MOBILE INTELLIGENCE ANALYTICS FOR URBAN SMART LIVING

by

ZIJUN YAO

A Dissertation submitted to the

Graduate School-Newark

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Management

written under the direction of

Dr. Hui Xiong

and approved by

_____

_____

_____

_____

Newark, New Jersey

May 2018

ABSTRACT OF THE DISSERTATION

MOBILE INTELLIGENCE ANALYTICS FOR URBAN SMART LIVING

By ZIJUN YAO

Dissertation Director: Dr. Hui Xiong

Today, as the sensing technology and mobile computing have been popularized, a variety of mobile data related to human mobility and urban geography have been accumulated in a large amount. This type of data comprehensively records the fine-grain events of our cities through "4W" aspects of information: What happened? Where it happened? When it happened? And who did it? By proper analysis, this data can be a rich source of mobile intelligence to support various location-based and real-time decision-making solutions for a broad range of urban smart living applications. Indeed, mobile intelligence analytics plays an important role in urban life because city residents often make choices under more uncertainty and can benefit more from personalized advice based on their preferences and contexts. Therefore, it is especially meaningful to develop data-driven methodologies which can effectively and efficiently guide users to make optimal decisions to achieve the goal of urban smart living.

In this dissertation, we aim to address the unique challenges of urban smart living in mobile and pervasive business environments from both theoretical and practical perspectives. Specifically, we first develop a safety-aware house ranking system by considering the impact of neighborhood criminal offenses on house values. The proposed framework extracts features regarding community safety conditions of different

houses, and utilizes multiply safety features to rank houses by unit value. To enhance safety-aware ranking, we introduce major characteristics of house profile to control the similarity between houses during pair-wise ranker learning. The experimental results show that the proposed method substantially outperforms the baseline learn-to-rank methods for safety-aware house ranking. Moreover, in the second study, we introduce an effective point-of-interest (POI) recommender system to consider the temporal compatibility between POI popularity and user regularity. We propose to use the massive human mobility data to profile the temporal pattern of POI popularity, and infer the regularity pattern of users based on the POI they visited through a modeling intuition "you are where you go". We demonstrate the effectiveness of the proposed model through the extensive experiments on the real-world datasets of New York City. Finally, we introduce a zone embedding framework to identify the urban functions of city zones by studying massive origin-destination transportation data. We focus on exploiting the idea of word embedding in natural language processing domain to learn zone functions in urban computing domain by developing a novel analog from word co-occurrence to zone co-occurrence using human mobility patterns. To incorporate the contexts of human mobility in our framework, we develop the directed and temporal co-occurrence for considering mobility direction and time, and the different importance of co-occurrence for considering travel distance and zone attractiveness. The evaluation validates the proposed method and shows that the learned embeddings can comprehensively capture the urban functions of city zones. From the three studies, we conclude that mobile intelligence analytics can be powerful at disclosing patterns, relations and hidden knowledge, and it is promising

to explore the power of mobile intelligence to provide location-based insights, and ultimately, to improve business performance.

ACKNOWLEDGEMENTS

I would like to express my great appreciation to all the people who provided me with tremendous support and help during my Ph.D. study.

First, I would like to express my deep gratitude to my advisor, Prof. Hui Xiong, for his continuous support, guidance, patience, and encouragement, which are necessary to survive and thrive the graduate school and beyond. I thank him for generously giving me motivation, support, time, assistance, opportunities, and friendship; for teaching me how to identify key problems with impact, and present and evaluate ideas. He helped making me a better writer, speaker, and scholar.

I also sincerely thank my other committee members: Prof. Guiling Wang, Prof. Spiros Papadimitriou and Prof. Xiaodong Lin. All of them not only provide constructive suggestions and comments on my work and this dissertation, but also offer numerous support and help in my career development, and I am very grateful for them.

I also want to thank my colleagues and collaborators. I would like to thank Prof. Yanjie Fu at Missouri University of Science and Technology, Dr. Bin Liu at IBM Research for years of research collaboration. Thanks are also due to Dr. Shoshana Loeb, Dr. K.C. Lee, Dr. Weicong Ding, Dr. Yifan Sun, Dr. Nikhil Rao, Dr. Deguang Kong. Dr. Xiao Bai and Dr. Miao Lu for their mentorship in my industrial

internships. It was a great pleasure working with all of them. I also owe a hefty amount of thanks to my colleagues and friends Yong Ge, Keli Xiao, Zhongmou Li, Chuanren Liu, Meng Qu, Constantine Alexander Vitt, Jingyuan Yang, Can Chen, Hao Zhong, Farid Razzak, Qingxin Meng, Junming Liu, Mingfei Teng, Yanchi Liu, Huang Xu, Xinjiang Lu, Renjun Hu, Haochao Ying, Jingjing Gu, Fei Yi, Yuanbo Xu, Xiao Zhang, Xiaoyi Deng, Kun Yuan, Fuzhen Zhuang, Ya Yang, and Xiaolong Deng for their help and valuable suggestion.

I would like to acknowledge the Department of Management Science and Information Systems (MSIS) for supplying me with the best equipment and working environment that helped me to accomplish much of this work. I want to paricularly thank Mrs. Luz Kosar and Mr. Goncalo Filipe for all of their assistance.

Finally, I would like to thank my wife Dr. Chao Chen, my boys Kevin and Kai, my father Prof. Guoxiang Yao and all our family members for their love, support and understanding. Without their encouragement and help, I would never be where I am now. Thank you for making my life so special and full of joy.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Recent years have witnessed an unprecedented advancement in the technologies of mobile computing and positioning. Mobile devices equipped with GPS, WiFi and LTE have enable us to collect a massive amount of location traces from multiple sources, such as public safety, social network and city transportation. These mobile data are fine-grained on location, time and description, and comprehensively record the major human activities in urban areas. Given this rich information resource, we have a tremendous opportunity to understand the hidden knowledge behind the data. As a result, we can deliver valuable intelligence for supporting location-based and real-time decision-making in various smart living scenarios.

Urban smart living refers to the tasks for helping users make optimal decisions in various urban scenarios. For example, considering the community safety conditions, a potential home buyer needs to find out which house has a higher unit value. For another example, based on the personal preference and context, a potential customer needs to decide which restaurant can meet his expectation to the most degree. Since this type of smart living applications can effectively and efficiently provide personalized, location-based, and real-time services, mobile intelligence analytics is expecting to be hugely beneficial for urban life. Alone this line, it is important to investigate the uniqueness that distinguishes mobile intelligence analytics from traditional intel-

ligence analytics. In this dissertation, we aim to exploit the patterns and knowledge of location data to contributes urban smart living in mobile and pervasive business environments from both theoretical and practical perspectives.

1.1 Research Motivation

In the first study, we develop a safety-aware house ranking system. We aim to access the local safety conditions with careful designed features and build the relationship between house value and neighborhood safety. Although there are a few studies which have investigated the impacts of crime on house appraisals (Gibbons, 2004; Pope, 2008; Linden & Rockoff, 2008), they suffer from two limitations: 1) naive crime statistics (e.g., counting crime cases) which fails to develop sophisticated crime evaluation features to get in-depth understanding of community safety; 2) traditional appraisal models (e.g., Hedonic regression) which include multiply influencing aspects in house appraisal function, mask the impacts of community safety. Unlike prior studies, we want to comprehensively learn the local criminal offense data with consideration such as repeatability and temporality. Moreover, we want to solely focus on the influence of community safety on house appraisal. To tackle these two research limitations, it is critical to generate effective features from crime data, and systematically modeling the impacts of community safety without effects of other aspects, such as neighborhood income level and rating of nearby schools.

In the second study, we develop a mobile recommender system to consider temporal matching between user regularity and point-of-interest (POI) popularity. Unlike traditional interest-oriented merchandise recommendation (e.g., books, films, etc.),

POI recommendation is more complex and challenging due to the contextual characteristics of location-based service. For example, besides personal interest, the recommended POI should also fit the user on time aspect. Recent studies have considered temporal influences on POI recommendation, such as time-aware POI recommendation which recommends different POIs to users at different time. For example, (Gao, Tang, Hu, & Liu, 2013a) applies user-item matrix factorization for each time slot and assumes every user has similar preferences in consecutive time slots for regularization. (Q. Yuan, Cong, Ma, Sun, & Thalmann, 2013a) computes user similarity via the same spatio-temporal check-ins in the past and conducts a user-based recommendation approach. (Xiong, Chen, Huang, Schneider, & Carbonell, 2010) adds the time dimension to user-item matrix and applies tensor factorization for recommendations. However, these studies overlooked that the temporal preference of users is effected by their regularity, and the popularity of POIs is varying according to locations and categories. They also didn't consider the influence of temporal compatibility between users and POIs. Moreover, they solely depended on the time input of history check-ins, and suffered from the sparsity problem of check-in data. In order to address these limitations, it is appealing to incorporate the temporal compatibility between user regularities and POI popularities into POI recommendation, and utilize human mobility data to boost recommendation performances.

In the third study, we develop a zone function identification system to quantitatively profile the similarity between city zones in terms of urban functions. While the literature has shown promising effectiveness of analyzing massive positioning data for urban exploration (Cranshaw, Schwartz, Hong, & Sadeh, 2012; J. Yuan, Zheng,

& Xie, 2012; Z. Cheng, Caverlee, Lee, & Sui, 2011; Silva, de Melo, Almeida, Salles, & Loureiro, 2012), there are limited studies aiming to provide an integrated and principled approach to the representation learning of city zones in terms of urban functions. In this study, we aim to propose an effective solution to learn the distributed and low-dimensional embeddings of city zones. Zones with similar urban functions are geometrically closer in the embedding space. Using zone embeddings, we are able to identify functional regions of cities which consist of several zones with similar functions. Furthermore, many analytic models can be empowered by using these extracted representations as inputs. To achieve this goal, it is important to tackle two critical challenges toward learning effective zone representations by inferring urban functions through inter-zone human mobility rather than intra-zone POI visiting activity, and effectively exploiting spatio-temporal contexts of human mobility patterns for effective zone embedding.

1.2 Contribution

In this dissertation, we study the unique characteristics of urban smart living tasks and demonstrate how to develop mobile intelligence analytic methods in different application scenarios. We list the research contribution of each study as follows:

- **House ranking system:** We identify two categories of discriminative crime evidences that comprehensively describe community safety. The first category is based on crime severity which focuses on property losses led by different burglary crimes. The second category is based on temporal correlation which considers the correlation of crimes to learn community safety. Moreover, we propose a

ranking model to understand the impacts of community safety. Since mostly investors want to compare houses rather than knowing the exact value, ranking houses from the perspective of community safety can help investors differentiate low value houses from the other houses. Specifically, we propose a house safety-aware (HSA) ranker which combines all extracted community safety features to rank houses according to house values. In addition, we integrate distinctive house profile such as neighborhood income, nearby school rating and house build year to differentiate the comparability of house pairs in pair-wise ranking objective function. Last, we optimize the ranking model by jointly preserving the house ranking consistency and maximizing the value prediction accuracy. Last, we validate our method with real-world dataset. Experiments shows that our proposed ranking method not only provides better explanations in safety impacts on house values, but also demonstrates a substantial improvement in ranking accuracy compared with baseline ranking methods (Yao, Fu, Liu, & Xiong, 2016).

- **Mobile recommender system:** We propose a Temporal Matching Poisson Factorization Model (TM-PFM) to profile the popularity of POIs, model the regularity of users, and incorporate the temporal matching between users and POIs into overall recommending consideration. We first present a new framework to profile a time-varying popularity of POIs (e.g., hourly visiting change in a day). Traditional methods usually capture this temporal variation by counting POIs' check-in frequencies therefore suffer from check-in data sparsity. We

utilize heterogeneous human mobility data to evaluate POI popularity. The benefits of employing human mobility data include (i) it is more abundant and less biased than check-in data, and (ii) it reveals which areas are currently active which is a determinant of POI popularity. Moreover, we further analyze POIs by categories and adopt a mixture model to obtain the final POI temporal popularity pattern. Secondly, except some particular events (e.g., parties, concerts), people's availability is usually determined by their routines, thus there is a predictable regularity. Therefore, we consider temporal regularity of each user which describes their daily regular availability for POI exploration. We learn the latent regularity patterns of users by finding the best match with the popularity patterns of visited POI based on check-in frequencies. We validate our proposed method with real-world LBSN check-in and human mobility datasets. The effectiveness of temporal matching in POI recommendation is proven by extensive experiments and a substantial improvement in recommendation performances over baseline methods is demonstrated (Yao, Fu, Liu, Liu, & Xiong, 2016).

- **Zone function identification:** We present a novel human mobility based zone embedding framework to represent urban functions with distributed and low-dimensional vectors. We treat the origin-destination pair of a mobility pattern as a co-occurrence of two zones for learning zone embeddings. Since urban functions are also jointly reflected by mobility direction, and departure/arrival time, we need the embedding method to take into account "leaving for" and

"arriving from" at "different time" for modeling a zone co-occurrence. We define a set of human mobility events which contain zone, time and status of mobility patterns, to serve as embedding "contexts" of target zones, for incorporating zone co-occurrence with spatio-temporal characteristics. Moreover, during the learning of zone embeddings, we give different importance to different co-occurrences by calculating the travel demand of origin-destination pairs with destination attraction (e.g., total mobility pattern arrivals) and travel distance (e.g., average mobility pattern length) information. Finally, we conduct extensive experiments with real-world urban datasets of New York City to show the effectiveness of the proposed method (Yao, Fu, Liu, Hu, & Xiong, 2018).

1.3 Overview

Chapter 2 addresses the community safety modeling for house ranking using spatio-temporal criminal offense data. Two types of features are extracted to effectively identify local conditions of safety. A novel ranking model which considers the profile similarity between houses is proposed to achieve optimal safety-aware ranking performance.

Chapter 3 presents a POI recommender system which focuses on the temporal matching between the POI popularity and the user regularity. Massive human mobility data is introduced to profile the popularity patterns of POIs. Meanwhile, we model users' temporal regularity by developing a user-POI temporal matching function into the overall preference estimation. Last, we propose a factorization based POI recommendation algorithm which combines the user-POI general interest and user-POI

temporal matching degree to improve the overall quality of recommendation.

Chapter 4 presents a zone embedding framework to identify zone functions using massive inter-zone transportation data. The framework learns the embedding of zones from its associated zones by considering the frequency of different origin-destination mobility patterns. To incorporate the contexts of mobility patterns, we define a set of human mobility events which contain zone, time and status of mobility patterns, to serve as co-occurrence of target zones. In addition, during the learning of zone embeddings, we give different importance to different co-occurrences by calculating the travel demand of origin-destination pairs with destination attraction (e.g., total mobility pattern arrivals) and travel distance (e.g., average mobility pattern length) information.

CHAPTER 2

THE IMPACTS OF COMMUNITY SAFETY ON HOUSE RANKING

It is well recognized that community safety which affects people's right to live without fear of crime has considerable impacts on housing investments. Housing investors can make more informed decisions if they are fully aware of safety related factors. To this end, we develop a safety-aware house ranking method by incorporating community safety into house assessment. Specifically, we first propose a novel framework to infer community safety level by mining community crime evidences from rich spatio-temporal historical crime data. Then we develop a ranking model which fuses multiply community safety features to rank house value based on the degree of community safety. Finally, we conduct a comprehensive evaluation of the proposed method with real-world crime and house data. The experimental results show that the proposed method substantially outperforms the baseline methods for house ranking.

2.1 Introduction

Community safety describes the degree that people live without fear of crime, such as the risk of being victimized in burglary, robbery, or assault. It has become a fundamental buying factor of houses nowadays. Community safety issues can severely damage the value of a house by: 1) endangering occupants and properties; 2) degrading living and business environments, e.g., people are less likely to rent for living or

business; 3) obstructing the development of the area, e.g., new houses or infrastructure are less likely to be built nearby. Therefore, from the perspective of investors, it is requisite to be aware of potential community safety issues.

Empirical studies have confirmed the importance of community safety factors for house appraisals. For example, a standard deviation increase in the local density of property crime causes a 10% decrease in the price of an average property in London (Gibbons, 2004); the move-in of a sex offender leads to a 2.3% fall in nearby housing prices in Hillsborough County, Florida (Pope, 2008). These evidences show that the value of houses can be significantly influenced by community safety issues.

Motivated by the above, it is appealing to provide a tool for investors to rank house values based on the degree of community safety. For investigating the impacts of community safety on house values, historical crime especially house burglary (also called break-in) is a valuable resource for the following reasons: 1) burglary directly threats houses, 2) burglary is commonly spread in all locations, 3) burglary provides sufficient historical cases for investigation. With rapid advances in positioning technology, data with fine-grained locations such as coordinates of crime records is now available. This allows us to appraise houses via their neighboring crimes which make direct impacts.

Although there are a few studies which have investigated the impacts of crime on house appraisals (Gibbons, 2004; Pope, 2008; Linden & Rockoff, 2008), they suffer from two limitations: 1) naive crime statistics (e.g., counting crime cases), which can be improved by sophisticated crime analysis to get in-depth understanding of community safety; 2) traditional appraisal models (e.g., Hedonic regression) which

include multiply influencing aspects in house appraisal function, mask the impacts of community safety. Unlike prior studies, we want to comprehensively learn local safety levels by employing factors not limited to naive crime statistics. Moreover, we want to solely focus on the influence of community safety on house appraisal. Therefore, we tackle two research challenges in this chapter, **Challenge 1**: what crime analysis can be done to generate in-depth understanding of community safety; **Challenge 2**: how to systematically model the impacts of community safety on house values without effects of other aspects, such as neighborhood income level and rating of nearby schools?

For **Challenge 1**, we identify two categories of discriminative crime evidences that comprehensively describe community safety. The first category is based on crime severity which focuses on property losses led by different burglary crimes. Since available crime information does not provide actual losses explicitly, we derive evidences to infer the severity of crimes implicitly: occurrence address evidence and occurrence time evidence of crimes. The second category is based on temporal correlation which considers the correlation of crimes to learn community safety. The crime temporal correlation reflects an important phenomenon called near repeat in criminology, which implies degraded community safety and increased victimization risk (Ratcliffe & Rengert, 2008; Kleemans, 2001). Therefore, we mine near repeat series to discover temporal correlations of crimes around houses. Based on near repeat series, we extract evidences to detect temporal correlations: series size evidence, series length evidence and series intensity evidence.

For **Challenge 2**, we propose a ranking model to understand the impacts of com-

munity safety. Since mostly investors want to compare houses rather than knowing the exact value, ranking houses from the perspective of community safety can help investors differentiate low value houses from the other houses. Therefore, we propose a house safety-aware (HSA) ranker which combines all extracted community safety features to rank houses according to house values. In addition, we integrate distinctive house profile such as neighborhood income, nearby school rating and house build year to differentiate the comparability of house pairs in pair-wise ranking objective for enhancing ranking accuracy of community safety features. Last, we optimize the ranking model by jointly preserving the house ranking consistency and maximizing the value prediction accuracy.

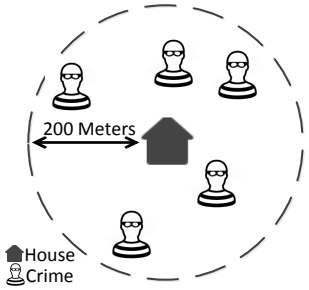
In summary, in this chapter we strategically leverage rich spatio-temporal crime data for effective house ranking. We highlight our key contributions as follows:

- We present an advanced crime analysis (e.g., crime evidence mining) to comprehensively infer the community safety by exploiting historical crime data.
- We develop a safety-aware ranking model by incorporating the comparability of house pairs into the optimization of pair-wise ranking objective, in a way that we better model the impact of community safety without the effects of other aspects, and thereby enhance the ranking accuracy.
- We validate our method with real-world dataset. Experiments shows that our proposed ranking method not only provides better explanations in safety impacts on house values, but also demonstrates a substantial improvement in ranking accuracy compared with baseline ranking methods.

2.2 Problem Definition

Usually, when people use house values to indicate the quality and benefits of a houses, they actually mean unit values (e.g., values per square footage) since total values are also affected by floor areas of houses. Moreover, from the perspective of investors, ranking houses in terms of their quality and benefits, instead of predicting absolute appraised values, is more needed for making investment decisions. Therefore, to provide investors with a tool to compare houses, we rank houses based on unit values by taking community safety degrees into account. In the rest of this chapter, "house value" will be used to represent the unit value of a house.

We are given a set of I houses $H = \{h_1, h_2, ..., h_I\}$ where each house has a location (e.g., latitude and longitude) and corresponding house values $Y = \{y_1, y_2, ..., y_I\}$ where y_i denotes the value per square footage in dollar of house h_i. We are also given profiles of houses in H where each house has several house characteristics such as neighborhood income, nearby school rating and house build year. Last, we are given the complete historical house burglary records of the area, each burglary crime is denoted by $< loc, add, t >$, which has a location loc, an address add and an occurrence timestamp t. The task is training a model to rank a testing set of houses in an ascending order according to their house value by exploiting historical burglary data. We propose to accomplish the task with a two-step framework: 1) extracting and aggregating community crime evidences for learning the community safety of different houses from historical crime data, 2) ranking houses by incorporating the impacts of community safety.



OFFENSE_ID:	20102001432202OO
OFFENSE_TYPE_ID:	burglary-residence-by-force
OCCURRENCE_DATE:	2010-04-28 08:29:59
INCIDENT_ADDRESS:	2909 N IVANHOE ST, DENVER
LATITUDE:	39.7586462
LONGITUDE:	-104.9214312

(a) (b)

Figure 2.1: Examples of (a) surrounding crimes of a house, and (b) a crime record.

2.3 Community Crime Evidence Extraction

In this section, we study how to extract and aggregate community crime evidence for learning community safety features of houses. Figure 2.1a shows an example how we collect five historical crimes around house h_i to form its crime sequence $C_i = \{c_1, c_2, ..., c_5\}$. To get the crime sequence of a house, we collect all the historical crimes which occurred within d meters (e.g., 200 meters) of the house and order the crimes by occurrence time from oldest to newest. We mine crime evidences for house h_i based on its specific crime sequence C_i. Figure 2.1b shows a sample of crime records. In the following, we will extract crime evidences in two categories: 1) crime severity, and 2) crime temporal correlation. Finally we aggregate crime evidences by each evidence type to generate house-level community safety features.

2.3.1 Category Based on Crime Severity.

Crime severity indicates the damage level made by a crime. For burglary, it is usually determined by the property losses in a crime. However, we do not have the explicit

description about the actual losses of valuables from available crime information. Therefore, we attempt to infer the severity of a burglary implicitly from other crime information. To this end, we propose to mine two evidences for a burglary to assess its severity.

Occurrence Address Evidence.

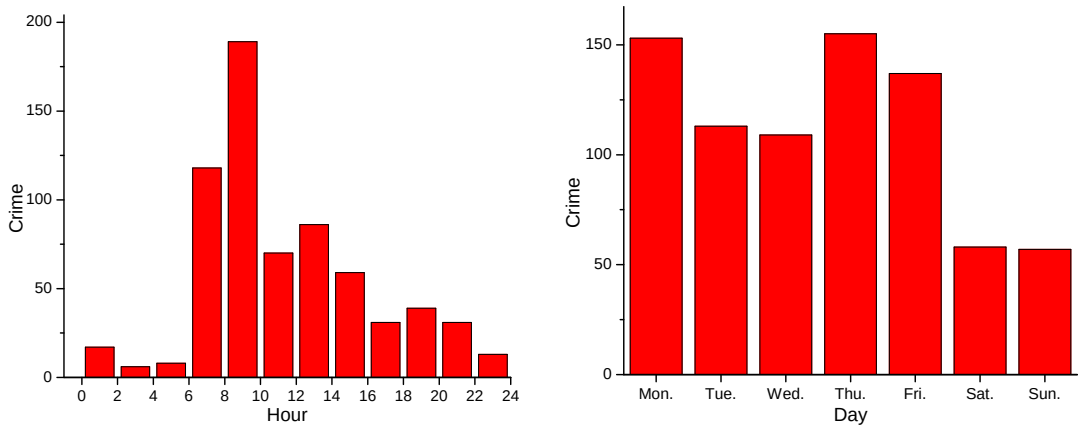
Knowing the detailed occurrence address of burglaries, we can retrieve the appraisal of the victimized houses (e.g., 265900 dollars). Intuitively, the loss led by a burglary is proportional to the appraisal of the victimized house. Higher appraisal usually means the victimized house has more rooms or the house is more luxury. Either possibility gives burglars a higher chance to collect more valuables. Based on this intuition, we can infer the possible loss in a burglary crime by knowing the appraisal of the victimized house via the burglary address. Therefore, we propose the first type of evidence:

$$E_1(c) = Appraisal(add_c), \tag{2.1}$$

where c is a burglary crime. add_c is the occurrence address of burglaries. *Appraisal* denotes the appraisal of the victimized house located at burglary address.

