times \mathcal{O}(D \times (|U| + |L| + |V| + D_h) + D_h^2 + |U|)$, where $Traj = \cup_{u_i \in U} Tra(u_i)$, denoting the whole set of trajectories in training set and $Iter$ is the number of iterations. The training time for ARNN scales with the number of trajectories and the total number of users, POIs and semantic words, the embedding size and the number of hidden units. That is, it generally scales with the dataset size and the dimension sizes.

6.3 Experiments

Extensive experiments have been carried out to investigate the following questions: (i) How does different meta-paths affect our model performance? (ii) How do the time threshold and the embedding dimensionality affect our model accuracy? (iii) What is the convergence property of our model? (iv) How does our approach compare with state-of-the-art methods?

Table 6.1: Statistics of datasets ($\Delta t = 12h$)

City	Users	POIs	Check-ins	Trajectories	Words
NY	490	19253	77853	9082	242
TK	1499	33530	247794	31180	235
SF	170	7340	32058	2953	306

6.3.1 Experimental Setup

Data Preprocessing. Following previous work (Yao et al., 2017), users with fewer than 5 trajectories and trajectories with fewer than 3 check-ins were removed. Previous works (Yao et al., 2017; Zhang et al., 2017b; Feng et al., 2018) also usually filter out inactive POIs with few visits. Removing those POIs from trajectories could ease the sparsity but induces the model to learn wrong transition information. Thus, no filtering was conducted to POIs here. The time threshold $\Delta t = 12h$ was adopted to create high-quality trajectories, because the experiments showed that it was the optimal setting as shown in Table 6.2. The earlier 80% check-ins of each user were selected as training set while the remaining for testing set. The statistics of datasets (including the numbers of users, POIs, check-in records, trajectories, and semantic words) is summarized in Table 6.1. The implementation of ARNN has been released⁵.

⁵<https://github.com/jeffreyguo24/ARNN>

Comparison Methods. To evaluate the performance of ARNN, it was compared with state-of-the-art methods:

- **UCF** (Sarwar et al., 2001): is a user-based collaborative filtering by using the user-POI matrix. It was implemented with Librec (Guo et al., 2015a);
- **FPMC** (Rendle et al., 2010): extends Markov Chain for sequential prediction. FPMC⁶ has been implemented by Rendle et al. (2010);
- **FPMC-LR** (Cheng et al., 2013): extends FPMC by considering geographical constraints. FPMC-LR was also implemented based on FPMC framework;
- **LSTM** (Hochreiter & Schmidhuber, 1997): is a popular variant model of RNN for sequential prediction. It is a built-in function in Pytorch library⁷;
- **ST-RNN** (Liu et al., 2016): is a recent RNN-based model that incorporates temporal and geographical information. Its implementation⁸ has been provided by Liu et al. (2016);
- **SERM** (Yao et al., 2017): is a state-of-the-art method combining multiple contexts in a recurrent model. Its implementation⁹ has been provided by Yao et al. (2017);
- **DeepMove** (Feng et al., 2018): is a state-of-the-art recurrent model capturing multi-level influence of previous check-ins. Its implementation¹⁰ has been given by Feng et al. (2018);

⁶<https://github.com/flaviovd/fpmc>

⁷https://pytorch.org/docs/stable/_modules/torch/nn/modules/rnn.html

⁸<https://github.com/yongqyu/STRNN>

⁹<https://github.com/yaodi833/serm>

¹⁰<https://github.com/vonfeng/DeepMove>

- **Caser** (Tang & Wang, 2018): is a state-of-the-art method that models the joint effects of previous visits based on CNN. Its implementation¹¹ has been provided by Tang & Wang (2018).

Evaluation Metrics. Following the existing works (Zhang et al., 2017b; Yao et al., 2017; Feng et al., 2018), prediction accuracy ($Acc@N$, $N = 1, 5, 10, 20$) was employed to check whether the ground-truth POI appears in the top- N recommendation list.

Parameter Settings. The parameters were tuned to achieve the best results, or set parameters as suggested by the original papers for all comparison methods. For ARNN, $\Delta d = 3km$, $walk_num = 50$ and $walk_len = 10$ for the meta-path based random walk. The epoch number was set as 20 and the learning rate was 0.01. The regularization parameter λ was chosen as 0.01. The number of hidden units, D_h , was set as 64 for all cities. The number of recurrent layers was 1. The hidden state and cell state were initialized as zero. Except the default values of above parameters, experiments have been conducted to find the optimal values of the embedding size and the number of iterations. First, a grid search in $\{10, 20, 40, \dots, 200\}$ was applied to find the optimal settings for the embedding size (that is, $D_u = D_l = D_t = D_s$). It was set as 120/160/120 for NY, TK and SF, respectively. We further applied a grid search to find the optimal number of iterations in $\{5, 10, 15, \dots, 40\}$.

6.3.2 Analysis of Meta-paths

To investigate the effect of various meta-paths, the selected meta-paths were cumulatively incorporated into neighbor discovery and record the performance change accordingly. The results with respect to $Acc@N$ ($N = 1, 5, 10, 20$) are presented in Figure 6.6, where *Geo*, *Sem* and *User* represent $L \rightarrow L$, $L \rightarrow V \rightarrow L$ and $L \rightarrow U \rightarrow L$, respectively. *None*

¹¹https://github.com/graytowne/caser_pytorch

Figure 6.6: Performance of variants of ARNN on the three datasets when various meta-path are cumulatively incorporated.

means no meta-path is used, that is, ARNN is degraded to LSTM. Besides, the results of SF were worse than NY and TK. As described in Figure 6.3 (b-d), the similarity between of both neighbors and non-neighbors in SF was much smaller than others. This implied that the transitional information is rarely shared among POIs in Gowalla (Cheng et al., 2013; Zhang et al., 2014b), which demoted the effectiveness of both sequential regularity and transition regularities of neighbors.

It could be observed that the performance became increasingly better as more meta-paths were incorporated into our approach. According to the results, all the ARNN variants outperformed LSTM, which confirms the effectiveness of utilizing transition regularities of neighbors. Meta-path $L \rightarrow U \rightarrow L$ delivered more significant enhancement than others, which reinforces the fact that the similar properties of two POIs can be better captured by user preference than geographical and semantic factors..

