UNIVERSITY OF TECHNOLOGY, SYDNEY
Faculty of Engineering and Information Technology

Exploring Heterogeneous Social Information Networks for Recommendation

by

Qinzhe Zhang

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Sydney, Australia

2017

Certificate of Authorship/Originality

I certify that the work in this thesis has not been previously submitted for a degree nor has it been submitted as a part of the requirements for another degree except as fully acknowledged within the text.

I also certify that this thesis has been written by me. Any help that I have received in my research and in the preparation of the thesis itself has been fully acknowledged. In addition, I certify that all information sources and literature used are quoted in the thesis.

This research is supported by an Australian Government Research Training Program Scholarship.

ABSTRACT

Exploring Heterogeneous Social Information Networks for Recommendation

by

Qinzhe Zhang

A basic premise behind our study of heterogeneous social information networks for recommendation is that a complex network structure leads to a large volume of implicit but valuable information which can significantly enhance recommendation performance. In our work, we combine the global popularity and personalized features of travel destinations and also integrate temporal sensitive patterns to form spatial-temporal wise trajectory recommendation. We then develop a model to identify representative areas of interest (AOIs) for travellers based on a large scale dataset consisting of geo-tagged images and check-ins. In addition, we introduce active time frame analysis to determine the most suitable time to visit an AOI during the day. The outcome of this work can suggest relevant personalized travel recommendations to assist people who are arriving in new cities.

Another important part of our research is to study how “local” and “global” social influences exert their impact on user preferences or purchasing decisions. We first simulate the social influence diffusion in the network to find the global and local influence nodes. We then embed these two different kinds of influence data, as regularization terms, into a traditional recommendation model to improve its accuracy. We find that “Community Stars” and “Web Celebrities”, represent “local” and “global” influence nodes respectively, a phenomenon which does exist and can help us to generate significantly better recommendation results.

A central topic of our thesis is also to utilize a large heterogeneous social information network to identify the collective market hyping behaviours. Combating

malicious user attacks is also a key task in the recommendation research field. In our study, we investigate the evolving spam strategies which can escape from most of the traditional detection methods. Based on the investigation of the advanced spam technique, we define three kinds of heterogeneous information networks to model the patterns in such spam activities and we then propose an unsupervised learning model which combines the three networks in an attempt to discover collective hyping activities. Overall, we utilize the heterogeneous social information network to enhance recommendation quality, not only by improving the user experience and recommendation accuracy, but also by ensuring that quality and genuine information is not overwhelmed by advanced hyping activities.

Dedication

I dedicate my dissertation work to my parents, my wife and my baby daughter who also born on my thesis submission date. A special feeling of gratitude to my loving parents, Yaran Zhang and Miliang Qin whose words of encouragement and push for tenacity ring in my ears. My wife, Qing Deng has never left my side and always encourage me to move forward during this tough but exciting journey, especially considering we are new immigrants in Australia hence she should have lots of responsibility in many aspects during this tough time. We are proud of what we achieve when we think back many difficulties we have overcome in these years. I also dedicate this dissertation to my family, especially my grandmothers, who always encourage me to stick to my goal although they do not understand what I was doing.

In addition, I dedicate this thesis to many friends of mine who have supported me throughout the process. I will always appreciate all they have done, especially they always keep listening to me, about my concern and frustration, as well as my success in publication.

Acknowledgements

I wish to thank my principal supervisor Professor Chengqi Zhang was more than generous with his expertise and precious time.

In addition, I would like to thank Dr. Guodong Long who supported me a lot in many aspects of my research. His support make my Phd studying being an enjoyable experience.

Finally, a special acknowledge and thanks to Dr. Peng Zhang whose office was always open whenever I ran into a trouble spot or had a question about my research or writing. His consistently allowed this thesis to be my own work, but steered me in the right the direction whenever he thought I needed it.

Qinzhe Zhang
Sydney, Australia, 2017.

List of Publications

Journal Papers

J-1. **Qinzhe. Zhang**, Jia. Wu, Guodong. Long, Peng. Zhang and Chengqi. Zhang, "Collective Hyping Detection System for Identifying Online Spam Activities," *IEEE Intelligent Systems*, 2017. (Accepted on 12th of January 2017)

J-2. **Qinzhe. Zhang**, Jia. Wu, Guodong. Long, Peng. Zhang and Chengqi. Zhang, "Dual Influence Embedded Social Recommendation," *Word Wide Web: Internet and Web Information Systems (WWW)*, 2017. (Accepted on 20th of July 2017)

Conference Papers

C-1. **Qinzhe. Zhang**, and Litao. Yu, Guodong. Long:, "SocialTrail: Recommending Social Trajectories from Location-Based Social Networks, *Australasian Database Conference (ADC 2015)*, pp. 314-317, May. 31, 2015.

C-2. **Qinzhe. Zhang**, and Qin.Zhang, Guodong. Long, Peng. Zhang and Chengqi. Zhang:, "Exploring Heterogeneous Product Networks for Discovering Collective Marketing Hyping Behavior, *The Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2016)*, pp. 40-51, Apr. 19-22, 2016.

C-3. **Qinzhe. Zhang**, and Jia.Wu, Guodong. Long, Peng. Zhang and Chengqi. Zhang:, "Global and Local Influence-based Social Recommendation, *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM 2016)*, pp. 1917-1920, Oct. 24-28, 2016.

Contents

Certificate ii

Abstract iii

Dedication v

Acknowledgments vi

List of Publications vii

List of Figures xii

Abbreviation xiv

Notation xv

1 Introduction 1

1.1 Research Objectives 2

1.2 Thesis Organization 4

1.3 Thesis overview and contributions 5

1.3.1 Part 1: Temporal and Spatial Aware Social Trajectory Recommendation 5

1.3.2 Part 2: Collective Marketing Hyping Behavior Detection . . . 7

1.3.3 Part-3: Dual Social Influence Embedded Recommendation . . 8

2 Literature Review 10

2.1 Part-1: Temporal and Spatial Aware Social Trajectory Recommendation 10

2.1.1 Traditional trajectory recommendation 10

2.1.2 Trajectory recommendation from geo-tagged images 11

2.1.3 Mining Areas of Interest from heterogeneous networks 13

2.2 Part-2: Collective Marketing Hyping Behavior Detection 14

2.2.1 Review Spam Detection 14

2.2.2 Spam Studies in OSNs 15

2.2.3 Temporal Feature in Spam Detection 16

2.2.4 Shapelet Learning Studies 17

2.3 Part-3: Dual Social Influence Embedded Recommendation 18

2.3.1 Traditional Recommender Systems 18

2.3.2 Matrix Factorization in Recommendation 19

2.3.3 Social Recommender Systems 20

2.3.4 The Effectiveness of Social Influence 21

3 Temporal and Spatial Aware Social Trajectory Recommendation 23

3.1 Introduction 23

3.2 Related work 27

3.3 Methodology 28

3.3.1 Problem Definition 28

3.3.2 Generating AOIs 29

3.3.3 Generating social trajectories 32

3.4 Framework and algorithm 41

3.5 Experiment 42

3.5.1 Dataset Analysis 43

3.5.2 Social trajectory recommendation evaluation 45

4 Collective Marketing Hyping Behavior Detection 50

4.1 Introduction 50

4.2 Related work 54

4.3 Methodology 55

4.3.1 Problem Definition 55

4.3.2 Shapelet Learning Model 57

4.3.3 Product Network Regularization 60

4.3.4 Collaborative Hyping Detection Model 62

4.4 Algorithm 63

4.5 Experiments 63

4.5.1 Dataset 64

4.5.2 User Correlation-based Product Network 65

4.5.3 Comparison with Existing Models 67

4.5.4 The impact of PBR/SBR/UCR in CHDM 68

4.5.5 Case Study 69

4.5.6 Collaborative Marketing Hyping Activities 71

5 Dual Social Influence Embedded Recommendation 73

5.1 Introduction 73

5.2 Related work 77

5.3 Methodology 78

5.3.1 Problem Definition 78

5.3.2 Social Influence Model 81

5.3.3 Dual Influence Embedded Social Recommendation Model . . . 90

5.4 Algorithm 91

5.5 Experiments 92

5.5.1 Experimental Settings 93

5.5.2 Time Complexity Comparison 94

5.5.3 Comparison with Existing Models 96

5.5.4 The impact of local/global influence in DISR 97

5.5.5 Top N Recommendation Evaluation 99

5.5.6 Parameter Analysis 103

6 Conclusion and Future Directions 104

6.1 Conclusion 104

6.2 Vision of the future 105

Bibliography 107

List of Figures

3.1 Geo-tagged Images and Check-ins 25

3.2 Effect of secondary clustering 32

3.3 Inference Transition 34

3.4 Combined Authority Analysis 37

3.5 Choose Next Stop 40

3.6 Social Trajectory Recommendation Framework 42

3.7 Recommendation Scores by Gender 46

3.8 AOI Visualization and Manipulation 47

3.9 Temporal and Spatial Aware Recommendation 47

3.10 Social Trajectory Recommendation 48

4.1 Evolving Marketing Hyping Ecosystem 52

4.2 Store-based Network and Matrix $\mathbf{G}_1$. 61

4.3 Product-based Network and Matrix $\mathbf{G}_2$. 61

4.4 User correlation-based Network and Matrix $\mathbf{G}_3$. 63

4.5 Framework of CHDM model 64

4.6 Case Study Analysis 70

4.7 Collaborative Marketing Hyping Activities. 71

5.1 An illustration of social influence. The target node w is mostly influenced by the *Global Influential Node* m and the *Local Influential Node* u, rather than by friends (w_1, w_2, w_3, w_4, and w_5). 74

5.2 (a) Node weight, and (b) Influence weight. 81

5.3 DISR framework We utilize the implicit social influence information from our GIM and LIM models, embedded in the matrix factorization process, to improve the accuracy of the missing value prediction in the original user-item rating matrix. 89

5.4 Performance comparison between five methods on a Top K recommendation task on the Mafengwo dataset. The y-axis indicates the probability of the 'Most Relevant Item' being ranked by a specific model compared to other items. The x-axis denotes the percentile of these other items. 101

5.5 Performance comparison between five methods on a Top K recommendation task on the Douban dataset. The y-axis indicates the probability of the 'Most Relevant Item' being ranked by a specific model compared to other items. The x-axis denotes the percentile of these other items. 101

5.6 Parameter study on Mafengwo data. 102

5.7 Parameter study on Douban data. 102

Abbreviation

1-SMC - First-order Sliding Mode Control

2-SMC - Second-order Sliding Mode Control

2-D: Two-dimensional

3-D: Three-dimensional

DF - Describing Function

FRF - Frequency Response Function

FSSMC - Frequency Shaped Sliding Mode Control

HOSM: Higher-order sliding modes

LTI: Linear time-invariant

MIMO: Multi input multi output

MR - Magnetorheological

MDoF - Multiple Degree of Freedom

RMSE - Root Means Square error

SDoF - Single Degree of Freedom

SISO Single input single output

SMC - Sliding Mode Control

SVD: Singular value decomposition

TF - Transfer Function.

VSC: Variable structure control

Nomenclature and Notation

Capital letters denote matrices.

Lower-case alphabets denote column vectors.

$(.)^T$ denotes the transpose operation.

I_n is the identity matrix of dimension $n \times n$.

0_n is the zero matrix of dimension $n \times n$.

$\mathbb{R}$, $\mathbb{R}^+$ denote the field of real numbers, and the set of positive reals, respectively.

Chapter 1

Introduction

Our main research interest is in understanding how to enhance recommendation performance using heterogeneous information network properties. A comprehensive recommender system should be able to not only increase the accuracy of the recommended items, but also detect and eliminate spam activities, for providing a better user experience. Thus, what information should be included in the heterogeneous information network? How can we model this information in light of traditional recommender models? How can we accurately identify ’hyping’ activities? The answers to these questions are crucial to a range of application areas from temporal and spatial aware travel recommendations, collective marketing hyping detection to ’web celebrity influence’ embedded recommender systems.

To this end, we study three such cases with a view to build a comprehensive recommender system:

- **Temporal and spatial aware recommender system:** Temporal and spatial properties and models play an important role in travel recommendation. By employing collective social information, such as geo-tagged images, check-in data, etc., we propose a novel system that not only integrates geo-tagged images and check-in data to discover meaningful social trajectories to enrich travel information, it also takes both temporal and spatial factors into consideration to make trajectory recommendation more accurate.
- **Collaborative marketing hyping detection system:** With the rapid increase in the usage of Web 2.0, online reviews and ratings have become vital

to users in their decision making process. To identify fake reviews and hyping behavior online, we propose a new **Collaborative Marketing Hyping Detection** solution which aims to identify spam comments generated by the Spam Reviewer Cloud and to detect products which adopt an evolving spam strategy for promotion. In general, we develop an unsupervised learning model which integrate heterogeneous product review networks as regularization terms in an attempt to discover collective hyping activities, which can also help us to improve recommendation quality.

- **Dual social influence embedded recommender system:** With the increasing number of users in various online communities, such as, Netflix, Douban, etc., people can easily connect with others. Thus, what assistance the underlying social influence could provide to improve recommendation accuracy is interesting and important. We first simulate the social influence diffusion in the network to find the global and local influence nodes and then embed this dual influence data into a traditional recommendation model to improve accuracy. Mathematically, we formulate the global and local influence data as new dual social influence regularization terms and embed them into a matrix factorization-based recommendation model.

1.1 Research Objectives

Traditional recommendation focuses on an 'increasing accuracy' point of view where researchers tend to answer questions such as how to minimize the root mean squared error (RMSE) or enhance the top-k ranking performance, but seldom consider the overall recommendation ecosystem. To build a comprehensive recommender system, our research investigates ways to improve recommendation quality from three aspects:

- A temporal and spatial aware recommendation model can be more user-friendly and provider a better user experience.
- An advanced collective hyping detection model can decrease web fraud and enhance user retention.
- A dual social influence embedded recommender system can increase recommendation accuracy for better user understanding.

No single online application is able to provide the required amount of information, but we can find online communities where some of this information is available to validate our research. For example, location-based social networks (LBSNs) are widespread and enable users to not only contribute to online reviews on a real-world location but also to suggest the most suitable time to visit such a place by sharing their check-in time. Such data can be collected from large online e-commercial platforms which is discussed in a later section. Furthermore, with the proliferation of online co-ratings or co-review communities, social influence has been proven to be very important for marketing, which forms a new "web celebrity effect" and it can easily affect user purchasing behaviors. Such online social networks have recently appeared, providing a good opportunity to study this new problem.

However, utilizing this information to improve recommendation quality is challenging. Specifically, most existing trajectory recommendation systems use traditional "editor-selected" trajectories to generate the top-ranked trips for users, without understanding the overall trip and tend to utilize homogeneous data only (e.g., geo-tagged images), failing to comprehensively utilize travel information as a whole and not taking into account many other important factors such as travel time, duration and sequence, etc. Furthermore, as e-commercial applications are everywhere, many online business providers use fake reviews to mislead users and to convince them to purchase their products or services. Although a large amount of research has been conducted on spam detection, we find evolving hyping strategy, i.e., User Cloud, can still easily escape from spam detection techniques. On these platforms, business owners can purchase anonymous comments generated by real users by paying for them. This makes spam detection very challenging as they can avoid most of the anti-spam algorithms. Last but not least, in many rating-based online communities, researchers tend to use explicit pieces of social information incorporating recommendations which normally adopt similarity measures (*e.g.* cosine similarity) to evaluate the explicit relationships in the network, but they do not consider the latent and implicit relationships in the network, such as social influence. A target user's purchase behavior or interests, for instance, is not always determined by their directly connected relationships and may be significantly influenced by the influential reputation of people they do not know in the network. How to discover latent social influence in social networks and utilize this information to improve recommendation accuracy is challenging.

1.2 Thesis Organization

In order to investigate each problem, we design a model which integrates a well-designed heterogeneous information network to enhance its performance and implement the following steps to solve each problem:

- In Step 1, we identify our research interest from real-world problem observation and discuss how can make improvement on existing methods.
- In Step 2, recent related work is surveyed to further identify the weakness of the existing research and why this work does not fully answer the research questions in this thesis.
- In Step 3, a heterogeneous information network is designed and models are built with regarding to solving research problems in our studies.

- In Step 4, algorithms are developed by utilizing the insights from the models that explain the observations.
- In Step 5, real-world data is analyzed and collected and solid validation experiments are conducted to compare the proposed model with other state-of-the-art research.

Thus, this thesis comprises three sections, each of which corresponds to a specific problem. Each section comprises five subsections. An overview of the thesis and its contribution is discussed in the following subsection.

1.3 Thesis overview and contributions

The thesis addresses a number of real-world problems regarding the properties of and the information within heterogeneous networks by revealing how their engagement can enhance recommendation quality.

This research focuses on using the heterogeneous information network to make personalized recommendations to enhance the user experience and accuracy. This thesis has a 3-by-5 structure: it focuses on three research domains, i.e., temporal and spatial aware social trajectory recommendation, collective hyping behavior detection and dual social influence embedded recommendation, each referring to one specific recommendation quality enhancement topic. Each is discussed in five sections: Introduction, Related work, Models, Algorithms and Experiments.

1.3.1 Part 1: Temporal and Spatial Aware Social Trajectory Recommendation

Questions: There are many recommendation applications associated with travel, however, these applications tend to suggest a single scenic location or a single trip, which motivates the research in this thesis to investigate ways to ensure that travel recommendations are not just referring to well-known tourist attractions but also include local destinations to cater for a variety of tastes. Furthermore, it is important to ensure the recommendation is flexible and accurate, instead of simply being a sequence of several points without considering the temporal factors related to travel and the best time to visit the recommended location?

Answers: To answer these questions, we first collect and fuse two types of data, geo-tagged images and check-in data, to represent traditional and personalized preferences, respectively, to form a heterogeneous network. Then, according to their geographic properties, these are clustered into small areas of interest(AOIs) for recommendation. In addition, each AOI is assigned a most suitable time to visit and the duration of the visit - which is learned from the temporal features in the real-world data.

Contributions:

- The personalized trajectory recommendations in the proposed system are derived from two types of social data - geo-tagged images and check-in data. The former represents traditional tourist attractions whereas the latter reflects local choices such as restaurants, shopping and night life.
- The recommended trajectories are not isolated locations, rather several AOIs, thus providing more tourism options for travellers.
- In addition to taking into account the popularity of the recommended destination, the proposed algorithm is able to support spatial and temporal aware trajectory ranking.

1.3.2 Part 2: Collective Marketing Hyping Behavior Detection

Questions: Many e-commerce platforms, such as Amazon.com and Taobao.com, report that many review writers charge businesses a fee to hype their product* which has been identified as one of the most serious issues on these platforms. What is the evolving spam strategy and how does it avoid detection by an anti-spam algorithm? What are the behavioural patterns of online hypers who randomly appear online? How can an effective model be designed to efficiently identify collective hyping behavior?

Answers: To answer these questions, we first collect real-world data from Taobao.com, as we wish to investigate whether hyping behaviour exhibits collective patterns. Then, we design several heterogeneous information networks among products, reviewers and online stores and integrate them into a shapelet learning model to make classifications based on both network correlation and the temporal features of the review itself.

Contributions:

- We devise a new solution called **Collaborative Marketing Hyping Detection** which uncovers the collective strategy of all the online competitors who engage in vicious hyping competition on e-commerce platforms. To the best of our knowledge, this is an area which has never previously been studied.
- We propose an unsupervised shapelet learning model which utilizes temporal patterns for recognizing collaborative fraud patterns.
- We incorporate three regularization terms into the shapelet learning model, to constrain the matrix factorization in this model.

*http://money.cnn.com/2015/10/18/technology/amazon-lawsuit-fake-reviews/

1.3.3 Part-3: Dual Social Influence Embedded Recommendation

Questions: Recently, the term “internet celebrity economy” has become increasingly familiar. An internet celebrity is someone who has become popular through their use of the Internet, possibly via online blogs or weblogs. Many may be expert in some specific area. This phenomenon has been amplified by the spread of social networks and as a result, internet celebrities can significantly affect the behaviours or tastes of those who follow them online. Thus, is user similarity, which is employed as part of recommendation, still suitable in these cases? What is the difference between the influence of “Online celebrities” and “Public Celebrities online”? How can this information be utilized to enhance recommendations?

Answers: Based on our observations, we develop two terms *Global Influential Nodes* and *Local Influential Nodes* to represent “public Celebrities online” and “online celebrities” respectively. Then, we propose a social influence maximization model to find globally and locally influential nodes for each target user. By integrating this information from social networks into a fundamental recommendation model, we develop our model and achieve better recommendation performance.

Contributions:

- We systematically analyze the difference between social influence and similarity in terms of the role that each plays in recommendation, decision making and marketing strategies. This observation motivates us to utilize social influence to improve recommendation quality.
- We develop a Global Influential Model (GIM) and Local Influential Model (LIM) to find the global/local influential nodes.
- We incorporate both the GIM and LIM as dual influence embedded regularization terms, to constrain the matrix factorization for social recommendation.

In the next chapter, a comprehensive literature review of the related work is provided.

Chapter 2

Literature Review

In this chapter, we survey a bunch of recent research works related to our research directions contain in this thesis.

2.1 Part-1: Temporal and Spatial Aware Social Trajectory Recommendation

First the limitations in the existing work on traditional trajectory recommendation are discussed, followed by an overview of how to build a heterogeneous network to mine areas of interest, which is useful for our research.

2.1.1 Traditional trajectory recommendation

Trip mining and recommendation has become increasingly important in recent years. Generally, the data sources used in travel recommender system can be roughly divided into three types: GPS trajectory data, travelogues (i.e., blogs), and geo-tagged photos.

In the early stages, global positioning systems(GPSs) provided trajectory data, which was widely utilized. Zheng et al. [113, 111, 112] obtained information on popular locations and travel sequences from GPS trajectory data to provide personalized friend and location recommendation by evaluating the similarity of users based on GPS record histories. One significant challenge for GPS trajectory-based methods is that it is difficult to extract data from an extremely large number of individuals. [113] first exploited HITS-based link analysis techniques to mine places of interest and trips. Zheng et al. [112] used a similar approach to recommend destinations and friends based on one's GPS history. Cao et al. [18] used GPS trajectories to discover and rank semantic locations, and also to recommend travel destinations [111]. Even though GPS data are very detailed and informative, there are still many limitations in using GPS data for research. For instance, for venues inside a shopping center, GPS devices are not able to provide accurate information to track people's movements inside a building due to limited car access.

There are also many other studies [46, 37, 32] which extract travel-related knowledge from content, such as blogs, to make travel recommendations. Ji et al. [46] use a graph-based method to discover city landmarks. Hao et al. [37] propose a probabilistic topic model to find topics from travelogues and then use appropriate topics to represent locations to recommend places of interest. Gao et al. [32] aim to automatically discover and rank landmarks using geo-tagged information, metadata of photos and user knowledge in Yahoo! Travel Guide to identify and rank landmarks. However, the weakness of using the travelogue-based method is in determining the location of travelogues, as usually, this type of data are unstructured and full of noise. Furthermore, these kinds of data are more or less out-of-date and people seldom continue to employ a GPS or a blog to share their experiences.

2.1.2 Trajectory recommendation from geo-tagged images

Recently, a variety of online media services, such as Flickr.com and Youtube.com, experienced a dramatic increase in use, which led to a large number of photos and videos becoming available on the Internet and many online communities were formed [108]. Together with the social textual and spatiotemporal metadata, these rich media data have stimulated research on discovering knowledge and patterns of online behaviors and communities. In [23], Crandall collected more than 35 million geo-tagged photos and predicted their location using visual, textual and temporal features. Kalogerakis et al. [51] identified geo-locations within a sequence of online user-shared images. Kennedy et al. [54] proposed a model which took spatial and temporal features into account to discover aggregate knowledge of a geographical area based on Flickr photos tags in specific city. In addition, Rattenbury et al. [81] and Yanai et al. [101] uncovered the correlation between photo tags, geographical locations and events by analyzing the spatiotemporal distribution of image tag data. Many trajectory recommendation systems have been developed to find personalized routes and locations. An excellent survey on the applications and approaches of using geo-tagged media in recommendation is given in [69]. The approaches in [59] and [67] define an itinerary as a list of landmarks that a person intends to visit, according to the geo-tagged photos they have previously shared for recommendation. The itinerary in the proposed work is defined as the spatio-temporal movement trajectory with well-designed granularity. *Trip Pattern Search* [4] employs large-scale geo-photos to identify travel recommendations, but it only focuses on travel between different cities. Hence, it does not provide information to help people explore one city. An-Jung Cheng combines several types of data, such as blogs, GPS logs, geo-tagged photos, and users' attributes (e.g., gender, age and race), to make personalized trip recommendations for different user types [22]. A tourism recommendation framework is presented in [17], in which geo-tagged images are clustered to find popular locations. The TravelScope [38], another interesting work, enables users to take virtual tours in certain regions. In addition to geo-tagged images, travel logs produced by users are also helpful for discovering points of interest. However, most geo-tagged images are related to traditional tourist attractions and do not provide information on local choices, such as restaurants, bars, etc.