Occurrence Time Evidence.

The occurrence time of crimes is another information for inferring the severity of a burglary. House burglary has a unique character that burglars always have to make sure there is no occupant at home to commit crimes. Since the most common cause of people leaving home is for work or school, the confidence for committing burglaries should be strongly correlated to people's working schedule. In Figure 2.2a



(a) (b)

Figure 2.2: Hourly and daily number of burglaries in a residential area during 2009-2014.

which shows the number of burglary by every two hours in workdays, most of the burglaries concentrate in the time range from 6:00 to 16:00 which is the time people are usually out for work. In Figure 2.2b which shows the number of burglary by every day in weeks, burglaries happened much more frequently in workday (Monday to Friday) compare to in weekend (Saturday, Sunday). Based on this observation, during different time slots, the general confidence of burglars for committing a crime should be different.

We propose a *time entropy* to model the burglary confidence of different time slots by analyzing how many houses were victimized during the time slots. Specifically, we define a time slot in two dimensions: 1) two hours of a day and 2) workday or weekend. For example, a time slot can be 8:00 to 10:00 in every workday. Then let k denotes a time slot, $C_{k,i}$ is the set of burglaries occurred in kth time slot at ith house, and C_k is the set of all burglaries occurred in kth time slot. The probability

that a randomly picked burglary occurred in kth time slot belongs to the ith house is $P_{k,i} = |C_{k,i}|/|C_k|$. We define the Shannon entropy of time slot k as follow:

$$Entropy(k) = -\sum_{i:P_{k,i}\neq 0} P_{k,i} \cdot logP_{k,i}. \tag{2.2}$$

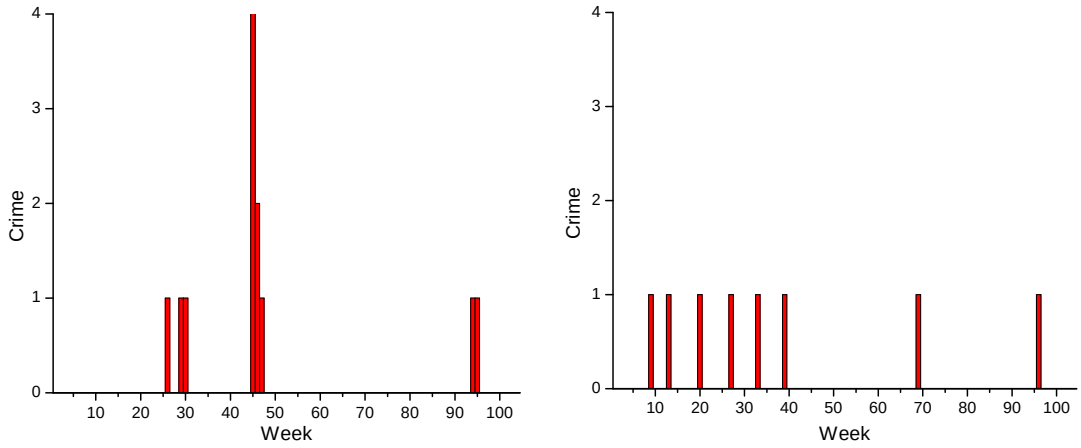
A higher time entropy implies a confident time slot during which houses' occupants are more possible to be not at home. On the other hand, a lower time entropy indicate a worse time slot during which occupants are less possible to be not at home. Therefore, we infer the severity of a burglary by the entropy of the time slot it occurred in. If a burglary occurred during a high entropy time slot, we consider it may result more losses since burglars can take longer time for searching valuables and have fewer chance to be discovered. We propose the second type of evidence:

$$E_2(c) = Entropy(ts_c), \tag{2.3}$$

where c is a burglary crime and ts_c is the time slot during which the burglary occurred.

2.3.2 Category Based on Temporal Correlation.

Given a crime sequence of a house, the temporal correlation evidence aims to consider the temporal proximity among burglaries. By analyzing temporal correlation of crimes, we can infer the local community safety. Figure 2.3 shows the weekly statistics of burglary happened within 400 meters of two houses respectively during 2013-2014. Figure 2.3a shows the burglary near a low value house, we can see that many burglaries tend to cluster together temporally. On the contrary, Figure 2.3b shows burglary near a normal value house, we can see that these crimes behave more independently. This temporal correlation can be explained by near repeat phenomenon in criminology



(a) Low value house

(b) Normal value house

Figure 2.3: Weekly number of burglaries near two houses during 2009-2010.

research (Kleemans, 2001; Ratcliffe & Rengert, 2008). Usually, the burglaries which repeatedly occurred in the same area within short interval means that they are very likely to be committed by the same burglars. Preference by returning burglars implies worse community safety issues and brings higher crime risks to the area (Hearnden & Magill, 2004). Therefore, we can expect that a stronger temporal correlation of crimes leads to worse community safety of houses.

For capturing the temporal correlation, we propose to mine *near repeat series* for houses. Basically, a near repeat series is mined from the crime sequence of a house and it is a set of crimes in which every crime happened very shortly after the previous one except the first. A house can have none to multiple near repeat series. Formally, we define a near repeat series as follows:

Definition 1 *A* ***near repeat series*** s *of a house* h_i *consists of* N *adjacent crimes* $s = \{c_1, c_2, ..., c_N\}$ *in which every crime* c_n *belong to the* h_i*'s crime sequence:* $c_n \in C_i$

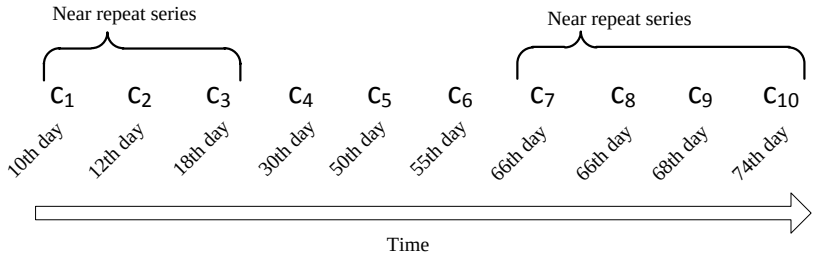


Figure 2.4: An example of near repeat series mining.

and there is no other near repeat series s' makes that $s \subseteq s'$. Every near repeat series meets two conditions: 1) the number of crimes in s should be not less than a minimum size threshold θ: $|s| \geqslant \theta$, and 2) the interval between any two adjacent crimes should be not longer than a maximum time threshold τ: $\forall n \in [2, N]$, $t_{c_n} - t_{c_{n-1}} \leqslant \tau$ where t_c represents the timestamp of a crime c.

Figure 2.4 shows an example of mining near repeat series. Given a crime sequence $C_i = \{c_1, ..., c_{10}\}$ with their occurrence date of a house h_i, c_1 is the oldest crime while c_{10} is the latest one. Suppose we define the maximum time threshold to be 7 days while the minimum size threshold to be 3 crimes. We can find the first near repeat series to be $s_1 = \{c_1, c_2, c_3\}$ since $t_{c_2} - t_{c_1} = 2$ and $t_{c_3} - t_{c_2} = 6$. Because $t_{c_4} - t_{c_3} = 12$, repeat series s_1 stops at c_3. Although the interval between c_5 and c_6 is less than 7 days, they can not find c_4 or c_7 to reach the minimum size threshold. Last, from c_8 to c_{10}, each of them has a less than 7 days interval from its previous one, therefore we find the second near repeat series $s_2 = \{c_7, c_8, c_9, c_{10}\}$ by qualified intervals of adjacent crimes and qualified series size.

We propose three evidences based on near repeat series to assess local community

safety.

Series Size Evidence.

Series size evidence which measures the number of crimes in a near repeat series, is a significant indicator for community safety situation. For a house with less safety issues, the nearby crimes should act independently and they are unlikely to form large size near repeat series. If a near repeat series has large size, there is high probability that the area of house has serious safety issues. Therefore, we propose the third evidence:

$$E_3(s) = |s|, \tag{2.4}$$

where s is a near repeat series consists of crimes.

Series Length Evidence.

Series length evidence measures the length of period a series lasts in order to learn the community safety situation. As the near repeat series lasts longer, it is more difficult for burglars to commit repeat crime because of increasing police attention. If near repeat series lasts long, it means that the area is promising for burglars, therefore it indicates a worse community safety situation for local houses. We propose the forth evidence:

$$E_4(s) = t_{c_N} - t_{c_1} + 1, \tag{2.5}$$

where t_{c_1} and t_{c_N} represent the date of the first crime c_1 and the last crime c_N of the near repeat series s.

Series Intensity Evidence.

Series intensity uses the shortest interval occurred in a near repeat series to learn the community safety situation. An intensive near repeat series means that at least two crimes of it have very short interval thus shows a dangerous sign of the area. For example, given two size-3 near repeat series with similar length, the first series has the interval $\{0-day, 7-day\}$ while the second series has the interval $\{3-day, 4-day\}$. The first series has a higher intensity since the first two crimes of it occurred in a the same day with 0-day interval. The second series has a lower intensity since the shortest interval in the second series is 3-day. Therefore, we propose the fifth evidence:

$$E_5(s) = \max_{2 \leq n \leq N} \{\tau - (t_{c_n} - t_{c_{n-1}}) + 1\}, \tag{2.6}$$

where $t_{c_n} - t_{c_{n-1}}$ represents the interval days between crime c_n and its previous adjacent crime c_{n-1} of series s, τ represents the maximum time threshold.

2.3.3 Evidence Aggregation.

We aggregate evidences by types for generating house-level community safety features. As we introduced, for a house h_i, we have a crime sequence $C_i = \{c_1, c_2, ..., c_{K_i}\}$ and a set of near repeat series $S_i = \{s_1, s_2, ..., s_{J_i}\}$. The evidence belongs to crime severity category $(E_1(c), E_2(c))$ is extracted by crime cases while the evidence belongs to temporal correlation category $(E_3(s), E_4(s), E_5(s))$ is extracted by near repeat series. Therefore, for a house h_i, we will have a community safety feature vector

$X_i = \{x_{i1}, x_{i2}, ..., x_{i5}\}$ as following:

$$x_{im} = \begin{cases} \sum_{c \in C_i} E_m(c) & \text{if } m \in (1,2) \\ \sum_{s \in S_i} E_m(s) & \text{if } m \in (3,4,5) \end{cases} \tag{2.7}$$

2.4 A Safety-Aware House Ranking Model

In this section, we propose a House Safety-Aware (HSA) model which ranks houses by incorporating the degrees of community safety.

2.4.1 Model Specification.

Let us define the input of the model to be X_i, y_i and P_i for a house h_i, where X_i denotes M-size vector of community safety features, y_i denotes the ground truth of house value and P_i denotes the L-size vector of house profile characteristics. We want to train a function $f_i = W^T X_i$ which formulates the house value by having $y_i = f_i + \epsilon$, where W denotes the vector of weights for safety features and ϵ denotes the error term which subjects to Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Thus, we have $y_i \sim \mathcal{N}(f_i, \sigma^2)$.

2.4.2 Objective Function.

We propose to jointly model the accuracy of house value prediction and the consistency of house ranking prediction in an objective function. Let the parameter for estimation to be W, model hyperparameter to be $\Phi = \{\sigma^2, b^2\}$, observed ground truth to be $\mathcal{O} = \{Y, R\}$ where Y and R denote the value and rank of houses. Then we have the posterior probability:

$$Pr(W|\mathcal{O}, \Phi) = Pr(\mathcal{O}|W, \Phi) Pr(W|\Phi). \tag{2.8}$$

First, let us address the probability of observed data. We model $Pr(\mathcal{O}|W,\Phi)$ as a joint probability of house value prediction $Pr(Y|W,\Phi)$ and house ranking prediction $Pr(R|W,\Phi)$. **Modeling prediction accuracy**, we use $Pr(Y|W,\Phi) = \prod_{i=1}^{I} \mathcal{N}(y_i|f_i,\sigma^2)$ to ensure value prediction accuracy of houses. **Modeling ranking consistency**, we adopt a pair-wise probability to ensure ranking correctness of all house pairs. Suppose the I houses has already been ranked by house value in ascending order. Given two index i, j which has $i < j$, we should always have $y_i < y_j$ and $h_i \rightarrow h_j$ which means the rank of house h_i is higher than the rank of h_j in ground truth. Therefore, we use $Pr(R|W,\Phi) = \prod_{i=1}^{I-1} \prod_{j=i+1}^{I} Pr(h_i \rightarrow h_j|W,\Phi)$ to represent the probability that h_i is correctly ranked higher than h_j by model for all house pairs. We adopt Sigmoid function to represent the probability of pair-wise ranking consistency: $Pr(h_i \rightarrow h_j) = \frac{1}{1+exp(-(f_j - f_i))}$.

Integrating house profile into ranking consistency. House is a kind of distinctive property which has various characteristics, such as the house profile shown in Table 2.1. If two houses have too large differences in house profile, their value difference does not help the model to learn the impacts of community safety. For example, if two houses have too different build year (e.g., 1950 vs. 2010), then the impact of community safety on house value may be overridden by the impact of build year. Therefore, including the rank observation of dissimilar house pair in optimization objective will jeopardize the ranking prediction capacity of community safety features. Based on this motivation, we propose to weight house pairs differently by the comparability of house pairs by exploiting profile data. Specifically, we use characteristics in Table 2.1 to compute the similarity between every house pair by $D_{ij} = -\sqrt{\sum_{l=1}^{L} |p_l^i - p_l^j|^2}$,

Table 2.1: Characteristics in house profile.

Characteristics	Description
Neighborhood Characteristics	
Household Income	Average annual household income
High Educated Ratio	Ratio of residents with least bachelors degree to residents who are at least 25 years old
Population Growth	Percentage growth of population from 2000 to 2010
Surrounding Characteristics	
School Rating	Average rating of the nearest public high, middle and primary schools (A school has rating 1 to 5)
Point-of-Interest (POI) Diversity	Number of diverse categorical tags extracted from all the POIs which locate within d meters of the house
Check-in Density	Average number of social network check-in within d meters in every workday after hours 6 PM to 6 AM
Build Characteristics	
Land Area (in sqft), Bedroom Number and Build Year	

where i and j denote the house pair, p_l denotes the lth characteristic in profile vectors. D_{ij} is normalized as a real number between 0 and 1. Then, we incorporate D_{ij} as the comparability into the pair-wise probability of ranking consistency. We have the new ranking consistency probability: $Pr(h_i \to h_j|W,\Phi)^{D_{ij}}$, where D_{ij} is assigned as an exponent to corresponding house pair's ranking consistency probability. **Benefit**: when the similarity D_{ij} between h_i and h_j is high such as the extreme 1, the impact of their ranking probability $Pr(h_i \to h_j)$ will be fully preserved in objective function. On the other hand, when D_{ij} is low such as the extreme 0, the impact of $Pr(h_i \to h_j)$ will be fully blocked outside of objective function. In this way, we differentiate the importance of different house pairs for objective function, thus we can better use community safety for house ranking.

We present the final probability of observed data:

$$\begin{aligned} &Pr(\mathcal{O}|W,\Phi) = Pr(Y|W,\Phi)Pr(R|W,\Phi) \\ &= \prod_{i=1}^{I} \mathcal{N}(y_i|f_i,\sigma^2) \cdot \prod_{i=1}^{I-1} \prod_{j=i+1}^{I} \left(\frac{1}{1+exp(-(f_j-f_i))}\right)^{D_{ij}}. \end{aligned} \tag{2.9}$$

Next, let us address the prior distribution of W which is the last part in posterior distribution. We model $Pr(W|\Phi)$ as Gaussian distribution with 0 mean, where b^2 represents the variance of parameter w_m. We have the prior distribution formally:

$$Pr(W|\Phi) = \prod_{m=1}^{M} \mathcal{N}(w_m|0,b^2). \tag{2.10}$$

2.4.3 Parameter Estimation.

Given the posterior distribution in Equation 2.8, we want to find the optimal W to maximize the probability. The log posterior distribution is:

$$\begin{aligned}\mathcal{L}(W|Y,R,\sigma^2,b^2) &= -\frac{1}{2\sigma^2}\sum_{i=1}^{I}(y_i - f_i)^2 \\ &+ \sum_{i=1}^{I-1}\sum_{j=i+1}^{I} D_{ij} ln \frac{1}{1+exp(-(f_j - f_i))} - \frac{1}{2b^2}\sum_{m=1}^{M} w_m^2.\end{aligned} \tag{2.11}$$

To maximize the log posterior, we utilize gradient ascent method to update parameter w_m by $w_m^{(t+1)} = w_m^{(t)} + \alpha \times \frac{\partial \mathcal{L}}{\partial w_m}$, where α is the learning rate and $\frac{\partial \mathcal{L}}{\partial w_m}$ is the derivatives according to Equation 2.11:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_m} &= \frac{1}{\sigma^2}\sum_{i=1}^{I}(y_i - f_i)x_{im} - \frac{1}{b^2}w_m \\ &+ \sum_{i=1}^{I-1}\sum_{j=i+1}^{I} D_{ij}\frac{exp(-(f_j - f_i))}{1+exp(-(f_j - f_i))}(x_{jm} - x_{im}).\end{aligned} \tag{2.12}$$

2.5 Experimental Results

In this section, we present a comprehensive experiment to evaluate the proposed method on real-world dataset.

2.5.1 Experimental Data.

All the data of houses and crimes are collected from Denver Open Data Catalog (*Denver Open Data Catalog*, n.d.). For houses dataset, since house comparisons usually happen in the same type with not very far distance, we restrict the house type to only single family detached home which is the major type in U.S., and collect 3000 houses evenly spread in a major residential region of north Denver which consists of five adjacent official neighborhoods as shown in Figure 2.5a, 2.5b. All the house

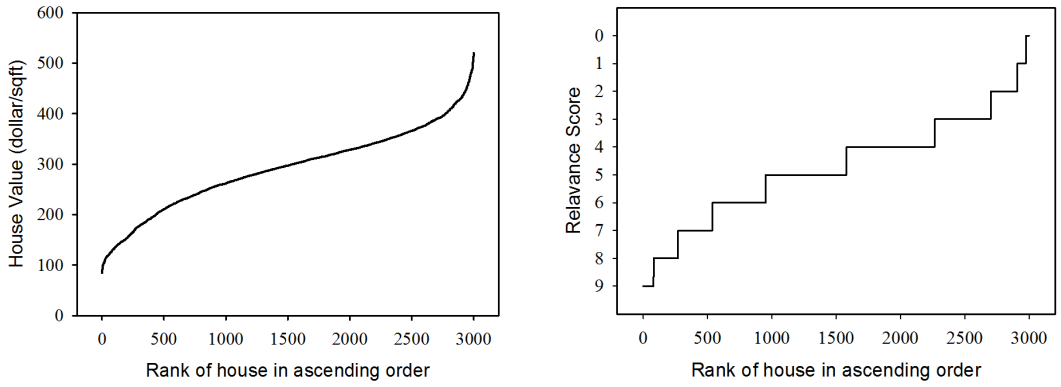
(a) (b) (c)

Figure 2.5: (a) Official neighborhoods. (b) Houses. (c) Burglary crimes.

values are appraised in 2015. Figure 2.6a shows the values of 3000 houses in ascending order. For ranking purpose, we evenly split of range of house value into 10 levels and give the level of lower values the higher relevance score. Therefore, as shown in Figure 2.6b, the real house value y_i in experiment is the relevance score from 0 to 9 which shows how low a house value is. For crime dataset, we collect residential forcible burglaries happened in Denver during 2009 to 2014. Totally, we find 1131 forcible burglary crimes which are related to our collected houses as shown in Figure 2.5c. For house profiles, we collect neighborhood data from demographic of 2010 & 2000 US Census, POIs and check-in (9/2010 to 1/2011) data from Foursquare, public school data from official public school rating (*School Performance Framework*, n.d.).

2.5.2 Baseline Algorithms.

To validate the effectiveness of our proposed method, we compare it with several traditional ranking algorithm: 1) LambdaMART (Burges, 2010), which employs the Lambda function of LambdaRank as gradients in the learning of Multiple Additive



(a) (b)

Figure 2.6: (a) Value of ranked houses. (b) Relevance score of ranked houses.

Regression Trees (MART). 2) AdaRank (Xu & Li, 2007), which plugs the evaluation measures into the framework for boosting optimization. 3) RankBoost (Freund, Iyer, Schapire, & Singer, 2003), which adopts AdaBoost (Freund & Schapire, 1997) for the pair-wise classification. 4) Coordinate Ascent (Metzler & Croft, 2007), which applies coordinate ascent technique in unconstrained optimization. 5) ListNet (Cao, Qin, Liu, Tsai, & Li, 2007), which defines the loss function by the probability distribution on permutations.

We adopt RankLib (*The Lemur Project/RankLib*, n.d.) for baseline algorithm implementation. For LambdaMART, we set number of trees = 300, number of leaves = 10. For AdaRank, we set number of round = 500, tolerance = 0.002. For RankBoost, we set number of rounds = 300. For Coordinate Ascent, we set number of random restarts = 5, tolerance = 0.001. We randomly split 3000 houses to be 4:1 where 2400 for training set and 600 for testing set.

2.5.3 Evaluation Metrics

Normalized Discounted Cumulative Gain (NDCG).

NDCG is obtained from Discounted Cumulative Gain (DCG) which measures the ranking quality by calculating the cumulative gain from the top of the result list to the particular rank position K. $DCG@K = rel_1 + \sum_{i=2}^{K} \frac{rel_i}{\log_2(i)}$ where rel_i represents the relevance score of the result at position i. Then we compute Ideal DCG (IDCN) which represents the maximum possible DCG till position K by sorting the result list by relevance. Last we obtain the normalized DCG: $NDCG@K = \frac{DCG@K}{IDCG@K}$.

Kendall's tau coefficient (Tau).

Kendall's Tau coefficient measures the ranking quality by rank correlation: the similarity of the orderings of houses between predicted ranking list and ground truth ranking list. Let $(\hat{r_i}, r_i)$ be the rank of house h_i in predicted ranking list and ground truth ranking list. Any pair of houses $(\hat{r_i}, r_i)$ and $(\hat{r_j}, r_j)$ are concordant if both $\hat{r_i} > \hat{r_j}$ and $r_i > r_j$ or if both $\hat{r_i} < \hat{r_j}$ and $r_i < r_j$. They are discordant, if $\hat{r_i} > \hat{r_j}$ and $r_i < r_j$ or if $\hat{r_i} < \hat{r_j}$ and $r_i > r_j$. We obtain the Kendall's tau coefficient: $Tau = \frac{\#concordant - \#discordant}{\#concordant + \#discordant}$.

Precision and Recall.