With the incorporation of all meta-paths, the performance was enhanced significantly by 14.69%, 18.05%, 15.68% and 16.32% averagely with respect to Acc@1, Acc@5, Acc@10

and Acc@20 across the three datasets. Nevertheless, the improvements were not always significant with $L \rightarrow V \rightarrow L$ and $L \rightarrow L$. With $L \rightarrow V \rightarrow L$, the random walker could reach neighbors that were far away from the POI. The distant POIs might have no common transition patterns, as the distance of two successive check-ins is often less than $10km$ (Feng et al., 2015). By following $L \rightarrow L$, although the random walker can find nearby neighbors, these neighbors (for example, a museum) may not share similar characteristics with the POI (for example, a gym club). Despite some unsatisfactory neighbors could be retrieved, their saliences can be distinguished automatically by the attention layer.

Table 6.2: Performance variation of ARNN with different time threshold (Δt) on three datasets evaluated by Acc@N. The best performance is highlighted in bold.

City	Metric	3h	6h	12h	1d	2d
NY	Acc@1	0.1816	0.2058	0.1738	0.1526	0.1220
	Acc@5	0.3437	0.348	0.3530	0.2943	0.2498
	Acc@10	0.4076	0.4087	0.4162	0.3317	0.2977
	Acc@20	0.4474	0.4571	0.4393	0.3544	0.3097
	Pct($\leq \Delta t$)	46.75%	54.45%	63.83%	80.04%	90.45%
TK	Acc@1	0.1811	0.1801	0.1867	0.1651	0.1321
	Acc@5	0.3448	0.3605	0.3657	0.3044	0.2283
	Acc@10	0.4121	0.4180	0.4285	0.3569	0.2637
	Acc@20	0.4625	0.4889	0.4864	0.4125	0.3392
	Pct($\leq \Delta t$)	58.96%	63.89%	71.74%	84.33%	92.18%
SF	Acc@1	0.0699	0.1297	0.1324	0.0816	0.0829
	Acc@5	0.1441	0.2010	0.2128	0.1780	0.1685
	Acc@10	0.1594	0.2318	0.2336	0.2126	0.2072
	Acc@20	0.1703	0.2431	0.2530	0.2336	0.2251
	Pct($\leq \Delta t$)	50.26%	56.68%	63.48%	80.34%	90.52%

6.3.3 Analysis of Time Threshold

Previous studies show that the time interval of two successive check-ins had critical influence on next POI recommendation (Zhang et al., 2014b; Feng et al., 2015). It is

thus necessary to investigate the impact of various settings of the time threshold (Δt) on the accuracy of our method, illustrated in Table 6.2. With $\Delta t = 12h$, ARNN generated better results than other settings. To better explain the results, the percentages of the successive check-in pairs that happened within various Δt were calculated, that is, $\text{Pct}(\leq \Delta t) = \frac{\sum_{i=1}^{|U|} |\{(r_k, r_{k+1}) \mid r_k \in \text{His}(u_i), t_{k+1} - t_k \leq \Delta t\}|}{\sum_{i=1}^{|U|} |\{(r_k, r_{k+1}) \mid r_k \in \text{His}(u_i)\}|}$, where $\text{His}(u_i)$ represents the historical sequences of u_i , r_k indicates a check-in and its following check-in is r_{k+1} .

As shown in Table 6.2, for about 50% and 60% of successive check-ins, their time interval is smaller than $3h$ and $6h$, respectively. This verifies previous finding that the majority of successive check-ins happen within a few hours (Cheng et al., 2013). The best results were achieved when $\Delta t = 12h$, implying the successive check-ins with the time interval in $6h - 12h$ still carry effective transition information. However, the performance deteriorated when Δt increased to $24h$, because the time interval exceeded to a point such that wrong transition information might be introduced in the trajectories. Overall, $\Delta t = 12h$ was a robust setting to generate high-quality trajectories for the model training on different cities.

6.3.4 Analysis of Hyper-parameters

This section demonstrate the experimental results about the impacts of various settings of hyper-parameters.

The Effect of Dimensionality. Figure 6.7 describes the recommendation performance with respect to Acc@1 for various dimensionality (that is, embedding size) with other optimal hyperparameters fixed on all the three datasets. Note similar trends can be observed for other metrics. The performance became stable when the dimension increased to a certain level. Besides, the optimal dimension size of Tokyo was higher than others, since there are much more POIs in Tokyo. Moreover, ARNN consistently outperformed other state-of-the-art methods when its results remained stable, indicating that it could well deal with the variation of dimensionality.

Figure 6.7: Analysis of Dimensionality

The Effect of the Number of Iterations. Furthermore, to understand the convergence of ARNN, the performance change was recorded with respect to the number of iterations. As shown in Figure 6.8, it could be observed that ARNN could converge within 20 iterations. In summary, with strong ability to dealing with the dimensionality and fast convergence speed, ARNN is a robust approach that can be applied in practical scenarios.

6.3.5 Comparative Results

With optimal settings of involved parameters, experiments were conducted to evaluate ARNN with state-of-the-art methods. Table 6.3 presents the performance of all comparison methods evaluated by $\text{Acc}@N$ with $N = 1, 5, 10, 20$. UCF performed worst since it only models user general interest without considering sequential regularity, the most important influence for next POI recommendation. By modeling sequential regularity, FPMC performed better than UCF. FPMC-LR outperformed FPMC by further fusing geographical information. However, the results of both Markov-based methods were still

Figure 6.8: Analysis of the Number of Iterations

poor, which could be explained by the two limitations of the MC technique. To illustrate, it fails to model high-order sequential regularity, as they only captures the Markov properties of trajectories. Experiments were also conducted on the Matrix Factorization technique, however, the result was even worse than UCF. MF could not learn effective latent features of POIs, due to the large-volume of POIs without sufficient information.

In general, RNN-based methods outperformed Markov-based methods because of their capability of memorizing long-term dependencies (Hochreiter & Schmidhuber, 1997). ST-RNN delivered decent results by further exploiting spatial and temporal contexts associated with each check-in in the sequence. However, ST-RNN was outperformed by LSTM in some cases due to the stronger ability of LSTM to model long sequence than basic RNN. By considering semantic context, SERM performed better than ST-RNN. Moreover, SERM also involved much fewer parameters by embedding various contexts into low-dimensional vectors, whereas ST-RNN learned time-specific and distance-specific transition matrices. DeepMove adopted a similar attention-based LSTM architecture applied in language translation (Luong et al., 2015), to capture the multi-level influence

Table 6.3: Performance of all the comparison methods on the three real-world datasets measured Acc@N ($N = 1, 5, 10, 20$). The best performance is highlighted in bold while the second best performance is marked by ‘†’. ‘Improve’ indicates the relative improvements that ARNN achieves relative to the best performance of other methods.