2.1.3 Mining Areas of Interest from heterogeneous networks

In travel planning, too much information can make the decision-making process difficult but limited information can also make it difficult to make a decision. One solution is to find areas of interes (AOI) by clustering geo-tagged images based on their geographic and social attributes. The work in [55] develops a k-means-based method to cluster images for landmark search results. It first conducts a k-means-based method on image geographic location data and then ranks the clusters and their representative images. [24] proposes a method to render multimedia travel stories in response to spatial queries. However, most of the existing work has similar limitations. Geo-tagged images, such as those posted on Panoramio.com, normally reflect peoples opinions on and experience with on traditional travel locations, such as popular sight-seeing attractions. However, as previously mentioned, a traveler's interest usually involves many other aspects apart from sight-seeing, including dining, shopping, and so on, but these are not well represented by geo-tagged images. Similar to geo-tagged images, researchers investigated the application of geo-tagged video search and advertising in [94, 44, 6, 7]. However, there are very limited data sources on geo-tagged videos.

Check-ins is another popular LBSN application, which enables people to post a check-in to tell their friends where they are and what they doing. By adding check-in data, aforementioned limitations and problems can be solved. Check-ins are actively shared by users as they have more semantic meaning and they are more concise than GPS data in representing real-world places. Although geo-tagged images and check-ins are similar to each other, they provide travel information of different granularity, thus, we build a check-in and geo-tagged image fused heterogeneous network to effectively discover AOIs and further recommend trajectories based on the temporal and spatial attributes of these AOIs.

2.2 Part-2: Collective Marketing Hyping Behavior Detection

In this section, we review three major research directions related to the spam detection problem: reviewing spam detection, spam studies on Online Social Networks (OSNs) and temporal features in spam detection. We also discuss shapelet learning studies which are proposed as the main part of our objective function.

2.2.1 Review Spam Detection

Jindal [48, 49] conducted the initial research into opinion spam and studied the problem of trustworthiness in review opinions. Three types of review fraud activities were defined - untrustworthy opinions, reviews on brands only and non-reviews. It is relatively easy to classify the last two activities by human labeling but the first type is very difficult to label manually. As a result of this limitation, only reviews that impair reputation and fame can be taken into consideration and many duplicate or near-duplicate spam reviews can almost certainly be discarded. However, reviews which dishonestly promote the product were not considered in this study. Currently, spammers can easily escape naive detection anti-spam methods. Li [62] initially analyzed several attributes related to spam behaviour, e.g., content features, sentiment features, product features, and metadata features. A two-view semi-supervised method was exploited to identify spam reviews. Feng [27] introduced the concept of the natural distribution of opinions and defined three types of reviewers: any-time reviewers, multi-time reviewers and single-time reviewers. These different types of distributional footprints of deceptive reviews were estimated statistically using neuro-linguistic programming (NLP) techniques. However, the number of single-time reviewers, which is the most suspicious user type, has been more or less stable in its ratio to multi-time reviewer since 2007, which means this method cannot find spammer groups, or collective behaviors. Jindal [50] defined several classes of expectations in his work, which employed certain unexpectedness measures to rank the rules for indicating unusual behaviors as spam activities. However, the aim of this study is to quantify the ratio of abnormal reviews in online e-commerce platforms, rather than detecting the trustworthiness of the reviews on certain products. Apart from understanding textual reviews, rating is another method used to detect spammers. In[26], Fei proposed a model to detect spammed products or product groups by comparing the differences in rating behaviors between suspicious and normal users. Xu [99] proposed a content-based approach for detecting email spam activities which unitize a fusion algorithm to dynamically capture a variety of spam patterns which may change with time. Jing conducted a survey on suspicious behavior detection in [47]. More than one hundred existing techniques have been researched which can be divided into three approaches: content-based, network-based and behavioral-based approaches. This work provides a detailed picture of how spam strategies have evolved and anti-spam techniques have developed.

2.2.2 Spam Studies in OSNs

Stein [90] proposed a model of a real-time adversarial learning framework to classify the read and write activities on Facebook. However, no further information was given on the features and policies used, which prevents comparison. Two offline investigations [31, 34] revealed large-scale spam activities on Twitter and Facebook respectively. As they focus on offline analysis, however, neither can be used to directly detect online spam activities. The method in [34] for securing a large number of individuals is time-consuming due to the adoption of a URL blacklist method while [31] utilizes a clustering method which needs to operate on an intact set of messages and hence has apparent limits on clustering efficiency. Thomas [93] designed a model to filter malicious URLs in OSNs in real-time to discover malicious messages, but this approach is simply a deep analysis of the URL's landing page which ignores the message content. There are also studies that focus on exploiting relationships for spam detection. Song [89] built a sender-receiver relationship to classify Twitter messages. The authors of [61, 91, 102, 104] designed and utilized machine learning techniques to discover spam groups and thus enable fraud identification. Anti-spam methods on YouTube and a social book-marking website were also detailed in [12, 72]. All these works select their most valuable features in the machine learning algorithm, for example, the key functionality in YouTube or the book-marking site in video and bookmark-sharing respectively, while the feature used in Twitter and Facebook is social network information. Thus, it is very difficult to select features in different spam detection scenarios.

2.2.3 Temporal Feature in Spam Detection

Tyler [76] studied the problem of temporal correlations between spam and phishing websites and conducted an empirical study. The study analyzed the temporal relationship between fast-flux attack spam and the lifetime of phishing websites. Shen et al. addressed the problem of 'link spam' [86] and their proposed model defined temporal information such as In-link Growth Rate and In-link Death Rate in a support vector machine (SVM) model for classification. Xie [97] proposed a model that only employs temporal features, with no semantic or rating behaviour analysis, to detect abnormal burst as the number of reviews increases. This work analyzed multi-dimensional time series by referring to singleton reviewers (users who have only written one review) for leveraging correlated anomalies. However, an obvious defect is that singleton reviewers may not always be spammers, and the scenario is not suitable for e-commerce platforms that require genuine transactions to take place before permission to comment is granted. This work defined review bursts and used Kernel Density Estimation with several features to detect them. It proposed a model based on Markov Random Field, and utilized a reviewer-reviews-store graph to detect spammers. These works inspire our belief that temporal analysis is a promising direction in spam detection, especially when other features are difficult to obtain or are immune to evolving spam strategies. In interesting research conducted by Xu in [100], a large-scale real-world dataset is collected from a telecommunication service provider and a temporal and user network information is combined to classify spammers in a Short Message Service (SMS).

2.2.4 Shapelet Learning Studies

Shapelets [105] are time series short segments that can best predict class labels. The basic idea of shapelet discovery is to consider all the segments of the training data and assess them according to a scoring function to estimate how predictive they are with respect to the given class labels [39]. The work in [105] built a decision tree classifier by recursively searching for informative shapelets measured by information gain. Based on the distance measure of information gain, several new measures such as F-Stat, Kruskall-Wallis and Mood's Median are used in shapelet selection [41, 64]. Since time series data usually have a large number of candidate segments, the runtime of brute-force shapelet selection is infeasible. A series of speed-up techniques have therefore been proposed. On the one hand, there are smart implementations using the early abandonment of distance computations and entropy pruning of the information gain heuristic [105]. On the other hand, many speed-ups rely on the reuse of computations and pruning of the search space [77], as well as pruning candidates by searching possibly interesting candidates on the SAX representation [80] or using infrequent shapelets [39]. Shapelets have been applied in a series of real-world applications. Instead of exhaustively searching shapelets, Rakthanmanon in [80] proposed learning optimal shapelets and reported statistically significant improvements in accuracy compared to other shapelet-based classifiers. Rather than restricting the pool of possible candidates to those found in the training data and simply searching them, they considered shapelets to be parameters that can be learned through regression learning. This type of learning method does not consider a limited set of candidates but can choose an arbitrary number of shapelets. In general, relative to our problem, the underlying collaborative temporal features(i.e., review time series) among products can be efficiently and effectively mined by using the shapelet learning method.

2.3 Part-3: Dual Social Influence Embedded Recommendation

In this section, we review three major research directions in recommender systems: traditional recommender systems, matrix factorization for recommendation, and social-based recommendation, all of which have attracted significant attention in the past. We also investigate the significance of social influence in networks, which is incorporated with recommendations in this chapter.

2.3.1 Traditional Recommender Systems

Recommender systems are software tools and techniques that provide suggestions for users [85]. Collaborative filtering has been adopted by many recent recommendation models and predicts an active user's preference for an unknown object based on the feedback of peers. Most existing methods can be categorized as either neighborhood-based methods and model-based methods. The former can be further divided into user-based methods [40] or item-based methods[63, 84]. The intuitive idea is to identify the similarities between general users and the target user based on the feedback of the former, where the similarity between two users is measured by the feedback on common items. However, this type of methodology faces the problems of data sparsity and cold start. Model-based methods alleviate the feedback scarcity issue by leveraging data mining or machine learning methods based on the training data and use the model to predict the active user's preference on the unknown item. Typical models include latent factor models [16], Bayesian models [110] and decision tree [14]. These methods provide foundational solutions to modern recommendation problems.

2.3.2 Matrix Factorization in Recommendation

The underlying assumption in this area is that matrix factorization can characterize users and items by vectors of factors inferred from rating patterns, and that a high correspondence between items and users will produce recommendation results.

One strength of matrix factorization is its ability to handle the sparse rating matrix. In[83], singular value decomposition (SVD) was exploited to reduce the dimensionality of the sparse rating matrix, which provides the best lower rank approximations of the original matrix. Another strength of matrix factorization is that it can be integrated with additional constraints. For instance, when explicit rating information is sparse, other implicit information (such as user purchasing history and browsing history) can be utilized to create a densely filled matrix. Recent studies illustrate that recommendation methods based on the bipartite network of items and users often exhibit outstanding performance. For example, using the non-negative matrix factorization method, [11] first calculates two non-negative matrices. In [28], a distribution framework is built based on collaborative filtering and the k-Nearest Neighbours algorithm with a fast response time in user and item partition.

Overall, matrix factorization assumes a latent factor space which can be mapped by users or items and can make predictions by minimizing the distance between the target rating matrix and the user and item vector. Our proposed model also adopts the matrix factorization method as the main element of the objective function by integrating it with a social information model based on dual influence-based maximization results.

2.3.3 Social Recommender Systems

The notion of social recommendation has attracted significant attention in both academia and industry. Due to the rapid evolution of social media platforms (*e.g.* Facebook, Twitter, and WeChat), the relationships between individuals in society have expanded remarkably.

In the early stage of social recommendation, social context information is widely used to improve the accuracy of prediction. Yao *et al.* [103] proposed a multi-layer context graph from user feedback data and developed two ranking methods (*i.e.* context-aware personalized random walk and semantic path-based random walk) to improve the recommendation. In [30], Gan *et al.* developed a data fusion approach that integrates historical and tag data for personalized recommendation, which improves the recommendation accuracy and diversity. Lu *et al.* [68] adopted the author's identity and social connections to design regularization constraints for integration with a text-based predictor. Their experiments proved that this combination improves the accuracy of review quality prediction even through the training data is sparse. In [74], social network analysis is fused with topic modeling, and this method can be utilized in many text mining problems, such as community discovery or spatial text mining. Many works also focus on using friends' network information in social data to improve recommendation performance. Only similar users are exploited in [98] for recommendation, with no consideration being given to social network information. In [36, 65], the proposed model either uses oversimplified heuristics or only analyzes neighbor relationships for recommendation. In [71], Ma *et al.* explored the incorporation of social friendships in recommender systems and designed two social regularization terms to enhance performance when missing ratings in the User-Item matrix are predicted. Based on the social regularization recommendation model, Ma *et al.* extended his work in [70], which employs users' implicit social relationships for recommendation. These implicit social relationships were defined between a user and other users according to the most similar or dissimilar rating behaviors. Interesting work in [92] indicates that most social recommendations only consider the local perspective of social content, seldom exploiting the global views of social relations. This work initially integrated global reputation into the proposed model and achieved improved prediction results; however, it mainly utilizes similarity when dealing with relationships from a local perspective. In [43], Hu *et al.* proposed a recommendation framework named MR3, which jointly models users' rating behaviors, social relationships, and review comments. In [66], Liu *et al.* proposed a probabilistic relational matrix factorization (PRMF) model which learns the optimal social dependency between users to improve the recommendation accuracy, with or without reference to users' social relationships.

2.3.4 The Effectiveness of Social Influence

Most social recommendation systems only make use of an individual's explicit relationships, such as those who connect directly with them, ignoring any analysis of the influence of the implicit relations of users. We therefore review work on social influence maximization.

Many works focus on retrieving influence maximization results. Wang *et al.* [95] discovered influential vertices on the basis of community detection and selection, while Barbieri *et al.* [9] proposed a social influence model based on topic modeling. In [56], two shortest route-based influence cascade models are proposed to improve efficiency. Another finding [20] is that influence diffusion increases with the degree of nodes; thus [19, 21] enhance efficiency by decreasing the computations on the local influence community. A new perspective is provided in [35] which aims to find the local influential nodes for each target user in the network. For instance, Bill Gates may not be the most influential person to someone who follows him on Facebook - the greatest effect may be exerted by someone's second-nearest connection such as a friend's friend.

As previously mentioned, global and local influence maximization affects different aspects, thus it is necessary to integrate these two different elements to provide more comprehensive social information.

Chapter 3

Temporal and Spatial Aware Social Trajectory Recommendation

In this part, we introduce a temporal and spatial aware social trajectory recommender system which aims to employ heterogeneous network information to help the user make more reasonable decisions when they arrive in new cities and thus enhance the user experience and stickiness.

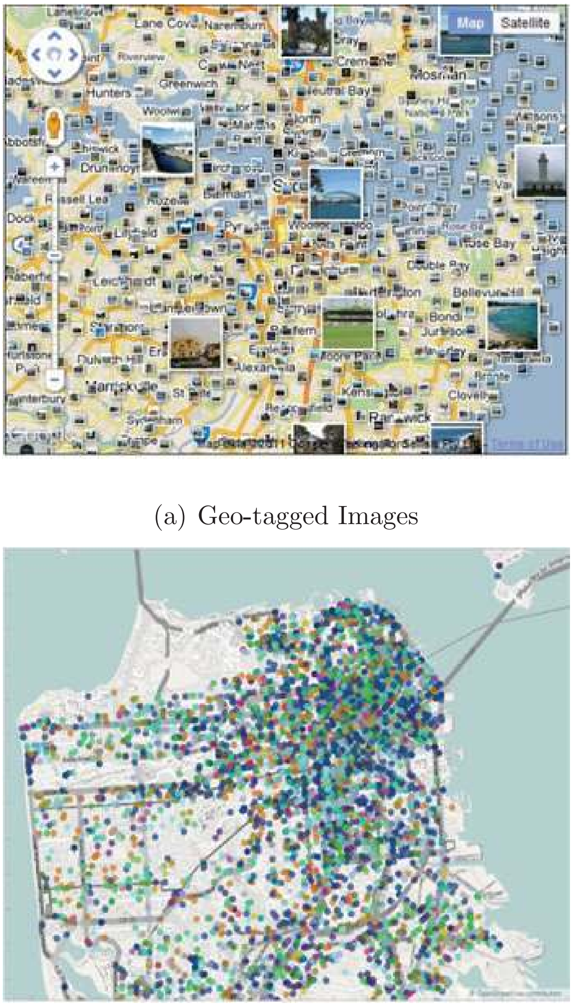
3.1 Introduction

Travel is a hot topic today and the role it plays during one's life has grown significantly, which can be seen in the drastic growth of online tourism communities and applications. Accelerated by advances in mobile devices, a wide variety of sophisticated online applications have been developed which encourage travelers to share details of their travel experiences, which are gradually replacing traditional travel logs which were mostly in the form of online travel communities. Different from text-based logs, modern techniques enable travel experiences to be shared in a much more vivid and efficient way, sharing the temporospatial information during their travel.

A GPS trail is one of the most representative forms of human trajectories. Though a GPS trail provides very detailed and continuous information about people's travel decisions and preferences, there are still several limitations when using such information for recommendation tasks. Specifically, sample size is one of the clear limitations, as GPS data are not concise enough to monitor trails inside a complex such as a shopping mall with all venues inside. Another limitation is that GPS data lack social information, such as the user's characteristics, which make it very difficult to offer personalized recommendations. Finally, GPS data tend to produce many unpredictable noises which make it difficult to recognize a real tourism position. For example, heavy traffic conditions can produce dense trails, however this seems to be of little interests to tourists.

The advent of the location-based service network (LBSN) provides another form of landmark information when travelling, such as geo-tagged image, check-ins, etc., which can resolve the limitation of relying only on GPS data. For instance, a geo-tagged image not only contains concise information on tourist attractions, but also personal social information. In addition, offering travel recommendations based on geo-tagged images can fully employ the collective intelligence of individuals who are travelling. However, the nature of geo-tagged images means that they are limited in terms of being only able to consider one specific type of travel information, i.e. sight-seeing information and they do not take into account other aspects of travel in which a tourist may be interested, such as dining and shopping, which are not included in geo-tagged images and make existing systems overly sensitive to the popularity of traditional tourist locations. For instance, if a business traveller has limited time and a specific geographic travelling area, then simply recommending trajectories based on geo-tagged images does not take into account the users true requirements.

To address these limitations, we introduce check-ins as another form of social information for mining trajectories. Check-ins is a new LBSN application and has become widely used online recently. With more detailed semantic denotation about a geographic location (i.e., the name of venues, such as "Starbucks coffee shop"), check-ins provide more informative data on real-world venues, such as restaurants and bars, than GPS data. Also importantly, temporospatial information and de-



(a) Geo-tagged Images

(b) Check-ins

Figure 3.1 : Geo-tagged Images and Check-ins

pendencies are stored in check-ins, making it very easy to find trajectories from the check-in transitions history.

These limitations in existing trip recommendation can be summarized as follows:

- The nature of GPS data limits more personalized and accurate recommendations, and it overly relies on data quality.

- Recommendations which only utilize geo-tagged images miss information on local areas of interest, such as restaurants and bars, thus, travellers are not able to access relevant information on local travel destination.

- Most trip recommendation applications recommend a sequence of single locations and do not take into account a traveller's temporal and geographic constraints.

To solve these problems, we combine data from geo-tagged images and check-ins (as shown in Figure 3.1) to discover areas of interest (AOI). Then we recommend a social trajectory which comprises several AOIs, instead of single locations. Finally, according to the individual's input in terms of their preferred start and end information(time and location), we can dynamically make recommendations to them. We comprehensively solve all the existing limitations and our contributions are summarized as follows:

- We build a geo-tagged image and check-in heterogeneous network to discover the AOIs, which represents both traditional and well-known tourist attractions and local lesser-known choices.
- Our recommended trajectories are not isolated locations but rather, several AOIs within a well-designed level of granularity, which enables travellers to make more flexible decisions.
- We can dynamically generate the recommended social trajectory based on user input, both temporal and spatial.

The rest of this chapter is organized as follows. Section 3.2 introduced the related work; Section 3.3 describes the proposed model; Section 3.4 overviews the framework; and the experiment is discussed in Section 3.5.

3.2 Related work

Recent research methodologies used in trip mining and recommendation can be roughly categorized into three groups: travelogues (i.e., blogs), GPS trajectory data and geo-tagged photos.

Many researchers seek ways of using textual information for trip recommendation. Ji et al. emphasize the mining of city landmarks using a graph-based method [46]. Hao et al. [37] propose a probabilistic topic model to generate topics from travelogues and then represent locations with appropriate topics for further destination recommendation and summarization. In [32], an interesting model was proposed to automatically recognize and rank the landmarks for travellers. However, unstructured travelogue-based data contain much noisy metadata. They also only play a role in destination recommendation by merely showing information about a location. Furthermore, these kinds of data are more or less out of date and people seldom share their experiences by writing blogs. In his pioneering work, Zheng et al. [113, 111, 112] employed GPS properties to discover popular and interesting locations and classical travel sequences to provide a personalized friend and location recommender using the similarities between users in terms of their location histories. The main obstacle of the trajectory-based method is that data resources are not easy to obtain from a large number of people. There are also concerns about the use of GPS trajectories for these tasks, for example, it is difficult to track a person's path inside a large shopping complex or identify an area of heavy traffic. Geo-tagged images have been exploited in trajectory recommendation due to the boom of LBSN service. An excellent survey on the application and approaches of geo-tagged media is given in [69]. The methods in [59] and [67] assume that all geo-tagged photos are remaining to tourists and tourist itineraries are also readily available. An itinerary is defined as a list of landmarks that a person visits, which is generated by mapping photo geospatial coordinates to the latitude/longitude of the given landmarks. In contrast, the itinerary in the proposed work is defined to be the spatio-temporal movement trajectory with much finer granularity. However, most of the geo-tagged images are limited in traditional tourist attraction data, which lacks information on local choices, such as restaurants, bars, etc. Therefore, it is difficult to generate an intact trajectory by simply utilizing geo-tagged image data.

In this chapter, to improve the quality of trajectory recommendation, we integrate check-in data with geo-tagged images to build a heterogeneous network to discover AOIs, to recommend trajectories instead of single locations.

3.3 Methodology

3.3.1 Problem Definition

In general, the problem we study is straightforward. We first define an AOI as a heterogeneous structure by combining geo-tagged images and check-in data. Then, we develop a model to dynamically recommend trajectories consisting of AOIs according to user predefined temporal and spatial constraints. In this chapter, two terms are defined referred to represent geo-tagged images and check-in respectively:

- tourist location (or scene) is a traditional popular travel destination, carried in a geo-tagged image. In an LBSN platform, the majority of geo-tagged images shared by people are tourist attractions.
- venue location(or venue)is carried in a check-in entry. In addition to its GPS coordinates, it also contains specific name of the location. For example, {‘Starbucks; 31.247378, 121.475067} point to a Starbucks coffee shop in a certain geographic location. As check-in data is shared by local people, or people who have lived somewhere for a long time, we assume these locations represent local choices in relation to dining, entertainment, etc., which is also

very important information for travellers.

Table 3.1 details the notation and symbols that are used in this chapter.

3.3.2 Generating AOIs

Travelers' interests are our primary consideration for recommending a trajectory to them. To cater for their interests in both traditional sight seeing and local events or activities, we extract tourist location from the geo-tagged images and venue location from the check-in data. Then, we perform density-based clustering on the combined heterogeneous network to generate the clusters, which are called AOIs in our application.

Table 3.1 : Symbols and notations

ς	reachability-distance threshold
k	maximum number of neighbors when cluster expand
$d(*)$	In this work, it refers to Euclidean distance
p_i	a tourist location
$p_{k,k'}(p_i)$	neighborhood venue location density
$l_a \to l_b$	original transition history
$\bar{t}_{p_j \to p_*}$	average transition history from p_j to other scenic locations
θ	angle of two trails
A	an AOI
A_a	AOI ranking score
A_F	AOI liveness vector
A_d	AOI duration
ϕ	travel time threshold
ω	travel time
G	transition graph
τ	transition histories
i	visiting time from 0 to 23
$\kappa(V)$	overall number of visitors

To this end, we adopt an improved density-based clustering algorithm based on DBSCAN [25] and OPTICS [3] to take both geo-tagged images and check-in data into account to generate AOIs. Here, through the clustering algorithm, geo-tagged images and check-in data are the same elements - pairs of geographic coordinators, and we cannot mix them together to get the AOIs. This is because there is more check-in data than geo-tagged images and this imbalanced distribution will impair the clusters' quality. In other words, check-in data is important support information for travel recommendation, but geo-tagged images are an integral part. Mixed data will bias the cluster to check-in data. Therefore, we adopt the two-step cluster method to discover AOIs. First, we conduct density-based clustering for tourist location area mining and then utilize secondary clustering to appropriately plug into the check-in data to form AOIs.

Primary Clustering

We first introduce primary clustering on a tourist location, and then we discuss secondary clustering to add in venues which are obtained from the tourism locations' neighbors to further include local choices. Accordingly, the input of primary clustering is a set of tourist locations. Through the algorithm, the density of tourist locations represents how popular this area is for travelers and the first clusters generated are considered as the initial AOIs. As AOIs are located in a particular city, a distance constraint will still be required to measure the cluster granularity size for the maximal distance among the scenes within the same initial AOI. According to the average tourist location density of our datasets, we set reachability distance ς to 300m and k = 25, which means that only 25 neighbors within 300m will be considered when a cluster grows. More details on basic density-based clustering can be found in [3]. After this primary clustering process, the venue locations are then joined to their nearest AOI in secondary clustering.