Precision measure the fraction of retrieved houses which are relevant. Recall measure the fraction of relevant houses that are retrieved. In our case, we consider the low value house with 7-9 relevance score as relevant, and consider other house with 0-6 relevance score as irrelevant. In retrieved top K ranking houses, we calculate the

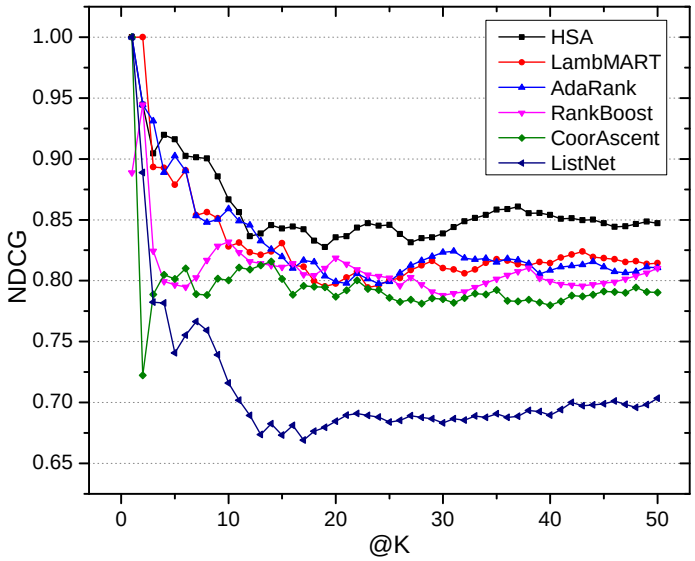


Figure 2.7: NDCG performance comparison.

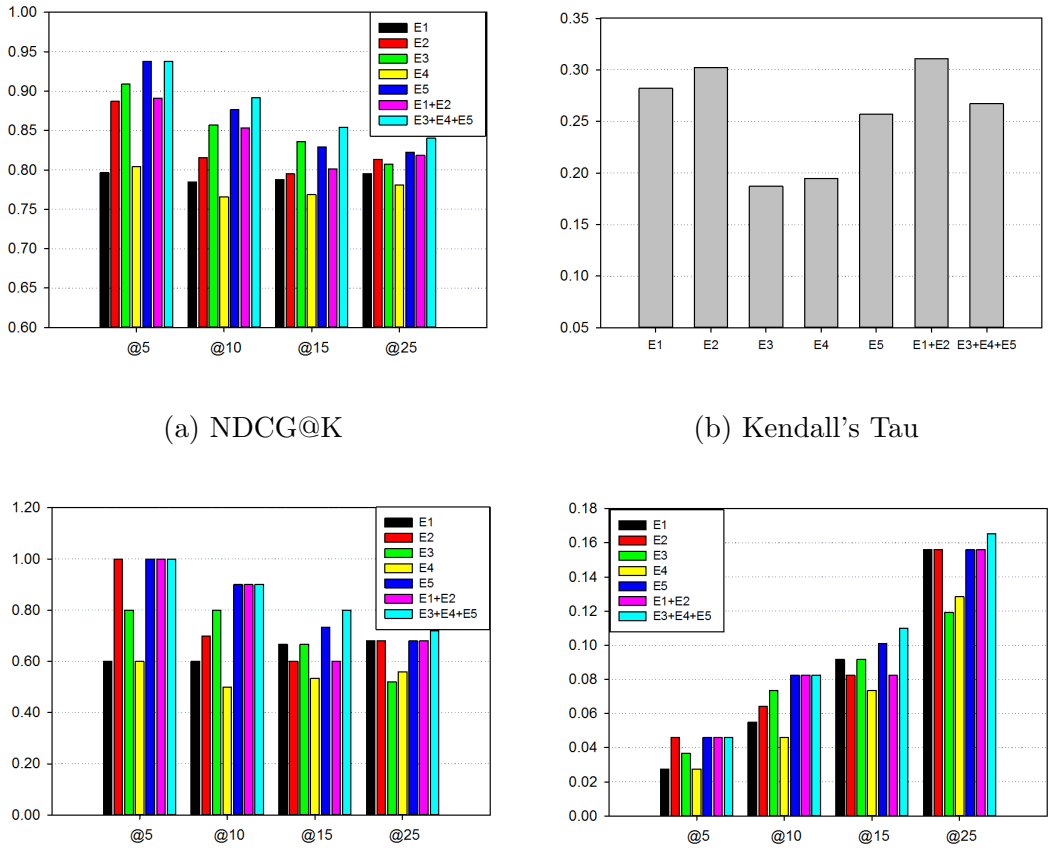
precision by $Precision@K = \frac{|h_K \cap h_{\geq 7}|}{|h_K|}$ and the recall by $Recall@K = \frac{|h_K \cap h_{\geq 7}|}{|h_{\geq 7}|}$, where h_K and $h_{\geq 7}$ denote the set of retrieved houses and the set of relevant houses.

2.5.4 Performance Evaluation on House Safety-Aware (HSA) Ranking.

We compare the performance of proposed HSA and baseline algorithms with the metrics of NDCG and Kendall's tau coefficient. Figure 2.7 shows the NDCG@K of each algorithm from K=1 to K=50. Overall, we can see that HSA outperforms all baselines. The improvement start to be obvious since K = 15. Second, we notice that LambdaRank and AdaRank which performance closely reach the second best overall performance. Compared to the rest of baseline algorithms, these two achieve obvious improvement of NDCG performance when K is smaller than 10. Moreover, RankBoost and Coordinate Ascent perform similarly. They both do not perform well when K is smaller than 15. Then their performance returns when K grows large. Last, ListNet makes the lowest performance, which is far behind other baseline algorithms. Table

Table 2.2: Performance of each algorithm.

Metrics	Lamb MART	Ada Rank	Rank Boost	Coor Ascnt	List Net	HSA
NDCG@5	0.8788	0.9023	0.7967	0.8015	0.7408	**0.9160**
NDCG@7	0.8537	0.8532	0.8026	0.7890	0.7664	**0.9013**
NDCG@10	0.8280	0.8590	0.8320	0.8003	0.7159	**0.8666**
NDCG@15	0.8309	0.8197	0.8113	0.8015	0.6733	**0.8429**
NDCG@25	0.8007	0.7993	0.8025	0.7859	0.6839	**0.8457**
Tau	0.2137	0.2733	0.1611	0.2326	0.2471	**0.3146**



(a) NDCG@K

(b) Kendall's Tau

(c) Precision@K

(d) Recall@K

Figure 2.8: Performance of different crime evidences.

2.2 shows the numerical result of NDCG at small K and Kendall's Tau coefficient. On NDCG @5, @7, @10, @15 and @25, HSA shows the consistent advantage. On Kendall's Tau coefficient, HSA achieves the best performance of 0.3146, the second best is AdaRank which gets 0.2733. Overall, HSA shows obvious advance in both NDCG and Kendall's Tau coefficient. In summary, the results shows that HSA model which incorporates house pairs comparability into ranking objective optimization can effectively increase the performance of house ranking with community safety.

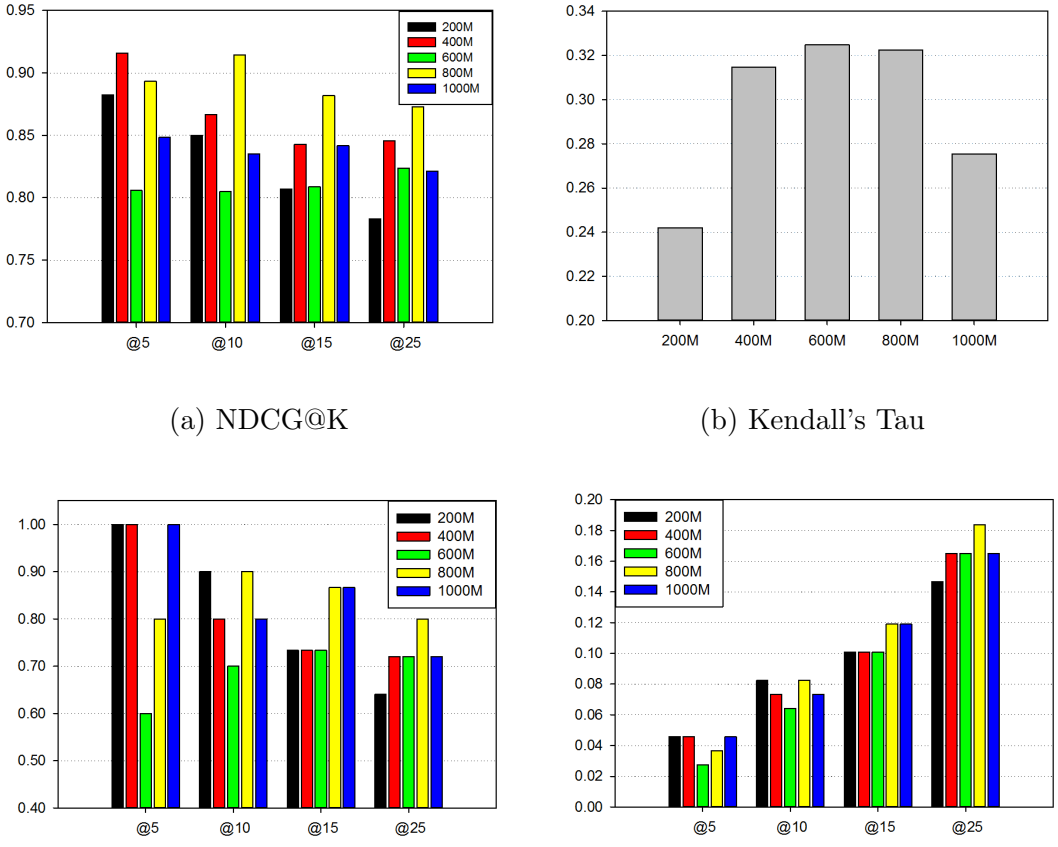
2.5.5 Performance of Different Crime Evidences.

We compare the performance of every single evidence type as well as two evidence combinations in NDCG, Tau, Precision and Recall. In Figure 2.8, E1 (Occurrence address), E2 (Occurrence time), E3 (Series size), E4 (Series length) and E5 (Series intensity) represent the five evidence types we extract. The combination of E1+E2 denotes the crime severity based category while the combination of E3+E4+E5 denotes the temporal correlation based category. First let us see the performance of single evidences. From the perspective of top K ranking measured by NDCG, E3 and E5 have the best performance and E2 preforms well too. In Precision and Recall, E5 performs the best. E2 and E3 perform well too. For the overall ranking consistency by Kendall's Tau, E1 and E2 show the best performance, E5 also does well. Then let us see the performance of evidence combinations based on two evidence categories. The crime severity based combination E1+E2 outperforms the single evidence E1 or E2 in ranking quality of both NDCG and Tau coefficient. The temporal correlation based combination E3+E4+E5 outperforms the single evidence E3, E4 or E5

in both of NDCG and Tau coefficient as well as Precision and Recall. Comparing E1+E2 with E3+E4+E5, we find that E3+E4+E5 consistently provides better top K ranking quality. Generally, the temporal correlation category evidences perform better than crime severity category evidences in differentiating low value houses via community safety conditions.

2.5.6 Performance of Different Crime Collection Radius.

Since we only consider the crimes which occurred within a certain distance of a house as impactful crimes, we want to explore what the proper distance is for learning a house's community safety. For example, if the radius is 400M, we will learn community safety by the crimes within 400 meter of a house. Figure 2.9 shows the performance on 5 different radius: 200M, 400M, 600M, 800M and 1000M in metrics of NDCG, Tau, Precision and Recall. From Kendall's Tau coefficient, we can see that radius in 400M, 600M and 800M outperform other distance in the quality of overall ranking. From NDCG, we can observe that 800M and 400M perform the best but the performance of 600M falls. One possible reason is that some houses can not cover more burglaries when the radius increases because it may cover non-residential blocks (e.g., square). When the radius continuously increases to 800M, the circle area overcomes the effects of non-residential blocks and reaches sufficient crimes for safety assessments. Therefore, we can have two insights from the results: 1) The distance from 400M to 600M which is a walking distance provides the most impactful crime for a house. 200M is too short to collect sufficient crimes while 1000M overly collects crime which do not generate real impacts. 2) Proper radius also depends on specific



(a) NDCG@K

(b) Kendall's Tau

(c) Precision@K

(d) Recall@K

Figure 2.9: Performance of different crime collection radius.

geographical situation which may disable some radius.

2.6 Related Work

This work can be grouped into three research categories. The first category is the study of the appraisal and ranking of real property. The works in (Pope, 2008; Linden & Rockoff, 2008) show that after a registered sex offender moves into a neighborhood, nearby housing prices would be declined in response. The work in (Gibbons, 2004) reports that property crimes have a significant negative impact on property price in

London area. The work in (Buonanno, Montolio, & Raya-Vílchez, 2013) concludes that decreases of perceived security level in victimization survey is associated with decreases of the real property valuation of a district. The works in (Fu, Xiong, et al., 2014; Fu et al., 2015) model the effects of geographical dependencies and function diversities for ranking estate investment values. The work in (Fu, Ge, et al., 2014) explores the effects of people's moving behaviors and online reviews on real estate ranking.

The second category belongs to criminology research. The work in (Polvi, Looman, Humphries, & Pease, 1991) finds that there is a dramatically enhanced risk of repeat burglaries for a house immediately after an initial burglary happened. The work in (Kleemans, 2001) shows that repeat victimization is more likely in high-crime than in low-crime areas, and that the re-committing by same offenders plays a key role in repeat victimization. The works in (Townsley, Homel, & Chaseling, 2003; Ratcliffe & Rengert, 2008) show that the elevated crime risk after the initial crime not only comes to the victims itself but also spread to nearby areas.

The last category is the research in Learning-to-rank (LTR) algorithm. There are three categories of LTR algorithm, point-wise ranking directly predicts the relevance degree of a document, such as (Cooper, Gey, & Dabney, 1992) which adopts regression to solve the problem of ranking. Pair-wise ranking output the relative order for a pair of two documents. The work in (Herbrich, Graepel, & Obermayer, 1999) applies the SVM technique to classify orders for document pairs. Last, list-wise ranking model the entire ranking of a whole set of documents. The work in (Cao et al., 2007) defines the loss function by using the probability distribution on permutations.

2.7 Concluding Remarks

In this chapter, we presented a systematic study on ranking house by leveraging spatio-temporal crime data. Specifically, we first extracted community crime evidences in two categories: crime severity and crime temporal correlation. Moreover we proposed effective approach to ranking houses based on value by incorporating the house specific features of community safety. Also, we integrated the impacts of popular house profile in optimization to enhance the proposed ranking model. Finally, extensive experimental results on real-world crime and house data validated the performance of the proposed method.

CHAPTER 3

POI RECOMMENDATION: A TEMPORAL MATCHING BETWEEN POI POPULARITY AND USER REGULARITY

Point of interest (POI) recommendation, which provides personalized recommendation of places to mobile users, is an important task in location-based social networks (LBSNs). However, unlike traditional interest-oriented merchandise recommendation, POI recommendation is more complex due to the *timing* effects: we need to examine whether the POI fits a user's availability. While there are some prior studies which included the temporal effect into POI recommendations, they overlooked the compatibility between time-varying popularity of POIs and regular availability of users, which we believe has a non-negligible impact on user decision-making. To this end, in this chapter, we present a novel method which incorporates the degree of temporal matching between users and POIs into personalized POI recommendations. Specifically, we first profile the temporal popularity of POIs to show when a POI is popular for visit by mining the spatio-temporal human mobility and POI category data. Secondly, we propose latent user regularities to characterize when a user is regularly available for exploring POIs, which is learned with a user-POI temporal matching function. Finally, results of extensive experiments with real-world POI check-in and human mobility data demonstrate that our proposed user-POI temporal matching method delivers substantial advantages over baseline models for POI recommendation tasks.

3.1 Introduction

The rapid development of GPS equipped mobile devices (e.g., smartphones) has powered large location-based social networks (LBSNs) (e.g., Foursquare), raised the number of mobile users, and enabled various location-based services (LBS). Using these LBS, users share their experiences of places, also known as Point of interests (POIs) such as restaurants or museums. Meanwhile, data collected through LBS activity enable better personalized recommendations of POIs. As a result, POI recommendation, which suggests personalized POIs to users, becomes an important component to improve user experiences and services provided by LBS.

Different from traditional interest-oriented merchandise recommendation (e.g., books, films, etc.), POI recommendation is more complex and challenging due to the unique characteristics of LBS. Firstly, besides personal interest, the timing of recommended POIs should be compatible with users' personal availability. For example, if a user is usually available to explore POIs during morning hours, he would be more likely to visit POIs with morning popularity (e.g., coffee shops, brunch restaurants). Similarly, if a POI is more popular during night hours (e.g., bars), it is more rational to recommend it to users who are available at nights. Secondly, area activity (or volume of people in an area) changes over time as people concentration to different places at different times throughout a day (e.g., work, entertainment). The area where recommended POIs reside should be active at a given time to increase the chance of visiting. For example, in the morning of weekdays, users are concentrated surrounding office/business locations, while at night time of weekends, nightlife and

restaurant regions are most active.

Recent studies have considered temporal influences on POI recommendation, such as time-aware POI recommendation which recommends different POIs to users at different time. For example, (Gao et al., 2013a) applies user-item matrix factorization for each time slot and assumes every user has similar preferences in consecutive time slots for regularization. (Q. Yuan et al., 2013a) computes user similarity via the same spatio-temporal check-ins in the past and conducts a user-based recommendation approach. (Xiong et al., 2010) adds the time dimension to user-item matrix and applies tensor factorization for recommendations. However, these studies overlooked temporal regularity of users, and time-varying popularity of POI. They also didn't consider the influence of temporal compatibility between users and POIs. In addition, they solely depended on the time input of history check-ins, and suffered from the sparsity problem of check-in data. Last, these studies didn't fully utilize spatio-temporal human mobility patterns which reflect the changes of areas' activity over time. In order to address these limitations, in this chapter, we introduce a novel model which incorporates the temporal compatibility between user regularities and POI popularities into POI recommendation, and utilize human mobility data to boost recommendation performances.

In this chapter, we propose a Temporal Matching Poisson Factorization Model (TM-PFM) to profile the popularity of POIs, model the regularity of users, and incorporate the temporal matching between users and POIs into overall recommending consideration. We first present a new framework to profile a time-varying popularity of POIs (e.g., hourly visiting change) in a day. Traditional methods usually capture

this temporal variation by counting POIs' check-in frequencies therefore suffer from check-in data sparsity. Previous studies (Zheng, Li, Chen, Xie, & Ma, 2008; Cho, Myers, & Leskovec, 2011) and (Y. Wang et al., 2015) have demonstrated that human mobility is highly regular and predictable, and human mobility data from heterogeneous sources display similar patterns. Therefore, we utilize heterogeneous human mobility data to evaluate POI popularity. The benefits of employing human mobility data include (i) it is more abundant and less biased than check-in data, and (ii) it reveals which areas are currently active which is a determinant of POI popularity. Moreover, we further analyze POIs by categories and adopt a mixture model to obtain the final POI temporal popularity pattern. Secondly, except some particular events (e.g., parties, concerts), people's availability is usually determined by their routines, thus there is a predictable regularity. Therefore, we consider temporal regularity of each user which describes their regular available time every day for POI exploration. We propose to learn the latent regularity patterns of users by finding the best match with the popularity patterns of visited POI based on check-in frequencies. Finally, with the learned user regularity, we are able to match users with POIs they have not visited yet, and evaluate the temporal matching degree and the general user-POI interest to make recommendations.

In summary, in this chapter we propose a novel temporal matching method between users and POIs for POI recommendation, and strategically leverage rich spatio-temporal human mobility data to boost the performance of the model. We highlight our key contributions as follows:

- We propose a factorization based POI recommendation model which incorporates the temporal matching between user regularity and POI popularity to improve POI recommendations.

- We present a novel framework which utilizes heterogeneous human mobility data to profile time-varying popularity of POIs which bypass the check-in data sparsity issue. Meanwhile, we model users' temporal regularity by incorporating user-POI temporal matching into preference estimation.

- We validate our proposed method with real-world LBSN check-in and human mobility datasets. The effectiveness of temporal matching in POI recommendation is proven by extensive experiments and a substantial improvement in recommendation performances over baseline methods is demonstrated.

3.2 Methodology Overview

We first provide some basic concepts in LBS, then formulate the problem of POI recommendation, and finally show the overview of the proposed temporal pattern matching based framework.

3.2.1 Preliminary

Definition 2 *(**Check-in**) A **check-in** is an event that a LBSN user reports his/her physical visit to a POI. Generally, a check-in contains the following information: LBSN user, check-in POI with location (e.g., longitude and latitude), category (e.g., Italian restaurant), and check-in timestamp.*

Definition 3 *(**Taxi trip**) A **taxi trip** is a route that a taxi delivers passengers*

Table 3.1: Mathematical notations.

Symbol	Size	Description
$\boldsymbol{Y}$	$M \times N$	user-POI check-in count matrix
$\boldsymbol{\mathcal{T}}$	1×2	day type $= \{wd(\text{weekday}), we(\text{weekend})\}$
$\boldsymbol{Q^*}$	$N \times S$	POI temporal popularity matrix, $* \in \boldsymbol{\mathcal{T}}$
$\boldsymbol{P^*}$	$M \times S$	user temporal regularity matrix, $* \in \boldsymbol{\mathcal{T}}$
$\boldsymbol{U}$	$M \times K$	user latent factor matrix
$\boldsymbol{V}$	$N \times K$	item latent factor matrix
$\boldsymbol{\mu^*}$	$1 \times M$	user temporal regularity parameter vector, $* \in \boldsymbol{\mathcal{T}}$

from one location to another. Every taxi trip starts with a passenger ***pick-up*** *event and ends with a passenger* ***drop-off*** *event. Each pick-up and drop-off contains the information of location and timestamp.*

3.2.2 Problem Definition

Let $U = \{u_1, u_2, ..., u_M\}$ be a set of LBSN users and $V = \{v_1, v_2, ..., v_N\}$ be a set of POIs where each POI has a location (e.g., latitude and longitude). Consider the existence of the historical check-ins where each record indicates a user u_i checked into a POI v_j once, we can extract the check-in number that u_i preformed check-in to v_j, named y_{ij}. **The objective** of personalized POI recommendation is to recommend POIs to users based on personal check-in history. In addition, we integrate the large-scale spatio-temporal taxi trip data, where each trip ends with a drop-off event which

indicates human arrivals with location and timestamp. We refer i as user and j as POI in following sections for simplicity. The important notations used in this chapter are listed in Table 3.1.

3.2.3 General Framework

We propose a two-step method which includes (i) profiling temporal patterns of POI popularity and (ii) modeling temporal matching of user-POI pairs.

Step 1: Profiling Temporal Patterns of POI Popularity. We aim at profiling the temporal popularity of POIs which describes how the popularity of a POI varies during a day. Specifically, we split a day into S equal-sized time slots (e.g., 24 hours), and each time slot is associated with a probability describing the ratio of the in time slot visit volume to the whole-day visit volume. We package these S probabilities chronologically as a vector which is the temporal pattern of popularity to be profiled. To achieve this, we first extract the area activity (e.g., how many active people in the area) around POI locations by utilizing human mobility data. Subsequently, we extract the category popularity by aggregating the check-in frequencies at the POI category level to refine the profiling. Lastly, we use a mixture model to smooth and further characterize the temporal pattern of POI popularity.

Step 2: Modeling Temporal Matching of user-POI pairs. We aim to develop a user-POI temporal matching model to infer the temporal regularity of users. First, we consider that each user has regular available times every day due to personal routines. Meanwhile, users are more likely to visit a POI at its popular times. We associate S equal-sized time slots with probabilities to show how likely a user may

explore POIs during a specific time slot in a day. By vectorizing these probabilities, a user's temporal regularities are defined. Furthermore, we present a function which matches a user's latent regularity with a POI's profiled popularity. We combine the temporal matching degree with the general interest as the overall preferences. Finally, we learn the users' temporal regularity by optimizing the distance between the estimated preferences and the frequencies of history check-in at POIs.