City	Metric	UCF	FPMC	FPMC-LR	LSTM	ST-RNN	SERM	DeepMove	Caser	ARNN	Improve
NY	Acc@1	0.0010	0.0214	0.0248	0.1384	0.1292	0.1571	0.1595	0.1608†	0.1738	8.08%
	Acc@5	0.0055	0.0386	0.0403	0.2378	0.2474	0.2880	0.3072	0.3210†	0.3530	9.97%
	Acc@10	0.0096	0.0735	0.0783	0.2752	0.2803	0.3230	0.3524	0.3767†	0.4162	10.49%
	Acc@20	0.0131	0.0872	0.0929	0.3058	0.3299	0.3437	0.3792	0.4016†	0.4393	9.39%
TK	Acc@1	0.0092	0.0196	0.0227	0.1458	0.1572	0.1608	0.1652	0.1709†	0.1867	9.25%
	Acc@5	0.0197	0.0312	0.0358	0.2554	0.2443	0.2899	0.3065	0.3302†	0.3657	10.75%
	Acc@10	0.0385	0.0597	0.0626	0.2989	0.2794	0.3407	0.3890†	0.3852	0.4285	10.15%
	Acc@20	0.0199	0.0686	0.0753	0.3399	0.3285	0.3842	0.4258	0.4387†	0.4864	10.87%
SF	Acc@1	0.0005	0.0122	0.0167	0.1116	0.1035	0.1205	0.1186	0.1220†	0.1324	8.52%
	Acc@5	0.0025	0.0245	0.0314	0.1667	0.1630	0.1949	0.1973†	0.1894	0.2128	7.86%
	Acc@10	0.0045	0.0426	0.0512	0.1845	0.1860	0.2098	0.2102	0.2177†	0.2336	7.30%
	Acc@20	0.0076	0.0541	0.0625	0.2009	0.2098	0.2321	0.2326	0.2345†	0.2530	7.89%

of distant check-ins in the sequence. The way of integrating attention layer in ARNN was different from DeepMove. To explain, the attention layer in ARNN captures the transition regularities of each neighbor at each time step while it captures the impacts of previous check-ins along the sequence. In general, Caser outperformed other RNN-based methods by modeling the effects of past check-ins jointly while others models previous check-ins independently.

However, either the methods of learning embeddings of different contexts (ST-RNN and SERM) or modeling the complex sequential regularity (DeepMove and Caser) were deficient in solving the sparsity issue. By modeling the transition regularities of various neighbors, ARNN performed the best with the average improvements are 9.48%, 10.26%, 7.89% on the three datasets with p -value < 0.01 . The significant improvement margins demonstrates the strong ability of ARNN in resolving the sparsity for more accurate recommendation.

6.4 Conclusion

This study explored the transition regularities of neighbors to resolve the sparsity issue for next POI recommendation, as the data analysis showed that neighbors have similar transition patterns. A meta-path based random walk on a novel heterogeneous information network was first designed to discover relevant neighbors based on heterogeneous factors. Next, both sequential regularity and transition regularities of neighbors were jointly modeled by proposing an attentional recurrent neural network framework, ARNN. ARNN was a composite architecture with LSTM that could model the sequential regularity while the transition regularities of the discovered neighbors were integrated via attention mechanism. Experimental results on multiple real-world datasets demonstrates the superiority of ARNN to state-of-the-art approaches.

Chapter 7

Conclusions and Future Work

This chapter first summarizes the contributions of this thesis and presents several promising directions for the future research.

7.1 Summary of Contributions

The huge amount of heterogeneous information in LBSNs creates great challenges to develop an effective POI recommender system. Previous studies have discovered that multiple influential factors have impacts on user check-in decision. Although some studies take efforts to partially model those influential factors, there is a lack of research that attempts to capture multiple factors. The geographical constraints, complexity and heterogeneity of user check-ins, bring severe challenges to model various influential factors in a unified framework. To overcome such challenges, this thesis proposed a series of novel POI recommender systems by considering two perspectives including representation and methodology.

The first study aims to explore geographical and content factors to retrieve more effective neighbors in CF framework with a homogeneous graph in Chapter 3. The proposed framework (TSLRS) first rely on the textual descriptions of POIs to distill the topics and infer the topic expertise by taking user topic similarity into account. The new measurement of user similarity is developed based on the target user topic

preference and the topic expertise of neighbors, so as to discover more effective neighbors by further considering the geographical distribution of user check-ins. The experimental evaluation shows the proposed approach performed better than state-of-the-art graph-based approaches.

Chapter 4 describes the second study which proposed a more comprehensive framework than Chapter 3 – the proposed graph-based approach, AGS-RW, exploits geographical, social and content factors in a unified manner. More specifically, a novel heterogeneous graph, AGS-IG, is built to embed the heterogeneous information regarding users, POIs and aspects. Based on AGS-IG, AGS-RW incorporates meta-paths into personalized PageRank by exploiting the semantics of AGS-IG to improve the recommendation performance. By doing so, AGS-RW captures the diversity property of user check-in decision by modeling those influential factors in a unified framework. The extensive experiments demonstrate that the proposed method significantly outperformed state-of-the-art algorithms.

Chapter 5 presents the third study which dived deeper into modeling various influential factors for the general POI recommendation task. User check-in behavior exhibits two properties: (i) diversity; and (ii) imbalance. Prior studies model the diversity property, but fail to capture the imbalance property. First, a meta-path based random walk process has been designed to efficiently generate neighbors of users or POIs based on different influential factors over AGS-IG. Then, a novel matrix factorization (MF) framework (AGS-MF) is proposed to integrate the regularizers of meta-path based neighbors into MF, while learn the personalized weights of meta-paths for each individual user and POI (Guo et al., 2019b). In this sense, both the diversity and imbalance properties of user check-in decision can be effectively captured. Furthermore, by incorporating the various kinds of neighbors, AGS-MF is also capable of learning more effective latent representations that preserve user preference, POI characteristics, as well as the heterogeneous

properties of AGS-IG. The empirical results demonstrate the significant improvements by AGS-MF in comparison with state-of-the-art methods.