Secondary Clustering

Secondary clustering mainly attaches a venue location density constraint when a scenic location is assigned to a cluster in the basic algorithm. As a result, the neighbouring venues to the tourist location will slightly regulate the AOIs. The first round density-based clustering procedure goes through the tourist locations as the clusters are expanded, rather than being processed with a specified order. However, this does not obtain ideal results. For instance, Figure 3.2 demonstrates two clusters, A and B, and a tourist location p between A and B (blue points are tourist locations and green points are venue locations). Assume that p is not a core point (the cluster cannot further expand from p) and it can be included in either A or B, depending on which cluster is processed first. Now if the clustering algorithm processes A first, p is a member of A after the clustering ends. However, we expect p to be part of B, because despite having similar tourist location densities, venue locations in B also much more dense, which is an important reason why we modified the clustering algorithm with a secondary density evaluation to reinforce the order and to constrain the choice of which clusters to assign to a boundary tourist location such as p in Figure 3.2. Therefore, we define the neighboring venue density as follows:

Given a tourist location p_i, its neighborhood venue location density, denoted as $p_{k,k'}(p_i)$, is computed as:

$$p_{k,k'}(p_i) = \overline{p_{k'}(p_j)}, \tag{3.1}$$

where p_j is one of pi's k-th nearest tourism locations, $p_{k'}(p_j)$ is the average distance of p_j to its $k^{'}$ nearest venues, and $\bar{\cdot}$ represents the average.

Neighborhood venue density measures how easily a traveler can access neighboring venues around a neighborhood of tourism locations. The k nearest tourist locations of each tourist location p_i are initially identified. Then, for each neighbor

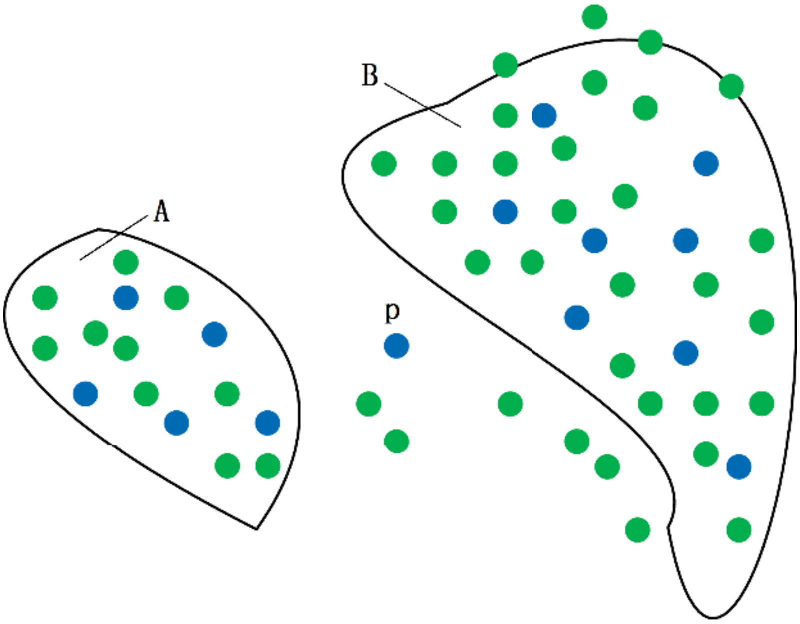


Figure 3.2 : Effect of secondary clustering

of p_i - p_j, all its $k^{'}$ nearest venues' locations are then determined and the average distance of p_j to its $k^{'}$ nearest venues is finally used to compute the neighborhood venue density of p_i by the average function. When a tourist location is attached to a cluster, its surrounding venues are also attached to the cluster if the venue is not yet assigned(note that a venue can only be assigned once). These changes ensure that tourist location cluster with higher neighborhood venue densities will be first processed, consequently in cases like Figure 3.2.

3.3.3 Generating social trajectories

After discovering the AOIs, we dynamically recommend social trajectories according to the AOIs popularity and the temporal and spatial constraints imputed by users. Thus, we first discuss how to represent an AOI by its overall popularity ranking score, liveness and duration information. Finally, we introduce a dynamic way of generating temporal and spatial aware social trajectories.

Representing AOIs

To make dynamic trajectory recommendation, we represent an AOI by three different pieces of information - its ranking score, liveness by time and average duration in that area. We discuss these three pieces of information separately in the following sub-sections.

Ranking Score information of AOIs

An AOI has a heterogeneous structure which consists of tourist locations and venue locations, so location density mainly indicates its popularity and the ranking of an AOI can be treated as web link importance analysis. Authority analysis is the traditional solution for such a problem, thus we first collect the original and inferred transition history from the datasets, and then propose a combined authority analysis algorithm to assess the ranking of AOIs. The first transition history - original transitions history - can be defined in Definition 2 as follows:

An original transition $l_a \rightarrow l_b$ indicates one traveller's movement from one place l_a to another l_b. It can be collected from a pair of geo-tagged images or check-ins referring to two consecutive locations l_a and l_b, if and only if the following conditions are met:

1. The pair of geo-tagged images or check-ins are posted by the same user.
2. The pair of geo-tagged images or check-ins are two consecutive entries in terms of their post time from that user.
3. The pair of geo-tagged images or check-ins are posted on the same day but carry different location information.

However, the mutual independence of geo-tagged images and check-in data makes the original transition history sparse and it is not possible to estimate the AOI

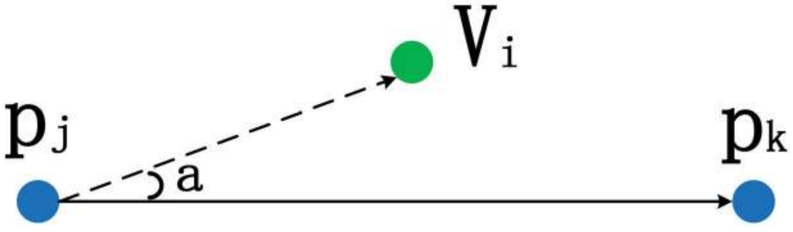


Figure 3.3 : Inference Transition

ranking score as key information. Therefore, we need to further explore additional information to evaluate the AOI ranking. Here, we design an inferred transitions history among locations from these two different datasets. Intuitively, if a user travels from one tourist location to another on the same day (an original transition), they probably have also visited these venues while on the original transition. Based on the above assumptions, we define Inferred Transition History in Definition 3 as follows:

v_* and p_* denote venues and tourist locations, respectively. Given an original transition from p_j to p_k (Figure 3.3), the dashed line from p_j to v_i is defined as an inferred transition $d(v_i, p_j) \leq d(p_j, p_k)$ if the angle α between these two transitions is not greater than a threshold θ. The inferred transition history from p_j to v_i is estimated as:

$$t_{p_j \to v_i} = cos\alpha \frac{d(p_j, p_k) - min(d(v_i, p_j), d(v_i, p_k))}{d(p_j, p_k)} t_{p_j \to p_k} \frac{\bar{t}_{v_* \to v_i}}{\bar{t}_{p_j \to p_*}} \tag{3.2}$$

where $\bar{t}_{v_* \to v_i}$ is the average transition history from other venues to v_i, and $\bar{t}_{p_j \to p_*}$ means the average transition history from p_j to other tourist locations. The inferred transition history from venues to tourist locations can be defined similarly.

The inferred transition history is made up of three components. For instance in Equation 2, the first component $cos\alpha \frac{d(p_j,p_k) - min(d(v_i,p_j),d(v_i,p_k))}{d(p_j,p_k)}$ models the probability of the user deviating on the way of $p_j \to p_k$ and stops at v_i. The more similar the directions are and the shorter the distance to either p_j or p_k, the larger chance of such as a detour. The second component $t_{p_j \to p_k}$ measures how significant the transition history is while the third component is the normalization of this significance. In our work, θ is set to $15°$. The inferred transition history is assumed to belong to users who contribute to the transition, i.e., the user who has the original transition history $p_j \to p_k$ in Equation 2. We denote τ as the intact transition history, including both original and inferred ones.

With the intact transition history, we utilize a HITS-based link analysis method to rank AOIs. We model locations and their relations in a graph. Formally, $G = L = l_1, ..., l_n, \tau = t_1, ..., t_n$, where each node l_i in the graph is either a tourist location or a venue, and the directional edge t_{ij} represents the transition from the node l_i to l_j with the weight as the transition history from l_i to l_j. The basic authority analysis uses the hub scores and authority scores to describe the nodes' significance outwards and inwards. We denote the hub scores and authority scores for all the locations as $h = (h_1, ..., h_n)^T$ and $a = (a_1, ..., a_n)^T$, and summarize the iterations of authority analysis as repeated applications of the following two mutually reinforcing operations, $AUTH(*) = T^T$ and $HUB(*) = T$, as:

$$a = AUTH(h) = T^T h \tag{3.3}$$

$$h = HUB(a) = Ta \tag{3.4}$$

$$a^{i+1} = AUTH(HUB(a^i)) = T^T T a^i \tag{3.5}$$

$$h^{i+1} = HUB(AUTH(h^i)) = T T^T h^i \tag{3.6}$$

where i is the number of iterations. Here, the adjacency matrix T is defined as the matrix form of the directional edges τ between each pair of nodes in L.

The basic authority analysis does not take the users' experiences into account, so each transition is equal in computation. However, we can observe in the real world that each transition should have different weights, according to the user's experience and their activeness. For instance, a transition from a user with more experience contributes more to the significance of transitions. Hence, we propose a combined authority analysis framework to improve the basic authority analysis. In this framework, we consider two hub scores, i.e., the hub score $h_l = (h_{l_1}, ..., h_{l_n})^T$ based on the location-location relationship and the hub score $h_u = (h_{u_1}, ..., h_{u_m})^T$ from the user-location perspective. In addition, the combined authority score is defined as $a = (a_1, ..., a_n)^T$. Here n is the number of unique locations and m is the number of unique users. In addition, despite the T_l adjacency matrix retrieved from τ , we also build another user-location matrix T_u of $m \times n$. The value of $t_{u_{ij}}$ is the i_{th} user who contributes $t_{u_{ij}}$ visits to the j_{th} location. We then propose the authority analysis framework as:

$$h_l = HUB_l(a) = T_l a \tag{3.7}$$

$$h_u = HUB_u(a) = T_u a \tag{3.8}$$

$$a = AUTH(h_l, h_u) = T_l^T h_l \diamond T_u^T h_u \tag{3.9}$$

where $\diamond$ is the element-wise product. Note the h_u increment the authority scores with the locations that a user has visited, then the $T_u^T h_u$ reproduces the authority scores for each location from the user's hub scores, as well as their visiting histories.

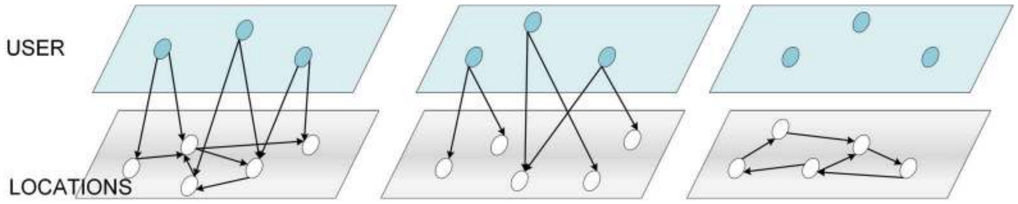


Figure 3.4 : Combined Authority Analysis

A quality hub is a location which has more outbound transitions to significant locations while a quality authority is a location which has more inbound transitions from significant locations. Moreover, user is a quality hub if they have more outbound transitions to significant locations whereas a user is a quality authority if the locations have more inbound transitions from significant users. The iterative process of Equation 9 is summarized as:

$$a^{i+1} = AUTH(HUB_l(a^i), HUB_u(a^i)) = T_l^T T_l a^t \diamond T_u^T T_u a^t \tag{3.10}$$

where we also have ${h_l}^{i+1} = T_l a^i$, and ${h_u}^{i+1} = T_u a^i$. After each iteration, the hub scores and authority scores are all normalized, i.e. ${a_j}^i = \frac{{a_j}^i}{\sqrt{\sum {a_*}^{i2}}}$, to guarantee convergence. In our experiments, convergence is produced within four iterations. From left to right as shown in Figure 3.4, we illustrate three different models: the combined authority analysis, the framework described in [113] and the basic authority analysis. Basic authority analysis only considers transitions between locations. The framework proposed in [113] only employs user-location relations without consideration of location-location transitions. We cover both these two relationships to provide the ranking score of AOIs.

Liveness information of AOIs

With the result of the combined authority analysis, we obtain the ranking of locations based on their authority scores. However, a recommended trajectory is different from a recommend single location, hence we also need to consider temporal appropriateness. For instance, a nightclub is very popular for travelers, thus it gains a higher ranking score, but it is not suitable for inclusion in a trajectory if people only want to tour a city during the day time.

Based on the above observation, we use a 24-dimension vector $F = f_0, ..., f_{23}$ to represent the liveness information of an AOI for each hour during one day. Here, each element f_i in F is a numeric value which can be calculated as follows:

$$f_i = \forall_i \kappa(V) \tag{3.11}$$

where i is the visiting time, for instance, 7 means $8:00am$. $\kappa(V)$ is the overall number of visitors who have checked-in or posted a geo-tagged image at time i. Here, we round the users' posting time by half an hour. For instance, given a posting time between $8:00$ and $8:30$, we count the visiting number into $i = 7$, otherwise $i = 8$. For each AOI A, we normalize its corresponding liveness vector F into the range [0,1].

$$A_F = \frac{f_i - min(F)}{max(F) - min(F)} \tag{3.12}$$

Duration information of AOIs

The last important piece of information to represent an AOI is the suitable duration time for the traveler to spend in this area. As an AOI comprises number of locations and people will not visit each location in this area, thus we assign a suitable duration time to an AOI, as a suggestion for users to make their own decision.

According to Definition 2, we defined the original transition history $t_{l_a \to l_b}$ to support AOI ranking. As previously mentioned, one of the conditions $t_{l_a \to l_b}$ which needs to be met is that such a transition must occur on the same day. In addition, each location in an AOI may correspond to many transition histories from different users. Hence, we calculate the duration for one location d as follows:

$$d_{l_a} = \forall_{l_a} \overline{\sigma(time_{l_a} - time_{l_b})} \tag{3.13}$$

where d_{l_a} is the duration time of location l_a in one specific AOI, $\sigma(time_{l_a} - time_{l_b})$ is the transition time interval in location l_a, and then we average the stay time of all the users who have ever been to l_a.

After we get d_{l_a}, we find the maximum duration time for this AOI as different users may have different plans in terms of the duration of their stay in a specific place. Thus, the maximum duration time may be too long for some users, but it is better than visiting somewhere with insufficient time. Thus, the duration information of each AOI A can be calculated as follows:

$$A_d = \forall_{l_a \in A} Max(d_{l_a}) \tag{3.14}$$

Overall, we get three different kinds of information: ranking, liveness and duration to build a vector to represent each AOI A as A_a, A_F, A_d. To this point in time, each AOI contains comprehensive information on not only their popularity, and the best visiting time but also a suitable duration period. Next, we discuss how to employ this information for dynamic social trajectory generation.

Dynamic social trajectory generation

Generally, to dynamically generate a social trajectory, the next AOI which will be included in the recommended trail relies on the information on the last AOI. At the beginning, our system will ask the user to input four pieces of information - start time $Time_s$, end time $Time_e$, departure location Loc_s and destination Loc_e. Thus, starting from Loc_s, our task is always related to how to find the next best stop from

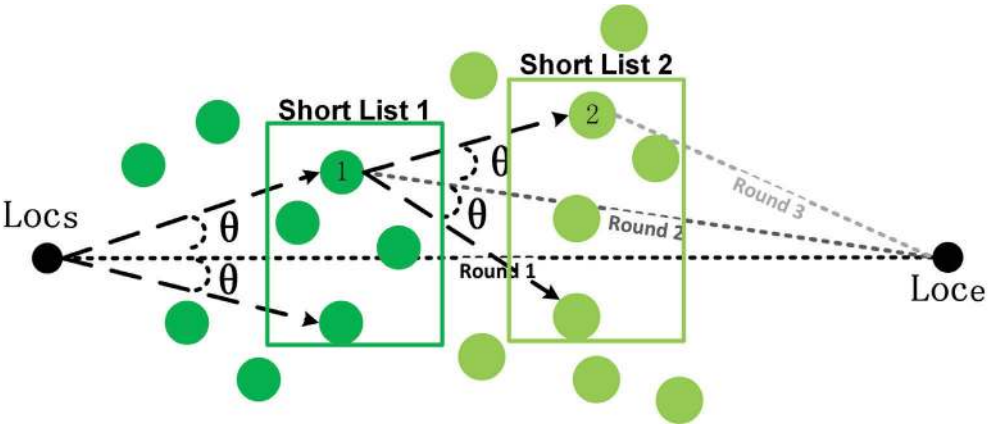


Figure 3.5 : Choose Next Stop

all the AOIs by considering its popularity, temporal and spatial features.

As seen in Figure 3.5, both the green and the light green circles represent AOIs. To determine the next stop, we always consider two aspects with reference to its geographic properties, range and distance. As we do not want the main direction of a trajectory to deviate too much, we first search for all the AOIs in a suitable range. Notice that each AOI has a center C_A which is a pair of geographic coordinates, hence we can calculate the angle θ of all AOIs, Loc_s and Loc_e as shown in Figure 3.5, and we set θ to 15° as well. Then, we filter some of those AOIs whose travel time exceeds the threshold ϕ. Here, we get the travel time ω from Loc_s and C_A by Google maps public transportation travel time API, and we set ϕ to 20 minutes. Then, we have a short list of AOIs to choose the first stop. The best one in this list is determined by the feature A_a and A_F. Note that for each AOI, A_F is a 24-dimension vector which represents the liveness of the AOI. According to $Time_s$, we can get the corresponding element $A_{F_{Time_s}}$ in A_F. Finally, we assign each AOI in the current short list a recommendation score η which is calculated as follows:

$$\eta = A_a \times A_{F_{Time_s}} \tag{3.15}$$

where $A_a \times A_{F_{Time_s}}$ means we consider the AOI's popularity and liveness with reference to the current time, simultaneously.

This is the process of next stop selection in the first round. After this, we update several elements and start the second round. Specifically, the second round start location is the first AOI (coordinates of its centre) as shown in Figure 3.5. Then, based on the duration time A_d of the first AOI, the first start time $Time_s$ and the first travel time ω, we update the next round start time ${Time_s}^{new}$ as follows:

$$ {Time_s}^{new} = {Time_s}^{old} + \omega + A_d \tag{3.16} $$

where only at the beginning of the second round, ${Time_s}^{old}$ is $Time_s$ which is assigned by the user, after which it is automatically calculated. This process will be ended if $({Time_s}^{new} - Time_e)$ is less than 30 minutes. Next, we discuss the algorithm and system framework.

3.4 Framework and algorithm

In the previous sections, we introduce how to generate AOIs and how to utilize three types of information to represent an AOI. Furthermore, we discuss how to dynamically generate a temporal and spatial aware social trajectory by considering AOI features and user input. Figure 3.6 shows the structure of our model.

Our Algorithm is pretty straightforward, for a pair of given start time and end time, with a pair of departure location and destination, we recursively general a next step according to start time and location, by considering the temporal and spatial features learned from our dataset. The whole process will end up when condition fulfilled, in other words, no more point can be added in to the trajectory when consider the end time and destination.

By considering all these factors, we develop a temporal and spatial aware social

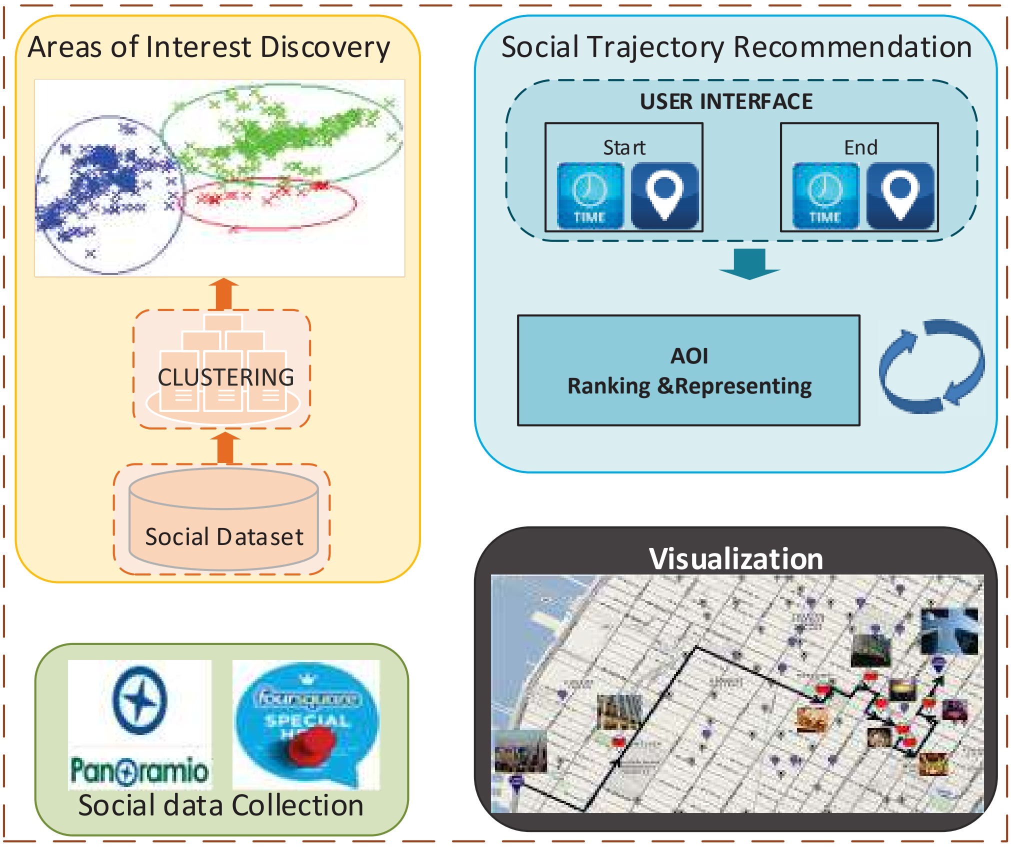


Figure 3.6 : Social Trajectory Recommendation Framework

trajectory model (TSASTM) Algorithm 1 to dynamically generate travel routes for users, which is depicted as follows:

3.5 Experiment

In this section, we first conduct an analysis on two datasets and then we evaluate the recommendation quality of social trajectory. As our social trajectory recommendation does not have absolute ground-truth, thus we obtain the ground-truth knowledge online and from human efforts online. Then, we provide two user cases to demonstrate the visualization of our system.

Algorithm 1: TSASTM Algorithm

```
Input  : $Time_s$, departure time;
         $Time_e$, end time;
         $Loc_s$, departure location;
         $Loc_e$, destination;
         $A$, all the AOIs;
         $\phi$, travel time threshold
         $\theta$, direction angle threshold
Output: Social trajectory from departure location to destination within assigned
         time frame.
$Initialize\ \ Time_left = Time_s - Time_e$
$Initialize\ \ SocialTrail = \{Loc_s\}$
while $Time_{left} > 30minutes$ do
    $1.Seeking\ next\ best\ stop\ with\ Time_s,\ Loc_s:$
        $Next_s = \forall_A Max(\eta)\ \ by\ \ Equ.3.15$
        $Time_s^{new} = Time_s^{old} + \omega + A_d\ \ by\ \ Equ.3.16$
        $SocialTrail\ \ += Next_s$
    $2.Update\ Loc_s\ and\ Time_s:$
        $Loc_s = Next_s$
        $Time_s = Time_s^{new}$
    $3.Recalculate\ left\ travelling\ time:$
        $Time_left = Time_s - Time_e$
end
    $SocialTrail\ \ += Loc_e$
$Output\ \ Visulazation\ Social\ Trajectory$
```

3.5.1 Dataset Analysis

We consider geo-tagged images in combination with check-ins to discover AOIs. Note that check-ins have several unique characteristics compared to geo-tagged images and raw GPS records. The location carried by a geo-tagged image does not include much venue information. A typical geo-tagged image is embedded with a title, while a GPS record is not usually associated with any semantic information. This means check-ins have richer and more accurate semantic information than the other two data types. Furthermore, check-ins represent more daily life activities, while geo-tagged images often show more tourist attractions. Raw GPS records could contain both, but in a vague and inaccurate fashion. Thus, in light of the fact that the integration of geo-tagged images and check-ins can provide more comprehensive travel information to tourists, we collect these two kinds of data in New York city, described as follows:

- CheckIn4sq: we obtained around 22,000 venues from 2 million check-ins from a public dataset[*]
- GeoPanoramio: we collect around 15,000 tourist location images from Panoramio.com

Next, we examine the statistical significance for both datasets. The check-in entries are contributed by 89,374 users, and 121,989 unique venues are covered. Naturally, of the vast user community, most of them remain rather inactive, as the average number of entries per user is only 22.3. If we examine the average number of entries regarding venues (16.39), it is even more scattered as there are more venues than users. In terms of activity and popularity among the users and venues, a long tail effect can be observed, that is the top 10% of active users contributed 85.4% of the total check-in entries, and the top 10% of visited venues received 65.9% of the total check-in entries. For the dataset GeoPanoramio, there are 2,553 users who have created an average number of 5.87 images. Again, the top 10% of active users shared 69.54% of all the uploaded images, with each top user uploading 35.78 images on average.