3.3 Profiling Temporal Patterns of POI Popularity

In this section, we introduce how to profile the temporal popularity for POIs. Intuitively, counting the check-in frequency during each time slot for a POI can complete this job. However, the POI level check-in records are too few to provide valid results. Thus we propose to alternatively analyze the temporal popularity in an implicit way. Generally, the current popularity of a POI is affected by two aspects: (i) how many active people are around the POI, and (ii) what type of service this POI provides. For the former one, we assess the area activity by mining how many people come to a POI's area during a time slot with taxi trip data. For the later aspect, we profile the category popularity by answering how many people visit a POI category in a time slot with check-in and POI category data. We combine these two effects to generate rough popularity patterns for POIs. Last, we utilize the mixture Gaussian model to smooth and characterize the popularity variations to obtain the final popularity patterns.

At the beginning, let us define the temporal popularity. We assign a unique popularity pattern to every POI to describe the visit volume changes over time slots

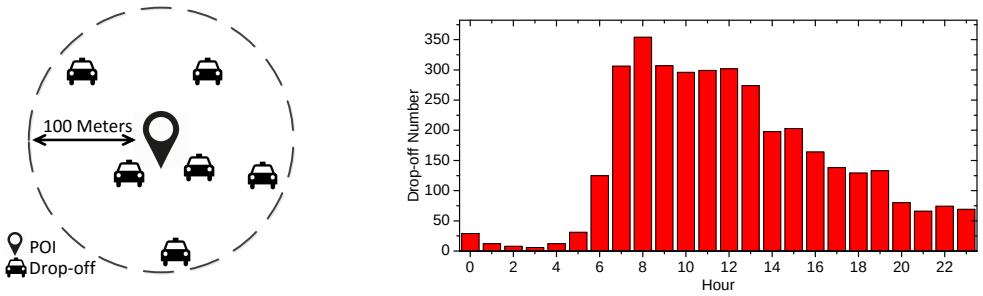
every day. To profile this temporal pattern, we use a size-S vector to represent the ratio of each time slot's visits to the whole day's visits. All the ratios are organized chronologically and their sum for a day equals to 1. Usually, the temporal pattern of a POI's popularity changes largely from weekdays (Monday to Friday) to weekends (Saturday, Sunday). Therefore, for each POI, we identify two types of temporal pattern: (1) weekday pattern $\boldsymbol{q_j^{wd}}$ and (2) weekend pattern $\boldsymbol{q_j^{we}}$. Formally, we denote temporal patterns of a POI's popularity as following:

$$\boldsymbol{q_j^*} = \{q^*_{j,1}, ..., q^*_{j,S}\}, \quad * \in \{wd, we\}, \tag{3.1}$$

where $q^*_{j,s}$ represents the probability that visitors will check-in to the POI j in the time slot s with respect to weekday wd or weekend we. For each $\boldsymbol{q_j^*}$ we have $\sum_{s=1}^{S} q^*_{j,s} = 1$ and $q^*_{j,s} \geq 0$.

3.3.1 Assessing Temporal Patterns with Area Activity

Every day, people concentrate to different places at different times for daily purposes (e.g., working, entertaining). Given a particular time, if a POI's is in the area where contains high volume of people, the POI is expecting to have more visits. Taxi is a fundamental transportation tool for people who live in large cities (e.g., New York City). Since each taxi trip ends with a destination, given massive and comprehensive taxi trips of a city, we are able to know where concentrates high volume of people at different times. Therefore, we collect taxi drop-offs which happened within walking distance (e.g., 100 meters) of each POI's location as shown in Figure 3.1a. The reason of choosing 100-meter for drop-off collection is that a longer distance makes the collection area too large that the unique characteristic of POI location can not



(a) (b)

Figure 3.1: (a) Method of collecting taxi drop-offs for a POI. (b) Time-varying taxi drop-off frequency around an office POI.

be captured, meanwhile, a shorter distance may not cover the nearest street crossing or road segment, thus POI locations may not be able to collect enough drop-offs to profile area activity. We count taxi drop-offs by time slots and day types (e.g., the drop-offs during 10AM-11AM in weekend days) as the example shown in Figure 3.1b. Through this taxi data processing, we profile the temporal pattern of area activity around POIs and denote them as:

$$\boldsymbol{\mathcal{D}_j^*} = \{\mathcal{D}_{j,1}^*, ..., \mathcal{D}_{j,S}^*\}, \quad * \in \{wd, we\}, \tag{3.2}$$

where $\mathcal{D}_{j,s}^*$ represents the portion of taxi drop-offs around POI j during s-th time slot in a type of day. For each $\boldsymbol{\mathcal{D}_j^*}$ we have $\sum_{s=1}^{S} \mathcal{D}_{j,s}^* = 1$ and $\mathcal{D}_{j,s}^* \geq 0$.

3.3.2 Refining Temporal Patterns with Category Popularity

At the same time, the popularity pattern of a POI is not only dominated by area activity but also related to its category. For example, at mid-night, even though an area may be highly active by having many visits, a museum at this place can not be popular. Therefore, the profiled patterns based on area activity need to be further

refined by integrating the category popularity. At the level of POI category (e.g., department stores), the sparsity problem of check-in data is alleviated. Therefore, we count check-ins frequencies for categories over time slots. We denote the category pattern as:

$$\boldsymbol{\mathcal{C}_v^*} = \{\mathcal{C}_{v,1}^*, ..., \mathcal{C}_{v,S}^*\}, \quad * \in \{wd, we\}, \tag{3.3}$$

where $\mathcal{C}_{v,s}^*$ represents the portion of check-in at the category v during s-th time slot in a type of day. For each $\boldsymbol{\mathcal{C}_v^*}$ we have $\sum_{s=1}^{S} \mathcal{C}_{v,s}^* = 1$ and $\mathcal{C}_{v,s}^* \geq 0$.

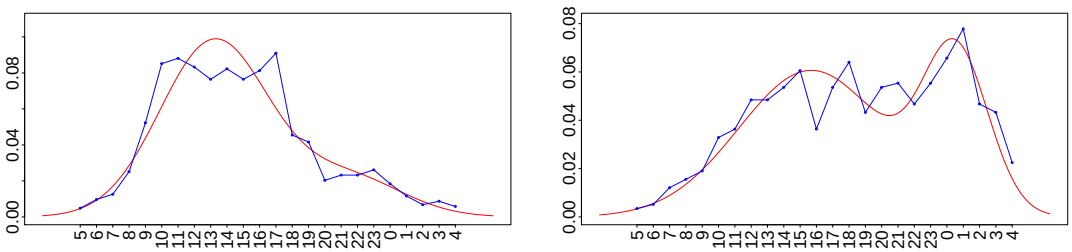
Next, by combining the effects of category popularity with the effects of area activity, we obtain the refined POI temporal popularity which is more close to the reality. We denote the combined temporal popularity of POIs as $\boldsymbol{q'^*_j}$, whose probability for each time slot is:

$$q'^*_{j,s} = \varphi \mathcal{D}^*_{j,s} + (1 - \varphi)\mathcal{C}^*_{c(j),s}, \quad * \in \{wd, we\}, \tag{3.4}$$

where $c(j)$ is the operation to get the category v of POI j, $0 < \varphi < 1$ controls the weights.

3.3.3 Enhancing Temporal Patterns with Mixture Model

In the last part of temporal popularity profiling, we want to describe each temporal pattern with a proper distribution. The first motivation is to smooth the visit probability over time slots because artificial spiting of drop-offs into time slots may cause volatile patterns especially in adjacent time slots as shown in Figure 3.2. Another motivation which is more important is that we want to strategically characterize the popularity pattern to be more discriminative by weakening the idle time slots and highlighting the popular time slots.



(a) (b)

Figure 3.2: Example of two POI popularity patterns in hours of day. Blue and Red: before and after GMM smoothing.

To achieve the requirements raised by above motivations, we propose to adopt Gaussian Mixture Model (GMM) to model popularity patterns for several advantages. First, GMM can express one or more visit peaks in a day as a POI usually behaves in reality. Second, the Gaussian distribution can well simulate the process of visit changes of POIs. For example, a POI's popularity often starts from idle to busy and gets back to idle. Usually one process last for several time slots and the popularity changes smoothly. Third, for the idle times, the visit probability are weaken. Meanwhile, for the busy times, probability are enhanced and concentrated to the peak point. Therefore, we formally define the probability over time slots of a POI popularity pattern $\boldsymbol{q}_j^*$ with GMM as:

$$q_{j,s}^* = \sum_{r=1}^{R} w_{j,r}^* \cdot \mathcal{N}(s|\mu_{j,r}^*, \sigma_{j,r}^{*2}), \quad * \in \{wd, we\}, \tag{3.5}$$

where s represents the s-th time slots and R represents the number of Gaussian components in a daily temporal pattern. $w_{j,r}^*$ represents the mixture weight of rth Gaussian distribution. In our observation, most of the POIs have no more than two visiting peaks in a day such as restaurants. Therefore we predefine the number

of Gaussian components $R = 2$ for all GMM modeling. Input the POI temporal patterns $\boldsymbol{q'^{*}_{j}}$ we obtained in previous step, we apply Expectation–Maximization (EM) algorithm to estimate the GMM for each pattern as shown in Figure 3.2. Last, we obtain the final popularity patterns $\boldsymbol{q^{*}_{j}}$ for each POI.

3.4 Recommendations via Temporal Matching

In this section, we first introduce how to model the temporal matching between user and POI, then we present the parameter estimation of the model.

3.4.1 Model Specification

To generate recommendation of a POI j for a user i, we assume the overall preference on the user-POI pair f_{ij} is impacted by (i) the user-POI general interest score, $\delta(i, j)$, and (ii) the user-POI temporal matching score, $m(i, j)$:

$$f_{ij} = \delta(i, j) \cdot m(i, j). \tag{3.6}$$

The user-POI general interest score $\delta(i, j)$ is learned from classic matrix factorization methods, by combining K-dimensional user latent factor vector $\boldsymbol{u_i}$ and POI latent factor vector $\boldsymbol{v_j}$ as follows: $\delta(i, j) = \boldsymbol{u_i}^{\top} \boldsymbol{v_j}$. The user-POI temporal matching score $m(i, j)$ is the degree of matching between users and POIs, based on S-dimensional user temporal regularity vectors $\boldsymbol{\rho^{*}_{i}}$ and POI temporal popularity vectors $\boldsymbol{q^{*}_{j}}$, where $* \in \{wd, we\}$, wd and we respectively represent the day type of weekday and weekend. Next, we present the detailed temporal matching modeling.

Capturing User Daily Temporal Regularity Except some special events, the available hours for exploring POIs are usually regular for users due to personal daily

routines. For example, if a user always have a long lunch break, thus he may regularly explore POIs during 12PM to 2PM. Therefore, we propose that every LBSN user has a latent daily-repeated personalized temporal regularity which decides when he/she is likely to explore POIs every day. Usually a individual's temporal regularities is different in weekday and weekend, we define two types of daily temporal regularities for each user:

$$\boldsymbol{\rho}_{\boldsymbol{i}}^{*} = \{\rho_{i,1}^{*}, ..., \rho_{i,S}^{*}\}, \quad * \in \{wd, we\}, \tag{3.7}$$

where $\rho_{i,s}^{*}$ represent user i's exploring probabilities during time slot s for weekdays wd or weekend we. For each regularity pattern $\boldsymbol{\rho}_{\boldsymbol{i}}^{*}$, we have $\sum_{s=1}^{S} \rho_{i,s}^{*} = 1$ and $\rho_{i,s}^{*} \geq 0$.

At the same time, we also want to regularize user's availability distribution over time slots. In reality, users usually plan one trip in a day and their availability does not fluctuate largely in adjacent time slots, therefore we assume one window per day for each user for POI exploration. We exploit a Gaussian distribution to regularize each regularity pattern. For $\boldsymbol{\rho}_{\boldsymbol{i}}^{*}$, we have the probability in each time slots as:

$$\rho_{i,s}^{*} = \mathcal{N}(s|\mu_{i}^{*}, \varepsilon_{i}^{*2}), \quad * \in \{wd, we\}. \tag{3.8}$$

Here we model the check-in probability of s-th time slot as the probability density at s (e.g., $s = 5$).

Modeling User-POI Temporal Matching Here we present how we match the user's temporal regularities with the POI's temporal popularities. The objective of temporal matching for a user-POI pair is to examine if the POI is well-timed for the user's temporal regularity. For example, for a user who explores POI in the morning time, a coffee shop is more well-timed than a bar. Since the popularity pattern can

indicate the optimum time slots of POIs, our method is to find out if the regularity pattern of users has any common time slots to favor a POI's popularity. We define the temporal matching score $m(i,j)$ for user i and POI j as following:

$$m(i,j) = \gamma \boldsymbol{\rho}_{\boldsymbol{i}}^{\boldsymbol{wd}\top} \boldsymbol{q}_{\boldsymbol{j}}^{\boldsymbol{wd}} + (1-\gamma) \boldsymbol{\rho}_{\boldsymbol{i}}^{\boldsymbol{we}\top} \boldsymbol{q}_{\boldsymbol{j}}^{\boldsymbol{we}}, \tag{3.9}$$

where $0 < \gamma < 1$ controls the weights of temporal matching score on weekday and weekend. For example, we can assume that the importance of each day of a week would be the same for each user, therefore $\gamma = \frac{5}{7}$ for five days of weekday and the rest $\frac{2}{7}$ for two days of weekend.

In this model, we have four latent variables to be learned: POI interest latent factors $\boldsymbol{v_j}$, user interest latent factors $\boldsymbol{u_i}$, and user daily temporal regularities $\boldsymbol{\rho_i^*}$, where $* \in \{wd, we\}$ for day types of weekday and weekend respectively. $\boldsymbol{v_j}$ and $\boldsymbol{u_i}$ are K-dimensional vectors while $\boldsymbol{\rho_i^*}$ are S-dimensional vectors. Since we model the regularity on every time slot s to be $\rho_{i,s}^* = \mathcal{N}(s|\mu_i^*, \varepsilon_i^{*2})$ for user temporal regularity $\boldsymbol{\rho_i^*}$ as Equation (3.8), we further translate user temporal regularity factors $\boldsymbol{\rho_i^*}$ into μ_i^* and ε_i^*. For reducing parameters to learn and improving computational efficiency, we predefine a unified ε for all user temporal regularities by referring a usual availability window of people (e.g., 4 hours). Therefore, we rewrite the temporal matching score $m(i,j)$ in Equation (3.9) as following:

$$m(i,j) = \gamma \sum_{s=1}^{S} \mathcal{N}(s|\mu_i^{wd}, \varepsilon^2) q_{j,s}^{wd} + (1-\gamma) \sum_{s=1}^{S} \mathcal{N}(s|\mu_i^{we}, \varepsilon^2) q_{j,s}^{we}. \tag{3.10}$$

Finally, to infer the latent factors $\boldsymbol{v_j}$, $\boldsymbol{u_i}$ and μ_i^*, we need to formulate the estimated user-POI preference f_{ij} to follow a probability distribution $\Pr(y_{ij}|f_{ij})$, where y_{ij} is the user-POI check-in count as the groundtruth of user preference. Also, since

all the user-POI visit count y_{ij} are non-negative, we expect our estimated preference f_{ij} to be non-negative. We use a Bayesian non-negative latent factor model.

Given the heavy skewness and wide range of discrete check-in count data as shown in Figure 3.3b, we adopt a Poisson distribution to model $\Pr(y_{ij}|f_{ij})$:

$$\begin{aligned} y_{ij} &\sim \text{Poisson}(f_{ij}) \\ \Pr(y_{ij}|f_{ij}) &= (f_{ij})^{y_{ij}} \frac{\exp\{-f_{ij}\}}{y_{ij}!}, \end{aligned} \tag{3.11}$$

where $f_{ij} = \boldsymbol{u_i}^\top \boldsymbol{v_j} \cdot m(i,j)$ refers to Equation (3.6), $m(i,j)$ refers to Equation (3.10).

Furthermore, v_{jk}, u_{ik} can be given Gamma distributions while μ_i^* can be given Gaussian distribution as empirical priors. Therefore, the user-POI preferences can be modeled as a generative process:

1. For each POI j, generate K-dim POI latent factor:

$$v_{jk} \sim Gamma(\alpha_V, \beta_V), \tag{3.12}$$

2. For each user i, generate K-dim user latent factor:

$$u_{ik} \sim Gamma(\alpha_U, \beta_U), \tag{3.13}$$

Also, generate user temporal regularity factor for weekday and weekend:

$$\mu_i^* \sim \mathcal{N}(\alpha_\mu, \sigma_\mu^2), \quad * \in \{wd, we\}, \tag{3.14}$$

3. For each user-POI pair $< i, j >$, generate response:

$$Pr(y_{ij}|\boldsymbol{v_j}, \boldsymbol{u_i}, \mu_i^{wd}, \mu_i^{we}) = (f_{ij})^{y_{ij}} \frac{\exp\{-f_{ij}\}}{y_{ij}!}, \tag{3.15}$$

where $\Theta = \{\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}}\}$ are parameters for estimation, and $\Phi = \{\alpha_V, \beta_V, \alpha_U, \beta_U, \alpha_\mu, \sigma_\mu^2\}$ are hyperparameters.

3.4.2 Parameter Estimation

Given the observations of user-POI check-in count $\boldsymbol{Y}$ and the hyperparameters Φ, according to Maximum a posteriori (MAP) estimation, we optimize parameters $\boldsymbol{V}$, $\boldsymbol{U}$, $\boldsymbol{\mu^{wd}}$, $\boldsymbol{\mu^{we}}$ by maximizing the posterior probability:

$$\Pr(\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}} | \boldsymbol{Y}, \Phi) \propto \Pr(\boldsymbol{Y} | \boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}}) \Pr(\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}} | \Phi). \quad (3.16)$$

For $Pr(y_{ij} | \boldsymbol{v_j}, \boldsymbol{u_i}, \mu_i^{wd}, \mu_i^{wd})$, we use Equation (3.15) to compute:

$$\Pr(\boldsymbol{Y} | \boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}}, \Phi) = \prod_{i=1}^{M} \prod_{j=1}^{N} (f_{ij})^{y_{ij}} \frac{\exp\{-f_{ij}\}}{y_{ij}!} \quad (3.17)$$

For $\Pr(\boldsymbol{v_j}, \boldsymbol{u_i}, \mu_i^{wd}, \mu_i^{we} | \alpha_V, \beta_V, \alpha_U, \beta_U, \alpha_\mu, \sigma_\mu^2)$ which are the prior distributions of $\boldsymbol{V}$, $\boldsymbol{U}$, $\boldsymbol{\mu^{wd}}$, and $\boldsymbol{\mu^{we}}$, we use Equation (3.12, 3.13, 3.14) to generate:

$$\begin{aligned}
\Pr(\boldsymbol{V} | \alpha_V, \beta_V) &= \prod_{j=1}^{N} \prod_{k=1}^{K} \frac{v_{jk}^{\alpha_V - 1} \exp(-v_{jk}/\beta_V)}{\beta_V^{\alpha_V} \Gamma(\alpha_V)} \\
\Pr(\boldsymbol{U} | \alpha_U, \beta_U) &= \prod_{i=1}^{M} \prod_{k=1}^{K} \frac{u_{ik}^{\alpha_U - 1} \exp(-u_{ik}/\beta_U)}{\beta_U^{\alpha_U} \Gamma(\alpha_U)} \\
\Pr(\boldsymbol{\mu^*} | \alpha_\mu, \sigma_\mu^2) &= \prod_{i=1}^{M} \frac{1}{\sigma_\mu \sqrt{2\pi}} \exp\{-\frac{(\mu_i^* - \alpha_\mu)^2}{2\sigma_\mu^2}\}, * \in \{wd, we\}.
\end{aligned} \quad (3.18)$$

Then we have the log posterior of Equation (3.16) as:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{\mu^{wd}}, \boldsymbol{\mu^{we}} | \boldsymbol{Y}, \Phi) = & \sum_{i=1}^{M} \sum_{j=1}^{N} (y_{ij} \ln f_{ij} - f_{ij}) \\
& + \sum_{j=1}^{N} \sum_{k=1}^{K} \Big((\alpha_V - 1) \ln v_{jk} - v_{jk}/\beta_V \Big) \\
& + \sum_{i=1}^{M} \sum_{k=1}^{K} \Big((\alpha_U - 1) \ln u_{ik} - u_{ik}/\beta_U \Big) \\
& + \sum_{i=1}^{M} \Big(-\frac{1}{2} \ln \sigma_\mu^2 - \frac{(\mu_i^{wd} - \alpha_\mu)^2}{2\sigma_\mu^2} \Big) \\
& + \sum_{i=1}^{M} \Big(-\frac{1}{2} \ln \sigma_\mu^2 - \frac{(\mu_i^{we} - \alpha_\mu)^2}{2\sigma_\mu^2} \Big) + \text{const.}
\end{aligned} \quad (3.19)$$

Taking derivatives on $\mathcal{L}$ with respect to v_{jk}, u_{ik}, μ_i^{wd} and μ_i^{we}, we have:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial v_{jk}} &= \frac{\alpha_V - 1}{v_{jk}} - \frac{1}{\beta_V} + \sum_{i=1}^{M} \Big(\frac{y_{ij}}{f_{ij}} - 1\Big) u_{ik} \cdot m(i,j) \\
\frac{\partial \mathcal{L}}{\partial u_{ik}} &= \frac{\alpha_U - 1}{u_{ik}} - \frac{1}{\beta_U} + \sum_{j=1}^{N} \Big(\frac{y_{ij}}{f_{ij}} - 1\Big) v_{jk} \cdot m(i,j) \\
\frac{\partial \mathcal{L}}{\partial \mu_i^*} &= -\frac{\mu_i^* - \alpha_\mu}{\sigma_\mu^2} + \sum_{j=1}^{N} \Bigg((\frac{y_{ij}}{f_{ij}} - 1) \cdot \boldsymbol{u_i}^\top \boldsymbol{v_j} \\
&\cdot \sum_{s=1}^{S} \Big(\frac{\gamma^* q_{js}^* (s - \mu_i^*)}{\varepsilon^3 \sqrt{2\pi}} \exp\{-\frac{(s-\mu_i^*)^2}{2\varepsilon^2}\} \Big) \Bigg), * \in \{wd, we\},
\end{aligned} \tag{3.20}$$

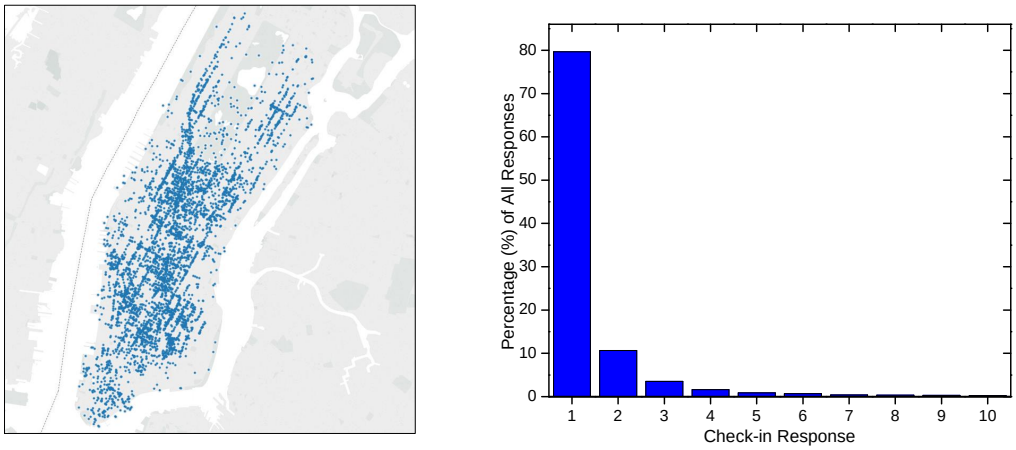
where γ^{wd} and γ^{we} are γ and $1 - \gamma$. We use gradient ascending method to infer the parameters. Specifically, we maximize the posterior by updating parameters as $\upsilon^{(t+1)} \leftarrow \upsilon^{(t)} + \epsilon \times \frac{\partial \mathcal{L}}{\partial \upsilon}$, where υ is an element in $\{\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{\mu^*}\}$, $\frac{\partial \mathcal{L}}{\partial \upsilon}$ is the derivatives according to Equation (3.20), and ϵ is the learning rate.