Existing studies mostly explore the sequential regularity of user check-in sequence, but inherently suffer from the data sparsity issue. Thus, the transition regularities of various kinds of POI neighbors are exploited to ease such data sparsity issue. The fourth study proposed a novel recurrent framework (ARNN) to achieve a joint modeling of both the sequential regularity and transition regularities of various neighbors to resolve the sparsity for better next POI recommendation in Chapter 6. Specifically, a multi-modal embedding layer is first designed to transfer the sparse features into dense embeddings fed into recurrent layer. To extract various types of neighbors, a novel heterogeneous graph is designed to represent the heterogeneous information including users, POIs and semantic information (that is, categories and tags of POIs) into a unified space. A *meta-path* based random walk process is then designed to efficiently discover the neighbors based on the heterogeneous influential factors. An attention layer is designed to capture the transition regularities of neighbors at each time step in the sequence, while automatically selects highly salient neighbors correlated to current POI. Finally, the attention layer is tailored to seamlessly cooperate with the recurrent layer as a unified recurrent framework. The experimental evaluation shows that the significant performance enhancement by ARNN compared with state-of-the-art methods.

In summary, a series of recommendation approaches have been proposed for both the general and next POI recommendation tasks. The main contributions of this thesis could be summarized as: (i) for representation of the heterogeneous information in a unified space, homogeneous graph and heterogeneous graph based models are investigated; and (ii) for methodologies of exploiting the heterogeneous information represented by graph structures, novel graph-based ranking methods are proposed to latent factor model, then deep learning method in order to model the heterogeneity of user check-in behavior.

7.2 Future Work

There are multiple potential directions for future work. In this thesis, two potential and promising directions are worth exploring in the future:

Sentiment Analysis. One possible direction is to exploit the sentiment analysis technique to further capture user preference in a finer granularity. To explain, Chapter 4 and Chapter 5 utilize the aspects extracted from user reviews, that is, they capture the frequency that the user mentions a certain aspect, which can estimate the importance level of an aspect for the user. To better understand the user’s opinion about an aspect for a POI, it is essential to figure out her sentiment towards the aspect. By applying sentiment analysis to user review, for example, “I really enjoy the pasta in this cafe, but it’s quite noisy around.”, the user’s preference can be captured in a detailed level, that is, (“pasta”, positive), (“environment”, negative). Therefore, the edge weight of the knowledge graph can be more precise for capturing the relations among various types of entities.

Deep Learning Methods. Another direction is pointed out from the perspective of techniques. In recent years, the revolutionary advances of deep learning technique have gained significant attention in various domains, such as speech recognition, image analysis and natural language processing, with recommender systems being no exceptions (Zhang et al., 2017a). Deep learning provides a better understanding of user’s demands, item’s characteristics and historical interactions between them, thus can be adapted to further enhance recommendation performance. Thus, deep learning models are expected to be adapted in the context of location-based social network (LBSN). The challenge for applying such idea is to combine the multi-modal features, including textual features, image feature and user preference based on check-in matrix. As they are represented in various ways, it would be such a difficult task to represent different kind of features in a unified space, which is a future direction worth exploring with deep learning models.

Real-time POI Recommendation. Location-based check-in data carries strong preference signals, providing us a new perspective to understand human mobility pattern. However, existing POI recommender systems are limited to modeling user preference only based on historical data, due to the lack of real-time contextual data. In fact, a user’s decision of visiting a place is often influenced by the real-time conditions. For instance, users may consider the weather condition to decide whether he/she will visit a POI. It brings a new concept of POI recommendation – real-time POI recommendation in location-based social networks (Bao et al., 2015). The real-time POI recommendation definition is similar to next POI recommendation, that is, given the user’s historical check-ins, the previous check-in sequence, the current time and the real-time contexts, recommend a list of POIs that the user prefers to visit right now. For example, given a check-in sequence before 19:00 in a day, *home(8:00) → bus stop(8:30) → coffee shop(9:00) → office(9:30) → ... → ?(now:19:00, raining, heavy traffic, ...)*, the real-time POI recommendation task is to predict what the user would like to visit now with the real-time contexts. There have been some studies on real-time location prediction with GPS-based trajectory data (Zheng et al., 2010; Zheng & Xie, 2011; Zheng et al., 2012). Nevertheless, there is a lack of research focusing on exploiting check-in data for the real-time POI recommendation task. In essence, we can consider real-time POI recommendation as an extension of next POI recommendation since they both need to use users’ past trajectory data for predicting where the user would visit in real time. The difference between the two tasks lies in that real-time POI recommendation need to consider real-time contextual information, such as weather, traffic, car park conditions and social events. As shown in Figure 7.1¹ (Park et al., 2007; Zheng et al., 2010; Bao et al., 2015), real-time POI recommendation is extremely difficult due to the variety of data sources.

In particular, there are several significant challenges of developing an effective real-time POI recommender system:

¹This figure is not directly given by related works, but is summarized based on them.

Figure 7.1: An overview of real-time POI recommendation

- The sparsity of check-in sequence create the handiness of learning useful transitional patterns.
- The lack of real-time contextual data. Intuitively, the user next movement could be determined by some real-time contexts. For example, if it is raining, a user prefers having lunch in a restaurant in his/her office building rather than somewhere far away.
- The lack of user profile information. The current datasets only give user IDs without other information about them, such as their demographics. However, additional information of users can provide valuable clues to understand their preference. For instance, with the age data, the POI recommender system can filter out some inappropriate places (such as night clubs) in the recommendation list for teenagers. Besides, people from different culture also visit different kind of POIs, for example, westerners mostly would have dinner at western restaurants after work while Indians prefer Indian cuisines.

- The lack of POI profile information. Except for some textual information (categories, tags and reviews), there is no other information, such as food price and promotion. If such information is available, the recommender system can push POIs with special offers for those price sensitive users;
- The difficulty of incorporating the influences of various kinds of information as well as measuring their respectively importance for a user's check-in decision. Currently, the research about next POI recommendation primarily focuses on developing advanced deep learning models with historical check-in trajectories and other side information (e.g., social network, POI descriptions and reviews).

In order to drive the real-time POI recommendation study, excessive research efforts need to be taken to:

- Identify more useful data sources about the real-time contextual information, users and POIs. For example, real-time weather and car park availability data of Singapore can be found on the government website². Since different location-based services operate different businesses, more complete POI profiles can be built by aggregating data from various LBSN platforms, for instance, Groupon³ may share more promotion or food price information which might not be obtained from Yelp and Foursquare. However, the full user profiles may be difficult to build due to the user privacy concerns. LBSN platforms thus need to take special efforts to carefully release user profile data with privacy protection.
- Match those additional datasets (real-time contextual, user and POI profiles) with the check-in dataset as there is no such dataset available for research including all types of data.