[*] http://www.public.asu.edu/h̃gao16/dataset.html.

Table 3.2 : Recommendation Ranking Score Criteria

$SocialTrajectoryQuality$	Criteria
5	I would definitely follow this recommendation.
4	I would follow most parts of this recommendation.
3	I would follow at most half of this recommendation.
2	I would not choose most parts of this recommendation.
1	I definitely would not agree with this recommendation.

3.5.2 Social trajectory recommendation evaluation

Similar to the problems of web page ranking [15] or GPS-trajectory-based location mining [113, 18, 114], our problem also does not have absolute ground-truth. Thus we obtain the ground-truth from human efforts online. Specifically, in our experiment, 103 annotators are involved in the annotations, these being 84 females and 19 males who have lived in New York City for more than 3 years. Each annotator randomly assigns their preferred departure time and location, as well as end time and destination. Then, our system will recommend a social trajectory according to the input information and displays this to the annotator. The annotator marks this social trajectory quality score between 1 to 5 according to what was recommended to them. The social trajectory quality score indicates how useful or interesting it is to the annotator. The criteria of the different quality scores are listed in Table 3.2.

The 103 users voted 231 times on recommendations and the average number of user voting was 2.2. Specifically, females made 186 recommendations and voted while males made 45 recommendations. The recommendation quality scoring is given in Table 3.3, showing that the most positive scores (score as 4 or 5) account for more than 70% of all the rating criteria, which indicates that the majority of our recommendations are useful to the users.

We also compare the rating results between the female and male groups. As

Table 3.3 : Recommendation Quality Scoring

RecommendationScoring	Proportion
5	0.277056277
4	0.45021645
3	0.181818182
2	0.082251082
1	0.008658009

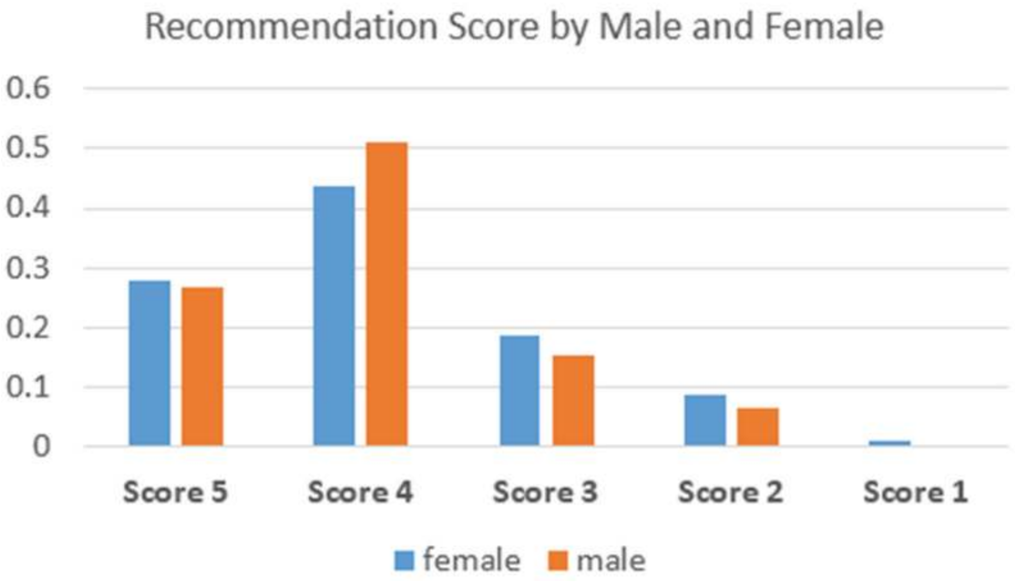


Figure 3.7 : Recommendation Scores by Gender

seen in Figure 3.7, the overall distribution of ratings in terms of user satisfaction is similar for both groups. It is interesting that none of the males gave the least satisfied score of 1 for our recommendations, whereas two recommendations received a score of 1 from the female group. Another interesting finding which can be seen in Figure 3.7 is that more males gave a rating score as 4 than females, which indicates that the males in this study were more satisfied with the recommendations.

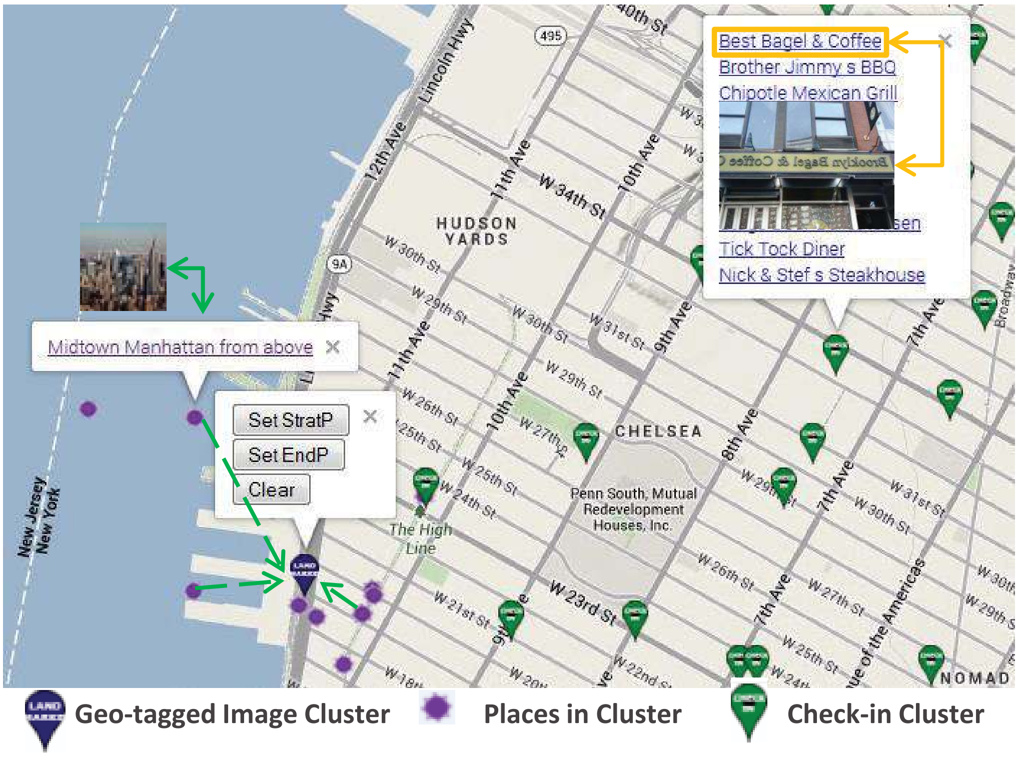
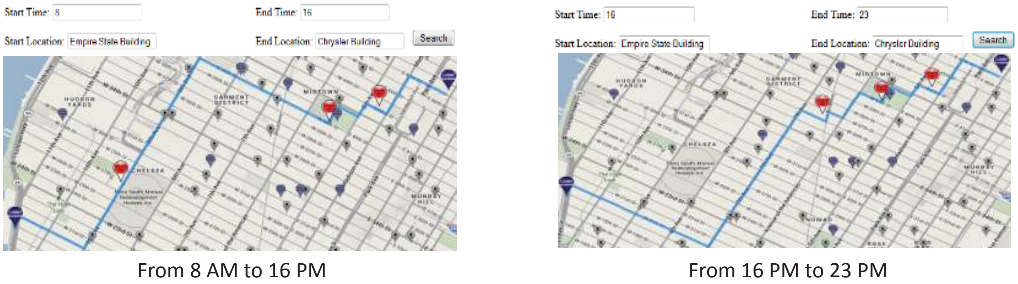


Figure 3.8 : AOI Visualization and Manipulation


Start Time: 8 End Time: 16
Start Location: Empire State Building End Location: Chrysler Building
Search

From 8 AM to 16 PM

Start Time: 16 End Time: 23
Start Location: Empire State Building End Location: Chrysler Building
Search

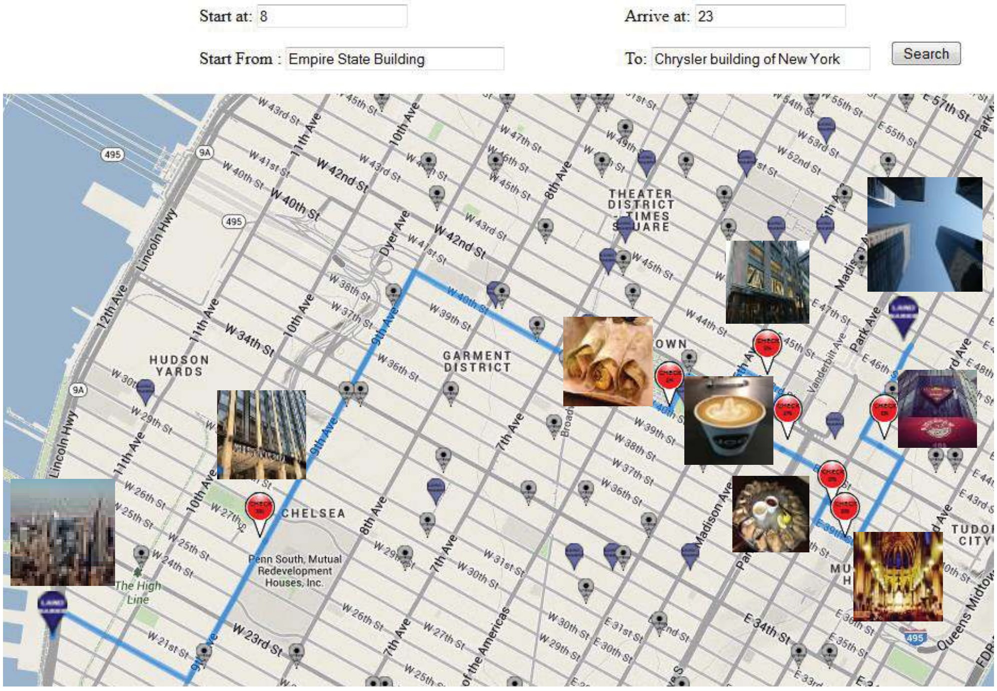
From 16 PM to 23 PM

Figure 3.9 : Temporal and Spatial Aware Recommendation

User Scenario Study

Several representative user scenarios of our recommender system are discussed in this subsection.

- As shown in Figure 3.8, the users click on a cluster(which represents an AOI)



Start at: 8

Arrive at: 23

Start From : Empire State Building

To: Chrysler building of New York

Figure 3.10 : Social Trajectory Recommendation

and they will be shown several points of interest which belong to this cluster. Then the users can click on these points of interest to get the link (represented by the place name) in order to browse the details on the official website.

- Also as shown in Figure 3.8, after clicking on a cluster, the user has access to a series of buttons by which they can set this AOI as the starting point or end point.

- Figure 3.9 shows that even when the same departure location and destination is set, for instance, from 'Empire State Building' to 'Chrysler Building', our system will provide a different trajectory according to the start and end time assigned by the user.

- Figure 3.10 shows an intact social trajectory recommendation case, where after choosing the starting point and destination with the start time and end time, the users can post queries and the recommended social trajectory will be presented on the map. In this case, we can see that throughout the trip, the system we not only consider traditional tourist attractions but also well-known local places, such as restaurants, coffee shops and BBQs, which makes the recommendation much more comprehensive and thus enhances the user experience.

Chapter 4

Collective Marketing Hyping Behavior Detection

4.1 Introduction

E-commerce and opinion-sharing websites are flourishing with the development of Web 2.0 technology. These online platforms encourage people to share their personal opinions, attitudes and feelings, which are not only related to products and services but also a variety of societal issues. These comments on specific products and online stores significantly affect customer purchase decisions [106, 60], therefore customer reviews are very valuable to individuals and online businesses. Sales volumes and profits, to some extent, rely on the number of positive reviews [42, 116]. As a result, some businesses resort to paying for fake reviews from an online service to unfairly hype themselves or denigrate competitors. Researchers have detected spam activities by analyzing important information about reviewers' behavior, user networks and semantic opinions [48, 49, 62, 27, 50, 26] and can successfully identify fraudsters or untrustworthy reviews that are operated in an automatic or semi-automatic way, in which spammers' online IDs are controlled by a few individuals who can post a massive number of false comments. It is relatively easy to apply anti-spam strategies in the above scenarios, as E-commerce websites apply many verification strategies to control user registration, and patterns of fraudulent behavior can be easily recognized.

Fake reviews aim to mislead users who shop online. Though existing anti-spam strategies effectively detect traditional spam activities, evolving spam schemes can successfully overcome conventional testing by buying comments written by genuine users which are sold on specific websites such as User Cloud. These spam activities become a kind of advertising campaign for business owners to maintain their top-ranking position. In this chapter, we propose a new **Collaborative Marketing Hyping Detection** solution, which aims to identify spam comments generated by the Spam Reviewer Cloud and to detect products which adopt an evolving spam strategy for promotion. We propose an unsupervised learning model which combines heterogeneous product review networks in an attempt to discover collective hyping activities. Our experiments validate the existence of the collaborative marketing hyping activities on a real-life e-commercial platform and also demonstrate that our model can effectively and accurately identify these advanced spam activities.

In the past, many approaches have been successfully developed to detect online spam. Li [62] initially analyzed several attributes related to spam behaviour, e.g., content features, sentiment features, product features, and metadata features. A two-view semi-supervised method was exploited to identify spam reviews. Feng [27] defined three types of reviewers (any-time reviewers, multi-time reviewers and single-time reviewers)and statistically made distributional footprints of deceptive reviews by using Neuro-Linguistic Programming (NLP) techniques. In [26], Fei proposed a model to detect spammed products or product groups by comparing the differences in rating behaviors between suspicious and normal users. These models rely on content features that can be easily avoided by inserting special characters. In addition, other features, such as temporal feature or networks information have been employed. Xu [100] collected large-scale real-world datasets from telecommunication service providers and combined temporal and user network information to classify spammers in Short Message Services (SMS). Xie [97] proposed a model that only employs temporal features, with no semantic or rating behavior analysis, for detecting abnormal bursts as the number of reviews increases. Tyler [76] studied a problem of temporal correlations between spam and phishing websites and con-

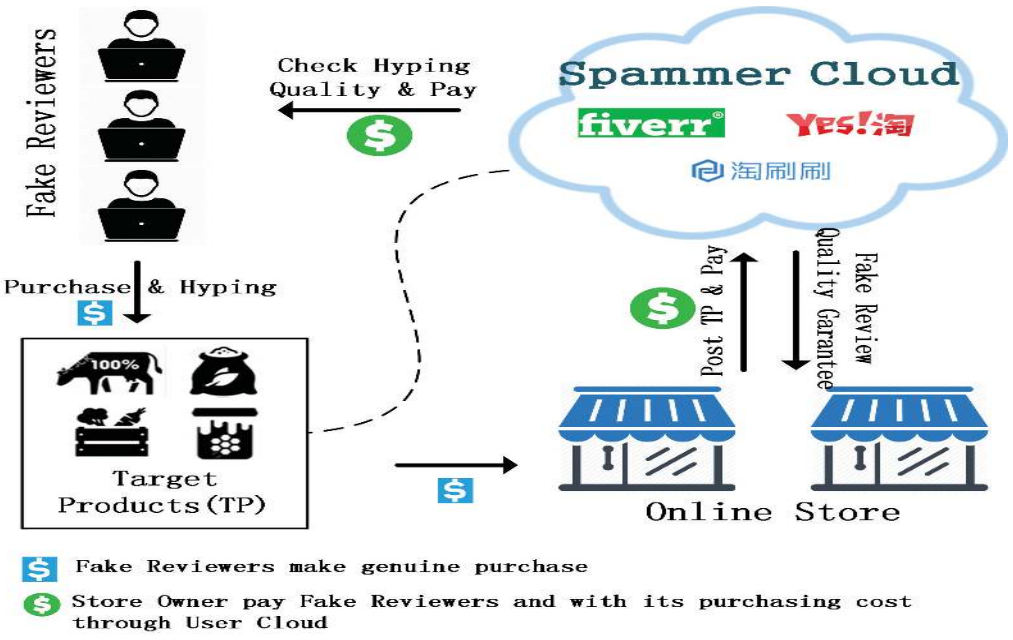

Fake Reviewers
Check Hyping Quality & Pay
Spammer Cloud
fiverr
YES!淘
淘刷刷
Purchase & Hyping
Target Products (TP)
Post TP & Pay
Fake Review Quality Garantee
Online Store
Fake Reviewers make genuine purchase
Store Owner pay Fake Reviewers and with its purchasing cost through User Cloud

Figure 4.1 : Evolving Marketing Hyping Ecosystem

ducted an empirical study. Intuitively, these works can also avoid sophisticated spam strategies.

Amazon has sued more than 1000 'fake' product reviewers who sell their fake reviews on Fiverr.com (one of the most famous spam reviewer clouds) [*]. On these types of user cloud platforms, business owners can purchase anonymous comments generated by real users by paying for them. This makes spam detection very challenging, as the advent of a massive number of apparently genuine fake reviewers (which we refer to as 'genuine fakes' in this thesis) makes the fraud pattern much more nebulous. To date, as shown in Figure 4.1, many third-party platforms have created various fake review markets(user cloud) for online product sellers and fake review providers. In real-world business processes, a massive number of random but genuine fake review providers conduct real transactions[†] and write positive com-

[*] http://money.cnn.com/2015/10/18/technology/amazon-lawsuit-fake-reviews/

[†] many e-commercial website think they can reduce spam review by allowing only real buyer to write reviews

ments to claim a bonus. Existing research ignores the latent connections in product networks, which are difficult to discover, especially when these spam activities have become a hyping and advertising investment which has gained increased popularity among homogeneous competitors online. Thus, anti-spam rules can be easily avoided, which also impairs the efficiency and effectiveness in detection performance.

In this work, we have coined a new solution - **Collaborative Marketing Hyping Detection**, which detects groups of online stores which simultaneously adopt marketing hyping. We discuss several challenges as follows:

- How can heterogeneous product information network be defined to infer their latent collaborative hyping behaviors? The network information may not be directly observed from the original data sets, so we need to build up an relationship matrix between products to represent their underlying correlation.
- What features need to be selected to best solve our problem? Traditional features, i.e., semantic clues or user relations may no longer suitable for discovering fraud due to the rapidly evolving spam strategies. Hence, we need to choose dedicated features according to our specific scenario.
- How can we design a model that will effectively identify the collaborative marketing hyping behavior? A model which can employ the power of heterogeneous product networks to discover collective hyping behaviour needs to be proposed.

To overcome these challenges, we propose an unsupervised shapelet learning model to discover the temporal features of product reviews, and then integrates the heterogeneous product network information as regularization terms to discover products which are subject to collaborative hyping. We define three regularization terms which reflect the underlying correlations between user, product and online store network.

We summarize our contributions in this chapter as follows:

- We coin a new solution called **Collaborative Marketing Hyping Detection**, which uncovers the collective strategy among all online competitors who engage in vicious hyping competition on current E-commerce platforms. To the best of our knowledge, this is an area which has never previously been studied.
- We propose an unsupervised shapelet learning model which utilizes temporal patterns for recognizing collaborative fraud patterns.
- We incorporate three regularization terms into the shapelet learning model, to constrain the matrix factorization in this model. The experiment results demonstrate that our model can identify collaborative hyping products or stores with high accuracy.

4.2 Related work

Spam detection is a hot topic and there are several representative studies in this filed. For instance, Jindal [48, 49] conducted initial research into opinion spam and studied the problem of trustworthiness in review opinions.In[26], Fei proposed a model to detect spammed products or product groups by comparing the differences in rating behaviors between suspicious and normal users. Stein [90] proposed a model of a real-time adversarial learning framework to classify the read and write activity in Facebook. However, no further information was given about the features and policies used, which prevents comparison. Two offline investigations [31, 34] have revealed large-scale spam activities in Twitter and Facebook respectively. As they focus on offline analysis, however, neither of them can be used to directly detect online spam activities. The method in [34] for securing a large number of individuals is time-consuming due to the adoption of a URL blacklist method while [31] utilizes a clustering method which needs to operate on an intact set of messages and hence has apparent limits on clustering efficiency. In addition, many researchers treat spam as anomaly detection according to users' temporal behaviours. For instance, Tyler [76] studied the problem of temporal correlations between spam and phishing websites and conducted an empirical study. The study analyzed the temporal relationship of fast-flux attack spam and the lifetime of phishing websites. Shen et al. addressed the problem of 'link spam' [86] and their proposed model defined temporal information such as In-link Growth Rate and In-link Death Rate in a support vector machine (SVM) model for classification. Xie [97] proposed a model that only employs temporal features, with no semantic or rating behavior analysis, for detecting abnormal bursts as the number of reviews increases. This work analyzed multi-dimensional time series by referring to singleton reviewers (users who had written only one review) for leveraging correlated anomalies. However, these work all have their own limitations in relation to evolving spam strategies, as most of the review content analysis methods may become ineffective in identifying them.

4.3 Methodology

In this section, we first define the problem, then discuss the shapelet learning model and then describe several well designed heterogeneous information networks for regularization. Lastly, we formulate the objective function.

4.3.1 Problem Definition

In 2015, fake product issues on Taobao were exposed on many public and social media platforms. Official investigations conducted by the Chinese Consumer Association(CCA) found that most of the fake products surprisingly maintained a top ranking position, which could continuously damage the interests of customers.

A key factor is that only individuals who successfully purchase a product can leave comments on that product on Taobao. CCA also reported several notorious fake review web markets in China, which formed a fake review industrial chain. On such a platform, for example, a Taobao store owner can post their request, say 1000 reviews at 10 RMB each, as 1000 tasks. Anyone in China who has the time can earn 10 RMB if they know such a web market. Ironically, these platform providers have their own mechanisms for preventing people from spamming the tasks, so they not only guarantee that a person can take only one task posted by a specific store, they can also ensure that the least amount of spam evidence (e.g., semantic clues, user behaviors, etc) is left in the comments. Traditional spam detection rules may thus be subtly escaped by such a spam strategy. Intuitively, it is found that these stores normally purchase fake reviews periodically, as individuals' needs change over time. For instance, by predicting the most active shopping periods for individuals, e.g., festivals, end-of-season, pre-season and so on, online merchants will buy fake reviews months beforehand to hold the top position until people start to make purchases. This forms a collaborative marketing hyping phenomenon among all homogeneous brands and disadvantages honest shop owners.

Mathematically, consider a set of online products $\mathbf{P}$ belonging to a group of stores, for each product $\mathbf{p}$ in $\mathbf{P}$, from which a review time series $\mathbf{t}$ can be obtained. Consider a set of time series $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\}$ that correspond to $\mathbf{P}$. Each time series $\mathbf{t}_i$ $(1 \leq i \leq n)$ contains an ordered set of real values denoted as $(\mathbf{t}_{i(1)}, \mathbf{t}_{i(2)}, \ldots, \mathbf{t}_{i(q_i)})$, where q_i is the length of $\mathbf{t}_i$. We wish to learn a set of top-k most discriminative shapelets $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$. Similar to the shapelet learning model [33], we set the length of the shapelets to expand r different length scales starting at a minimum l_{min}, i.e. $\{l_{min}, 2 \times l_{min}, \ldots, r \times l_{min}\}$. Each length scale $i \times l_{min}$ contains k_i shapelets and $k = \sum_{i=1}^{r} k_i$. Clearly, $\mathbf{S} \in \bigcup_{i=1}^{r} \mathbf{R}^{k_i \times (i \times l_{min})}$ and $r \times l_{min} \ll q_i$ to keep the shapelets compact. Our shapelet learning model uses

Table 4.1 : Symbols and notations

O	A series of target online stores
P	A group of products belonging to O
T	Time series data set generated from P
S	Top-k most discriminative shapelets
X	Shapelet transformation matrix
d_{ij}	Distance between shapelets i and j
l_i	Shapelet length
q_j	Time series length
E	Shapelet similarity matrix
V	Pseudo-class label matrix
U	Classification boundary under V
G_1	Store-based network
G_2	Product-based network
G_3	User Correlation-based network
$R(SBR)$	Store-based regularization
$R(PBR)$	Product-based regularization
$R(UCR)$	User Correlation-based regularization

matrix factorization techniques and thus all products can be classified on the latent spaces according to their temporal features. Additionally, we do not only consider singular spam activities in one store but aim to detect the collaborative hyping behaviors, thus, three different product information networks have been defined as regularization terms to constrain matrix factorization. We summarize the symbols and notations used in this chapter in Table 4.1.