3.5 Experiments

In this section, we empirically evaluate the performance of our proposed methods. We perform all the experiments on real-world datasets: LSBN data from Foursquare, human mobility data from taxi trip records of New York City.

3.5.1 Experimental Data

For LBSN dataset, we use the Foursquare dataset which is formulated in work (Yang, Zhang, Zheng, & Yu, 2015a). The dataset includes the check-in data in New York City (NYC) for 10 months (April 2012 to February 2013). Each check-in contains the information such as user ID, POI ID, location, timestamp and POI category. To work with NYC taxis which mainly drive in the city area, we limit the POIs to the most densely populated borough - Manhattan. Also, we remove the users and POIs with



(a) (b)

Figure 3.3: (a) POI geographical distribution. (b) Check-in response distribution.

too few check-ins (e.g., less than 3) from our dataset to avoid cold start problem. We finalized a dataset of 975 users for 4722 POIs with 64702 check-in observations. The user-POI check-in count matrix has a sparsity of 99.24 percent. Each user performs 66 check-ins to POIs on average. The number of check-ins for a POI ranges from 1 to 257. Figure 3.3 provides the geographical distribution of POIs as well as the distribution of user-POI check-in responses.

For human mobility data, we use yellow cab trip records from NYC taxi & limousine commission[1] covering the time range of check-in dataset. Due to the large size of taxi trips in NYC, we randomly sample 2 million trips in Manhattan. Each taxi trip contains an origin and a destination with information of location and timestamp.

[1]http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

3.5.2 Experimental Metrics

In our experiments, we recommend each user a list of N POIs which have the highest predicted values but are not visited in training set. Then we evaluate the lists based on the recommended POIs which are actually visited by users in testing set.

Precision and Recall: Given a top-N recommendation list of POIs $L_{N,rec}$, precision and recall are defined as:

$$\begin{aligned} \text{Precision@}N &= \frac{|L_{\text{N,rec}} \cap L_{\text{visited}}|}{N} \\ \text{Recall@}N &= \frac{|L_{\text{N,rec}} \cap L_{\text{visited}}|}{|L_{\text{visited}}|}, \end{aligned} \tag{3.21}$$

where $L_{\text{N,rec}}$ represents the recommended list of N POIs for a user, and L_{visited} represents the visited POIs of the user in test set. By averaging the precision and call value of all users, we obtain the overall precision and recall for a recommender system.

F-measure: F-measure is the harmonic mean of precision and recall. We adopt a unbalance F-measure F_β which put more emphasis on precision than recall by setting $\beta = 0.5$:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \text{Precision} + \text{Recall}}. \tag{3.22}$$

3.5.3 Baseline Algorithms

The experimental study compares our proposed temporal matching Poisson factor model (TM-PFM) with state-of-the-art factor-based models. Specifically, we compare our proposed TM-PFM model with following algorithms:

- Probabilistic Matrix Factorization (PMF) (Mnih & Salakhutdinov, 2008): a widely used probabilistic factor-based model with Gaussian observation noise.

- Non-negative Matrix Factorization (NMF)(Lee & Seung, 2001): a matrix factorization model with the constrain of non-negative latent variables.

- Bayesian Probabilistic Tensor Factorization (BPTF)(Xiong et al., 2010): a model which introduces time dimension to traditional user-item factor-based collaborative filtering method.

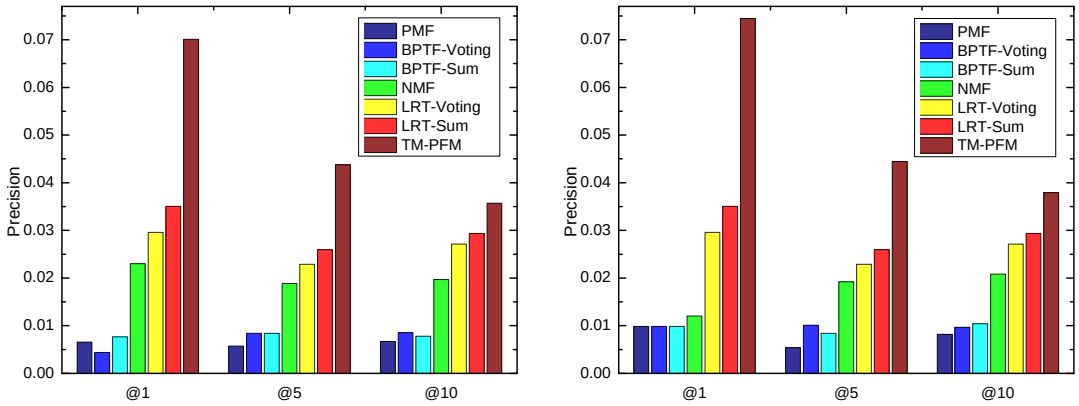
- Location Recommendation with Temporal effects (LRT)(Gao et al., 2013a): a factor-based model which learns users' time-aware preferences at separated time slots and use the preference similarity in consecutive times as regularization.

Since BPTF and LRT are temporal recommendation model, therefore, we need to obtain the overall preference for POIs. We aggregate the preference at each time slot by two ways.

- Sum: we consider a user's overall preference on a POI as the sum of his preference at each time slot.

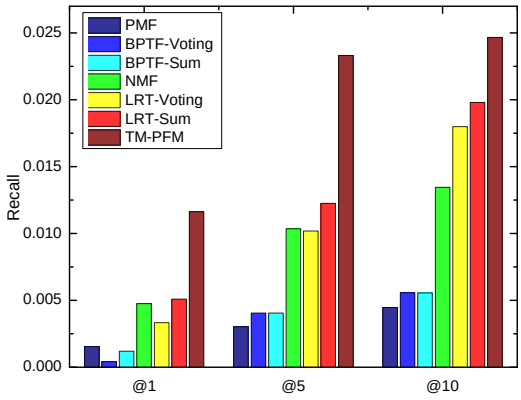
- Voting: for each time slot, we make a separate recommendation list and give the recommended POIs a nomination. The overall preference on a POI is obtained by the number of nominations.

For the experiment setup, we randomly divided the user-item check-in count data into 80 percent for training and 20 percent for testing. We set $\lambda_U = \lambda_V = 0.005$ for PMF. We set $\nu_\alpha = W_\alpha = \beta = 1$ for BPTF. For TM-PFM, we set $\alpha_U = \alpha_V = 4$ when $K = 10$, and $\alpha_U = \alpha_V = 3$ when $K = 20$. For both K, we set $S = 24$, $\varphi = 0.6$, $\gamma = \frac{5}{7}$, $\beta_U = \beta_V = 0.2$, $\varepsilon = 3$, $\alpha = 11.5$, and $\sigma_\mu = 3.5$.

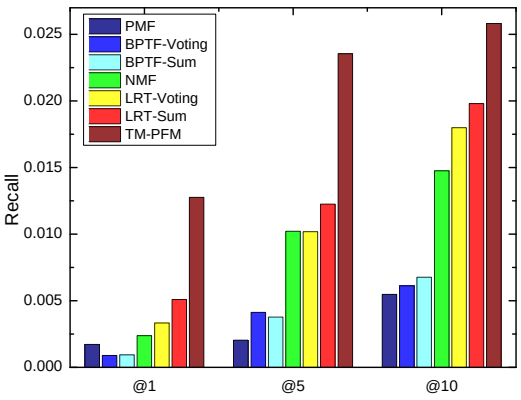


(a) Precision, K=10.

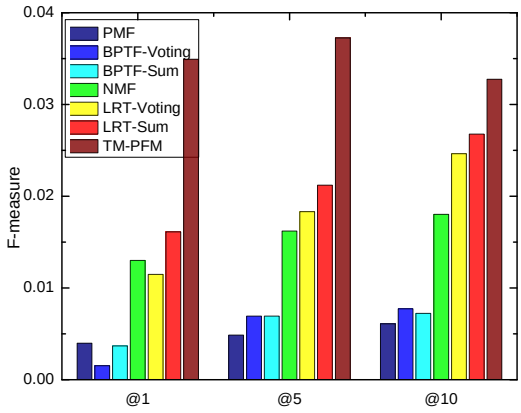
(b) Precision, K=20.



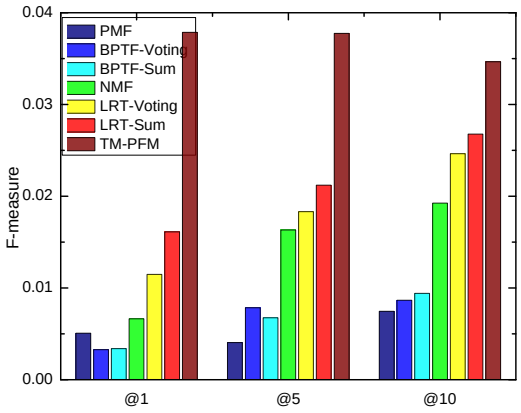
(c) Recall, K=10.



(d) Recall, K=20.



(e) F_β measure, K=10.



(f) F_β measure, K=20.

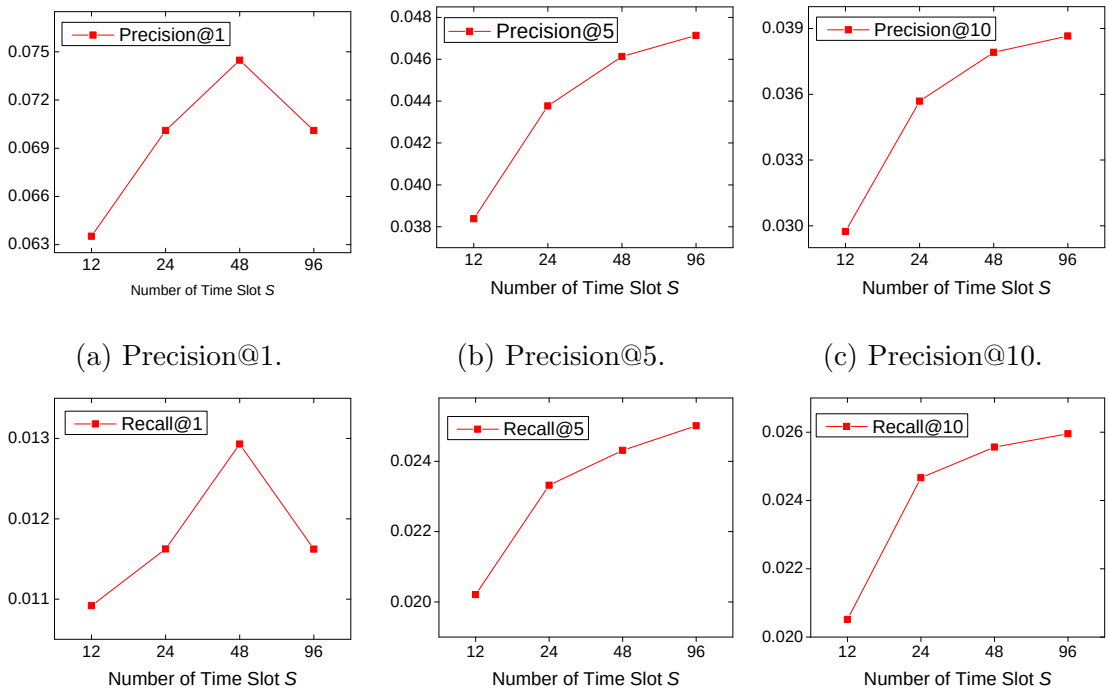
Figure 3.4: Precision, recall, and F_β measure @1, @5 and @10 with two different latent dimensions K.

3.5.4 Overall Performances

Figure 3.4 shows the precision@N, recall@N and F_β measure@N ($\beta = 0.5$) of all compared approaches on our dataset. For top-N position, we examine $N = 1, 5, 10$. For latent factor dimension, we explore $K = 10$ and $K = 20$.

Generally, we can see that our proposed approach TM-PFM consistently outperform baseline methods, including traditional recommendation models (PMF, NMF) as well as the temporal recommendation model (BPTF, LRT) for different N and different K. Specifically, we find that PMF performances similarly as BPTF approach with aggregation rule of voting or sum. NMF outperforms the previous three approach (PMF, BPTF-Voting, BPTF-Sum) by making latent variables non-negative. Furthermore, LRT approach (LRT-Sum, LRT-Voting) with either voting or sum rule outperforms NMF by learning time-aware preferences and assuming similarities for consecutive time slots. For temporal recommendation models BPTF and LRT, we find that the sum aggregation rule generally performs better than voting aggregation rule, especially on LRT with quilt significant differences. Last, our proposed TM-PFM model further outperforms LRT significantly on precision, recall and F_β measure, with respect to $K = 10$ and $K = 20$.

At the same time, from the experiment results we can see that the non-negative factor models (NMF, LRT and TM-PFM) preform better than the regular factor models (PMF, BPTF). One reason is that regular models are more suitable for explicit response (e.g., rating), but for implicit response such as check-in count data which is heavily skewed to 1, non-negative models provide better performance. Comparing



(a) Precision@1. (b) Precision@5. (c) Precision@10.

(d) Recall@1. (e) Recall@5. (f) Recall@10.

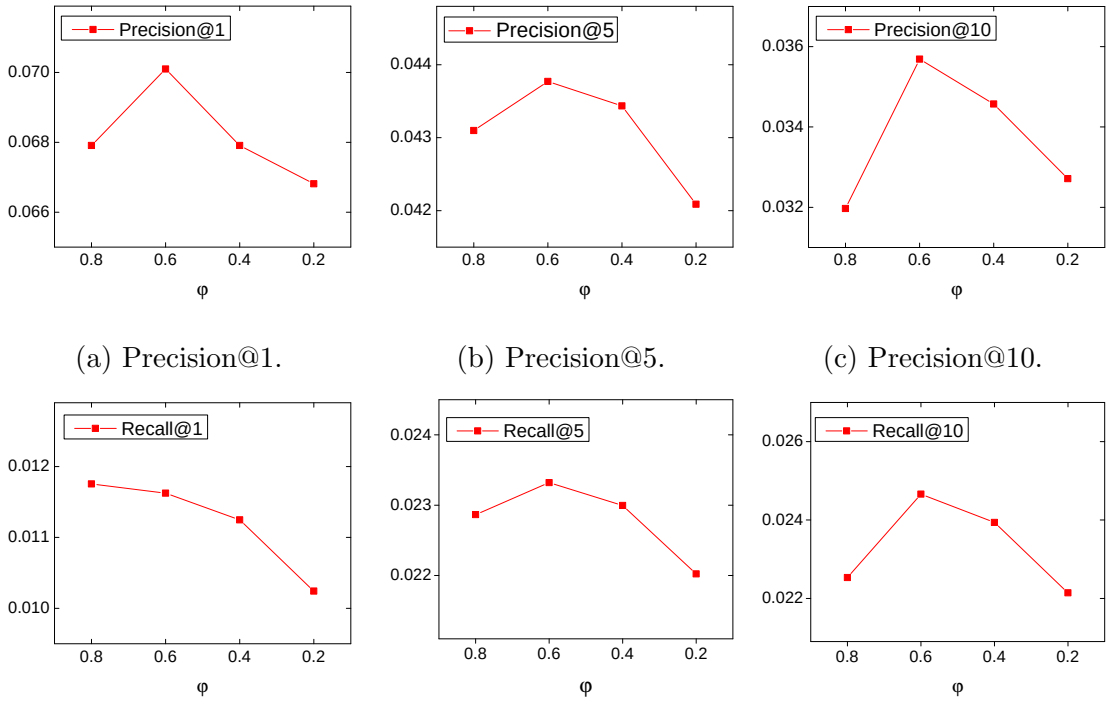
Figure 3.5: Precision and recall of proposed model with different time slot number S of temporal patterns (K=10).

our proposed model with other non-negative models (NMF, LRT), our model which adopts Poisson observation noise is more appropriate for modeling count data. Also, while the other non-negative models can only apply an approximation of probabilistic generative process, our proposed model provides a more authentic way. Compare to all baseline methods, our model demonstrates the effectiveness of incorporating user-POI temporal matching consideration into POI recommendations.

3.5.5 Performance with Different Time Slot Number

We study the model performance in different time slot numbers as shown in Figure 3.5. As we equally split one day into multiple time slots to construct temporal patterns,

the number of time slots S decide the length of each single slot. We compare four different numbers of time slots in this study: 12 time slots for 2-hour/slot, 24 time slots for 1-hour/slot, 48 time slots for 30-minute/slot and 96 time slots for 15-minute/slot. The larger number of time slots means the patterns of popularity or regularity are more fine-grained. Figure 3.5 shows the precision and recall performance of model at top-N position 1,5, and 10. We have two observations. Firstly, we can see that, as the number goes higher, the model achieve better performance. The only exception appears at precision@1 and recall@1 where performance decrease a few from 48 time slots to 96 time slots. However, the overall increasing trend still exists. The reason is that the popularity patterns characterize every POI to be more distinctive as the number of time slot goes up. By matching users' regularity with their visited POIs, the regularity patterns can be inferred with finer resolution. Therefore, the performance can be boosted by larger time slot numbers generally. Secondly, we can find out that the increase slows down when time slot number goes large. The largest increase usually happened at 12 time slots. After that, the performance does not increase strongly as before. One reason is that each time slot starts to lack sufficient observations (e.g., taxi drop-offs) for popularity profiling as the number of time slot becomes large. On the efficiency aspect, larger time slot number means more computation in temporal matching analysis, thus increase the training time. Therefore, 24 (1 hour per slot) and 48 (30-minute per slot) are relatively optimal time slot numbers which take account of both model performance and computation efficiency.



(a) Precision@1. (b) Precision@5. (c) Precision@10.

(d) Recall@1. (e) Recall@5. (f) Recall@10.

Figure 3.6: Precision and recall of proposed model with different weight φ of area-activity for mixing with category popularity $(1 - \varphi)$ (K=10).

3.5.6 Tuning the Weight φ for Area Activity

As shown in Figure 3.6, we tune the weight φ of area activity pattern to test the performances of our model. For profiling the final popularity patterns of POIs, we propose to combine the patterns of both area activity and category popularity with a mixing parameter φ. Recalling Equation (3.4), φ decides the mix ratio of the two patterns. For example, $\varphi = 0.8$ means we combine 0.8 times area activity effects and 0.2 times POI category effects to generate popularity patterns. Here we study what φ value gives good performances with four φ configuration: from $\varphi = 0.8$ which emphasizes more on area activity to $\varphi = 0.2$ which favors more on POI category. Here

we can see that $\varphi = 0.6$ provides the best performance and $\varphi = 0.4$ gets the second place. From the observation, we can conclude that both the area activity and the category popularity provide the important knowledge for POI popularity profiling. Specifically, area activity studied by taxi trips makes the larger contribution in POI profiling.

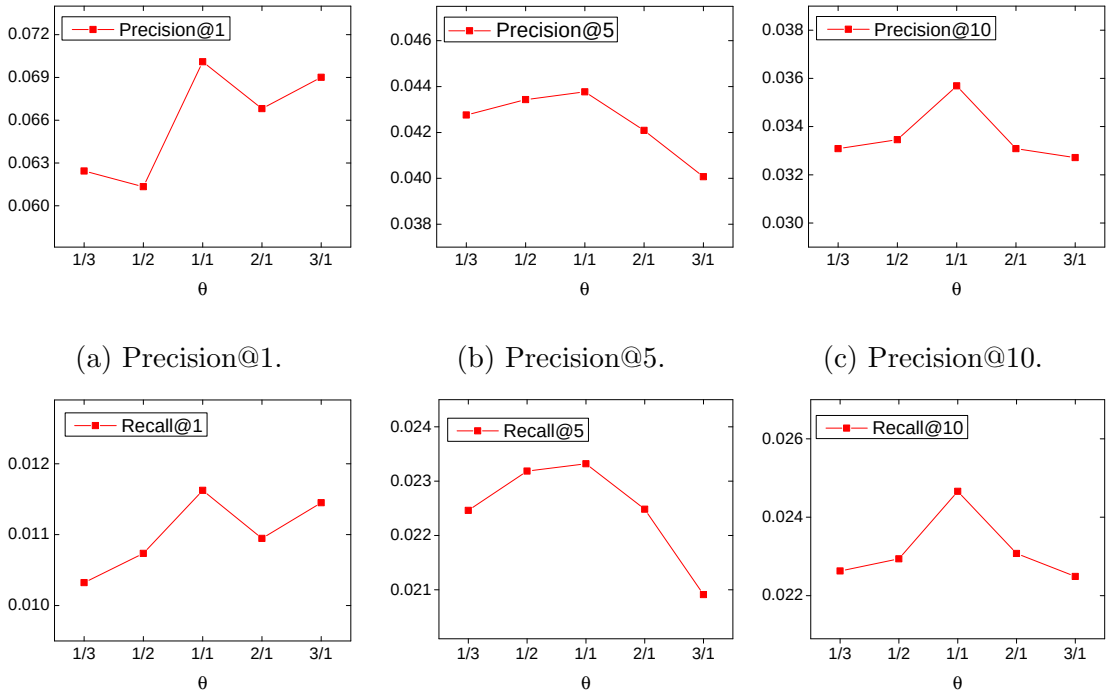
3.5.7 Tuning the Ratio θ of Weekday to Weekend

In our original configuration, we assume that each day of the week has the same weight for modeling user-POI temporal matching. Therefore, we set the matching score of a user-POI pair $m(i, j)$ in Equation (3.9) to have $\gamma = \frac{5}{7}$ because of 5 weekdays in a week while $1 - \gamma = \frac{2}{7}$ comes from two days in weekend. However, people may have unbalance weight on the days of weekdays or weekends. In this study, we want to tune the trade-off between weekdays and weekend to explore the day importance of LBSN users in New York City. We use the ratio θ to denote the trade-off:

$$\theta = \frac{Day^{wd}}{Day^{we}}, \tag{3.23}$$

where Day^{wd} denotes the weight of a day of weekday and Day^{we} denotes the weight of a day of weekend. For example, if $\theta = \frac{1}{2}$ which means a day of weekend is twice important than a day of weekday, we have $\gamma = \frac{5}{9}$ for weekday and $1 - \gamma = \frac{4}{9}$ for weekend by considering 5 days as weekday and 2 days as weekend in a week.

Figure 3.7 shows the performance comparison of different weight ratios θ. We test θ from 1/3 which means a day of weekend is three times more important than a day of weekday to 3/1 which means the opposite. The top-N performances in terms of precision, recall are visualized. We can observe that the performance at $\theta = 1/1$



(a) Precision@1. (b) Precision@5. (c) Precision@10.

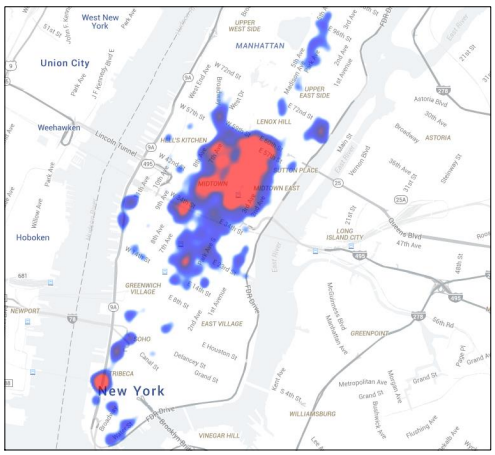
(d) Recall@1. (e) Recall@5. (f) Recall@10.

Figure 3.7: Precision and recall of proposed model with different weekday/weekend ratio θ for computing temporal match scores (K=10).