²<https://data.gov.sg/developer>

³<http://partner-api.groupon.com/help/>

- Analyze how different factors contribute to users' next movement with both historical check-ins and real-time contextual data.
- Develop a machine learning model being able to capture different factors. There have extensive research studies working on related topics, such as traffic prediction (Abdel-Aty & Pemmanaboina, 2006; Min & Wynter, 2011; Liao et al., 2018a), and social events detection in the social media platform (Koesdwiady et al., 2016; Chae et al., 2012; Zhang et al., 2018). Therefore, researchers can take those related works as reference to adjust current POI recommender system for real-time POI recommendation.

List of Publications

- **Guo, Q.**, Sun, Z., Zhang, J., Theng, Y.-L. (2020). An Attentional Recurrent Neural Network for Personalized Next Location Recommendation. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2020.
- **Guo, Q.**, Sun, Z., & Theng, Y.-L. (2019). A Tutorial on Exploiting Side Information for Recommendation. *the 19th International Conference on Web Engineering*, 2019.
- Sun, Z., **Guo, Q.**, Yang, J., Fang, H., Guo, G., Zhang, J., Robin, B. (2019). Recommendation with Side Information: A Survey and New Research Directions. Submitted to *Electronic Commerce Research and Applications*, 2019.
- **Guo, Q.**, Sun, Z., Zhang, J., & Theng, Y.-L. (2019). Modeling Heterogeneous Influences for Point-of-Interest Recommendation in Location-Based Social Networks. Submitted to *the 19th International Conference on Web Engineering*, 2019.
- **Guo, Q.**, Sun, Z., Zhang, J., Chen, Q., & Theng, Y.-L. (2017). Aspect-aware point-of-interest recommendation with geo-social influence. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 17–22. ACM.
- **Guo, Q.**, Huang, Y., & Theng, Y.-L. (2015). Topic-sensitive location recommendation with spatial awareness. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WIIAT)*, volume 1, pages 237–243. IEEE.

References

- Abdel-Aty, M. A. & Pemmanaboina, R. (2006). Calibrating a real-time traffic crash-prediction model using archived weather and ITS traffic data. *IEEE Transactions on Intelligent Transportation Systems*, 7(2):167–174.
- Adomavicius, G. & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, (6):734–749.
- Anzai, Y. (2012). *Pattern recognition and machine learning*. Elsevier.
- Arain, Q. A., Memon, H., Memon, I., Memon, M. H., Shaikh, R. A., & Mangi, F. A. (2017). Intelligent travel information platform based on location base services to predict user travel behavior from user-generated GPS traces. *International Journal of Computers and Applications*, 39(3):155–168.
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Bao, J., Zheng, Y., & Mokbel, M. F. (2012). Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 199–208. ACM.
- Bao, J., Zheng, Y., Wilkie, D., & Mokbel, M. (2015). Recommendations in location-based social networks: a survey. *GeoInformatica*, 19(3):525–565.
- Bell, R. M. & Koren, Y. (2007a). Lessons from the Netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, 9(2):75–79.

- Bell, R. M. & Koren, Y. (2007b). Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights. In *icdm*, volume 7, pages 43–52. Citeseer.
- Bellogín, A., Cantador, I., Díez, F., Castells, P., & Chavarriaga, E. (2013). An empirical comparison of social, collaborative filtering, and hybrid recommenders. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):14.
- Bhargava, P., Phan, T., Zhou, J., & Lee, J. (2015). Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In *Proceedings of the 24th International Conference on World Wide Web*, pages 130–140. ACM.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.
- Bottou, L. (1991). Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, 91(8):12.
- Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *the 19th International Conference on Computational Statistics*, pages 177–186. Springer.
- Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 43–52. Morgan Kaufmann Publishers Inc.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-adapted Interaction*, 12(4):331–370.
- Burke, R. (2007). Hybrid web recommender systems. In *The Adaptive Web*, pages 377–408. Springer.
- Cai, L., Xu, J., Liu, J., & Pei, T. (2018). Integrating spatial and temporal contexts into a factorization model for POI recommendation. *International Journal of Geographical Information Science*, 32(3):524–546.

- Chae, J., Thom, D., Bosch, H., Jang, Y., Maciejewski, R., Ebert, D. S., & Ertl, T. (2012). Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE VAST*, pages 143–152.
- Cheng, C., Yang, H., King, I., & Lyu, M. R. (2012). Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, volume 12, page 1.
- Cheng, C., Yang, H., Lyu, M. R., & King, I. (2013). Where You Like to Go Next: Successive Point-of-Interest Recommendation. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, volume 13, pages 2605–2611.
- Cheng, Z., Caverlee, J., Lee, K., & Sui, D. Z. (2011). Exploring millions of footprints in location sharing services. *The 5th International AAAI Conference on Web and Social Media*, 2011:81–88.
- Cho, E., Myers, S. A., & Leskovec, J. (2011). Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1082–1090. ACM.
- Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., et al. (2010). The YouTube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 293–296. ACM.
- Ding, Z., Li, X., Jiang, C., & Zhou, M. (2018). Objectives and State-of-the-Art of Location-Based Social Network Recommender Systems. *ACM Computing Surveys (CSUR)*, 51(1):18.
- Eirinaki, M., Vazirgiannis, M., & Kapogiannis, D. (2005). Web path recommendations based on page ranking and markov models. In *Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management*, pages 2–9. ACM.

- Feng, J., Li, Y., Zhang, C., Sun, F., Meng, F., Guo, A., & Jin, D. (2018). DeepMove: Predicting Human Mobility with Attentional Recurrent Networks. In *Proceedings of the 2018 World Wide Web Conference*, pages 1459–1468. International World Wide Web Conferences Steering Committee.
- Feng, S., Cong, G., An, B., & Chee, Y. M. (2017). POI2Vec: Geographical Latent Representation for Predicting Future Visitors. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, pages 102–108.
- Feng, S., Li, X., Zeng, Y., Cong, G., Chee, Y. M., & Yuan, Q. (2015). Personalized Ranking Metric Embedding for Next New POI Recommendation. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2069–2075.
- Gao, H., Tang, J., Hu, X., & Liu, H. (2013). Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 93–100. ACM.
- Gao, H., Tang, J., Hu, X., & Liu, H. (2015). Content-Aware Point of Interest Recommendation on Location-Based Social Networks. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, pages 1721–1727.
- Gao, R., Li, J., Li, X., Song, C., & Zhou, Y. (2018). A personalized point-of-interest recommendation model via fusion of geo-social information. *Neurocomputing*, 273:159–170.
- Ghazanfar, M. & Prugel-Bennett, A. (2010). An improved switching hybrid recommender system using Naive Bayes classifier and collaborative filtering. *The 2010 International Conference on Data Mining and Applications*.
- Gori, M., Pucci, A., Roma, V., & Siena, I. (2007). ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, volume 7, pages 2766–2771.
- Griffiths, T. L. & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235.