4.3.2 Shapelet Learning Model

Shapelet-transformed Representation

According to Lines's [64] work, *shapelet transformation* was proposed to downsize a time series into a short feature vector in the shapelet feature space. Time series are orderless but they can be uniformly represented by shapelet-transformation which preserves the most relevant information for classification.

For instance, given a set of time series $\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\}$ and a set of shapelets $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$, we use $\mathbf{X} \in \mathbf{R}^{k\times n}$ to denote the shapelet-transformed matrix, where each element $\mathbf{X}_{(\mathbf{s}_i,\mathbf{t}_j)}$ denotes the distance between shapelet $\mathbf{s}_i$ and time series $\mathbf{t}_j$. We use $\mathbf{X}_{(ij)}$ to represent $\mathbf{X}_{(\mathbf{s}_i,\mathbf{t}_j)}$ which can be calculated as in Eq. (4.1),

$$\mathbf{X}_{(ij)} = \min_{g=1,\ldots,\overline{q}} \frac{1}{l_i} \sum_{h=1}^{l_i} (\mathbf{t}_{j(g+h-1)} - \mathbf{s}_{i(h)}), \tag{4.1}$$

where $\overline{q} = q_j - l_i + 1$ is the quantity of segments with length l_i from series $\mathbf{t}_j$, and q_j, l_i are the lengths of time series $\mathbf{t}_j$ and shapelet $\mathbf{s}_i$ respectively.

Given a set of time series data $\mathbf{S}$, $\mathbf{X}_{(ij)}$ is a function that refers to all candidate shapelets $\mathbf{S}$, i.e. $\mathbf{X(S)}_{(ij)}$. Here, we elide the variable $\mathbf{S}$ and use $\mathbf{X}_{(ij)}$ instead.

Based on the work in [33], we approximate the distance function using the *soft minimum function* as in Eq. (2),

$$\mathbf{X}_{(ij)} \approx \frac{\sum_{q=1}^{\overline{q}} d_{ijq} \cdot e^{\alpha d_{ijq}}}{\sum_{q=1}^{\overline{q}} e^{\alpha d_{ijq}}}, \tag{4.2}$$

where $d_{ijq} = \frac{1}{l_i} \sum_{h=1}^{l_i} (\mathbf{t}_{j(q+h-1)} - \mathbf{s}_{i(h)})$, and α controls the precision of the function. The soft minimum approaches the true minimum when $\alpha \to -\infty$. In our experiments, we set $\alpha = -100$.

Pseudo-class label

Our dataset has been human labeled, and to evaluate the accuracy of our model, we introduce *pseudo-class labels* for unsupervised learning. In this chapter, c denotes the number of pseudo classes. The pseudo-class label matrix $\mathbf{V} \in \mathbf{R}^{c\times n}$ contains c labels, where $\mathbf{V}_{(ij)}$ indicates the probability of the j-th time series candidate categorized into the i-th class. If $\mathbf{V}_{(\bar{i}j)} > \mathbf{V}_{(i,j)}, \forall i$, then the time series example $\mathbf{t}_j$ belongs to the cluster $\bar{i}$.

Shapelet Similarity Minimization

To maximize the variance of the shapelets, we penalize the model if similar shapelets are generated. We denote the shapelet similarity matrix as $\mathbf{E} \in \mathbf{R}^{k \times k}$, where each element $\mathbf{E}_{(\mathbf{s}_i,\mathbf{s}_j)}$ represents the similarity between two shapelets $\mathbf{s}_i$ and $\mathbf{s}_j$. $\mathbf{E}_{(ij)}$ represents $\mathbf{E}_{(\mathbf{s}_i,\mathbf{s}_j)}$ as seen in Eq. (4.3),

$$\mathbf{E}_{(ij)} = e^{-\frac{\|d_{ij}\|^2}{\sigma^2}}, \tag{4.3}$$

where d_{ij} is the distance between shapelet $\mathbf{s}_i$ and shapelet $\mathbf{s}_j$. d_{ij} can be calculated by Eq. (4.2).

Shapelet Learning Model

We measure the least squares error between the original shapelet transformation matrix X and the pseudo-class labels by minimizing their distance as follows:

$$\min_{\mathbf{U}} \ \|\mathbf{X} - \mathbf{UV}\|_F^2 \tag{4.4}$$

where $\mathbf{U} \in R^{k \times c}$ is classification boundary corresponding to pseudo-class labels V. Overall, this is a joint optimization problem with respect to variables S , U and V for this model, as in Eq. (5),

$$\min_{\mathbf{S},\mathbf{U},\mathbf{V}} \frac{1}{2}\|\mathbf{X} - \mathbf{UV}\|_F^2 + \frac{\lambda_3}{2}\|\mathbf{H}(\mathbf{s})\|_F^2 + \frac{\lambda_4}{2}\|\mathbf{U}\|_F^2, \tag{4.5}$$

4.3.3 Product Network Regularization

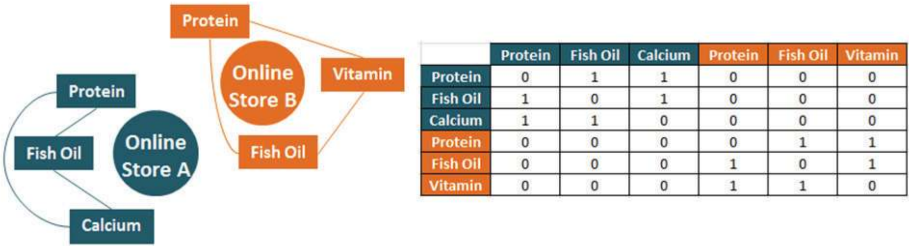
The product network provides correlation information about all the online stores. We model three types of heterogeneous information networks as regularization terms, namely store-based regularization, product-based regularization and user-correlation regularization.

Store-based Regularization

Online sellers normally hype their products periodically to retain their top ranking position. Homogeneous competitors always observe the hyping action of their peers when they prepare to purchase fake reviews. Collaborative marketing hyping is essentially a new type of ranking position competition between homogeneous stores and product owners: for instance, store A, which sells protein powder will start to hype when they find that their competitors have begun to seek spammers. Thus, similar products belonging to different stores may share common hyping behaviors in terms of their temporal features. Based on the above analysis, we design a store-based regularization term $R(SBR)$ to connect all the same products within different stores,

$$R(SBR) = \mathbf{V}\mathbf{G}_1\mathbf{V}^\top. \tag{4.6}$$

where $\mathbf{G}_1$ is a store-based network matrix. As shown in Figure 4.2, we set up the connection values as 1 for every product belonging to the same merchants; otherwise, it is 0. Store-based regularization terms based on the assumption of homogeneous merchandise within different stores share a similar hyping pattern with respect to their time series features. Thus, there is a very large possibility that they will be categorized into the same cluster. However, not all high-ranking products enhance 'reputation' and profit by adopting a spam method. Therefore, we discuss product-



	Protein	Fish Oil	Calcium	Protein	Fish Oil	Vitamin
Protein	0	1	1	0	0	0
Fish Oil	1	0	1	0	0	0
Calcium	1	1	0	0	0	0
Protein	0	0	0	0	1	1
Fish Oil	0	0	0	1	0	1
Vitamin	0	0	0	1	1	0

Figure 4.2 : Store-based Network and Matrix $\mathbf{G}_1$.

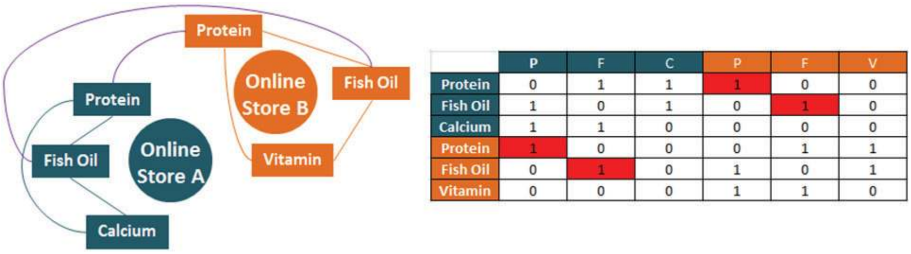
based regularization in the next section.

Product-based Regularization

In contrast to the external comparison in store-based regularization, product-based regularization focuses on the internal comparison of different products within the same store. An online seller who decides to use fake reviews will not only hype a single product in their stores, hence, we introduce the product-based regularization term $R(PBR)$, to indicate a homogeneous competitor's products within different stores,

$$R(PBR) = \mathbf{V}\mathbf{G}_2\mathbf{V}^\top \tag{4.7}$$

where $\mathbf{G}_2$ is the product connection matrix and $G_{ij} = 1$ when these two products i and j are within the same store; otherwise, $G_{ij} = 0$. Intuitively, these merchants



	P	F	C	P	F	V
Protein	0	1	1	1	0	0
Fish Oil	1	0	1	0	1	0
Calcium	1	1	0	0	0	0
Protein	1	0	0	0	1	1
Fish Oil	0	1	0	1	0	1
Vitamin	0	0	0	1	1	0

Figure 4.3 : Product-based Network and Matrix $\mathbf{G}_2$.

will adopt unfair techniques to promote most of their products, rather than only hyping one or two of them. Such products may also be more likely to share similar temporal patterns, hence this step is an ideal supplement to the first regularization model. We describe the product-based network and the matrix in Figure 4.3.

User Correlation-based Regularization

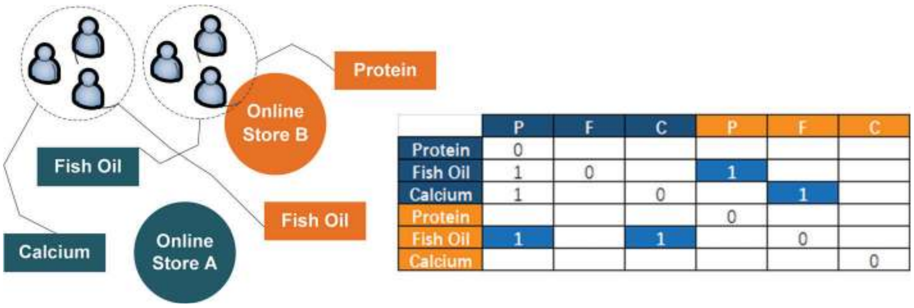
User-correlation regularization provides network information from another perspective. Spammers may accept multiple fake review tasks corresponding to different products at the same time. Thus, during the positive review burst period, these spammers may simultaneously emerge in the review list of hyping-oriented products. Ordinary customers do not normally purchase the same product from different stores at the same time, nor do they buy different products in different stores (to save transportation costs). Hence, we introduce the user correlation-based regularization term $R(UCR)$ to minimize the difference between products reviewed by same user in a specified period as follow:

$$R(UCR) = \mathbf{V}\mathbf{G}_3\mathbf{V}^\top \tag{4.8}$$

where $G_{ij} = 1$ in $\mathbf{G}_3$ when these two products i and j are reviewed by the same users in a nominated period; otherwise, $G_{ij} = 0$. Product network information based on the evidence of spammer groups is also very important for avoiding information loss. We represent the user-correlation product network in Figure 4.4.

4.3.4 Collaborative Hyping Detection Model

We propose our **Collaborative Hyping Detection Model** (CHDM), to solve the **Collective Marketing Hyping** problem defined in Section 4.1. This model integrates all the regularization terms we have defined into a shapelet learning model, which utilizes the temporal features and product network information for clustering.



	P	F	C	P	F	C
Protein	0					
Fish Oil	1	0		1		
Calcium	1		0		1	
Protein				0		
Fish Oil	1		1		0	
Calcium						0

Figure 4.4 : User correlation-based Network and Matrix $\mathbf{G}_3$.

The objective function is given as follows:

$$\begin{aligned}\min_{\mathbf{S},\mathbf{U},\mathbf{V}} \frac{1}{2}\|\mathbf{X}-\mathbf{U}\mathbf{V}\|_F^2 + \frac{\lambda_1}{2}\|\mathbf{V}\mathbf{G}_1\mathbf{V}^\top\| + \frac{\lambda_2}{2}\|\mathbf{V}\mathbf{G}_2\mathbf{V}^\top\| \\ + \frac{\lambda_3}{2}\|\mathbf{V}\mathbf{G}_3\mathbf{V}^\top\| + \frac{\lambda_4}{2}\|\mathbf{H}(\mathbf{s})\|_F^2 + \frac{\lambda_5}{2}\|\mathbf{U}\|_F^2,\end{aligned} \tag{4.9}$$

4.4 Algorithm

Our proposed CHDM model is very straightforward and integrates all the network information generated from store, product and user correlation into a shapelet learning model. The framework of our Collevtive Hyping Detection Model (CHDM) is depicted in Fig. 4.5. Specifically, we employ an unsupervised learning model to cluster the target stores based on their comments in relation to the temporal features. In addition, we incorporate the three different pieces of network information (e.g., store, product and user correlation) as regularization terms to enhance the clustering accuracy.

To achieve a local minimum of the CHDM objective function given by Eq. (4.9), we conduct the coordinate gradient descent to iteratively solve the three variables as in Algorithm 2.

4.5 Experiments

In this section, we validate our method from two key aspects:

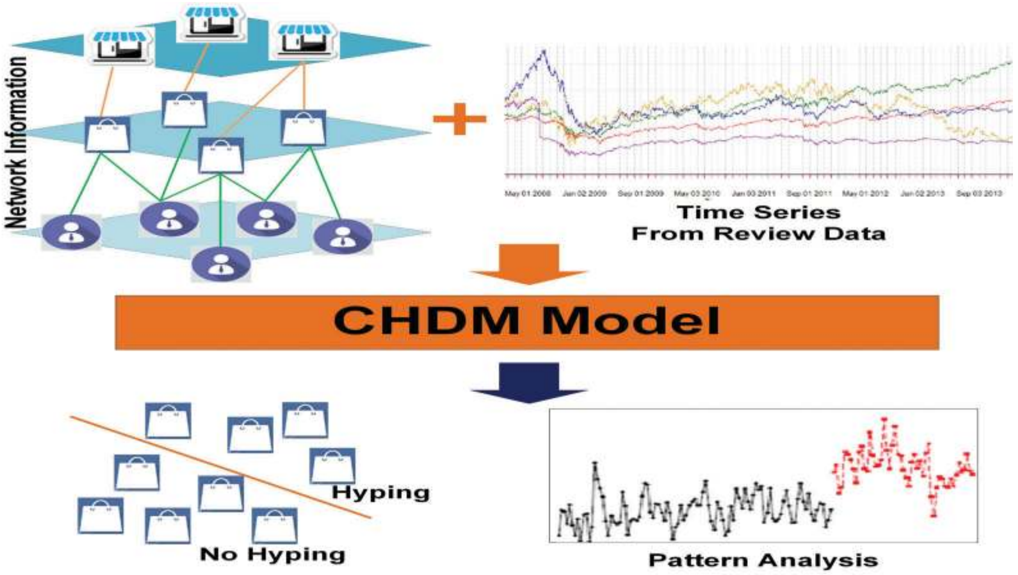


Figure 4.5 : Framework of CHDM model

1. How well does our CHDM model outperform other state-of-the-art spam detection techniques which also utilize temporal features?

2. What is the respective contribution of each of the defined regularization terms (PBR, SBR and UCR) to our proposed model? We first describe the dataset, and then discuss how to build the user correlation-based product information network. We analyze our investigation into the above two questions in the following subsections.

4.5.1 Dataset

The counterfeit crisis on Taobao.com caused a stir in 2015 and the Chinese Consumer Association reported on the top 10 fake goods that were being sold on Taobao, which included clothing, makeup, and digital devices, among others. We therefore collect data from these stores in these nominated industries. It should be noted that our goal is not to detect fake products; however, these reported high ranking products are all very susceptible to hyping. In Table 4.2, we describe the

```
Algorithm 2: CHDM Algorithm
Input  : $T$, review sequential data;
         $c$, number of class;
         $l_{min}$, $k$, length & number of sequential features;
         $i_{max}$, number of internal iterations;
         $\eta$, the learning rate;
         $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$, and $\alpha, \sigma$, parameters
Output: Sequential feature $\mathbf{S}$ and class label $\mathbf{V}$

Initialize $\mathbf{S}_0, \mathbf{V}_0, \mathbf{U}_0$
while Not convergent do
    $1. UpdateV withFixedU andS:$
        $\mathbf{V}_{ij}^{t+1} = \mathbf{V}_{ij}^{t}\sqrt{\frac{(X_t^T U_t)_{ij}}{[(\lambda_1 G_1^T + \lambda_2 G_2^T + \lambda_3 G_3^T + V_t V_t^T) X_t^T U_t]_{ij}}}$
    $2. UpdateU withFixedV andS:$
        $\mathbf{U}_{t+1}^{i} = \left\{ \begin{array}{ll} 0 & otherwise \\ (1 - \frac{\lambda_4}{\|(\mathbf{X}_t \mathbf{V}_t^{-1})^i\|})(\mathbf{X}_t \mathbf{V}_t^{-1})^i & if \ \ \|(\mathbf{X}_t \mathbf{V}_t^{-1})^i\| > \lambda_4 \end{array} \right\}$
    $3. UpdateSwithFixedU andV ::$
        $\mathbf{S}^{t+1} = \mathbf{S}^{t} - \alpha[(\mathbf{X}_s - \mathbf{U}\mathbf{V})\frac{\alpha \mathbf{X}_s}{\alpha \mathbf{S}} + \mathbf{H}_s \frac{\alpha \mathbf{H}_s}{\alpha \mathbf{S}}]$
end

Output $\mathbf{S}^* = \mathbf{S}_{t+1}; \mathbf{U}^* = \mathbf{U}_{t+1}; \mathbf{V}^* = \mathbf{V}_{t+1}$
```

statistics of our real-world dataset.

4.5.2 User Correlation-based Product Network

User name is a key piece of evidence for recognizing users, but the user information in Taobao is anonymized and IDs cannot be acquired, so we can only use an approximate match method to identify spammer users. For instance, a user name on a review page may appear as 'D***d', which indicates that only the initial and last characters in the name were kept. By matching the characters in these two positions, we can at least approximately identify the same users or similar users.

However, the above name evidence is insufficient and may cause noise and inaccuracy, hence we introduce another important piece of evidence - user level. Taobao applies very strict mechanisms in user level upgrades. Only users who have successfully completed a transaction with online shop owners can accumulate the required score to upgrade to a higher level. The higher the user level, the higher the score is needed for an upgrade, thus, additional information in the user matching process decreases the inaccuracy caused by only using user name information.

We build a user correlation-based product network in the following steps:

1. For a pair of randomly selected products, we first observe their review burst period and put all the related reviewers into two separate lists.

2. We conduct a matching process, using reviewers' names as well as their user level and place matched users on a short list

3. If the shortlist contains more than five matched users, we set the connection value of these two products to 1 in the user correlation-based product network matrix which is defined in Section 4.2.3. Otherwise, the value is set to 0.

4. Iteratively, we match all the products in our dataset to build a user correlation-based product network.

Table 4.2 : Dataset Statistics

Dataset	**Store**	**Products**	**Reviews**
Clothing Products	25	186	215892
Cosmetics Products	22	177	225823
Electronic Products	18	159	209654
Food Products	19	165	208639
Healthy Products	24	201	248536
Footwear Products	20	199	190953

Table 4.3 : Comparisons with Benchmark Method.

	Precision	**Recall**	**Accuracy**
MSSD	0.872±0.0086	0.834±0.0107	0.826±0.0093
CoSD	0.938±0.0082	0.903±0.0059	0.898±0.0064
CHDM	**0.966±0.0091**	**0.950±0.0077**	**0.945±0.0085**

4.5.3 Comparison with Existing Models

As part of our experiment, we invite 20 experienced online buyers with high user levels to label the products according to their evaluation. Some of these invited users are domain experts who have previously written fake reviews for various stores.

To validate our proposed CHDM model, we compare it with two representative spam detection techniques which utilize temporal features as follows:

- **MSSD** [97] detects singleton reviewers who appear in an assigned time window as abnormal evidence of spam activities.
- **CoSD** [109] employs a temporal feature classification technique with product related network information for collective spam detection.

Table 4.3 shows that our method significantly outperforms the two baseline techniques, and we make the following observations:

- Our CHDM model can achieve about 5% to 10% higher accuracy than CoSD and MSSD models respectively
- The spam temporal feature is implicit, and the significant drop in MSSD indicates that latent information cannot be fully discovered in the human-assigned time window. We discuss this as a case study in later sections.

4.5.4 The impact of PBR/SBR/UCR in CHDM

We define three types of regularization terms and we evaluate their influence on our model in this section. By setting parameters λ_1, λ_2 or λ_3 in our objective function, regularization term integration can be categorized as follows:

- $CHDM_{NoReg}$ does not consider any regularization terms by setting λ_1, λ_2 and λ_3 to 0.
- $CHDM_S$ considers only store-based regularization terms by setting λ_2 and λ_3 to 0.
- $CHDM_P$ considers only product-based regularization terms by setting λ_1 and λ_3 to 0.
- $CHDM_U$ considers only user correlation-based regularization terms by setting λ_1 and λ_2 to 0.
- $CHDM_{P+S}$ considers store-based and product- based regularization terms by setting λ_3 to 0, which also equal to the CoSD model[109].
- $CHDM_{S+U}$ considers store-based and user correlation-based regularization terms by setting λ_2 to 0.
- $CHDM_{P+U}$ considers product-based and user correlation-based regularization terms by setting λ_1 to 0.
- $CHDM$ considers all the regularization terms by setting a set of best fitting parameters.

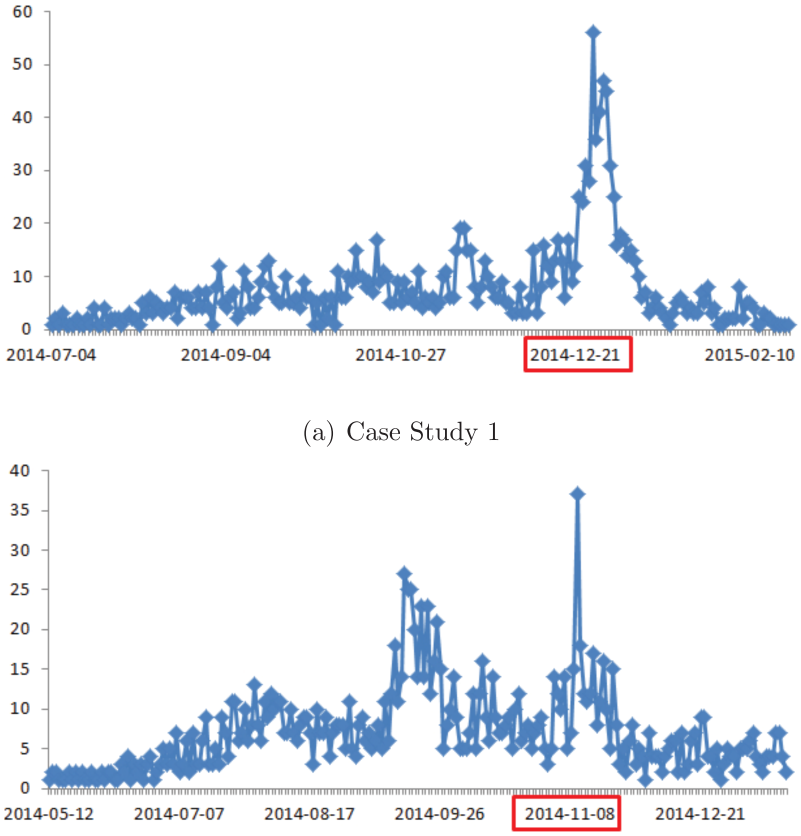
Table 4.4 demonstrates that $CHDM_{NoReg}$ returns the worst result, whereas our CHDM model outperforms all its counterparts. Additionally, by comparing two other singular regularization integrated models - $CHDM_S$ and $CHDM_P$ - we observe that a slightly enhancement in $CHDM_U$. This indicates that user correlation-based information is very important, which can also be observed in the comparison of the dual regularization integration models -$CHDM_{P+S}$, $CHDM_{P+U}$ and $CHDM_{S+U}$. In general, these observations prove the existence of collaborative hyping activities in Taobao and our model successfully takes spam reviewer correlation into account to find the products affected by collective spamming.