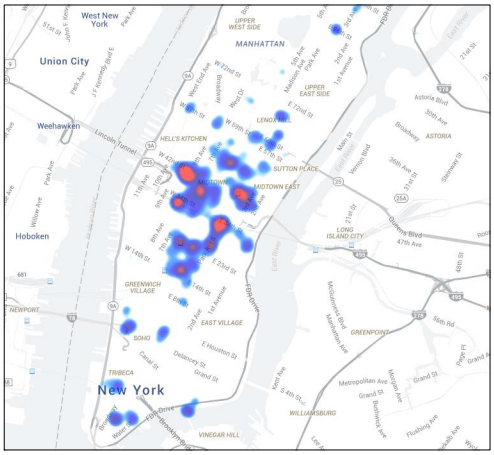
achieves the highest, which means the importance of weekdays and weekends are almost the same for modeling the user-POI temporal compatibility. Also, as the ratio θ goes more and more unbalance, the performance becomes worse generally, except precision and recall @1 from $\theta = 2/1$ to $\theta = 3/1$. From this study, we can see that large city such as NYC provides rich lifestyles in weekdays as in weekends.

3.5.8 Correlation between Taxi Trips and LBSN Check-ins

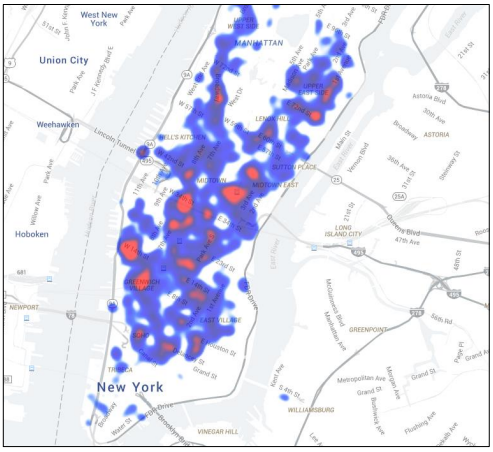
Here we conduct a case study to explore the spatio-temporal correlation between heterogeneous taxi rider mobility and LSBN user check-in behavior. We randomly sample taxi drop-offs and POI check-ins during different time period of weekday and



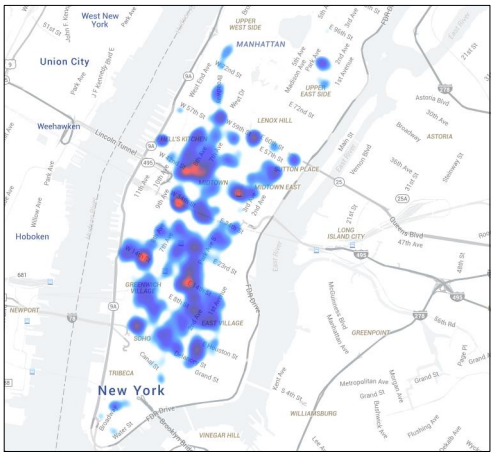
(a) Taxi drop-off 8–9AM.



(b) Check-in 8–9AM.



(c) Taxi drop-off 8–9PM.



(d) Check-in 8–9PM.

Figure 3.8: Heatmap of mobility of taxi riders and LBSN users at different time slot of weekday.

plot their locations to make the heatmaps. Figure 3.8 shows the heatmaps of two different period: 8AM–9AM and 8PM–9PM in weekdays. For visualization purpose, we only show the heat color on relatively high density areas, therefore the plain areas do not mean there are no events of drop-off or check-in.

Figure 3.8a, 3.8b show the human mobility of taxi riders and LBSN users in morning hour 8AM to 9AM. We can find that in the morning people are mainly concentrated in two business regions of NYC: Midtown (area around Rockefeller Center) and Financial District (area around Wall Street). Because of worse traffic, people who take taxi to Financial District are fewer than those take taxi to Midtown. Compare the taxi drop-off with check-in in morning hour, we can see that the taxi riders are relatively strong co-located with LBSN user mobility. Figure 3.8c, 3.8d show the taxi drop-offs and check-ins in night hour 8PM to 9PM. We can see that people leave business regions where they work in daytime. Meanwhile, they are going to residential areas (e.g., Upper Each, Upper West) where do not have many POIs for check-in as well as restaurant & nightlife areas (e.g., Greenwich Village, East Village, Fashion District) where have dense check-ins. Significantly, the check-in distribution is spatio-temporal correlated with the mobility trend of taxi riders. This case study supports our idea that the temporal patterns of LBSN users' check-in behavior are predictable via heterogeneous massive human mobility data, which can be utilized for boosting recommendation performance.

3.6 Related Work

In this section, we introduce the related work from three research angle: personalized recommendation methodology especially latent factor model, temporal influence en-

hanced POI recommender system, and human mobility analysis especially in LBSN environment.

Collaborative filtering technique, especially the factorization based approach has shown its importance to the field of recommender system. It has been widely used for various classic recommendation algorithms. The basic factorization algorithms include matrix factorization (Koren, Bell, & Volinsky, 2009), probabilistic matrix factorization (Mnih & Salakhutdinov, 2008) and its Bayesian version (Salakhutdinov & Mnih, 2008), as well as other variants (Agarwal & Chen, 2009; Koren, 2008). Most of these algorithms are majorly developed for explicit user response (e.g., rating), and assumes that the responses follow a Gaussian distribution over the predicted preferences. As more and more emerging recommendation applications which only have implicit user responses (e.g., count of web-click or check-in) came to the research filed, recommender systems are also required to infer user preferences from these heavy skew and wide range data. However, Gaussian-based latent factor models show their limitation on prediction performance. Under this circumstance, researchers developed latent factor models which is more suitable for implicit responses by setting non-negative constraints on latent variables (Gu, Zhou, & Ding, 2010; Lee & Seung, 2001; Zhang, Wang, Ford, & Makedon, 2006), which aims to force the predicted preferences into a wider range to adapt implicit responses. Furthermore, by better modeling heavy skew data and providing rigorous probabilistic generative process, Poisson distribution became popular in recommendation modeling especially for implicit response (Ma, Liu, King, & Lyu, 2011; Chen, Kapralov, Canny, & Pavlov, 2009; Liu, Xiong, Papadimitriou, Fu, & Yao, 2015; Liu, Kong, et al., 2015).

The second group is more specific to incorporate temporal influences into recommender system for better understanding users' temporal preferences. The first category can be summarized as time-aware recommendation which learns temporal preferences to recommend items for specific time slots (e.g., an hour of a day). The early work in (Koren, 2010; Ding & Li, 2005) discover the dynamic of user preference or interests over time. More recently, researchers start to investigate periodic patterns of user preferences (e.g., hourly interests of every day). One direct solution is to add an time dimension to user-item matrix and apply tensor factorization (Xiong et al., 2010; Bhargava, Phan, Zhou, & Lee, 2015). The work in (Gao et al., 2013a) considers a user's separated latent variables at different time slots, and preserves the similarity of personal preference in consecutive times. The work in (Q. Yuan et al., 2013a) makes time-aware recommendations by a user-based collaborative filtering method which computes the similarity between users by finding the same POIs at the same times in their check-in history. The work in (McInerney, Zheng, Rogers, & Jennings, 2013) learns temporal preferences by adopting topic model and training unique temporal features for each topic. Relevant work can also be found in (Q. Yuan, Cong, Ma, Sun, & Thalmann, 2013b; Gao, Tang, Hu, & Liu, 2013b). Our work is mostly related to this category. Meanwhile, there exists the other category which can be concluded as successive POI recommendation. The objective of this category aim to learn sequential patterns to predict user preferences for next POI, such as the work in (C. Cheng, Yang, Lyu, & King, 2013; Mathew, Raposo, & Martins, 2012) which train personalized Markov chain to capture sequential check-in preferences. More relevant work can be found in (Noulas, Scellato, Lathia, & Mascolo, 2012; Gao, Tang, & Liu,

2012).

The last group of research concentrates on human mobility analysis of LBSN users. The work in (D. Wang, Pedreschi, Song, Giannotti, & Barabasi, 2011) shows that users' mobility similarity is strongly correlated with their social proximity. The work in (Cho et al., 2011) utilizes Gaussian mixture model to capture users' periodic mobility at different states (e.g., home/work). The work in (Noulas, Shaw, Lambiotte, & Mascolo, 2015) explores the connectivity among urban places via the mobility of LBSN users. The work in (Y. Wang et al., 2015) uses heterogeneous mobility data to measure a static connectivity among areas for boosting the performance of user location prediction. The work in (N. J. Yuan et al., 2013) explores and categorize urban lifestyles with the mobility of LBSN users. More relevant work can be found in (Song, Qu, Blumm, & Barabási, 2010; Gonzalez, Hidalgo, & Barabasi, 2008).

3.7 Concluding Remarks

In this chapter, we developed a POI recommendation model by considering the temporal matching between users and POIs. Firstly, we presented a method to profile the temporal popularity of POIs by (i) mining area activity patterns with taxi trips, (ii) integrating category popularity pattern with POI category level check-ins, and (iii) enhancing patterns with mixture mode. Moreover, we learned the latent temporal regularity of users by incorporating the temporal matching degrees of user-POI pairs into user overall preference estimation. Finally, we conducted extensive experiments with POI check-in and human mobility data. As demonstrated by the experimental results, the consideration of temporal matching between users and POIs can bet-

ter model LBSN users' choosing processes. The performance improvement of our proposed method is substantial compared to benchmark methods.

CHAPTER 4

REPRESENTING URBAN FUNCTIONS THROUGH ZONE EMBEDDING WITH HUMAN MOBILITY PATTERNS

Urban functions refer to the purposes of land use in cities where each zone plays a distinct role and cooperates with each other to serve peoples various life needs. Understanding zone functions helps to solve a variety of urban related problems, such as increasing traffic capacity and enhancing location-based services. Therefore, it is beneficial to investigate how to learn the representations of city zones in terms of urban functions, for better supporting urban analytic applications. To this end, in this chapter, we propose a framework to learn a vector representation (embedding) for city zones by exploiting large-scale taxi trajectories. Specifically, we extract human mobility patterns from taxi trajectories, and use the "co-occurrence" of origin-destination zones to learn zone embeddings. To utilize the spatio-temporal characteristics of human mobility patterns, we incorporate mobility direction, departure/arrival time, destination attraction, and travel distance into the modeling of zone embeddings. We conduct extensive experiments with real-world urban datasets of New York City to evaluate our proposed method. Experimental results demonstrate the effectiveness of the proposed embedding model to represent urban functions of zones with human mobility data.

4.1 Introduction

A city consists of a variety of zones providing different functions to support diverse demands of urban residents, such as working, recreation, and residence. Studying the urban functions of city zones provides indispensable information which is useful in solving many urban challenges, therefore plays a critical role in urban analytics. Recent years, the advent of sensing technologies and mobile computing has accumulated a variety of data related to human mobility in urban areas. As a result, data-driven approaches have been increasingly applied to explore urban functions of cities.

While the literature has shown promising effectiveness of analyzing massive positioning data for urban exploration (Cranshaw et al., 2012; J. Yuan et al., 2012; Z. Cheng et al., 2011; Silva et al., 2012), there are limited studies aiming to provide an integrated and principled approach to the representation learning of city zones in terms of urban functions. In this chapter, we aim to propose an effective solution to learn the distributed and low-dimensional embeddings of city zones. Zones with similar urban functions are geometrically closer in the embedding space. Using zone embeddings, we are able to identify functional regions of cities which consist of several zones with similar functions. Furthermore, many analytic models can be empowered by using these extracted representations as inputs.

Generally, there are two critical challenges toward learning effective zone representations of urban functions: (i) how to infer urban functions: through intra-zone human activity or inter-zone human mobility; (ii) how to effectively exploit human mobility patterns containing spatio-temporal characteristics for the zone embedding.

First, by providing intelligence to profile human activity categories (e.g., shopping) within city zones, location-based social network (LBSN) data have been widely used in urban analytics. However, LBSN data has reliability issues due to (i) biased check-ins in entertaining related Point-of-Interest (POI) categories; (ii) sparse check-ins outside hot areas or time periods; and (iii) significant discrepancies between check-ins and actual user mobility (G. Wang, Schoenebeck, Zheng, & Zhao, 2016). More importantly, by analyzing intra-zone human activities, it is difficult to obtain association information between any two zones, which is essential in zone embedding learning for capturing "contexts". On the contrary, human mobility data (e.g., vehicle trajectories) reveal important associations between any two zones through origin-destination human mobility patterns which are mainly function dependent (e.g., commuters usually travel to an office zone around 9 a.m. from residential or transportation zones). In addition, human mobility data usually cover a wider range of areas and time periods. Therefore, analyzing human mobility has a potential to better learn urban functions with zone embeddings.

Second, an effective framework is highly needed to learn zone embeddings with human mobility patterns across city zones. For this purpose, we bring in the idea of word2vec (Mikolov, Chen, Corrado, & Dean, 2013; Mikolov, Sutskever, Chen, Corrado, & Dean, 2013), which is originally a Natural Language Processing (NLP) model for word semantic learning. In word2vec, the embedding of a word is learned from its co-occur words which appear nearby in sentences. Therefore, two semantically similar words are likely to share similar vector representations. For example, "queen" and "king" would be close in the embedding space because both co-occur with the

same nearby word "kingdom" frequently. Similarly, by analyzing a zone based on its "co-occur" zones between which human mobility patterns exist (e.g., a residential zone and an office zone "co-occur" when people travel from one to another), we can learn zone embeddings by analyzing association strength of every zone pair. With zone embeddings, zones with similar functions are geometrically close, whereas zones with different functions are distant.

Based on the above idea, a potential solution is to treat the origin-destination pair of a mobility pattern as a co-occurrence of two zones for learning zone embeddings. However, urban functions are also jointly reflected by mobility direction, and departure/arrival time. In other words, the embedding method should be able to take into account "leaving for" and "arriving from" at "different time" for modeling a zone co-occurrence. To that end, we define a set of human mobility events which contain zone, time and status of mobility patterns, to serve as embedding "contexts" of target zones, for incorporating zone co-occurrence with spatio-temporal characteristics. In addition, during the learning of zone embeddings, we give different importance to different co-occurrences by calculating the travel demand of origin-destination pairs with destination attraction (e.g., total mobility pattern arrivals) and travel distance (e.g., average mobility pattern length) information.

Along these lines, in this chapter, we present a novel human mobility based zone embedding framework to represent urban functions with distributed and low-dimensional vectors. Specifically, we develop a co-occurrence based method to learn zone embeddings with human mobility patterns. Based on that, we take into account mobility direction and departure/arrival time to model spatio-temporal co-occurrence

of zones, and jointly incorporate destination attraction and travel distance to give different co-occurrences different importance in the embedding learning. Finally, we conduct extensive experiments with real-world urban datasets of New York City to show the effectiveness of the proposed method.

4.2 Preliminary

Recent years, representation learning have been making large progress in NLP (Bengio, Ducharme, Vincent, & Jauvin, 2003; Collobert & Weston, 2008; Bengio, Courville, & Vincent, 2013). Learning vector representation of words aims to extract useful semantic information from corpus by embedding vocabulary into vector space. The semantic similarity between any two words can be measured by the distance between two corresponding vectors. Word2vec method (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013) which trains a two-layer neural network to learn distributed embedding of words has been successful in many NLP tasks such as word analog. The idea to utilize the sequential order of words appear in sentences or documents, by assuming that every word and its surrounding words (i.e., context words) have dependencies with each other. By optimizing word embeddings based on observed word-word dependencies (i.e., co-occurrence), we extract the vector representations of semantic meanings of every word. Next we will introduce the techniques that are related to our work.

4.2.1 Skip-Gram

Proposed in (Mikolov, Sutskever, et al., 2013), Skip-gram is one of the original models of word2vec. For each word, it learns vector representation which can well predict its

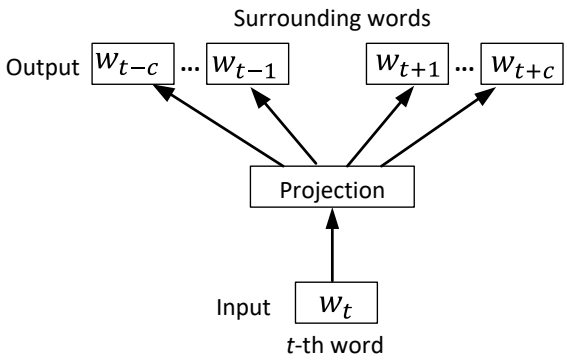


Figure 4.1: Skip-gram model.

surroundings words. As shown in Figure 4.1, from a training sentence or document, we can observe a sequence of words $w_1, w_2, w_3, ..., w_T$. The objective of Skip-gram is to maximize following average log probability:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t), \tag{4.1}$$

where c is a predefined size of context window. Then the conditional probability $p(w_{t+j}|w_t)$ can be defined using softmax function:

$$p(w_{t+j}|w_t) = \frac{\exp({v'_{w_O}}^{\top} v_{w_I})}{\sum_{w=1}^{W}\exp({v'_w}^{\top} v_{w_I})}, \tag{4.2}$$

where v_w and v'_w are the *input* and *output* vector representations of w while W is the vocabulary size. For each word, two vectors v_w and v'_w are learned to maximize the softmax probability of the co-occurrence of every word and its surrounding words. Finally, we take trained v_w as a word's embedding. In real training, since computing softmax for large vocabulary is too expensive, two techniques (i) Hierarchical softmax, and (ii) Negative sampling are developed to accelerate learning process. Here we will not extend the details of them.

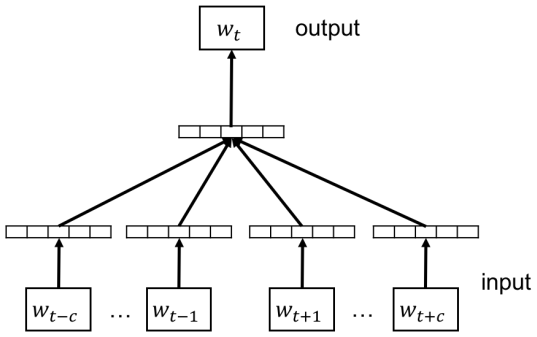


Figure 4.2: Continuous bag-of-word (CBOW) Model. The embedding v_{w_t} of each word w_t is learned by maximizing the co-occurrence of input word sequence $\{w_{t-c}, \cdots, w_{t-1}, w_{t+1}, \cdots, w_{t+c}\}$ with output word w_t.

4.2.2 Continuous Bag-of-Word (CBOW)

Continuous bag-of-word (CBOW) (Mikolov, Chen, et al., 2013) is another effective model of word2vec. As shown in Figure 4.2, a two-layer neural network aims to learn word embeddings from a sentence where the sequences of words $w_1, w_2, w_3, ..., w_T$ can be obtained. Each word w_t serves as the predicting target as the model output, the input of the model are the contexts (neighboring words) $w_{t-c}...w_{t-1}, w_{t+1}...w_{t+c}$ where c is a window size (e.g., 5) for contexts observation. The objective of CBOW is to predict targeting word given its context words. Formally, it aims to maximize the following log-likelihood:

$$\sum_{t=1}^{T} \log p(w_t | w_{t-c}...w_{t-1}, w_{t+1}...w_{t+c}), \tag{4.3}$$

where c is the predefined context window, and the conditional probability $p(w_t|w_{t-c}...w_{t-1}, w_{t+1}...w_{t+c})$ is generally modeled using a softmax function:

$$p(w_t | w_{t-c}...w_{t-1}, w_{t+1}...w_{t+c}) = \frac{\exp({v'_{w_t}}^{\top} v_I)}{\sum_{w=1}^{W} \exp({v'_w}^{\top} v_I)}, \tag{4.4}$$

where v and v' are the *input* and *output* vector representations respectively, and W is the vocabulary size. The input vector v_I is generated by averaging all context words' vector:

$$v_I = \frac{1}{2c} \sum_{-c \leq j \leq c, j \neq 0} v_{t+j}. \tag{4.5}$$

For each word in the sequence, input vector v_w and output vector ${v'}_w$ of words are optimized to maximize the conditional probability of the co-occurrence of every word and its surrounding words. After training all the words in corpus, input vector v_{w_t} is viewed as the word embedding for word w_t.

4.3 Problem Statement

We first define some concepts in our work, then we proceed to the problem statement of zone embedding learning.

Definition 4 *(**Human mobility pattern**) Given a taxi trip, we extract the human mobility pattern with the following information: (i) the origin-destination (O-D) pair of zone; (ii) the time of the departure (e.g., taxi passenger pick-up) and the arrival (e.g., taxi passenger drop-off); and (ii) travel distance.*

Definition 5 *(**Human mobility event**) Given a human mobility pattern, we extract two human mobility events: one for the departure and one for the arrival with the following information: (i) event occurrence zone; (ii) event occurrence time; and (iii) event status: a departure or an arrival.*

Given a set of city zones $Z = \{z_1, z_2, ..., z_N\}$ and a set of taxi trips. Each trip contains a passenger travel with the information of locations and timestamps for the

departure and the arrival. From taxi trips, we extract a set of zone level mobility patterns $P = \{p_1, p_2, ..., p_m\}$, where each mobility pattern $p = (p.z_O, p.z_D, p.t_O, p.t_D)$ includes the zones of origin $p.z_O$ and destination $p.z_D$, as well as the time of departure $p.t_O$ and arrival $p.t_D$. Each time is converted from a timestamp to a $\langle timeslot, daytype \rangle$ combination. We define $\{ts_1, ts_2, ..., ts_J\}$ to be a set of time slots of a day (e.g., 24 hours), and $\{wd, we\}$ to be a set of day types: weekday and weekend. Figure 4.3 illustrates the departure locations of human mobility events, and the city zones in urban areas.

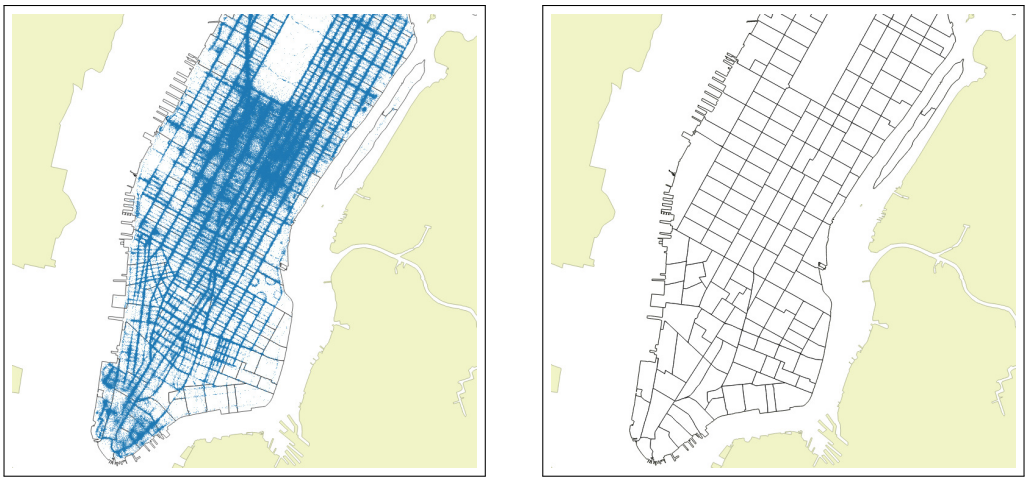
The objective is to learn the distributed and low-dimensional embeddings of city zones through the spatio-temporal human mobility patterns to represent their urban functions in a city.

4.4 Methodology

In this section, we start from the word2vec model. Then we show how to learn zone embeddings by incorporating spatio-temporal characteristics. Last, we present the model specification.