- Gruslys, A., Munos, R., Danihelka, I., Lanctot, M., & Graves, A. (2016). Memory-efficient backpropagation through time. In *Advances in Neural Information Processing Systems*, pages 4125–4133.
- Guo, G., Zhang, J., Sun, Z., & Yorke-Smith, N. (2015a). LibRec: A Java Library for Recommender Systems. In *UMAP Workshops*, volume 4.
- Guo, G., Zhang, J., & Yorke-Smith, N. (2013). A Novel Bayesian Similarity Measure for Recommender Systems. In *The International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2619–2625.
- Guo, Q., Huang, Y., & Theng, Y.-L. (2015b). Topic-Sensitive Location Recommendation with Spatial Awareness. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 1, pages 237–243. IEEE.
- Guo, Q., Sun, Z., & Theng, Y.-L. (2019a). Exploiting Side Information for Recommendation. In *International Conference on Web Engineering*, pages 569–573. Springer.
- Guo, Q., Sun, Z., Zhang, J., Chen, Q., & Theng, Y.-L. (2017). Aspect-aware point-of-interest recommendation with geo-social influence. In *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization (UMAP)*, pages 17–22. ACM.
- Guo, Q., Sun, Z., Zhang, J., & Theng, Y.-L. (2019b). Modeling Heterogeneous Influences for Point-of-Interest Recommendation in Location-Based Social Networks. In *International Conference on Web Engineering*, pages 72–80. Springer.
- Haveliwala, T. H. (2002). Topic-sensitive pagerank. In *Proceedings of the 11th International Conference on World Wide Web (WWW)*, pages 517–526. ACM.
- He, J., Li, X., Liao, L., Song, D., & Cheung, W. K. (2016). Inferring a Personalized Next Point-of-Interest Recommendation Model with Latent Behavior Patterns. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, pages 137–143.
- He, X., Chen, T., Kan, M.-Y., & Chen, X. (2015). Trirank: Review-aware explainable recommendation by modeling aspects. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*.

- Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*.
- Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Hong, L., Ahmed, A., Gurumurthy, S., Smola, A. J., & Tsioutsoulouklis, K. (2012). Discovering geographical topics in the twitter stream. In *Proceedings of the 21st International Conference on World Wide Web (WWW)*.
- Hu, B. & Ester, M. (2014). Social topic modeling for point-of-interest recommendation in location-based social networks. In *2014 IEEE International Conference on Data Mining (ICDM)*, pages 845–850. IEEE.
- Hu, G., Shao, J., Ni, Z., & Zhang, D. (2018). A graph based method for constructing popular routes with check-ins. *World Wide Web*, 21(6):1689–1703.
- Hu, M. & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*.
- Hwang, W.-S., Lee, H.-J., Kim, S.-W., & Lee, M. (2012). On using category experts for improving the performance and accuracy in recommender systems. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2355–2358. ACM.
- Jamali, M. & Ester, M. (2010). A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the 4th ACM Conference on Recommender systems*, pages 135–142. ACM.
- Jiang, B. & Yao, X. (2006). Location-based services and GIS in perspective. *Computers, Environment and Urban Systems*, 30(6):712–725.

- Jin, Z., Shi, D., Wu, Q., Yan, H., & Fan, H. (2012). LBSNRank: personalized pagerank on location-based social networks. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 980–987. ACM.
- Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Koesdwiady, A., Soua, R., & Karray, F. (2016). Improving traffic flow prediction with weather information in connected cars: a deep learning approach. *IEEE Transactions on Vehicular Technology*, 65(12):9508–9517.
- Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 8:30–37.
- Lampropoulos, A. S., Lampropoulou, P. S., & Tsihrintzis, G. A. (2012). A cascade-hybrid music recommender system for mobile services based on musical genre classification and personality diagnosis. *Multimedia Tools and Applications*, 59(1):241–258.
- Lee, S., Park, S., Kahng, M., & Lee, S.-G. (2013). Pathrank: Ranking nodes on a heterogeneous graph for flexible hybrid recommender systems. *Expert Systems with Applications*, 40(2).
- Lekakos, G. & Caravelas, P. (2008). A hybrid approach for movie recommendation. *Multimedia Tools and Applications*, 36(1-2):55–70.
- Lemire, D. & Maclachlan, A. (2005). Slope one predictors for online rating-based collaborative filtering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 471–475. SIAM.
- Li, H., Ge, Y., Hong, R., & Zhu, H. (2016). Point-of-interest recommendations: Learning potential check-ins from friends. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 975–984. ACM.

- Li, X., Cong, G., Li, X.-L., Pham, T.-A. N., & Krishnaswamy, S. (2015). Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 433–442. ACM.
- Li, Z., Ding, B., Han, J., Kays, R., & Nye, P. (2010). Mining periodic behaviors for moving objects. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1099–1108. ACM.
- Lian, D., Ge, Y., Zhang, F., Yuan, N. J., Xie, X., Zhou, T., & Rui, Y. (2015). Content-aware collaborative filtering for location recommendation based on human mobility data. In *2015 IEEE International Conference on Data Mining (ICDM)*, pages 261–270. IEEE.
- Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E., & Rui, Y. (2014). GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*.
- Liao, B., Zhang, J., Wu, C., McIlwraith, D., Chen, T., Yang, S., Guo, Y., & Wu, F. (2018a). Deep sequence learning with auxiliary information for traffic prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 537–546. ACM.
- Liao, G., Jiang, S., Zhou, Z., Wan, C., & Liu, X. (2018b). POI Recommendation of Location-Based Social Networks Using Tensor Factorization. In *2018 19th IEEE International Conference on Mobile Data Management (MDM)*, pages 116–124. IEEE.
- Lieberman, H. et al. (1995). Letizia: An agent that assists web browsing. *The International Joint Conference on Artificial Intelligence (IJCAI)*, 1995:924–929.
- Liu, B., Fu, Y., Yao, Z., & Xiong, H. (2013). Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1043–1051. ACM.