4.5.5 Case Study

As previously discussed, the MSSD model [97] identifies spam stores or products one by one through the detection of abnormal singleton reviewers attending in an assigned time window. However, this method misses latent information that underlies evolving hyping activities. We identify two representative cases depicted in Figure 4.6, which were tagged as 'spam' by the MSSD model, but which our CHDM model placed in a 'clean' class. It can be clearly seen that there is a remarkable purchasing burst for both, with 80% of buyers in this time window being singleton reviewers. In our experiment, we define customers who have made less than five transactions

Table 4.4 : Comparisons with Benchmark Method.

	Precision	**Recall**	**Accuracy**
$CHDM_{NoReg}$	0.903±0.0121	0.874±0.0107	0.867±0.0082
$CHDM_S$	0.915±0.0029	0.887±0.0059	0.880±0.0117
$CHDM_P$	0.917±0.0038	0.890±0.0114	0.883±0.0036
$CHDM_U$	0.924±0.0119	0.901±0.0065	0.894±0.0098
$CHDM_{P+S}$	0.938±0.0082	0.903±0.0059	0.898±0.0064
$CHDM_{S+U}$	0.945±0.0067	0.919±0.0018	0.915±0.0049
$CHDM_{P+U}$	0.953±0.0074	0.928±0.0063	0.922±0.0108
CHDM	**0.966±0.0091**	**0.950±0.0077**	**0.945±0.0085**



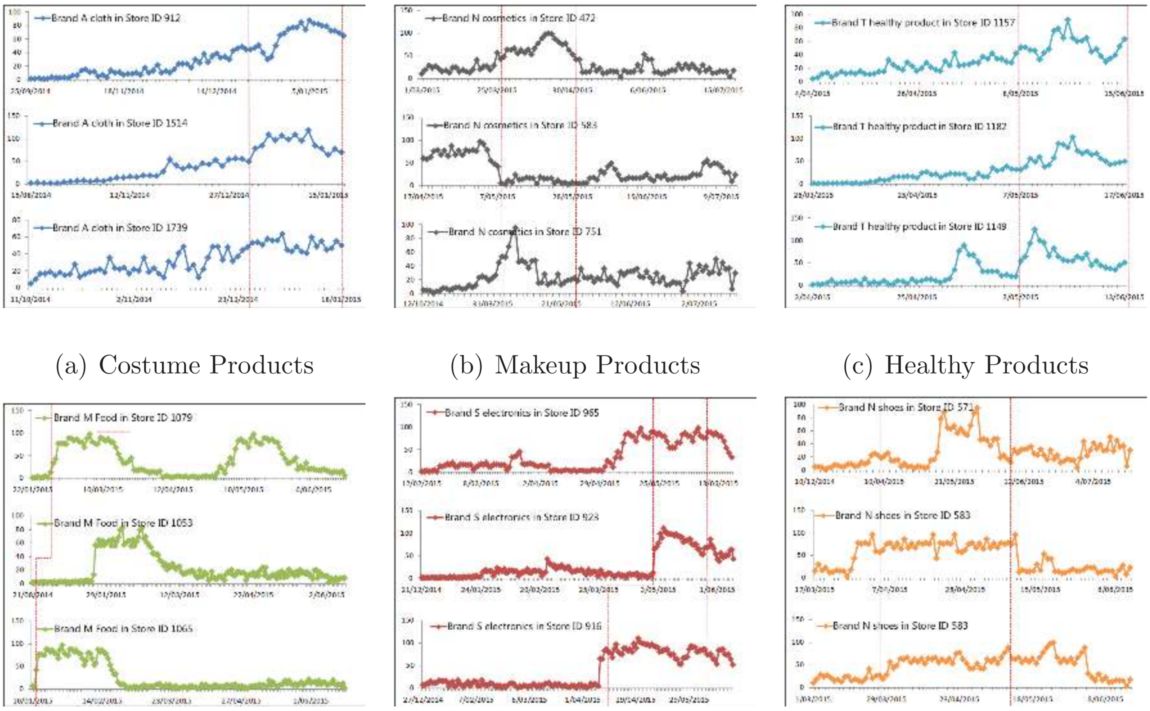
(a) Case Study 1

(b) Case Study 2

Figure 4.6 : Case Study Analysis

online since their registration as singleton reviewers. Because of Taobaos different customer level segmentation strategies and privacy policies, this provides the best match with the definition of *singleton reviewers* for the MSSD model.

For validation purposes, we asked domain experts to recheck these two cases. Most are of the opinion that it is a normal situation as one of the burst periods is close to Christmas while another is close to 'Double 11' (The on-line Boxing Day in China hyped by Taobao).



(a) Costume Products (b) Makeup Products (c) Healthy Products

(d) Provisions Products (e) Digital Products (f) Footwear Products

Figure 4.7 : Collaborative Marketing Hyping Activities.

4.5.6 Collaborative Marketing Hyping Activities

For each industry involved in spam activities exposed by the Chinese Customer Association, we identify several examples to demonstrate their collaborative marketing hyping activities, as depicted in Figure 4.7. For instance, in the health product industry, our proposed CHDM model recognizes similar temporal patterns between May and June of 2015. We can clearly observe a gradually ascending curve in terms of sales volume for three different products, which means that store owners no longer adopt the previous kinds of abrupt spam strategies to escape the detection algorithm applied by Taobao. Rather, they gradually increase the number of hyping purchasers at the beginning of May so that their product is ranked in the top position by Mother's Day which is in the middle of May or June of 2015, or Fathers Day which is in June of 2015, by carefully fitting the Taobao ranking algorithm. A similar situation can be observed in other industries, verifying that our model can successfully identify collaborative marketing hyping activities.

Chapter 5

Dual Social Influence Embedded Recommendation

5.1 Introduction

With the rapidly growing amount of information available on the World Wide Web, it becomes increasingly important to have tools and methods that can help individuals to find relevant information. A variety of recommender systems has been proposed over the years to assist users to efficiently and effectively identify their potential interests. Conventional recommender systems mainly utilize information filtering techniques for recommendation, such as collaborative filtering approaches [40] and content-based filtering approaches [75]. These methods exploit ratings collected from users with similar profiles, or ratings of similar items. For this type of method, however, cold-start and data sparsity are two significant and long-standing problems as specified in [84]. The explosion of social media provides us with an opportunity to use social information to solve these problems and improve the performance of traditional recommender systems. Social correlation theory, such as homophily [73], indicates that two socially linked users can be exploited for recommendation through their common tastes, hence the similarities between socially linked users have been successfully integrated into traditional recommendation systems to improve performance [87, 2, 65, 71, 92]. A large amount of recent research has shown that recommendation approaches based on the simulation of a diffusion process can outperform classical collaborative filtering methods[79, 29] while a heat-spreading process improves recommendation diversity[115]. However, these methods

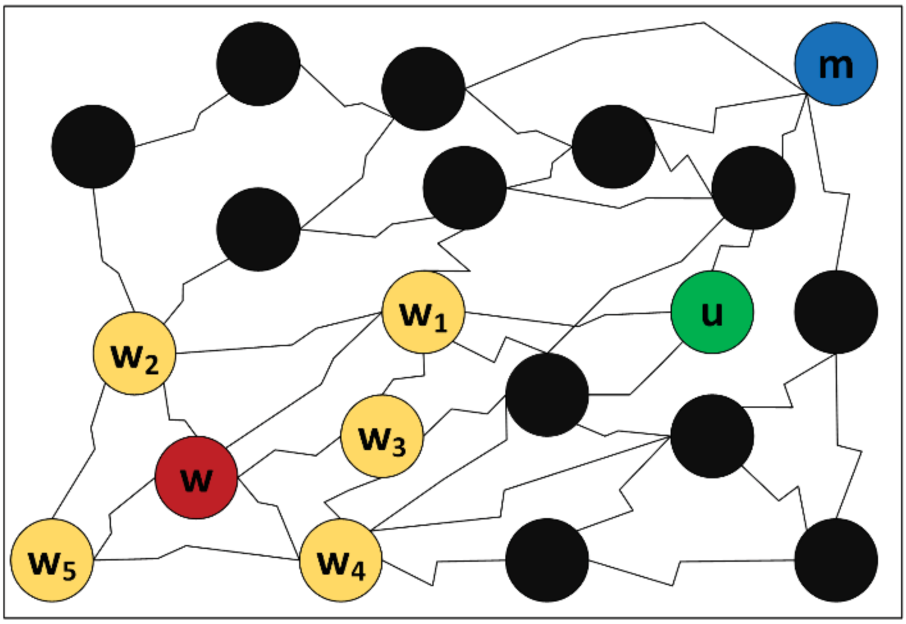


Figure 5.1 : An illustration of social influence. The target node w is mostly influenced by the *Global Influential Node* m and the *Local Influential Node* u, rather than by friends (w_1, w_2, w_3, w_4, and w_5).

neither consider the impact of other user correlation evidence (such as social influence) nor the influence of the diffusion process. Indeed, social influence performs better in user interaction evaluation than similarity, especially in decision making and opinion changing [13, 82], therefore relying only on directly connected relationships (such as friends) to make inference could impair recommendation performance. The potential mutual interaction in implicit user relationships of underlying network structures should be considered for recommendation.

Many researches on business and marketing strategies have found that social influence has long had great commercial value [5] and has been successfully used in marketing and innovation propagation [8, 96]. Such word-of-mouth influence and the role-model effects of social media users are increasingly gaining strength in both online and offline society. For example, a “Web Celebrity”, such as the late Steve Jobs, can exert much greater influence over innovation, new products or digital devices in our society than an unknown reviewer. The recent phenomenon of "Community Stars" has also become widespread. Anyone using social media may now find real life information about, say, books or restaurants, from people who have become "famous" within a certain domain, rather than from their online friends. The social influence-based strategy thus offers a promising enhancement to recommender systems and has outstanding potential to enrich performance. Nevertheless, one significant challenge for an influence-based recommendation approach is how to define and find the most influential nodes in the network and simultaneously incorporate them in the recommendation process. To address this issue, we propose a model named DISR (Dual Influence embedded in Social Recommendation framework) which incorporates the maximum effect of social influence (in both a global and local sense) imposed on users into a matrix factorization process for recommendation.

We first simulate the influence diffusion process to find both global and local influential nodes. We develop two terms, *Global Influential Nodes* and *Local Influential Nodes*. The difference between these two kinds of nodes is that the scope of their influence can be exerted and spread in a specific network. *Global Influential Nodes* indicate which nodes have much more influence in the network than other nodes. This measurement is viewed from a global perspective but it does not evaluate whether such a global node is also the most influential one to every single candidate in the network, especially when the size of the network is large. On the other hand, the philosophy behind the discovery of *Local Influential Nodes* takes a personal (or local) perspective to find those who affect them the most, especially for those nodes situated at the verge of the network, for which only assigning a selection of *Global Influential Nodes* to indicate the most influential people for them is definitely not enough. We illustrate an example as shown in Fig. 5.1 and give formal definitions of these two terms as follows [*]:

*Definition 1 (**Global Influential Nodes**):* Given a network $G = (V, E)$ under a diffusion model and a positive integer $k > 0$, the nodes in the set S^* are called top k **global influential nodes** if and only if S^* is the solution of the discrete optimization problem

$$S^* = \arg \max_{S \subset V, |S| = k} f(S),$$

where $f(S)$ is the influence that S exerts on the entire network.

*Definition 2 (**Local Influential Nodes**):* Given a network $G = (V, E)$ under a diffusion model and a positive integer $k > 0$, for a target node i, the nodes in the set $S^*(i)$ are called top k **local influential nodes** to i if and only if $S^*(i)$ is the solution of the discrete optimization problem

$$S^*(i) = \arg \max_{S \subseteq V, |S| = k} f_{S \to i},$$

where $f_{S \to i}$ represents the influence that the seed nodes in S can exert on node i.

We then integrate this information as regularization term to constrain the matrix factorization model [58] which has been proved to be one of the most effective and accurate methods in recommendation.

The contributions of our work are summarized as follows:

- We systematically analyze the difference between social influence and similarity in terms of the role that each plays in recommendation, decision making and marketing strategies. This observation motivates us to utilize social influence to improve recommendation quality.

- We develop a Global Influential Model (GIM) and Local Influential Model (LIM) to find the global/local influential nodes.

[*] we further discuss these two definition later section

- We incorporate both the GIM and LIM as dual influence embedded regularization terms, to constrain the matrix factorization for social recommendation.

5.2 Related work

Recommender systems are software tools and techniques that provide suggestions for users [85]. Collaborative filtering has been adopted by many recent recommendation models and predicts an active user's preference for an unknown object based on the feedback of peers. Most existing methods can be categorized into neighborhood-based methods and model-based methods. The former can be further divided into user-based methods [40] or item-based methods[63, 84]. Model-based methods alleviate the feedback scarcity issue by leveraging data mining or machine learning methods based on the training data and use the model to predict the active user's preference for the unknown item. Typical models include latent factor models [16], Bayesian models [110] and decision tree [14]. These methods provide the foundational solutions to modern recommendation problems. As real-world rating information may be very sparse, in[83], Singular Value Decomposition (SVD), was exploited to reduce the dimensionality of the sparse rating matrix, which provides the best lower rank approximations of the original matrix. Another strength of matrix factorization is that it can be integrated with the additional constraints. In [28], a distribution framework is built based on collaborative filtering and the k-Nearest Neighbours algorithm with a fast response time in user and item partition. The notion of social recommendation has attracted significant attention in both academia and industry. Due to the rapid evolution of social media platforms (*e.g.* Facebook, Twitter, and WeChat), the relationships between individuals in society have expanded remarkably. Another interesting work in [92] indicates that most social recommendation only considers the local perspective of social content, seldom exploiting the global views of social relations. This work initially integrated global reputation into the proposed model and achieved improved prediction results; however, it mainly utilizes similarity when dealing with relationships from a local perspective. In [43], Hu *et al.* proposed a recommendation framework named MR3, which jointly models users' rating behaviors, social relationships, and review comments. In [66], Liu *et al.* proposed a probabilistic relational matrix factorization (PRMF) model which learns the optimal social dependency between users to improve recommendation accuracy, with or without reference to users' social relationships. Most social recommendation systems only make use of an individual's explicit relationships, such as those who connect directly with them, ignoring any analysis of the influence of the implicit relations between users. We therefore also review work on social influence maximization.

5.3 Methodology

In this section, we first formally defined the problem we are studying after which we discuss the social influence model and then formulate the objective function.

5.3.1 Problem Definition

To illustrate our motivation theoretically, we introduce two principles from the social context of the social environment [10] as follows.

Principle 1 A person who has expertise in the domain has greater social influence and gains more credibility in recommendation than a person who has little or less knowledge in that domain [1].

Principle 2 The activities of participants in a social network can be categorized into different domains[88]. A person can have different preferences in contrasting interaction contexts within alternative domains.

Principle 1 explains why we aim to find the most influential people in the network for recommendation, while Principle 2 inspires us to define influential people in a global ("Web Celebrity") or local ("Community Stars") sense as the former can provide general suggestions while the latter can advise on niche domains.

Table 5.1 summarizes the symbols and notations used in this chapter. From the user-item rating matrix $R = (U, V)$, where U and V are the user vector and item vector respectively, we can build up a user social network $G = (U, E, w_{v \to u})$, where E represents all the edges that explicitly connect a pair of users in U while $w_{v \to u}$ $\in$[0,1] is the influence weight carried on the direct edge or path $v \to u$. For instance, given a pair of nodes u and v in G linked by e_{uv}, the influence weight that v exerts on u can be denoted as $w_{v \to u}$ [†]. Based on the independent cascade model [‡], a widely used model in the information propagation area, we propose a GIM model which retrieves a set of nodes S^* that maximally activates the nodes in the network G, and a LIM model which finds a set of nodes $S^*(i)$ that exerts the largest effect on a specific user i in G according to Principle 2. Based on Principle 1, we integrate S^* and $S^*(i)$, which stand for the influence of global expertise and local expertise respectively, into the a matrix factorization model to enhance recommendation performance. Combining global and local influential nodes can comprehensively model how social influence affects our real-life preferences in different granularity, thus we incorporate dual social influence through GIM and LIM with a collaborative filtering model in recommendation to improve rating prediction accuracy.

In summary, we study the problem of how effectively and efficiently the implicit social influence information improves the accuracy of missing value prediction in the rating matrix.

Table 5.1 : Symbols and notations

R	Rating matrix, users' rating score on items
U	l-rank factors of user matrix
V	l-rank factors of item matrix
G	User-User Graph of U
E	Edges in G
e_{vu}	a specific edge connecting v and u
S^*	global influence maximization set
$S^*(i)$	local influence maximization set
i	a given target node
$N(u)$	the neighbor set of u
X	one possible activation result throughout G
w_v	node weight of v
$w_{v\to u}$	influence weight on e_{vu}
$P(v \to u)$	all possible paths connect two indirectly connected nodes v and u
$\Sigma(\{S^*\})$	the number of nodes activated in one certain activated result
$\varphi(\{S^*\} \to i)$	indicator function evaluates whether i can be activated
k	the node number of the seed set
D	the number of simulations

5.3.2 Social Influence Model

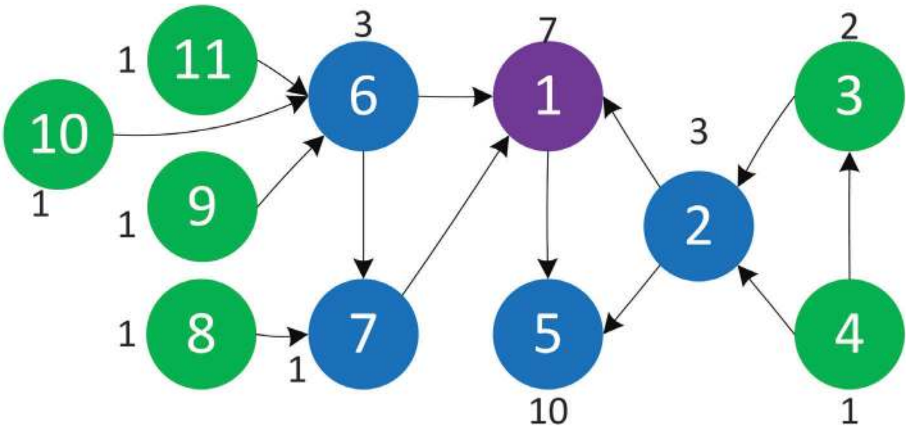
Social Influence Initialization

Node weight represents the node significance as

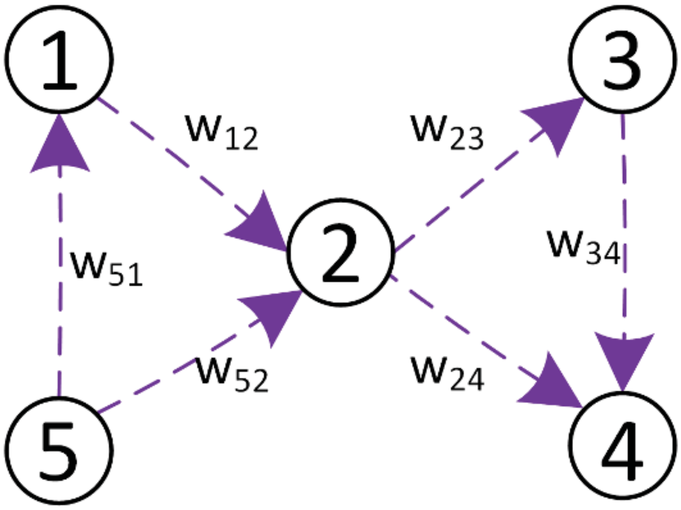
$$w_v = \begin{Bmatrix} \sum_{u \in d_{in}(v)} w_u & \text{if } d_{in}(v) \neq \emptyset \\ 1 & \text{otherwise} \end{Bmatrix}, \tag{5.1}$$

[†] Defined in the subsection of "influence weight" Section 5.3.2

[‡] Defined in the subsection of "independent cascade model" Section 5.3.2



(a) →: Follow Direction



(b)

Figure 5.2 : (a) Node weight, and (b) Influence weight.

where $d_{in}(v)$ is a node set directly linked to user v, that is, a node's social impact is determined by the social followers. As shown in Fig. 5.2(a), the weight of node ⑥ is a weight summation of the followers ⑨, ⑩, and ⑪ (*i.e.,* $w_6 = w_9 + w_{10} + w_{11}$). If a node (*e.g.,* node ⑨) does not have any followers, the weight is 1.

Influence weight indicates the impact degree of two users, and we define the influence weight as $w_{v\to u}$, which represents the influence that user v exerts on user u. To calculate $w_{v\to u}$ in the network, we need to first initialize all *direct link weights* w_{vu} which means that v can reach to u by a direct path. There are many examples in Fig. 5.2(b), such as ⑤ → ②, ⑤ → ①, ⑤ → ② and so on. In the above cases, v has a greater impact on its followers if weight w_v is relatively large, but node u will be affected relatively less if it follows many users. Thus, w_{vu} is in proportion to w_v, but in inverse proportion to the out degree of u, which we denote as $d_{out}(u)$, and the direct link weight can be calculated as

$$w_{vu} = c_1 e^{-\frac{c_2 d_{out}(u)}{w_v}}, \tag{5.2}$$

where c_1, c_2 and ε have been set to 0.0110000512, 0.00211794 and 0.001 respectively, to stabilize w_{vu} across the unit range $[0.01-\varepsilon, 0.01+\varepsilon]$to achieve the best influence estimation across the whole network. According to most of the social influence maximization research work [52, 78, 35], we also set w_{vu} to around 0.01 to achieve the best performance. In addition, the reasons why we assign c_1, c_2 to these two specific numeric values can be seen as follows:

$$c_1 e^{-\frac{c_2 d_{out}(u)}{w_v}} \in [0.01-\varepsilon, 0.01+\varepsilon] \tag{5.3}$$

$$e^{-\frac{c_2 d_{out}(u)}{w_v}} \in \left[\frac{(0.01-\varepsilon)}{c_1}, \frac{(0.01+\varepsilon)}{c_1}\right] \tag{5.4}$$

$$-c_2 \frac{d_{out}(u)}{w_v} \in \left[ln\frac{(0.01-\varepsilon)}{c_1}, ln\frac{(0.01+\varepsilon)}{c_1}\right] \tag{5.5}$$

$$\frac{d_{out}(u)}{w_v} \in \left[-\frac{1}{c_2}ln\frac{(0.01-\varepsilon)}{c_1}, -\frac{1}{c_2}ln\frac{(0.01+\varepsilon)}{c_1}\right] \tag{5.6}$$

where $\frac{d_{out}(u)}{w_v} \in [lower, upper]$, then we can have:

$$\frac{d_{out}(u)}{w_v} \in \left\{ \begin{array}{l} lower = -\frac{1}{c_2}ln\frac{(0.01+\varepsilon)}{c_1} \\ upper = -\frac{1}{c_2}ln\frac{(0.01-\varepsilon)}{c_1} \end{array} \right\}, \tag{5.7}$$

where ε is a very small positive numeric value which we set to 0.001 to keep w_{vu} around 0.01. The lower and upper values depend on our data network structure, according to which we can obtain a minimum and maximum value of $\frac{d_{out}(u)}{w_v}$. Though different datasets have a different value range of $\frac{d_{out}(u)}{w_v}$, our two datasets [§] are very similar in terms of this value range. The lowest $\frac{d_{out}(u)}{w_v}$ in the Douban dataset comes from a node with $d_{out}(u) = 54$ and $w_v = 24298$ while the lowest $\frac{d_{out}(u)}{w_v}$ in the Mafengwo dataset comes from a node with $d_{out}(u) = 38$ and $w_v = 17073$. In addition, the largest $\frac{d_{out}(u)}{w_v}$ in the Douban dataset comes from a node with $d_{out}(u) = 379$ and $w_v = 4$ while the largest $\frac{d_{out}(u)}{w_v}$ in the Mafengwo dataset comes from a node with $d_{out}(u) = 271$ and $w_v = 3$. We therefore uniformly assign the maximum and minimum value of $\frac{d_{out}(u)}{w_v}$ in both datasets to $[lower, upper] = [0.0022, 94.75]$.