4.4.1 Word Embedding

A fundamental observation in word embedding literature is that semantically similar words often have similar "contexts" (i.e., the words appear around them) in sentences (Levy & Goldberg, 2014; Mikolov, Sutskever, et al., 2013) in a corpus. By modeling the association strength of each word pair based on the frequencies they co-occur within a small context window (e.g., a window size of 5 means 5 words behind and 5 words ahead of the target word are contexts), the embedding v_w of word w can be



(a) (b)

Figure 4.3: (a) Locations of taxi pick-up in New York City (NYC). (b) City zones of NYC.

learned by

$$v_w{}^{\intercal} v'_c \approx \mathrm{PMI}(\mathcal{D})_{w,c}, \tag{4.6}$$

where v'_c is the embedding of *context word* c.

$\mathrm{PMI}(\mathcal{D})$ is a $|W| \times |W|$ pointwise mutual information (PMI) matrix calculated by word co-occurrence frequencies in corpus $\mathcal{D}$ with vocabulary size $|W|$. Each PMI value in a $\langle w, c \rangle$ entry is computed as

$$\mathrm{PMI}(\mathcal{D})_{w,c} = \log \left(\frac{\#(w,c) \cdot |\mathcal{D}|}{\#(w) \cdot \#(c)} \right), \tag{4.7}$$

where $\#(w,c)$ counts the frequency that words w and c co-occur, and $\#(w)$, $\#(c)$ counts the number of single occurrences of words w and c. $|\mathcal{D}|$ is the total number of word-context pairs in the corpus.

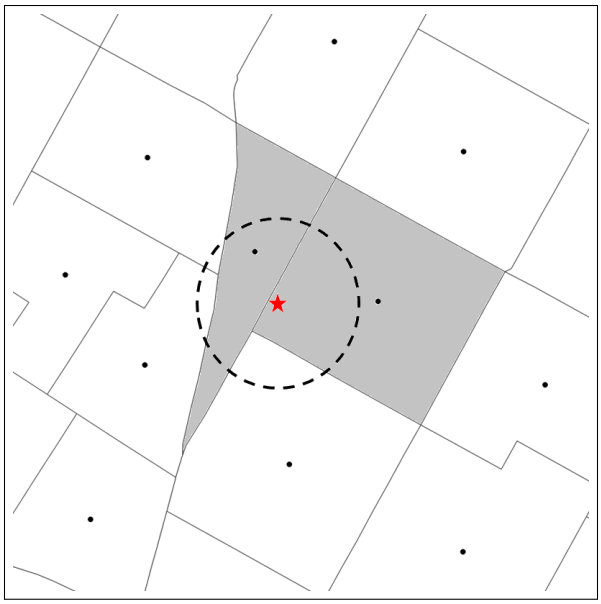


Figure 4.4: An example of finding relevant zones (gray zones) for pick-up/drop-off locations (red star). Dots are zone centers.

4.4.2 Extracting Zone Level Mobility Patterns

Given the city zones and taxi trips, we need to extract zone level mobility patterns, i.e., the pairs of zones as the origin and the destination of trips. A simple way can be only extracting the zones where the pick-up and drop-off of a trip locate. However, one characteristic of taxi trip data is that the pick-up and drop-off locations tend to concentrate around road network. As a result, many of them locate near the boundary of zones. Therefore, it is reasonable to consider adjacent city zones which may be relevant as well for pick-up and drop-off locations.

Motivated by the above, we first calculate the center location of city zones. Secondly, in addition to the zone where each pick-up/drop-off locates, we use a walking distance (e.g., 200 meters) to examine whether the centers of other zones are also located in the covered area. If yes, we count covered zones as relevant as well, as

shown in Figure 4.4.

Therefore, for each trip's pick-up or drop-off location, we consider all potentially relevant zones (one or more):

$$zone(l) = \{l.z_1, l.z_2, l.z_3..., l.z_N\}, \tag{4.8}$$

where l means the location of a pick-up/drop-off, $l.z$ means the relevant zones of it.

Last, by pairing each relevant zone of the pick-up with each relevant zone of the drop-off in a trip, we obtain the zone level human mobility patterns:

$$p = (p.z_O, p.z_D, p.t_O, p.t_D), \tag{4.9}$$

where $p.z_O, p.z_D$ are city zones of origin and destination. $p.t_O, p.t_D$ are times of departure and arrival which are converted from timestamps to $\langle timeslot, daytype \rangle$ combinations. We define $T = \{t_1, t_2, ..., t_S\}$ to be a set of time slots of a day (e.g., 24 hours), and $D = \{wd, we\}$ to be a set of day types: weekday and weekend.

4.4.3 Spatio-Temporal Zone Embedding

We propose to learn the embedding of zones from its associated zones based on human mobility. People leave zones and arrive at zones at different time for different trip purposes, which reveals a function dependent association between zones. As shown in Figure 4.5, a transportation zone of Pennsylvania train station in NYC usually shows a heavy mobility volume heading to office zones in the midtown in weekday mornings (8-9 a.m.), and shows a huge mobility volume returning from office zones after working (5-6 p.m.). Therefore, we could define that when two zones form a origin-destination pair, one zone co-occurrence happens. By counting the number

(a) (b)

Figure 4.5: An example of top mobility patterns in a transportation zone (red): (a) people move out to office zones (green) in 8-9 a.m., and (b) move back (yellow) in 5-6 p.m..

of origin-destination patterns, the frequency of zone co-occurrence can be obtained. Then zone embeddings can be learned by factorizing PMIs of zone pairs.

However, a problem of using the above simple zone co-occurrence is that the critical spatio-temporal characteristics of mobility patterns (e.g., mobility direction and departure/arrival time) are not incorporated into the embedding learning. For example, given a mobility pattern which involves zone A and zone B, to study the zone functions, the embedding framework not only needs to incorporate the co-occurrence of A and B, but also the corresponding mobility direction (e.g., A $\rightarrow$ B) and the departure/arrival time (e.g., at night).

To address this problem, we define human mobility events E to serve as embedding contexts (i.e., the mobility events appear with zones in mobility patterns) so that the

model includes zone co-occurrence, mobility direction, and departure/arrival time in the embedding learning. Specifically, a mobility event e is defined as follows:

$$e = (e.z, e.t, e.sta), \tag{4.10}$$

where $e.z$ is the zone, $e.t$ is the time by time slot and day type, and $e.sta \in \{arrive, leave\}$ is the mobility status which shows direction of the human mobility pattern. For example, a human mobility event can be defined as $\langle z_i, 12p.m.\text{-}weekday, arrive\rangle$ which means a human mobility pattern arrive at zone z_i during the time slot of 12 p.m.-1 p.m. on weekday.

From each mobility pattern $p = (p.z_O, p.z_D, p.t_O, p.t_D)$, we obtain two co-occurrences consisting of a zone as embedding target and a mobility event as embedding context:

zone	mobility event
$p.z_O$	$(p.z_D, p.t_D, arrive)$
$p.z_D$	$(p.z_O, p.t_O, leave)$

Using this new type of co-occurrences, we incorporate mobility direction and departure/arrival time of human mobility patterns to enable zone embedding.

Table 4.1 shows an analog between word embedding and zone embedding. In word embedding, every word serves as a target word and a context word. The association between a target word and a context word is learned from the frequency of word co-occurrences in sentences. Unlike traditional word embedding framework, city zones only serve as target "words" and mobility events only serve as context "words" in zone embedding. Accordingly, we use the co-occurrences of a zone and a mobility event in human mobility patterns to learn the pairwise zone associations.

Table 4.1: Analog between word embedding and zone embedding.

	word embedding	zone embedding
target	word	zone
context	word	mobility event
co-occur in	sentence	mobility pattern

4.4.4 Modeling Importance of Co-occurrences

A unique characteristic of zone embedding is that different zones in context has different impacts based on how attractive the destination is and how far the mobility travels. Based on that, we propose to give different zone co-occurrences different importance for better optimizing zone embeddings.

For each origin-destination (O-D) pair of zones, we calculate the travel demand between them based on two factors: (i) destination attraction (e.g., total mobility patterns arrival at destination zones) and (ii) travel distance (e.g., average mobility pattern travel distance) with the gravity model (Cascetta, Pagliara, & Papola, 2007) in traffic analysis. Then we use the travel demand of the O-D zone pair to guide its importance of co-occurrence in the embedding learning with a 0-1 weight.

Specifically, we define G as a $|Z| \times |Z|$ gravity matrix ($|Z|$ is the total zone number) where row dimension means each zone as an origin, column dimension means each zones as a destination. Each row of G is a distribution for the 0-1 probabilities of every destination zone z_D to attract a mobility pattern from a specific origin zone z_O,

with sum of 1.

In detail, a $\langle z_O, z_D \rangle$ entry of G is calculated as

$$G(z_O, z_D) = \frac{A_{z_D} F_{z_O, z_D}}{\sum_{z \in Z} A_z F_{z_O, z}}, \tag{4.11}$$

where A_z denotes the total number of mobility patterns arrive at zone z . F_{z_O, z_D} is the friction factor to serve as a cost for traveling between two zones, which is calculated based on the travel distance with a negative exponential function:

$$F_{z_O, z_D} = e^{-\beta d_{z_O, z_D}}, \tag{4.12}$$

where d_{z_O, z_D} is the travel distance between zone z_O and z_D calculated by averaging the travel distances of mobility patterns. β is the parameter which is obtained by minimizing

$$\sum_{z_O \in Z} \sum_{z_D \in Z} \left(T_{z_O, z_D} - \hat{T}_{z_O, z_D}\right)^2 \tag{4.13}$$

with a genetic algorithm (Deb, Pratap, Agarwal, & Meyarivan, 2002), where T_{z_O, z_D} is the observed mobility pattern number from z_O to z_D. $\hat{T}_{z_O, z_D}$ denotes the estimated mobility pattern number calculated by $\hat{T}_{z_O, z_D} = P_{z_O} \frac{A_{z_D} F_{z_O, z_D}}{\sum_{z \in Z} A_z F_{z_O, z}}$, where P_{z_O} is the total number of mobility patterns leave z_O. Also, since the mobility distributions are usually different on weekday and weekend, we calculate the gravity matrices G^{wd} and G^{we} by day types.

4.4.5 Model Specification

To learn the zone embeddings proposed in our framework, we minimize the following objective function over all co-occurrence of zones and mobility events:

$$\min_{V, V'} \frac{1}{2} \sum_{z \in Z} \sum_{e \in E} \left(M(z, e) - {v_z}^\intercal {v'}_e\right)^2 \cdot G^*(z_O, z_D), \tag{4.14}$$

where M denoted the $|Z| \times |E|$ matrix of positive pointwise mutual information (PPMI) which measures every co-occurrence of zones and mobility events, and $G^* \in \{G^{we}, G^{wd}\}$ denotes the gravity matrix for every origin-destination zone pair by day types.

In the first part, each value of M is computed as $M(z,e) = max\Big(0, \log(\frac{\#(z,e)\cdot|\mathcal{T}|}{\#(z)\cdot\#(e)})\Big)$, where $\#(z,e)$ counts the number of times that zone z and mobility event e co-occur, $\#(z), \#(e)$ count the numbers of single occurrence of z and e, and $|\mathcal{T}|$ is the observed number of co-occurrence from all mobility patterns.[1] v_z is the D-dimensional embedding of zone z and ${v'}_e$ is the D-dimensional embedding of mobility event e. We factorize PPMI matrix M into v_z and ${v'}_e$ by minimize the square error for all co-occurrences.

In the second part, $G^*(z_O, z_D)$ denotes the weight corresponding to the current co-occurrence which is retrieved by the origin-destination zones z_O,z_D and the day type $* \in \{wd, we\}$ interpreted from the current target zone and mobility event. We use it to apply different importance on co-occurrences to guide the optimization of zone embeddings

Parameter Estimation: Given the objective function in Equation 4.14, we take derivatives with respect to v_z and ${v'}_e$, and adopt gradient descent method to optimize embeddings. For experimental setup, we empirically set embedding dimension $D = 50$. Gravity matrices G^{wd} and G^{we} are calculated with $\beta^{wd} = 0.4674$ and $\beta^{we} = 0.3881$.

[1] We use PPMI to obtain a stable value because PMI can result in large negative values caused by log operation.

4.5 Evaluation

In this section, we empirically evaluate the performance of our proposed methods. We perform all the experiments on real-world datasets of New York City (NYC). We choose NYC because of following reasons: (1) NYC contains massive population, diverse urban functions and highly active economics. (2) Our experimental data has the best coverage in NYC, therefore we are able to have convincing results.

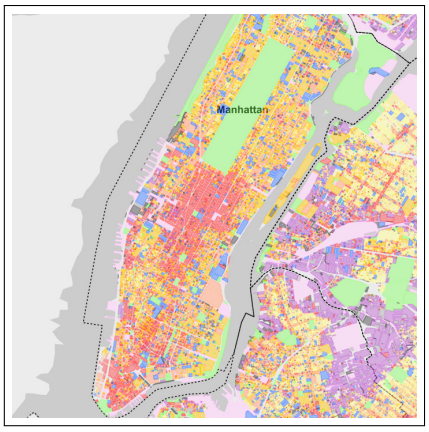
4.5.1 Data Description

The first dataset is the human mobility data. We use the trip records of yellow taxi from NYC taxi & limousine commission[2] to obtain people's mobility patterns. Since people in NYC seldom own cars, taxi is one of the most frequent and representative ground transportation choice. We use complete trip records of three month (June to August) of 2013. Finally we obtain 33,842,934 trips as our training data. Each taxi trip contains the locations of an origin and a destination as well as timestamps and trip distance.

The second dataset is the zone data. We use the city zones designed by US Census Bureau[3] for zone embedding learning. Using these zones has three benefits: (i) they are professionally designed to contain homogeneous population and environment; (ii) they are separated by major road network therefore they make interpretable study results; (iii) their areas are relatively small therefore can provide relatively fine resolution of analytic results. At last we obtain 193 city zones for urban function

[2]http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

[3]https://catalog.data.gov/dataset



1-One & Two Family Buildings
2-Multi-Family Walk-Up Buildings
3-Multi-Family Elevator Buildings
4-Mixed Residential & Commercial Buildings
5-Commercial & Office Buildings
6-Industrial & Manufacturing Buildings
7-Transportation & Utility
8-Public Facilities & Institutions
9-Open Space & Outdoor Recreation
10-Parking Facilities
11-Vacant Land

(a) Land use map (b) Land use types

Figure 4.6: Land use of NYC for validation.

study.

Since we compare proposed model against state-of-the-art baseline methods which utilize Location-base social network (LBSN) data, such as check-ins and Point-of-interests (POIs), the last dataset is the Foursquare data formulated by the work in (Yang, Zhang, Zheng, & Yu, 2015b). The dataset includes the check-in data in NYC for 10 months during 2012 to 2013. Each check-in contains the information such as user ID, POI ID, location, timestamp and POI category. Finally we have 17,009 POIs across 139 fine grained categories, 1,046 LBSN users, and 109,073 check-ins.

4.5.2 Evaluation Metrics

For evaluation, we conduct k-means clustering on zone embeddings with multiple cluster number setting K. We wish that zones with similar functions are assigned to the same clusters. To validate zone clustering performance, we utilized the official land use dataset[4] of NYC as ground-truth of zone function. This dataset describes

[4]maps.nyc.gov/zola

the land use designation of every lot (e.g., land of a building) in 11 land use types as shown in Figure 4.6. We aggregate all lots on the zone level. Finally, for each zone, we have a vector which contains percentages of zone area for 11 different land use types. For obtaining the ground-truth cluster label for city zones, we apply k-means clustering on land use vectors with different cluster number K. Then we use following metrics to evaluate clustering results on proposed zone embeddings as well as baselines:

• **Normalized Mutual Information (NMI)**, defined as

$$\text{NMI} = \frac{I(L;C)}{[H(L) + H(C)]/2}, \tag{4.15}$$

where L is the set of true labels and C is the set of clusters. $I(L;C)$ denotes the sum of mutual information between any cluster c_i and any label l_j. $H(L)$ and $H(C)$ denote the entropy for labels and clusters, respectively. This metric evaluates the purity of clustering results from an information-theoretic perspective.

• **Adjusted Rand Index (ARI)**, defined as

$$\text{ARI} = \frac{RI - Expected\_RI}{Max\_RI - Expected\_RI}, \tag{4.16}$$

which is the corrected-for-chance version of the Rand index (RI). By viewing every pair of zones as a series of decisions, we can calculate correctness by $RI = \frac{TP+TN}{TP+FP+TN+FN}$, where TP/FP denotes true/false positive and TN/FN denotes true/false negative. ARI has a score between -1.0 and 1.0 that random labeling has an ARI close to 0 and 1 stands for perfect match.

• **F-measure**, defined as

$$F_\beta = \frac{(\beta^2 + 1) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}, \tag{4.17}$$

where $Precision = \frac{TP}{TP+FP}$ and $Recall = \frac{TP}{TP+FN}$. Similarly to ARI, we view the clustering result of each pair of zones as a decision, then we can have precision and recall. F-measure is the harmonic mean of precision and recall. We put more emphasis on precision than recall by $\beta = 0.5$.

• **Cluster Internal Difference (CID)**, defined as

$$\text{CID} = \frac{1}{N} \sum_{i}^{N} d(z_i, c_{z_i}), \tag{4.18}$$

where N is the total number of zones. z_i is the land use vector of zone i and c_{z_i} is the average land use vector of i's assigned cluster. $d(x, y)$ calculates the Euclidean distance between the two vectors. Finally, we obtain the average distance from each zone's land use to its cluster's average land use. A smaller value means the zones assigned to the same cluster are more similar in terms of urban functions.

4.5.3 Baseline Approaches

The experimental study compares our proposed Mobility-based Zone Embeddings (**ZE-Mob**) with the following zone representations for zone clustering.

• **TF-IDF (POI)**: A direct approach is to represent urban functions is using in-zone human activity types from LBSN data (e.g., POI category). We use Term Frequency-Inverse Document Frequency (TF-IDF) to measure the importance of different POI categories ("term") to a zone ("document"). Specifically, each zone can be represented by a $|C|$-dimensional vector where $|C|$ is the total number of unique POI categories.

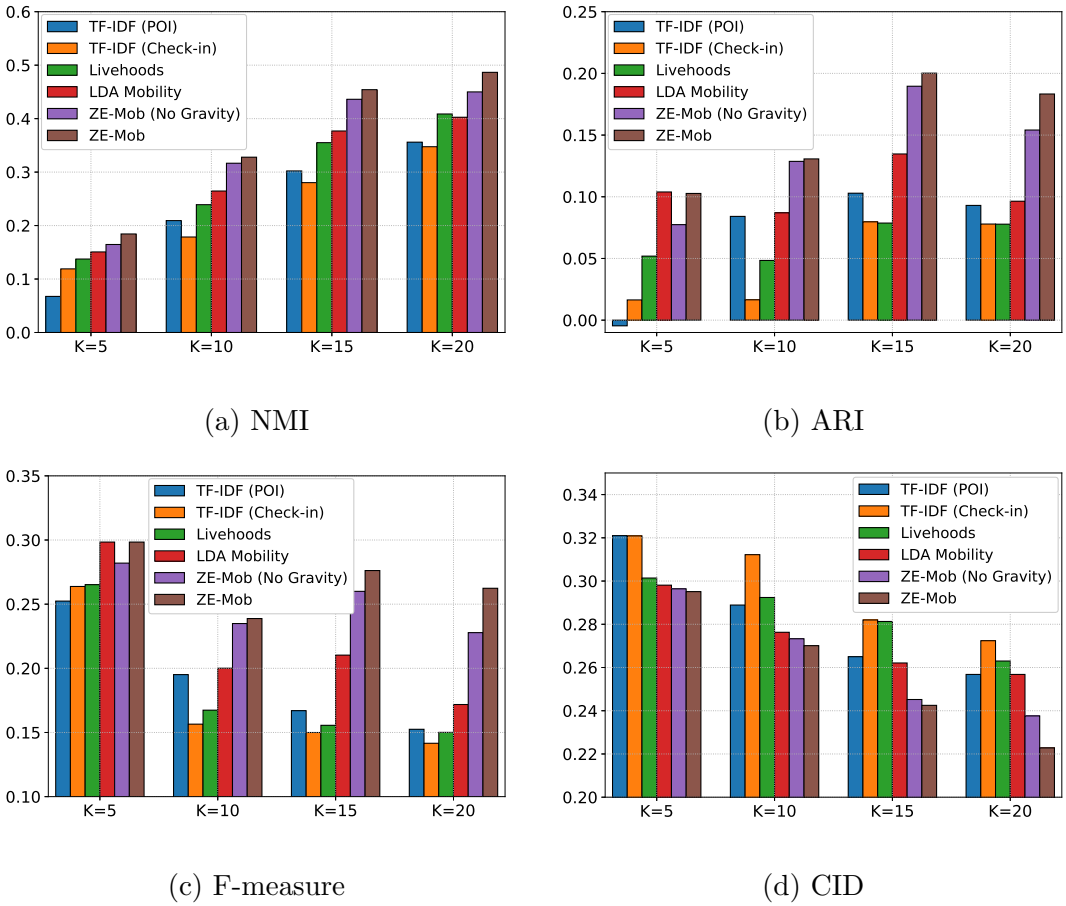
Each value indicates the TF-IDF of a category which is calculated based on the number of its corresponding POIs. Then we apply k-means on TF-IDF vectors for zone clustering.

• **TF-IDF (Check-in)**: This baseline adopts the same method to calculate TF-IDF. The difference is that we use check-in instead of POI to count the term frequency of categories.

• **Livehoods**: The work in (Cranshaw et al., 2012) proposes a clustering method to POIs by utilizing social similarity (e.g., check-in by the same user) and geographical proximity (e.g., distance between two POIs) in LSBN data. We aggregate POIs on the zone level and set locations of zones by zone centers to apply this baseline.

• **LDA Mobility**: The work in (J. Yuan et al., 2012) proposes to mine human mobility data to learn zone urban function by Latent Dirichlet allocation (LDA) model. The idea is to view zones as "document" and mobility events as "word". By having each zone as a "bag of mobility events" with observed frequencies, we learn a vector of latent urban function topics. Then we apply k-means on topic vectors for zone clustering.

• **ZE-Mob (No Gravity)**: This baseline is the same with propose method, except that we remove the impact of gravity matrices G^*. Therefore, we do not model the importance of zone co-occurrence in embedding learning with incorporation of destination attraction and travel distance.



(a) NMI

(b) ARI

(c) F-measure

(d) CID

Figure 4.7: Performance comparisons on different cluster number K.