- Liu, B. & Xiong, H. (2013). Point-of-interest recommendation in location based social networks with topic and location awareness. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 396–404. SIAM.
- Liu, B., Xiong, H., Papadimitriou, S., Fu, Y., & Yao, Z. (2014a). A general geographical probabilistic factor model for point of interest recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 27(5):1167–1179.
- Liu, F. & Lee, H. J. (2010). Use of social network information to enhance collaborative filtering performance. *Expert Systems with Applications*, 37(7):4772–4778.
- Liu, Q., Wu, S., Wang, L., & Tan, T. (2016). Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, pages 194–200.
- Liu, Y., Pham, T.-A. N., Cong, G., & Yuan, Q. (2017). An experimental evaluation of point-of-interest recommendation in location-based social networks. *Proceedings of the VLDB Endowment*, 10(10):1010–1021.
- Liu, Y., Wei, W., Sun, A., & Miao, C. (2014b). Exploiting geographical neighborhood characteristics for location recommendation. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 739–748. ACM.
- Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*, pages 73–105. Springer.
- Luong, M.-T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Ma, H., Zhou, D., Liu, C., Lyu, M. R., & King, I. (2011). Recommender systems with social regularization. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 287–296. ACM.

- Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., & Stumme, G. (2009). Evaluating similarity measures for emergent semantics of social tagging. In *Proceedings of the 18th International Conference on World Wide Web (WWW)*, pages 641–650. ACM.
- McNee, S. M., Kapoor, N., & Konstan, J. A. (2006). Don’t look stupid: avoiding pitfalls when recommending research papers. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, pages 171–180. ACM.
- Melville, P., Mooney, R. J., & Nagarajan, R. (2002). Content-boosted collaborative filtering for improved recommendations. *The Association for the Advancement of Artificial Intelligence (AAAI)*, 23:187–192.
- Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010). Recurrent neural network based language model. In *Interspeech*, volume 2, page 3.
- Min, W. & Wynter, L. (2011). Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, 19(4):606–616.
- Miyahara, K. & Pazzani, M. J. (2000). Collaborative filtering with the simple Bayesian classifier. In *Pacific Rim International Conference on Artificial Intelligence*, pages 679–689. Springer.
- Miyahara, K. & Pazzani, M. J. (2002). Improvement of collaborative filtering with the simple Bayesian classifier. *Information Processing Society of Japan*, 43(11).
- Mnih, V., Heess, N., Graves, A., et al. (2014). Recurrent models of visual attention. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS)*, pages 2204–2212.
- Monreale, A., Pinelli, F., Trasarti, R., & Giannotti, F. (2009). Wherenext: a location predictor on trajectory pattern mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 637–646. ACM.

- Noulas, A., Scellato, S., Lathia, N., & Mascolo, C. (2012). A random walk around the city: New venue recommendation in location-based social networks. In *Proceedings of the 2012 ASE/IEEE International Conference on Social Computing and ASE/IEEE International Conference on Privacy, Security, Risk and Trust*, pages 144–153. IEEE.
- Ogundele, T. J., Chow, C.-Y., & Zhang, J.-D. (2017). SoCaST: Exploiting Social, Categorical and Spatio-Temporal Preferences for Personalized Event Recommendations. In *Pervasive Systems, Algorithms and Networks & 2017 11th International Conference on Frontier of Computer Science and Technology & 2017 Third International Symposium on Creative Computing (ISPAN-FCST-ISCC), 2017 14th International Symposium on*, pages 38–45. IEEE.
- Ostapuk, N., Yang, J., & Cudré-Mauroux, P. (2019). Activelink: deep active learning for link prediction in knowledge graphs. In *The World Wide Web Conference*, pages 1398–1408. ACM.
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab.
- Park, M.-H., Hong, J.-H., & Cho, S.-B. (2007). Location-based recommendation system using bayesian user’s preference model in mobile devices. In *International conference on ubiquitous intelligence and computing*, pages 1130–1139. Springer.
- Pennock, D. M., Horvitz, E., Lawrence, S., & Giles, C. L. (2000). Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 473–480. Morgan Kaufmann Publishers Inc.
- Qiao, S., Han, N., Zhou, J., Li, R.-H., Jin, C., & Gutierrez, L. A. (2018). SocialMix: A familiarity-based and preference-aware location suggestion approach. *Engineering Applications of Artificial Intelligence*, 68:192–204.
- Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*.

- Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pages 811–820. ACM.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533.
- Salakhutdinov, R. & Mnih, A. (2007). Probabilistic Matrix Factorization. In *In Advances in Neural Information Processing Systems*, volume 1, pages 2–1.
- Salton, G. & McGill, M. J. (1986). Introduction to modern information retrieval.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Application of dimensionality reduction in recommender system-a case study. Technical report, Minnesota Univ Minneapolis Dept of Computer Science.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW)*, pages 285–295. ACM.
- Scellato, S., Noulas, A., & Mascolo, C. (2011). Exploiting place features in link prediction on location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1046–1054. ACM.
- Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 253–260. ACM.
- Schiller, J. & Voisard, A. (2004). *Location-based services*. Elsevier.
- Scott, D. W. (2015). *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.

- Shi, Y., Larson, M., & Hanjalic, A. (2014). Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)*, 47(1):3.
- Sizov, S. (2010). Geofolk: latent spatial semantics in web 2.0 social media. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM)*.
- Sontag, D. & Roy, D. (2011). Complexity of inference in latent dirichlet allocation. In *Advances in neural information processing systems*, pages 1008–1016.
- Steiniger, S., Neun, M., Edwardes, A., & Lenz, B. (2008). Foundations of LBS. *CartouCHe-Cartography for Swiss Higher Education*. *Obtido em*, 20:2010.
- Sun, Y., Han, J., Yan, X., Yu, P. S., & Wu, T. (2011). Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment*, 4(11):992–1003.
- Sun, Z., Guo, Q., Yang, J., Fang, H., Guo, G., Zhang, J., & Burke, R. (2019). Research commentary on recommendations with side information: A survey and research directions. *Electronic Commerce Research and Applications*, 37:100879.
- Symeonidis, P., Nanopoulos, A., & Manolopoulos, Y. (2009). A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 22(2):179–192.
- Tang, J. & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 565–573. ACM.
- Virrantaus, K., Markkula, J., Garmash, A., Terziyan, V., Veijalainen, J., Katanosov, A., & Tirri, H. (2001). Developing GIS-supported location-based services. In *Proceedings of the 2nd International Conference on Web Information Systems Engineering*, volume 2, pages 66–75. IEEE.
- Vucetic, S. & Obradovic, Z. (2005). Collaborative filtering using a regression-based approach. *Knowledge and Information Systems*, 7(1):1–22.