Based on the direct link weights w_{vu}, we can then calculate influence weight $w_{v\to u}$ case by case according to the *Inclusion Exclusion Principle*[107] as,

$$w_{v\to u} = \sum\nolimits_{P_i \in P(v\to u)} w_{P_i} - \sum\nolimits_{\cap P(v\to u)} w_{\cap P(v\to u)}, \tag{5.8}$$

where $P(v \to u)$ consists of all possible paths from v to u, with P_i denoting the ith path while $\sum_{\cap P(v\to u)} w_{\cap P(v\to u)}$ removes the overlapping probability of all paths in $P(v \to u)$.

We explain the above equation using a simple case as illustrated in Fig. 5.2(b), ⑤ can reach ② either by $P_1 = P(⑤ \to ① \to ②)$ or by $P_2 = P(⑤ \to ②)$. Thus,

[§] we will introduce our dataset in later section.

according to Eq. (5.8), $w_{5\to 2}$ can be calculated as follows,

$$\begin{aligned} w_{5\to 2} =& w_{P_1} + w_{P_2} - w_{P_1 \cap P_2} \\ =& w_{51}w_{12} + w_{52} - w_{51}w_{12}w_{52}, \end{aligned} \tag{5.9}$$

where w_{51}, w_{12}, w_{52} are all initialized direct link weights, which can be computed by Eq. (5.2).

In summary, to calculate $w_{v\to u}$, direct link weights w_{vu} need to be initialized first by Eq. (5.2), after which we follow Eq. (5.8) to calculate the influence weight. Note that, in some cases, influence weight $w_{v\to u}$ is equal to direct link weight w_{vu} when only one path to connect u and v exists.

Independent Cascade Model

The independent cascade model is the most basic model in the information propagation area. For a given social network graph $G = (U, E)$, where U and E denote vertices and links respectively, each node $u \in U$ is situated as having either inactive or active status. The status can change from inactive to active, but not vice versa. It is initiated by a seed node set $S \in U$ when a node u is activated for the first time at step t_0 and the influence flows independently to u's current inactivated neighbours $N(u) = v_1, v_2, ...$ There is an activation probability along the link (u, v_i) and only one opportunity for u to activate its neighbour v_i at step t_1. If it is unsuccessful, no further activation attempts on v_i are allowed. The process will remain quiescent until there is no further possibility of activating more nodes in U and the activation result is X in this round.

Local Influence Model

The Local Influence Model (LIM) aims to find the set of nodes $S^*(i)$ that most influence the target node i. The optimization function for LIM is defined as:

$$S^*(i) = \underset{S\subseteq V, \|S\|=k}{\arg\min}\; w_{S\to i}, \tag{5.10}$$

where $w_{S\to i}$ represents the influence that the seed nodes in S can have on node i. Given a target node $i = \textcircled{2}$ with the seed set size $k = 2$, there are 6 probabilities for generating the node seed set S as shown in Fig. 5.2(b), i.e., $S(\textcircled{2}) = \{\textcircled{1}, \textcircled{3}\}$, $S(\textcircled{2}) = \{\textcircled{1}, \textcircled{4}\}$, $S(\textcircled{2}) = \{\textcircled{1}, \textcircled{5}\}$, $S(\textcircled{2}) = \{\textcircled{3}, \textcircled{4}\}$, $S(\textcircled{2}) = \{\textcircled{3}, \textcircled{5}\}$, $S(\textcircled{2}) = \{\textcircled{4}, \textcircled{5}\}$. For $S(\textcircled{2}) = \{\textcircled{1}, \textcircled{5}\}$, $\textcircled{2}$ can be influenced by either $\textcircled{1}$ or $\textcircled{5}$. $w_{S(\textcircled{2})\to\textcircled{2}}$ is calculated as:

$$w_{S(\textcircled{2})\to\textcircled{2}} = w_{\{\textcircled{1},\textcircled{5}\}\to\textcircled{2}} = w_{12} + w_{52} - w_{12}w_{52}. \tag{5.11}$$

There are two challenges in solving the objective function when handling large network structures. The problem of computing $S^*(i) = \arg\min_{S\subseteq V, |S|=k} w_{S\to i}$ is NP-hard while the computation of $w_{S(i)\to i}$ is a #P-hard problem, which has been proved in [35]. It is impractical to solve this NP-hard problem using a brute-force method, especially when the social network is large. However, the optimization function in Eq. (5.10) has a sub-modular property [52, 78, 35], which can be employed to develop a $1 - 1e(\approx 63\%)$ approximate algorithm, such as the greedy algorithm. As local influence diffusion is #P-hard, we adopt the Monte-Carlo method to simulate the influence cascade and approximate the calculation of $w_{S(i)\to i}$.

Now we introduce the Local Greedy Algorithm (LGA) in Algorithm 3. This algorithm starts with an empty seed set and repeatedly adds a node that provides the maximum marginal increase into the set, until k nodes have been obtained. In Algorithm 3, the input is a graph G, a target node i and a positive number k which is the expected size of the selected nodes in seed set $S^*(i)$. j represents the current simulation round of the Monte-Carlo process. LGA first adds one node as the seed in each round, so that this node can maximize the marginal influence weight on i, as well as maximize the influence diffusion on i with the current seed set. Here,

Algorithm 3: Local Greedy Algorithm (LGA)

```
Input  : $k$, a positive number; $i$, a target node ;
         $G$, a graph; $D$, the number of simulations.
Output: $S^*(i)$, the top-k influential nodes for $i$

$S^*(i) \longleftarrow \phi$
while $|S^*(i)| < k$ do
    for each node $v \in V \setminus S^*(i)$ do
        $w_{S^*(i) \cup \{v\} \to i} = 0$
        for $j = 1$ to $D$ do
            $w_{S^*(i) \cup \{v\} \to i} + = \varphi(S^*(i) \cup \{v\} \to i)$
        end
        $w(S^*(i) \cup \{v\} \to i) / = D$
    end
    $S^*(i) = S^*(i) \bigcup \arg\min_{v \in V \setminus S^*(i)} w_{S(i) \cup \{v\} \to i}$
end
Output $S^*(i)$
```

$\varphi(I_l(i) \cup \{v\} \to i)$ is an indicator which will be 1 if target node i can be activated by one of the nodes in $I_l(i) \cup \{v\}$ under the activation results in the current Monte-Carlo process. After D times of repeated simulation of the Monte-Carlo process, the most influential node v for target node i can be found. By continuing to add new nodes in node set V until its size reaches k, a set of nodes is obtained which can maximally affect i, represented by $S^*(i)$ as the output.

Global Influence Model

The purpose of the Global Influence Model (GIM) is to find the seed set S^* which extends maximum influence across the entire network,

$$S^* = \underset{S \subseteq V, |S|=k}{\arg\min}\, \sigma(S), \tag{5.12}$$

where $\sigma(S)$ is the expected influence spread by the seed set S and is defined as,

$$\sigma(S) = E[I(S)] = \sum\nolimits_j jp(I(S) = j), \tag{5.13}$$

where $I(S)$ is a variable that represents the number of nodes influenced by the current seed set S. When addressing the global influence maximization problem, we can use the above function as it is not limited to network topology structure and is suitable for any network structure.

Here, we take a simple example to make Eq. 5.13 clearer. Given $S = \{\textcircled{2}\}$ in Fig. 5.2(b), $I(S)$ can be 1, 2 or 3, as ② can influence one or more nodes in $\{\textcircled{2}, \textcircled{3}, \textcircled{4}\}$. In the case of $I(S) = 1$, then ② only influences itself, and the influence probability is

$$p(I(S) = 1) = (1 - w_{23})(1 - w_{24}). \tag{5.14}$$

In the case of $I(S) = 2$, ② influences $\{\textcircled{2}, \textcircled{3}\}$ or $\{\textcircled{2}, \textcircled{4}\}$, giving

$$\begin{aligned} p(I(S) = 2) =& p\{\textcircled{2}, \textcircled{3}\} + p\{\textcircled{2}, \textcircled{4}\} \\ =& w_{23}(1 - w_{24})(1 - w_{34}) + w_{24}(1 - w_{23}). \end{aligned} \tag{5.15}$$

In the case of $I(S) = 3$, which means that $\{\textcircled{2}, \textcircled{3}, \textcircled{4}\}$ are all influenced by ②,

$$\begin{aligned} p(I(S) = 3) = p\{\textcircled{2}, \textcircled{3}, \textcircled{4}\} &= w_{23} w_{\{\textcircled{2}, \textcircled{3}\} \to \textcircled{4}} \\ &= w_{23}(w_{24} + w_{34} - w_{24} w_{34}) \end{aligned} \tag{5.16}$$

Hence, $\sigma(2) = p(I(S) = 1) + 2 * p(I(S) = 2) + 3 * p(I(S) = 3)$.

Similar to the LIM problem, it has been proved that obtaining S^* in Eq. (5.12) is an NP-hard problem while the computation of $\sigma(S)$ in Eq. (5.13) is #P-hard

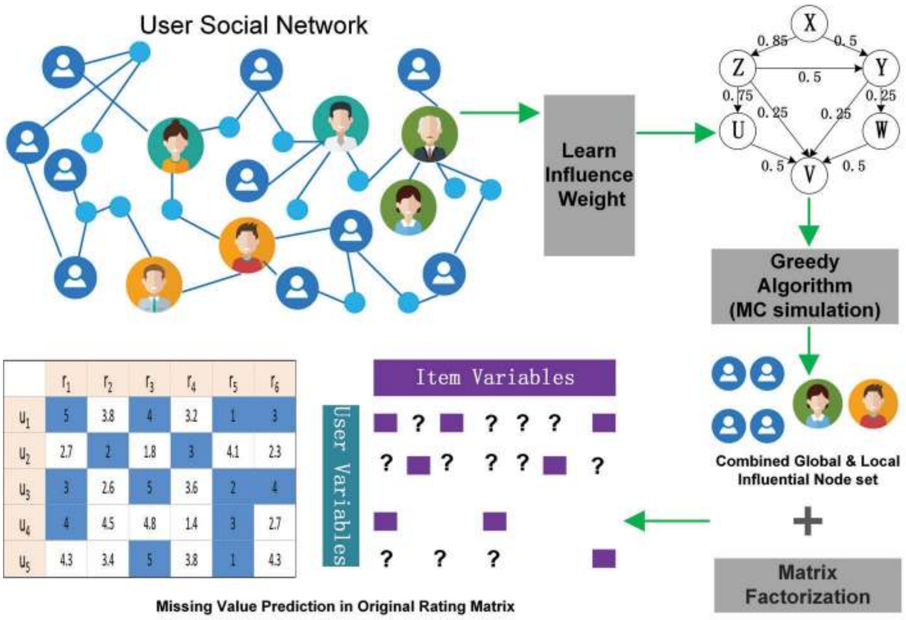
```
Algorithm 4: Global Greedy Algorithm (GGA)
Input  : $k$, a positive number; $G$, a graph;
         $D$, the number of simulations.
Output:  $S^*$, top-k global influential nodes

$S^* \longleftarrow \phi$
while $|S^*| < k$ do
    for each node $v \in V \setminus S^*$ do
        $\sigma(S^* \cup v) = 0$
        for $j = 1$ to $D$ do
            $\sigma(S^* \cup \{v\}) + = \Sigma(S^* \cup \{v\})$
        end
        $\sigma(S^* \cup \{v\}) / = D$
    end
    $S^* = S^* \bigcup \arg \max_{v \in V \setminus S^*} \sigma(S^* \cup \{v\})$
end
Output $S^*$
```

problem[52, 78], especially when the network becomes large and complicated. Thus, we also use Monte-Carlo simulation and further approximate the optimal result by employing the Global Greedy Algorithm (GGA) in Algorithm 4.

The objective function (Eq. (5.12)) also has the sub-modular property [52, 78] and we still exploit the greedy algorithm as an approximate algorithm. The Monte-Carlo method has also been utilized to solve the #P-hard problem of calculating $\sigma(S)$.

Algorithm 4, starts with an empty seed set and repeatedly adds a node that



	r_1	r_2	r_3	r_4	r_5	r_6
u_1	5	3.8	4	3.2	1	3
u_2	2.7	2	1.8	3	4.1	2.3
u_3	3	2.6	5	3.6	2	4
u_4	4	4.5	4.8	1.4	3	2.7
u_5	4.3	3.4	5	3.8	1	4.3

Figure 5.3 : DISR framework We utilize the implicit social influence information from our GIM and LIM models, embedded in the matrix factorization process, to improve the accuracy of the missing value prediction in the original user-item rating matrix.

provides the maximum marginal increase to the entire network, until k nodes have been obtained. In GGA, the input is a graph G and a positive number k, which is the expected size of the selected nodes in the seed set S^*. j represents the current simulation round of the Monte-Carlo process. GGA first adds one node as the seed in each round, and this node maximizes the marginal influence weight on the entire social network together with the current seed set. $\sigma(S^* \cup \{v\})$ calculates the number of activated nodes by $S^* \cup \{v\}$ under the current activation result. The whole process will stop when k satisfies the presetting and outputs a node set S^* which maximizes the spread of influence across the whole network.

5.3.3 Dual Influence Embedded Social Recommendation Model

We formulate the objective function by embedding our proposed model with a classical low-rank matrix factorization, shown as follows,

$$\begin{aligned} \min F(R,U,V) = &\frac{1}{2}\sum\nolimits_{i=1}^{m}\sum\nolimits_{j=1}^{n} I_{ij}(R_{ij}-{U_i}^{t}V_j)^2 \\ &+\frac{\beta}{2}\sum\nolimits_{i=1}^{m}\sum\nolimits_{k\in S^*\cup S^*(i)} w_{k\to i}\|U_i-U_k\|_F^2 \\ &+\frac{\lambda_1}{2}\|U\|_F^2+\frac{\lambda_2}{2}\|V\|_F^2, \end{aligned} \tag{5.17}$$

where V_j represents the set of items rated by user U_i and I_{ij} is an indicator function (*i.e.,* $I_{ij}=1$ *if user* u_i *rates item* v_j *and 0 otherwise*). The overall top-k influential nodes $k \in S^* \cup S^*(i)$ are discovered by combining the results of GIM and LIM. $w_{k\to i}$ indicates the influence weight that node u_k exerts on node u_i, whether the pair of nodes connect directly or indirectly. Parameter β aims to balance the mutual effect of the dual social influence embedded model (the second term) and the collaborative filtering model used by many recommender systems (the first term) while λ_1 and λ_2 prevent the problem of overfitting. We integrate the implicit social influence information model as a regularization term to constrain the matrix factorization and achieve a more accurate prediction in R. U_k is a fixed node set to the target nodes in the network and our objective function is similar to two related studies in [71] and [92] which have been proved can converge. Accordingly, a local minimum of the objective function given by Eq. (5.17) can be calculated by performing gradient descent with respect to the feature vector U_i and V_j in Eq. (5.18) and Eq. (5.19) respectively,

$$\begin{aligned} \frac{\partial F}{\partial U_i} = &\sum\nolimits_{j=1}^{n} I_{ij}({U_i}^{T}V_j-R_{ij})V_j+\lambda_1 U_i \\ &+\beta\sum\nolimits_{k\in S^*\cup S^*(i)} w_{k\to i}(U_i-U_k) \\ &+\beta\sum\nolimits_{j\neq i} I(i\in S^*\cup S^*(i))w_{k\to i}(U_i-U_j) \end{aligned} \tag{5.18}$$

$$\frac{\partial F}{\partial V_j} = \sum\nolimits_{i=1}^{m} I_{ij}({U_i}^{T}V_j-R_{ij})U_i+\lambda_2 V_j \tag{5.19}$$

5.4 Algorithm

The framework of our proposed dual influence embedded social recommendation model (DISR) is depicted in Fig. 5.3. After learning the influence weights in the network, we conduct our greedy algorithms in the Monte-Carlo simulation to find the global and local influential nodes, which are then embedded as the regularization term to constrain the matrix factorization. We incorporate the social influence in a user-user network with matrix factorization techniques to enhance the prediction performance in the user-rating matrix. In this section, we first give the LGA and GGA, which respectively correspond to the LIM (detailed in Section 5.3.2) and GIM (detailed in Section 5.3.2) model. We then discuss our DISR algorithm with its optimizing process in detail.

Algorithms 3 and Algorithms 4 find the local influential nodes and global influential nodes respectively, and they both adopt an independent cascade model [53] and apply the Monte-Carlo simulation to compute $w(S^*(i) \to i)$ and $\sigma(S)$ in Eq. (5.10) and Eq. (5.12) respectively. The complexity of LGA and GGA is the same. In one random simulation, the local influence spread calculation from each node in G takes $O(m)$ time. They both estimate the random diffusion process and sample the resulting active sets with D repeated simulations. Thus, the time complexity of selecting one seed from each of them is $O(nDm)$, where n and m are the number of nodes and edges respectively in G. For k seed nodes in U , the time complexity is $O(knDm)$.

Algorithm 5 initially obtains $S^*(i)$ and S^* via Algorithms 3 and Algorithm 4 respectively. To optimize the overall DISR objective function in Eq. (5.17), we conduct gradient descent with respect to the feature vector U_i and V_j according to Eq. (5.18) and Eq. (5.19), respectively.

Algorithm 5: DISR Algorithm

```
Input  : $k$, a positive number; $i$, a target node;
         $G$, a graph; $D$, the number of simulations;
         $R$, rating matrix; $\lambda$, $\lambda_1$, $\lambda_2$, $\beta$, parameters.
Output: $U$,$V$

$Initialize\ \ U_0, V_0$
$S^*(i) \longleftarrow LGA(k, i, G, D);$        //Algorithm 3
$S^* \longleftarrow GGA(k, G, D);$             //Algorithm 4
while Not convergent do
    1.$Update\ U\ with\ fixed\ V,\ S:$
        ${U_i}^{t+1} = {U_i}^{t} - \lambda * \frac{\partial F}{\partial U_i}\ in\ Eq.$ (5.18) with $S^* \cup S^*(i)$
    2.$Update\ V\ with\ fixed\ U,\ S:$
        ${V_j}^{t+1} = {V_j}^{t} - \lambda * \frac{\partial F}{\partial V_j}\ in\ Eq.$ (5.19) with $R$
end
$Output\ \ U^* = U_{t+1}, V^* = V_{t+1}$
```

5.5 Experiments

In this section, we validate our method from three aspects:

1. How significantly does our DISR model outperform other state-of-the-art recommendation systems?

2. What is the contribution of both GIM and LIM respectively to our proposed model?

3. What is the difference between improving the quality of Top N recommendations and enhancing the accuracy of the rating prediction? We first describe the experimental settings and then analyze our investigation into these three

questions in the following subsections.

5.5.1 Experimental Settings

Benchmark data. Two real world datasets are collected from Mafengwo and Douban. The social network information on these websites is relatively sparse since these are not social media platforms like Weibo or Facebook. To alleviate this problem, we pre-process these two datasets as follows:

- Choose 10-15 users who have more than 500 followers as the seed set on the corresponding website (Mafengwo or Douban).
- In each iteration, we employ a *Depth First Search* method to collect the followers from the seed set. A new seed user who has more than 500 followers will be added to the set.
- After building the user social network, a *Breadth First Search* method is employed to collect all the items rated by collected users.
- Lastly, we eliminate users whose rated items are less than five to obtain the clean dataset.

In summary, we collect and organize 11,498 unique user ratings on 1,582 travel locations with round 475,900 edges in Mafengwo and 12,563 unique user ratings on 1,700 movies with approximately 188,500 edges in Douban. In each case, we mark at least 40% of user items as the training set to evaluate recommendation performance.

Metrics. Two classic metrics are used to measure recommendation performance - mean absolute error (MAE) and root mean square error (RMSE), which are defined respectively as

$$MAE = \frac{1}{T}\sum\nolimits_{i,j}\left|R_{ij} - \overline{R}_{ij}\right| \tag{5.20}$$

$$RMSE = \sqrt{\frac{1}{T}\sum\nolimits_{i,j}(R_{ij} - \overline{R}_{ij})^2} \tag{5.21}$$

where T is the size of the rating set in the testing dataset and R_{ij} represents the rating score that user i gives to item j, while $\overline{R}_{ij}$ denotes the predicted value by one specific method. A smaller RMSE or MAE value indicates higher accuracy in rating value prediction and even a small improvement in these two metrics can have a notable effect on the quality of Top K recommendation [57].

Parameter Settings. The iteration step-size λ is set to 0.01. λ_1 and λ_2 are set to 0.001. The two important parameters β and k are respectively set to 0.01 and 50.

5.5.2 Time Complexity Comparison

We compare our proposed DISR model with several representative recommendation baselines in terms of their own time complexity, as in 5.2. It is clear that $UserMean$ and $ItemMean$ are the most time efficient models, while $DISR$, $LOCABAL$, SR^{u+-}_{i+-}, $SRPCC$, and NMF are at the same time cost level though relatively larger than either $UserMean$ or $ItemMean$ does. $PRMF$ accordingly has the largest time complexity.

As our aim is to find the best model for accuracy and recommendation quality, time complexity is not the key consideration when selecting the best model. Nevertheless, our model is still very competitive in terms of time consumption. It takes slightly longer than using $LOCABAL$, SR^{u+-}_{i+-}, $SRPCC$, *SocialMF*, but the accuracy is better than that of these four models and it is worth sacrificing this amount of time in running the model. In addition, though NMF, $UserMean$ and $ItemMean$ are significantly faster than our model, their experiment performance is much worse. Lastly,, our model performance is similar to that of $PRMF$, but is much faster. In summary,, our model is very competitive in terms of time efficiency.

Table 5.2 : Time Complexity Comparison

	DISR	**PRMF [66]**	**LOCABAL [92]**	**SR$_{i+-}^{u+-}$ [70]**	**SRPCC [71]**	**SocialMF [45]**	**NMF [58]**	**ItemMean**	**UserMean**
Time Complexity	O(knDm)	O(Kdm^2)	O(knm)	O(knm)	O(knm)	O(N$\bar{r}$K) + N$\bar{t}$K)	O(knm)	O(n)	O(m)
Time Consumed (Seconds)	5507.76	8319.89	5035.23	5255.47	5245.39	2749.88	3576.57	1.44	34.56

5.5.3 Comparison with Existing Models

We compare the proposed DISR model with several representative recommendation techniques as follows:

- **UserMean** uses the mean value of every user's rating score for prediction.
- **ItemMean**, similar to UserMean, uses the mean value of every item to predict the unassigned items.
- **NMF** is a collaborative filtering method for recommendation, where only user-item is considered [58].
- **SocialMF** improves recommendation accuracy by considering the social trust relationship between users. It always uses all social links available in the dataset [45].
- **SRPCC** is a representative method in social-based recommender systems with mainly uses average-based social regularization with a Pearson correlation coefficient as a similarity function [71].
- $\mathbf{SR}_{i+-}^{u+-}$ is a social network recommendation approach which exploits both implicit similar and dissimilar social information (e.g. item and user) in its recommendation model [70].
- **LOCABAL** is an improved version of the social relationship-based recommendation framework which considers the cosine similarity of friendship in a social network and also embeds the global reputation to achieve better social recommendation results [92].
- **PRMF** learns the optimal social dependency between users to improve the recommendation accuracy [66], and outperforms several state-of-the-art methods, including LOCALBAL [92] and MR3 [43].

The experimental results answer Question 1 well. Our proposed DISR significantly outperforms other baseline methods in terms of MAE and RMSE measures, and we make the following observations:

- Our proposed DISR is typically around 5% to 15% more accurate than the social recommender baseline methods, **SRPCC** and **LOCABAL**. This indicates that social influence, one of the best sources of social information, and is very important in building social recommendation systems.

- All the social recommendation systems, including our DISR model, significantly outperform other traditional recommender systems, *e.g.,* **NMF**, **UserMean**, and **ItemMean**. This demonstrates that social information plays a very important role in improving traditional recommendation systems.

5.5.4 The impact of local/global influence in DISR

To answer Question 2, we investigate the superiority of the proposed dual social influence embedding, and also compare the performance of our proposed DISR model with its two singleton versions by independently incorporating the results from GIM and LIM, labeled respectively DISR$_G$ and DISR$_L$.

- DISR$_G$ removes the local social influence by setting $k \in S^*$ in Eq. (5.17).

- DISR$_L$ removes the global social influence by setting $k \in S^*(i)$ in Eq. (5.17).

Table 5.3 : Experimental Results of the compared methods *w.r.t* MAE and RMSE on Mafengwo (MFW) and Douban (DB) datasets.