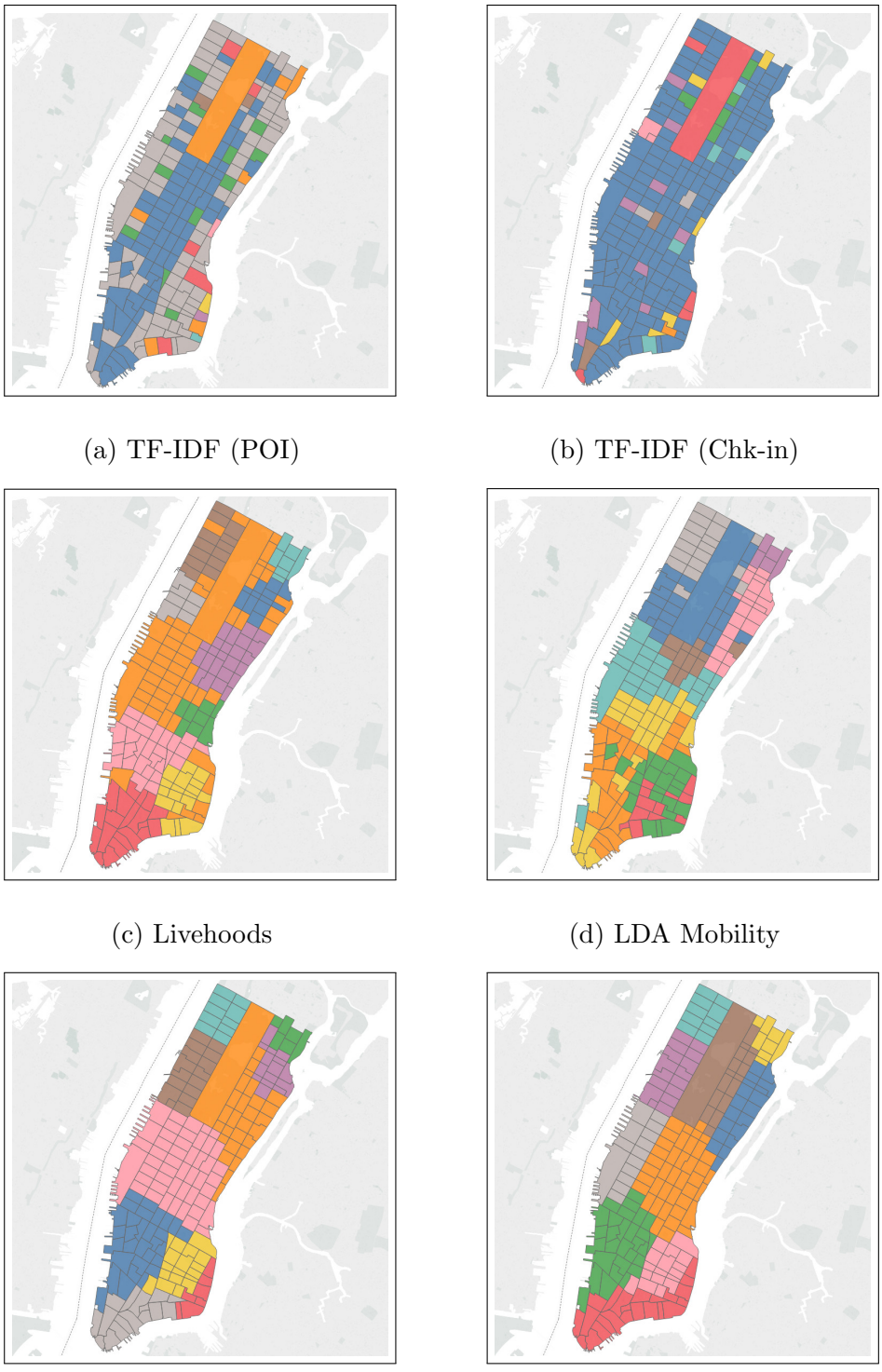
4.5.4 Performance Comparisons

Figure 4.7 shows the Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), F-measure, and Cluster Internal Difference (CID) of zone clustering results on all approaches with 4 different cluster number K: 5, 10, 15, and 20. Overall, we can see that our proposed approach ZE-Mob outperforms baseline methods on all metrics and all cluster number K.

Specifically, TF-IDF (POI) does not perform well. An important reason is that POIs in unpopular places such as residential areas are sparse. Meanwhile, since most of POIs concentrate in several categories (e.g., dinning), even the zones with sufficient POIs can not be differentiated by TF-IDF vector effectively. TF-IDF (check-in) performs the worst because check-in records are quite uneven on different POIs, which makes popular zones and unpopular zones more indistinguishable. Livehoods gives similar or better results than TF-IDF (POI) methods by modeling affinity value based on user sharing and geographical proximity. LDA Mobility gives better performance by learning latent urban function topics. Last, ZE-Mob and ZE-Mob (No Gravity) achieve the best performances. Based on the better performance of ZE-Mob with gravity, we validate that the incorporation of co-occurrence importance effectively improves zone embeddings.

4.5.5 Identifying Functional Regions

We perform k-means clustering (K=10) on zone embeddings to identify functional regions of NYC, such as residential regions, entertaining regions, and so on. As shown in Figure 4.8, each cluster is denoted by a color and used to identify a functional



(a) TF-IDF (POI)

(b) TF-IDF (Chk-in)

(c) Livehoods

(d) LDA Mobility

(e) ZE-Mob (No G.)

(f) ZE-Mob

Figure 4.8: City zone clustering ($K = 10$) by different methods. Each cluster is denoted by a unique color.

region. With TD-IDF methods (POI and check-in), the functional regions are weakly identified since the city is mainly partitioned into the zone clusters with POI and the zone clusters without POI. With livehoods, the clustering does better job than TF-IDF using the same LBSN data, because it considers geographic proximity in the clustering analysis. However, its algorithm prevents POIs with far distances to be in the same cluster, as a result, the shape of clusters can not be extended which does not fit the real situation of NYC. LDA Mobility gives better performance than all LBSN based methods by utilizing mobility data with topic modeling. We can see that it makes more reasonable identification than Livehood. However, some clusters are still mixed with each other, which make functional regions not clear enough. ZE-Mob and ZE-Mob (No Gravity) give the most satisfied functional region identification. For both approaches, we can see that upper town, middle town, lower town are clearly identified, and the lower town is correctly separated into the Financial District and the East/West Village (two famous entertaining regions). Compared with ZE-Mob, ZE-Mob (No Gravity) assigns all mid-town zones into a single cluster, but different functions exist on the east side (manufacturing region) and the west side (commercial region) refers to the land use map in Figure 4.6a. Also, ZE-Mob (No Gravity) incorrectly merges the 5th Ave shopping region into the Upper East region which mainly provides a residential function.

4.6 Related Work

In the section, we introduce the related works from three research angles: urban function learning, human mobility analysis, and embedding learning techniques.

Urban function study is an important research topic for city planners and urban designers in a long time for supporting decision making of city development. Early studies mainly rely on classic theory, long-term observation, and case-by-case survey for investigation. The work in (Goddard, 1970) surveys the taxi flow to analyze complex linkage system exists in center of London to study the location of activities. The work in (Putnam, 2001) discusses the change of community in a perspective of people's social interactions. More recently, a series of work (Cranshaw et al., 2012; J. Yuan et al., 2012; N. J. Yuan et al., 2015; Kling & Pozdnoukhov, 2012) use large-scale positioning data to perform data-driven urban function analysis.

Human mobility study has attracted many attentions in urban data analytics. Early work in (McFadden, 1974) develops a multiple dimensions to analyze people's travel demands from their traveling behavior. More recently, availability of vehicle GPS traces has empower many urban related applications. The work in (Zheng, Liu, Yuan, & Xie, 2011) uses trajectories of taxicabs to detect flawed urban planning areas, such as O-D region pairs with traffic issues. The work in (Ge et al., 2010; J. Yuan, Zheng, Zhang, Xie, & Sun, 2011) analyzes spetiotemporal patterns of city taxi's pick-up and drop-up behavior and driving route to find the optimal strategy for helping taxi driver increase their cars' occupancy rate. The work in (Y. Wang et al., 2015) uses heterogeneous mobility data including taxi, bus, and subway to measure spatial connectivity among areas for boosting the performance of user location prediction.

Word embedding learning has been studied in recent years through deep neural networks (Bengio et al., 2003; Collobert & Weston, 2008). Later, the work in GloVE (Pennington, Socher, & Manning, 2014) and word2vec (Mikolov, Sutskever,

et al., 2013; Mikolov, Chen, et al., 2013) which utilize word dependency for semantic analysis, have shown greatly increasement on performance in key NLP tasks, like document clustering (Kusner, Sun, Kolkin, & Weinberger, 2015), and word similarity (Levy, Goldberg, & Dagan, 2015). Recenltly, the work in (Levy & Goldberg, 2014) shows an equivalence of matrix factorization of a shifted PMI matrix to word2vec skip-gram.

4.7 Concluding Remarks

In this chapter, we presented a city zone embedding framework using human mobility patterns to represent urban functions with distributed and low-dimensional vectors. For this purpose, we exploited the human mobility patterns from massive taxi trajectories to model the embeddings with zone associations. Specifically, we developed a spatio-temporal embedding model for incorporating mobility directions and departure/arrival times for learning zone dependencies, and calculated the travel demands of origin-destination zones with destination attraction and travel distance for guiding the importance of zone co-occurrence in embedding optimization. Extensive experiments on real-world datasets demonstrated the effectiveness of proposed method with a consistent performance improvement over baseline methods.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this dissertation, we identified several unique challenges of mobile intelligence analytics for urban smart living, and then introduced how we use advanced data mining techniques to address these challenges.

First, we presented a systematic study on ranking house by leveraging spatio-temporal crime data. Specifically, we first extracted community crime evidences in two categories: crime severity and crime temporal correlation. Moreover, we proposed an effective approach to ranking houses based on value using the house specific features of community safety. Also, we integrated the impacts of major house profile in optimization to enhance the proposed ranking model. Finally, extensive experimental results on real-world crime and house data validated the performance of the proposed method.

Second, we developed a POI recommendation model by considering the temporal matching between users and POIs. Firstly, we presented a method to profile the temporal popularity of POIs by (i) mining area activity patterns with taxi trips, (ii) integrating category popularity pattern with POI category level check-ins, and (iii) enhancing patterns with mixture mode. Moreover, we learned the latent temporal regularity of users by incorporating the temporal matching degrees of user-POI pairs into user overall preference estimation. Finally, we conducted extensive experiments

with POI check-in and human mobility data. As demonstrated by the experimental results, the consideration of temporal matching between users and POIs can better model LBSN users' choosing processes. The performance improvement of our proposed method is substantial compared to benchmark methods.

Last, we presented a city zone embedding framework using human mobility patterns to represent urban functions with distributed and low-dimensional vectors. For this purpose, we exploited the human mobility patterns from massive taxi trajectories to model the embeddings with zone associations. Specifically, we developed a spatio-temporal embedding model for incorporating mobility directions and departure/arrival times for learning zone dependencies, and calculated the travel demands of origin-destination zones with destination attraction and travel distance for guiding the importance of zone co-occurrence in embedding optimization. Extensive experiments on real-world datasets demonstrated the effectiveness of proposed method with a consistent performance improvement over baseline methods.

5.1 Future Work

In the era of big data, scientific discovery and business solution are demanded to be more detailed, updated, and intelligent. Therefore, data-driven approaches is being increasingly critical for more research and application problems. Meanwhile, data sources are becoming more diverse. Data from multiple and heterogeneous sources are increasingly being incorporated together for solving problems.

I will extend, generalize, and deepen my previous work on mobile intelligence analytics for urban smart living. First, I plan to extend my urban function research from

static modeling to dynamic modeling, therefore we can track the function changing of city areas. Furthermore, I want to incorporate local business data such as point-of-interest prosperity to assess the impact of function change for local business. Second, I plan to enhance the research of mobile recommender systems by jointly considering Who (personalized preference), What (purpose characteristics), When (temporal dependency), Where (geographic influence), and introducing deep learning techniques to recommendation models. Last, I will generalize the modeling of safety awareness of real estate to more geographical items (e.g., retail stores, restaurants, etc.) for business site selection and taxi routing optimization.

BIBLIOGRAPHY

Agarwal, D., & Chen, B.-C. (2009). Regression-based latent factor models. In *Proceedings of the 15th acm sigkdd international conference on knowledge discovery and data mining* (pp. 19–28).

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1798–1828.

Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *journal of machine learning research*, *3*(Feb), 1137–1155.

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* (pp. 130–140).

Buonanno, P., Montolio, D., & Raya-Vílchez, J. M. (2013). Housing prices and crime perception. *Empirical Economics*, *45*(1), 305–321.

Burges, C. J. (2010). From ranknet to lambdarank to lambdamart: An overview. *Learning*, *11*, 23–581.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., & Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on machine learning* (pp. 129–136).

Cascetta, E., Pagliara, F., & Papola, A. (2007). Alternative approaches to trip distribution modelling: a retrospective review and suggestions for combining different approaches. *Papers in regional Science*, *86*(4), 597–620.

Chen, Y., Kapralov, M., Canny, J., & Pavlov, D. Y. (2009). Factor modeling for advertisement targeting. In *Advances in neural information processing systems* (pp. 324–332).

Cheng, C., Yang, H., Lyu, M. R., & King, I. (2013). Where you like to go next: Successive point-of-interest recommendation. In *Proceedings of the twenty-third international joint conference on artificial intelligence* (pp. 2605–2611). AAAI Press.

Cheng, Z., Caverlee, J., Lee, K., & Sui, D. Z. (2011). Exploring millions of footprints in location sharing services. In *Proceedings of the fifth international conference on weblogs and social media.*

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* (pp. 1082–1090).

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on machine learning* (pp. 160–167).

Cooper, W. S., Gey, F. C., & Dabney, D. P. (1992). Probabilistic retrieval based on staged logistic regression. In *Proceedings of the 15th annual international acm sigir conference on research and development in information retrieval* (pp. 198–210).

Cranshaw, J., Schwartz, R., Hong, J. I., & Sadeh, N. (2012). The livehoods project: Utilizing social media to understand the dynamics of a city. In *Proceedings of the sixth international conference on weblogs and social media* (p. 58).

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, *6*(2), 182–197.

Denver open data catalog. (n.d.). `http://data.denvergov.org/`.

Ding, Y., & Li, X. (2005). Time weight collaborative filtering. In *Proceedings of the 14th acm international conference on information and knowledge management* (pp. 485–492).

Freund, Y., Iyer, R., Schapire, R. E., & Singer, Y. (2003). An efficient boosting algorithm for combining preferences. *The Journal of machine learning research*, *4*, 933–969.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119–139.

Fu, Y., Ge, Y., Zheng, Y., Yao, Z., Liu, Y., Xiong, H., & Yuan, N. J. (2014). Sparse real estate ranking with online user reviews and offline moving behaviors. In *Proceedings of 2014 ieee international conference on data mining (icdm)* (pp. 120–129).

Fu, Y., Liu, G., Papadimitriou, S., Xiong, H., Ge, Y., Zhu, H., & Zhu, C. (2015). Real estate ranking via mixed land-use latent models. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining* (pp. 299–308).

Fu, Y., Xiong, H., Ge, Y., Yao, Z., Zheng, Y., & Zhou, Z.-H. (2014). Exploiting geographic dependencies for real estate appraisal: A mutual perspective of ranking and clustering. In *Proceedings of the 20th acm sigkdd international conference on knowledge discovery and data mining* (pp. 1047–1056).

Gao, H., Tang, J., Hu, X., & Liu, H. (2013a). Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th acm conference on recommender systems* (pp. 93–100).

Gao, H., Tang, J., Hu, X., & Liu, H. (2013b). Modeling temporal effects of human mobile behavior on location-based social networks. In *Proceedings of the 22nd acm international conference on conference on information & knowledge management* (pp. 1673–1678).

Gao, H., Tang, J., & Liu, H. (2012). Exploring social-historical ties on location-based social networks. In *Icwsm*.

Ge, Y., Xiong, H., Tuzhilin, A., Xiao, K., Gruteser, M., & Pazzani, M. (2010). An energy-efficient mobile recommender system. In *Proceedings of the 16th acm sigkdd international conference on knowledge discovery and data mining* (pp. 899–908).

Gibbons, S. (2004). The costs of urban property crime. *The Economic Journal*, *114*(499), F441–F463.

Goddard, J. B. (1970). Functional regions within the city centre: A study by factor analysis of taxi flows in central london. *Transactions of the Institute of British Geographers*, 161–182.

Gonzalez, M. C., Hidalgo, C. A., & Barabasi, A.-L. (2008). Understanding individual human mobility patterns. *Nature*, *453*(7196), 779–782.

Gu, Q., Zhou, J., & Ding, C. H. (2010). Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In *Proceedings of the 2010 siam international conference on data mining* (pp. 199–210).

Hearnden, I., & Magill, C. (2004). *Decision-making by house burglars: offenders' perspectives*. Home Office. Research, Development and Statistics Directorate.

Herbrich, R., Graepel, T., & Obermayer, K. (1999). Large margin rank boundaries for ordinal regression. *Advances in neural information processing systems*, 115–132.

Kleemans, E. R. (2001). Repeat burglary victimisation: results of empirical research in the netherlands. *Crime prevention studies*, *12*, 53–68.

Kling, F., & Pozdnoukhov, A. (2012). When a city tells a story: urban topic analysis. In *Proceedings of the 20th international conference on advances in geographic information systems* (pp. 482–485).

Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th acm sigkdd international conference on knowledge discovery and data mining* (pp. 426–434).

Koren, Y. (2010). Collaborative filtering with temporal dynamics. *Communications of the ACM*, *53*(4), 89–97.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*(8), 30–37.

Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning* (pp. 957–966).

Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems* (pp. 556–562).

The lemur project/ranklib. (n.d.). `http://sourceforge.net/p/lemur/wiki/RankLib/`.

Levy, O., & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems* (pp. 2177–2185).

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, *3*, 211–225.

Linden, L., & Rockoff, J. E. (2008). Estimates of the impact of crime risk on property values from megan's laws. *The American Economic Review*, 1103–1127.

Liu, B., Kong, D., Cen, L., Gong, N. Z., Jin, H., & Xiong, H. (2015). Personalized mobile app recommendation: Reconciling app functionality and user privacy preference. In *Proceedings of the eighth acm international conference on web search and data mining* (pp. 315–324).

Liu, B., Xiong, H., Papadimitriou, S., Fu, Y., & Yao, Z. (2015). A general geographical probabilistic factor model for point of interest recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *27*(5), 1167–1179.

Ma, H., Liu, C., King, I., & Lyu, M. R. (2011). Probabilistic factor models for web site recommendation. In *Proceedings of the 36th international acm sigir conference on research and development in information retrieval* (pp. 265–274).

Mathew, W., Raposo, R., & Martins, B. (2012). Predicting future locations with hidden markov models. In *Proceedings of the 2012 acm conference on ubiquitous computing* (pp. 911–918).

McFadden, D. (1974). The measurement of urban travel demand. *Journal of public economics*, *3*(4), 303–328.

McInerney, J., Zheng, J., Rogers, A., & Jennings, N. R. (2013). Modelling heterogeneous location habits in human populations for location prediction under data sparsity. In *Proceedings of the 2013 acm international joint conference on pervasive and ubiquitous computing* (pp. 469–478).

Metzler, D., & Croft, W. B. (2007). Linear feature-based models for information retrieval. *Information Retrieval*, *10*(3), 257–274.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Mnih, A., & Salakhutdinov, R. R. (2008). Probabilistic matrix factorization. In *Advances in neural information processing systems* (pp. 1257–1264).

Noulas, A., Scellato, S., Lathia, N., & Mascolo, C. (2012). Mining user mobility features for next place prediction in location-based services. In *2012 ieee 12th international conference on data mining (icdm)* (pp. 1038–1043).

Noulas, A., Shaw, B., Lambiotte, R., & Mascolo, C. (2015). Topological properties and temporal dynamics of place networks in urban environments. In *Proceedings of the 24th international conference on world wide web companion* (pp. 431–441).

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

Polvi, N., Looman, T., Humphries, C., & Pease, K. (1991). The time course of repeat burglary victimization. *British Journal of Criminology*, *31*(4), 411–414.

Pope, J. C. (2008). Fear of crime and housing prices: Household reactions to sex offender registries. *Journal of Urban Economics*, *64*(3), 601–614.

Putnam, R. D. (2001). *Bowling alone: The collapse and revival of american community*. Simon and Schuster.

Ratcliffe, J. H., & Rengert, G. F. (2008). Near-repeat patterns in philadelphia shootings. *Security Journal*, *21*(1), 58–76.

Salakhutdinov, R., & Mnih, A. (2008). Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on machine learning* (pp. 880–887).

School performance framework. (n.d.). `http://spf.dpsk12.org/`.

Silva, T. H., de Melo, P. O. V., Almeida, J. M., Salles, J., & Loureiro, A. A. (2012). Visualizing the invisible image of cities. In *2012 ieee international conference on green computing and communications (greencom)* (pp. 382–389).

Song, C., Qu, Z., Blumm, N., & Barabási, A.-L. (2010). Limits of predictability in human mobility. *Science*, *327*(5968), 1018–1021.

Townsley, M., Homel, R., & Chaseling, J. (2003). Infectious burglaries. a test of the near repeat hypothesis. *British Journal of Criminology*, *43*(3), 615–633.

Wang, D., Pedreschi, D., Song, C., Giannotti, F., & Barabasi, A.-L. (2011). Human mobility, social ties, and link prediction. In *Proceedings of the 17th acm sigkdd international conference on knowledge discovery and data mining* (pp. 1100–1108).

Wang, G., Schoenebeck, S. Y., Zheng, H., & Zhao, B. Y. (2016). " will check-in for badges": Understanding bias and misbehavior on location-based social networks. In *Proceedings of the tenth international conference on web and social media* (pp. 417–426).

Wang, Y., Yuan, N. J., Lian, D., Xu, L., Xie, X., Chen, E., & Rui, Y. (2015). Regularity and conformity: Location prediction using heterogeneous mobility data. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining* (pp. 1275–1284).

Xiong, L., Chen, X., Huang, T.-K., Schneider, J., & Carbonell, J. G. (2010). Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the 2010 siam international conference on data mining* (pp. 211–222).

Xu, J., & Li, H. (2007). Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international acm sigir conference on research and development in information retrieval* (pp. 391–398).

Yang, D., Zhang, D., Zheng, V. W., & Yu, Z. (2015a). Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *45*(1), 129–142.

Yang, D., Zhang, D., Zheng, V. W., & Yu, Z. (2015b). 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, Z., Fu, Y., Liu, B., Hu, W., & Xiong, H. (2018). Representing urban functions through zone embedding with human mobility patterns. In *Proceedings of 27th international joint conference on artificial intelligence (ijcai).*

Yao, Z., Fu, Y., Liu, B., Liu, Y., & Xiong, H. (2016). Poi recommendation: A temporal matching between poi popularity and user regularity. In *2016 ieee 16th international conference on data mining (icdm)* (pp. 549–558).

Yao, Z., Fu, Y., Liu, B., & Xiong, H. (2016). The impact of community safety on house ranking. In *Proceedings of the 2016 siam international conference on data mining (sdm)* (pp. 459–467).

Yuan, J., Zheng, Y., & Xie, X. (2012). Discovering regions of different functions in a city using human mobility and pois. In *Proceedings of the 18th acm sigkdd international conference on knowledge discovery and data mining* (pp. 186–194).

Yuan, J., Zheng, Y., Zhang, L., Xie, X., & Sun, G. (2011). Where to find my next passenger. In *Proceedings of the 13th international conference on ubiquitous computing* (pp. 109–118).

Yuan, N. J., Zhang, F., Lian, D., Zheng, K., Yu, S., & Xie, X. (2013). We know how you live: exploring the spectrum of urban lifestyles. In *Proceedings of the first acm conference on online social networks* (pp. 3–14).

Yuan, N. J., Zheng, Y., Xie, X., Wang, Y., Zheng, K., & Xiong, H. (2015). Discovering urban functional zones using latent activity trajectories. *IEEE Transactions on Knowledge and Data Engineering*, *27*(3), 712–725.

Yuan, Q., Cong, G., Ma, Z., Sun, A., & Thalmann, N. M. (2013a). Time-aware point-of-interest recommendation. In *Proceedings of the 36th international acm sigir conference on research and development in information retrieval* (pp. 363–372).

Yuan, Q., Cong, G., Ma, Z., Sun, A., & Thalmann, N. M. (2013b). Who, where, when and what: discover spatio-temporal topics for twitter users. In *Proceedings of the 19th acm sigkdd international conference on knowledge discovery and data mining* (pp. 605–613).

Zhang, S., Wang, W., Ford, J., & Makedon, F. (2006). Learning from incomplete ratings using non-negative matrix factorization. In *Proceedings of the 2010 siam international conference on data mining* (Vol. 6, pp. 548–552).

Zheng, Y., Li, Q., Chen, Y., Xie, X., & Ma, W.-Y. (2008). Understanding mobility based on gps data. In *Proceedings of the 10th international conference on ubiquitous computing* (pp. 312–321).

Zheng, Y., Liu, Y., Yuan, J., & Xie, X. (2011). Urban computing with taxicabs. In *Proceedings of the 13th international conference on ubiquitous computing* (pp. 89–98).