- Wang, H., Terrovitis, M., & Mamoulis, N. (2013). Location recommendation in location-based social networks using user check-in data. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 374–383. ACM.
- Wang, W., Pan, S. J., Dahlmeier, D., & Xiao, X. (2016). Recursive Neural Conditional Random Fields for Aspect-based Sentiment Analysis. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Wang, Y., Yuan, N. J., Lian, D., Xu, L., Xie, X., Chen, E., & Rui, Y. (2015a). Regularity and conformity: Location prediction using heterogeneous mobility data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1275–1284. ACM.
- Wang, Z.-S., Juang, J.-F., & Teng, W.-G. (2015b). Predicting poi visits with a heterogeneous information network. In *2015 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pages 388–395. IEEE.
- Xiao, B. & Benbasat, I. (2007). E-commerce product recommendation agents: use, characteristics, and impact. *Management Information Systems Quarterly*, 31(1):137–209.
- Xie, M., Yin, H., Xu, F., Wang, H., & Zhou, X. (2016). Graph-based metric embedding for next poi recommendation. In *International Conference on Web Information Systems Engineering*, pages 207–222. Springer.
- Yang, C., Bai, L., Zhang, C., Yuan, Q., & Han, J. (2017). Bridging collaborative filtering and semi-supervised learning: a neural approach for poi recommendation. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1245–1254. ACM.
- Yang, D., Zhang, D., Yu, Z., & Wang, Z. (2013). A sentiment-enhanced personalized location recommendation system. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*.

- Yang, D., Zhang, D., Zheng, V. W., & Yu, Z. (2015). Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):129–142.
- Yao, D., Zhang, C., Huang, J., & Bi, J. (2017). SERM: A recurrent model for next location prediction in semantic trajectories. In *The 26th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2411–2414. ACM.
- Yao, W., He, J., Huang, G., Cao, J., & Zhang, Y. (2015). A Graph-based model for context-aware recommendation using implicit feedback data. *World Wide Web*, 18(5):1351–1371.
- Ye, M., Yin, P., & Lee, W.-C. (2010). Location recommendation for location-based social networks. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*.
- Ye, M., Yin, P., Lee, W.-C., & Lee, D.-L. (2011). Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 325–334. ACM.
- Yin, H., Sun, Y., Cui, B., Hu, Z., & Chen, L. (2013). LCARS: a location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 221–229. ACM.
- Yin, H., Wang, W., Wang, H., Chen, L., & Zhou, X. (2017). Spatial-aware hierarchical collaborative deep learning for POI recommendation. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 29(11):2537–2551.
- Yu, F., Che, N., Li, Z., Li, K., & Jiang, S. (2017). Friend recommendation considering preference coverage in location-based social networks. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 91–105. Springer.
- Yuan, Q., Cong, G., Ma, Z., Sun, A., & Thalmann, N. M. (2013). Time-aware point-of-interest recommendation. In *Proceedings of the 36th International ACM SIGIR*

- Conference on Research and Development in Information Retrieval (SIGIR)*, pages 363–372. ACM.
- Yuan, Q., Cong, G., & Sun, A. (2014). Graph-based point-of-interest recommendation with geographical and temporal influences. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 659–668. ACM.
- Yuan, Q., Cong, G., Zhao, K., Ma, Z., & Sun, A. (2015). Who, where, when, and what: A nonparametric bayesian approach to context-aware recommendation and search for twitter users. *ACM Transactions on Information Systems (TOIS)*, 33(1):2.
- Zhang, J., Kong, X., & Yu, P. S. (2014a). Transferring heterogeneous links across location-based social networks. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, pages 303–312. ACM.
- Zhang, J.-D. & Chow, C.-Y. (2013). iGSLR: personalized geo-social location recommendation: a kernel density estimation approach. In *Proceedings of the 21st ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 334–343. ACM.
- Zhang, J.-D. & Chow, C.-Y. (2015). GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 443–452. ACM.
- Zhang, J.-D. & Chow, C.-Y. (2016). CRATS: an lda-based model for jointly mining latent communities, regions, activities, topics, and sentiments from geosocial network data. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 28(11):2895–2909.
- Zhang, J.-D., Chow, C.-Y., & Li, Y. (2014b). Lore: Exploiting sequential influence for location recommendations. In *Proceedings of the 22nd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 103–112. ACM.

- Zhang, J.-D., Chow, C.-Y., & Li, Y. (2015a). iGeoRec: A personalized and efficient geographical location recommendation framework. *IEEE Transactions on Services Computing*, 8(5):701–714.
- Zhang, J.-D., Chow, C.-Y., & Zheng, Y. (2015b). ORec: An opinion-based point-of-interest recommendation framework. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM)*, pages 1641–1650. ACM.
- Zhang, S., Yao, L., & Sun, A. (2017a). Deep learning based recommender system: A survey and new perspectives. *arXiv preprint arXiv:1707.07435*.
- Zhang, Y., Dai, H., Xu, C., Feng, J., Wang, T., Bian, J., Wang, B., & Liu, T.-Y. (2014c). Sequential Click Prediction for Sponsored Search with Recurrent Neural Networks. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, pages 1369–1375.
- Zhang, Y., Zhang, H., Zhang, M., Liu, Y., & Ma, S. (2014d). Do users rate or review?: Boost phrase-level sentiment labeling with review-level sentiment classification. In *Proceedings of the 37th ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR)*.
- Zhang, Z., He, Q., Gao, J., & Ni, M. (2018). A deep learning approach for detecting traffic accidents from social media data. *Transportation research part C: emerging technologies*, 86:580–596.
- Zhang, Z., Li, C., Wu, Z., Sun, A., Ye, D., & Luo, X. (2017b). Next: a neural network framework for next POI recommendation. *arXiv preprint arXiv:1704.04576*.
- Zhao, K., Cong, G., Yuan, Q., & Zhu, K. Q. (2015). SAR: A sentiment-aspect-region model for user preference analysis in geo-tagged reviews. In *IEEE 31st International Conference on Data Engineering (ICDE)*.
- Zhao, S., King, I., & Lyu, M. R. (2013). Capturing geographical influence in POI recommendations. In *International Conference on Neural Information Processing*, pages 530–537. Springer.

- Zhao, S., King, I., & Lyu, M. R. (2016). A Survey of Point-of-interest Recommendation in Location-based Social Networks. *arXiv preprint arXiv:1607.00647*.
- Zheng, V. W., Zheng, Y., Xie, X., & Yang, Q. (2010). Collaborative location and activity recommendations with GPS history data. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pages 1029–1038. ACM.
- Zheng, V. W., Zheng, Y., Xie, X., & Yang, Q. (2012). Towards mobile intelligence: Learning from GPS history data for collaborative recommendation. *Artificial Intelligence*, 184:17–37.
- Zheng, Y. & Xie, X. (2011). Learning travel recommendations from user-generated GPS traces. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(1):2.