	UMean		**IMean**		**NMF**		**SocialMF**		**SRPCC**		**SR$_{i+-}^{u+-}$**		**LOCABAL**		**PRMF**		**DISR**	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MFW 80%	0.7012	0.8653	0.6354	0.8123	0.5830	0.7395	0.5710	0.7189	0.5619	0.6993	0.5575	0.6963	0.5429	0.6783	0.5386	0.6699	**0.5129**	**0.6503**
MFW 60%	0.7088	0.8701	0.6385	0.8160	0.5880	0.7420	0.5758	0.7297	0.5679	0.7045	0.5634	0.7012	0.5502	0.6867	0.5446	0.6794	**0.5234**	**0.6621**
MFW 40%	0.7195	0.8770	0.6401	0.8213	0.5936	0.7479	0.5799	0.7371	0.5752	0.7098	0.5718	0.7057	0.5584	0.6907	0.5529	0.6849	**0.5297**	**0.6759**
DB 80%	0.7123	0.8732	0.6501	0.8089	0.5938	0.7625	0.5831	0.7479	0.5729	0.7357	0.5674	0.7299	0.5510	0.7198	0.5461	0.7108	**0.5235**	**0.6687**
DB 60%	0.7196	0.8785	0.6583	0.8103	0.6005	0.7698	0.5881	0.7584	0.5792	0.7401	0.5746	0.7359	0.5593	0.7226	0.5527	0.7183	**0.5307**	**0.6913**
DB 40%	0.7221	0.8834	0.6615	0.8153	0.6039	0.7729	0.5941	0.7628	0.5821	0.7436	0.5803	0.7384	0.5654	0.7344	0.5602	0.7286	**0.5509**	**0.7097**

Table 5.4 : Comparisons *w.r.t* single influence models.

	Mafengwo		**Douban**	
	MAE	RMSE	MAE	RMSE
DISR$_G$	0.5498	0.6831	0.5603	0.7195
DISR$_L$	0.5359	0.6781	0.5521	0.7089
DISR	**0.5129**	**0.6503**	**0.5235**	**0.6687**

As can be seen in Table 5.4, DISR$_L$ reduces recommendation error more effectively than DISR$_G$. This is because global influence nodes apply their impact across the whole network, while local influence nodes exert an effect on each user in the network, resulting in more information being received from DISR$_L$. However, neither achieves the low error performance demonstrated by DISR, which embeds both local and global influence information simultaneously.

5.5.5 Top N Recommendation Evaluation

So far, we have evaluated rating prediction accuracy on two real world datasets using RMSE and MAE measures and have investigated the difference in impact of two kinds of social influence. However, when a “best sell list” is generated in a descending order of predictions, the question (*i.e.* Question 3 is whether user experience improvement should be totally reliant on lowering the RMSE or MAE score, or in other words, whether lower prediction error always represents the most relevant recommendation list.

To shed light on this question, we use other evaluation methods to estimate the Top N recommendation performance. Here, we simulate the experiment in Koren’s work [57] and evaluate all the comparison methods in Section 6.3. We randomly split at least 20% of the data for validation in both the Mafengwo dataset, which consists of 2299 user ratings of 316 items, and the Douban dataset, which consists of 2512 user ratings on 340 items. We then choose all the 5-star ratings from the test dataset to represent the most relevant items for that specific user. Overall, our test dataset contains 3895 5-star ratings in Mafengwo and 4273 5-star ratings in Douban.

Our goal is to find the relative ranking position of these "Most Relevant Items" ordered by the predicted rating score of all the items for a specific user. For each 5-star rating item i rated by user u, we select an additional 100 random items from the test dataset. Then, we make rating predictions on i and the other 100 items of user u and sort these 101 items based on the rating score in descending order. Clearly, the best result is that i will antecede the other 100 items. Hence, there are 101 possible rankings for i, ranging from the best case random items (0%) which precede i, to the worst case in which all 100 items (100%) appear before i. In practice, there is little value in ranking a 5-star item beyond the top 20 position in the list, since in most cases, users only pay attention to the top 20 items in a recommendation list. In this experiment, therefore, we only consider cumulative rank distribution between 0% and 20% (the top 20 ranked items out of 100). Since the number 100 is arbitrary, rank positions on the X-axis are in percentiles (0% - 20%), rather than in absolute ranks (0 - 20).

As can be seen in Figure 5.4, our proposed $DISR$ model has 0.405 probability of ranking a 5-star movie before all other 100 randomly picked items (rank=0%) in the Mafengwo dataset, which is four times better than $SocialMF$ to achieve the same and is also about two times better than the performance of the SR_{i+-}^{u+-} model and $SRPCC$ model. The other two methods, $PRMF$ and $LOCABAL$, have a probability of around 0.318 and 0.264 respectively of achieving the same result.

In the Douban dataset (Figure 5.5), $DISR$ has a 0.357 probability of ranking the "Most Relevant Items" in the first position, which is about 3.5 times better than $SocialMF$ and $SRPCC$ can achieve. The other three methods, SR_{i+-}^{u+-}, $LOCABAL$ and $PRMF$, have a probability of 0.163, 0.211 and 0.269 respectively of achieving the same results as $DISR$.

Overall, there are remarkable differences between the performance of the $DISR$

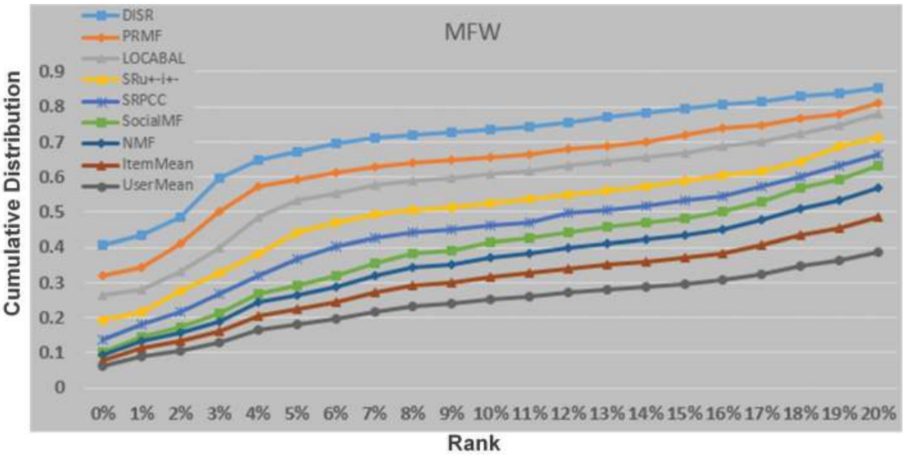


Figure 5.4 : Performance comparison between five methods on a Top K recommendation task on the Mafengwo dataset. The y-axis indicates the probability of the 'Most Relevant Item' being ranked by a specific model compared to other items. The x-axis denotes the percentile of these other items.

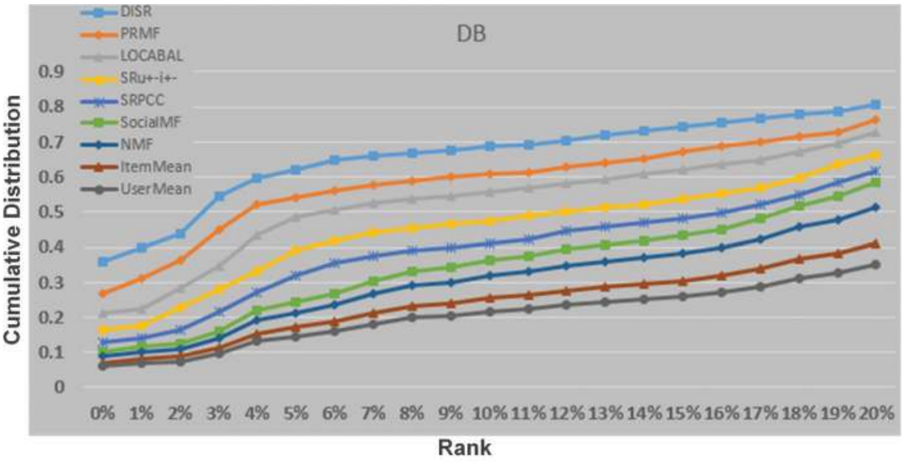
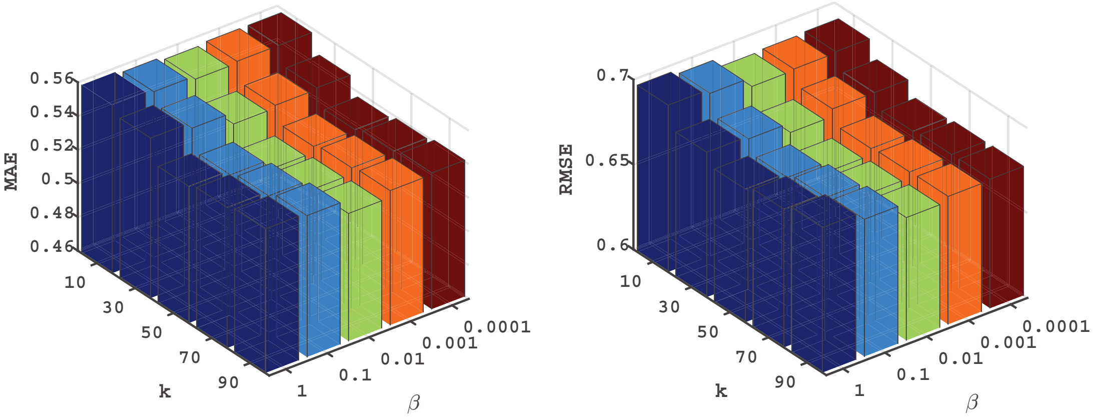


Figure 5.5 : Performance comparison between five methods on a Top K recommendation task on the Douban dataset. The y-axis indicates the probability of the 'Most Relevant Item' being ranked by a specific model compared to other items. The x-axis denotes the percentile of these other items.

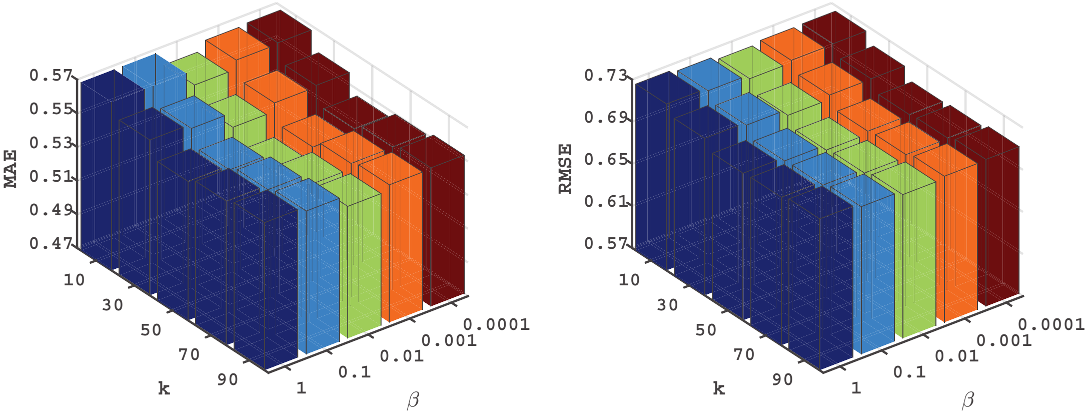
model and the three basic models, $UserMean$, $ItemMean$ and NMF, not only in terms of the RMSE and MAE score, but also in the Top N recommendation



(a)

(b)

Figure 5.6 : Parameter study on Mafengwo data.



(a)

(b)

Figure 5.7 : Parameter study on Douban data.

evaluations. Our method outperforms the other models by having a remarkably higher probability of ranking the “Most Relevant Items” in the top 20 positions and especially in the top 10 positions, though they have relatively smaller distance in terms of RMSE and MAE.

5.5.6 Parameter Analysis

We adopt cross-validation to choose the parameters for our algorithms. The validation data is constructed by 10% of the ratings randomly chosen from the training data. For matrix factorization, we set the dimensionality of the latent space to 10, search the value of β from $\{0.0001, 0.001, 0.01, 0.1, 1\}$ and vary the value of k as $\{10, 30, 50, 70, 90\}$. As shown in Figs. 5.6 and 5.7, the performance continues to rise with the increasing value of β from 0.0001 to 0.01, with the exception of 0.01 to 1. This observation suggests that the best balance for the social influence model and traditional collaborative filtering model is achieved when $\beta = 0.01$. Furthermore, the best performance is achieved with the setting $k = 50$, the middle value in its range. The parameters of other models are similarly set in corresponding works.

Chapter 6

Conclusion and Future Directions

6.1 Conclusion

The ubiquity and the emergence of Web 2.0 provides unique opportunities not only to study but also to design and build more quality recommender system. To achieve this goal, our study follows three perspectives:

1. Our first study investigates travel recommendation which is discussed in Chapter 3. In this work, we combine geo-tagged images and check-ins to discover AOIs which include both well-known tourist attractions and lesser-known local ones. AoIs not only cater the tastes of travellers in general, but also for locals, making travel information more integrated. Moreover, unlike other trip applications, e.g., Every Trail.com or e.g., My Trail.com, our social trajectory recommendation model can dynamically generate a number of AOIs by taking both temporal and spatial features into consideration simultaneously, to meet users' requirements when they are travelling. Our experiment proves that our model can enhance the user experience when it offers recommendations.

2. Another problem we defined and studied is collective marketing hyping, which focuses on discovering fraudulent reviews resulting from evolving spam strategies, e.g., Spam Reviewer Cloud. It is crucial to solve this problem as it presents a significant challenge and creates a crisis of confidence in online business. To resolve this problem, we not only exploit the shapelet learning method to detect the pattern of user comments in terms of their temporal features, we also detect spam activities in a collective way by using the latent heterogeneous network information as three designed regularization terms. The experimental results show that the heterogeneous network information plays

an important role in enhancing classification accuracy. We also validate the existence of collaborative hyping behavior on a real-life dataset.

3. The last problem we studied is how to effectively and efficiently employ social influence to improve recommendation quality. Social influence provides us with a new perspective on enhancing the traditional recommender system performance, but it also presents several challenges. In this work, we investigated how to embed local and global social influence for recommendation. To find the local and global influential nodes, we argued that user preferences and interests can be significantly affected by global or local influential individuals. We modeled GIM and LIM to find these influential nodes and embedded them as dual social influence information to address social recommendation problems. In our proposed model, the social influence information mainly works as a regularization term to constrain the matrix factorization of the recommendation model. Experiments and comparisons on large, real-world datasets show that the proposed dual social influence-based DISR approach significantly outperforms current baselines.

6.2 Vision of the future

Our long-term research direction is to harness large-scale information network to enhance the quality of recommender systems. For each part of our research work in this thesis, there are still several directions which can be explored.

First, we can exploit user relationship information in social trajectory recommendation to increase its user preference awareness. In addition, user preferences may change over time which means AOI ranking also need to be considered such as dynamic evolving information. Thus, how to design a model to rank the AOIs to adapt with time movement is also very interesting and challenging.

Secondly, as sematic information has not been taken into consideration in identifying the collaborative market hyping problem , it will be interesting to combine this type of information into our model. Moreover, we can employ more pieces of information to match users, for example, the user location or their review sentiment analysis. This can help us to detect the spam groups. Besides, with the developing of Artificial Intelligence(AI), many spam platform start to use deep learning method to generate human comments and it will bring very large challenge to existing anti-spam technique, how to recognize the AI-driven spam message will be a very new but valuable research area.

Last but not least, in terms of social influence, as local influence in particular may evolve over time, taking temporal information into consideration in our proposed model will be interesting. Second, negative and distrustful relationships are seldom researched, thus we can also explore negative social information in recommendations.

Bibliography

[1] P. S. Adler, “Market, hierarchy, and trust: The knowledge economy and the future of capitalism,” *Organization Science*, vol. 12, no. 2, pp. 215–234, 2001.

[2] V. Agarwal and K. Bharadwaj, “A collaborative filtering framework for friends recommendation in social networks based on interaction intensity and adaptive user similarity,” *Social Network Analysis and Mining*, vol. 3, no. 3, pp. 359–379, 2013.

[3] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, “Optics: Ordering points to identify the clustering structure,” in *ACM Sigmod Record*, 1999, pp. 49–60.

[4] Y. Arase, X. Xie, T. Hara, and S. Nishio, “Mining people’s trips from large scale geo-tagged photos,” in *Proceedings of the ACM International Conference on Multimedia*, 2010, pp. 133–142.

[5] J. Arndt, *Word of Mouth Advertising: A Review of the Literature.* Advertising Research Foundation, 1967.

[6] S. Arslan Ay, L. Zhang, S. H. Kim, M. He, and R. Zimmermann, “Grvs: A georeferenced video search engine,” in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2009, pp. 977–978.

[7] S. A. Ay, R. Zimmermann, and S. H. Kim, “Relevance ranking in georeferenced video search,” *Multimedia Systems.*

[8] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, “Everyone’s an influencer: Quantifying influence on twitter,” in *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, 2011, pp. 65–74.

[9] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware social influence propagation models," *Knowledge and Information Systems*, vol. 37, no. 3, pp. 555–584, 2013.

[10] E. Barnett and M. Casper., "A definition of 'social environment',," *American Journal of Public Health*, 2001.

[11] A. B. Barragáns-Martínez, E. Costa-Montenegro, J. C. Burguillo, M. Rey-López, F. A. Mikic-Fonte, and A. Peleteiro, "A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition," *Information Sciences*, vol. 180, no. 22, pp. 4290–4311, 2010.

[12] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, and M. Gonçalves, "Detecting spammers and content promoters in online video social networks," in *Proceedings of the international ACM conference on Research and Development in Information Retrieval (SIGIR)*, 2009, pp. 620–627.

[13] P. Bonhard and M. A. Sasse, "Knowing me, knowing you - using profiles and social networking to improve recommender systems," *BT Technology Journal*, vol. 24, no. 3, pp. 84–98, 2006.

[14] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, 1998, pp. Morgan Kaufmann, pages 43–52.

[15] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, pp. 107–117, 1998.

[16] J. Canny, "Collaborative filtering with privacy," in *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, 2002, pp. 45–57.

[17] L. Cao, J. Luo, A. Gallagher, X. Jin, J. Han, and T. S. Huang, "Aworldwide tourism recommendation system based on geotaggedweb photos," in *IEEE In-*

ternational Conference on Acoustics Speech and Signal Processing (ICASSP), 2010, pp. 2274–2277.

[18] X. Cao, G. Cong, and C. S. Jensen, "Mining significant semantic locations from gps data," *Proceedings of the VLDB Endowment*, pp. 1009–1020, 2010.

[19] W. Chen, C. Wang, and Y. Wang, "Scalable influence maximization for prevalent viral marketing in large-scale social networks," in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2010, pp. 1029–1038.

[20] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2009, pp. 199–208.

[21] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2010, pp. 88–97.

[22] A.-J. Cheng, Y.-Y. Chen, Y.-T. Huang, W. H. Hsu, and H.-Y. M. Liao, "Personalized travel recommendation by mining people attributes from community-contributed photos," in *Proceedings of the ACM International Conference on Multimedia*, 2011, pp. 83–92.

[23] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg, "Mapping the world's photos," in *Proceedings of the International Conference on World Wide Web (WWW)*, 2009, pp. 761–770.

[24] G. C. de Silva and K. Aizawa, "Retrieving multimedia travel stories using location data and spatial queries," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2009, pp. 785–788.

[25] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Proceedings of the ACM International Conference on Knowledge Discovery and Data mining*, 1996, pp. 226–231.

[26] G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh, "Exploiting burstiness in reviews for review spammer detection." *International Conference on Web and Social Media (ICWSM)*, pp. 175–184, 2013.

[27] S. Feng, L. Xing, A. Gogar, and Y. Choi, "Distributional footprints of deceptive product reviews." *International Conference on Web and Social Media (ICWSM)*, pp. 98–105, 2012.

[28] V. Formoso, D. Fernández, F. Cacheda, and V. Carneiro, "Distributed architecture for k-nearest neighbors recommender systems," *Proceedings of the ACM International Conference on World Wide Web (WWW)*, vol. 18, no. 4, pp. 997–1017, 2015.

[29] F. Fouss, L. Yen, A. Pirotte, and M. Saerens, "An experimental investigation of graph kernels on a collaborative recommendation task," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2006, pp. 863–868.

[30] M. Gan, "Taffy: Incorporating tag information into a diffusion process for personalized recommendations," *World Wide Web*, pp. 1–23, 2015.

[31] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao, "Detecting and characterizing social spam campaigns," in *Proceedings of the ACM Conference on Internet Measurement (SIGCOMM)*, 2010, pp. 35–47.

[32] Y. Gao, J. Tang, R. Hong, Q. Dai, T.-S. Chua, and R. Jain, "W2go: a travel guidance system by automatic landmark ranking," in *Proceedings of the ACM international conference on Multimedia (ACM MM)*, 2010, pp. 123–132.

[33] J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme, "Learning time-series shapelets," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2014, pp. 392–401.

[34] C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@ spam: the underground on 140 characters or less," in *Proceedings of the ACM Conference on Computer and Communications Security*, 2010, pp. 27–37.

[35] J. Guo, P. Zhang, C. Zhou, Y. Cao, and L. Guo, "Personalized influence maximization on social networks," in *Proceedings of the ACM International Conference on Information & Knowledge Management (CIKM)*, 2013, pp. 199–208.

[36] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman, "Personalized recommendation of social software items based on social relations," in *Proceedings of the ACM International Conference on Recommender systems (RECSYS)*, 2009, pp. 53–60.

[37] Q. Hao, R. Cai, C. Wang, R. Xiao, J.-M. Yang, Y. Pang, and L. Zhang, "Equip tourists with knowledge mined from travelogues," in *Proceedings of the International Conference on World Wide Web (WWW)*, 2010, pp. 401–410.

[38] Q. Hao, R. Cai, J.-M. Yang, R. Xiao, L. Liu, S. Wang, and L. Zhang, "Travelscope: Standing on the shoulders of dedicated travelers," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2009, pp. 1021–1022.

[39] Q. He, F. Zhuang, T. Shang, Z. Shi *et al.*, "Fast time series classification based on infrequent shapelets," in *International Conference on Machine Learning and Applications (ICMLA)*, 2012, pp. 215–219.

[40] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 1999, pp. 230–237.

[41] J. Hills, J. Lines, E. Baranauskas, J. Mapp, and A. Bagnall, "Classification of time series by shapelet transformation," *Data Mining and Knowledge Discovery*, pp. 851–881, 2014.

[42] N. N. Ho-Dac, S. J. Carson, and W. L. Moore, "The effects of positive and negative online customer reviews: Do brand strength and category maturity matter?" *Journal of Marketing*, pp. 37–53, 2013.

[43] G.-N. Hu, X.-Y. Dai, Y. Song, S.-J. Huang, and J.-J. Chen, "A synthetic approach for recommendation: Combining ratings, social relations, and reviews," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[44] Z. Huang, B. Hu, H. Cheng, H. T. Shen, H. Liu, and X. Zhou, "Mining near-duplicate graph for cluster-based reranking of web video search results," *ACM Transactions on Information Systems (TOIS)*.

[45] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the ACM International conference on Recommender Systems (RECSYS)*. ACM, 2010, pp. 135–142.

[46] R. Ji, X. Xie, H. Yao, and W.-Y. Ma, "Mining city landmarks from blogs by graph modeling," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2009, pp. 105–114.

[47] M. Jiang and a. C. F. Peng Cui, "Suspicious behavior detection: Current trends and future directions," *IEEE Intelligent Systems*, vol. 31, no. 1, pp. 31–39, 2016.

[48] N. Jindal and B. Liu, "Review spam detection," in *Proceedings of the International Conference on World Wide Web (WWW)*, 2007, pp. 1189–1190.

[49] ——, "Opinion spam and analysis," in *Proceedings of the International Conference on Web Search and Data Mining (WSDM)*, 2008, pp. 219–230.

[50] N. Jindal, B. Liu, and E.-P. Lim, "Finding unusual review patterns using unexpected rules," in *Proceedings of the ACM international conference on Information and knowledge management (IKM)*, 2010, pp. 1549–1552.

[51] E. Kalogerakis, O. Vesselova, J. Hays, A. A. Efros, and A. Hertzmann, "Image sequence geolocation with human travel priors," in *IEEE 12th International Conference on Computer Vision*, 2009, pp. 253–260.

[52] D. Kemp, J. Kleinber, and E. Tardos, “Maximizing the spread of influence in a social network,” in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2003, pp. 137–146.

[53] D. Kempe, J. Kleinberg, and É. Tardos, “Influential nodes in a diffusion model for social networks,” in *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP)*, 2005, pp. 1127–1138.

[54] L. Kennedy, M. Naaman, S. Ahern, R. Nair, and T. Rattenbury, “How flickr helps us make sense of the world: Context and content in community-contributed media collections,” in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2007, pp. 631–640.

[55] L. S. Kennedy and M. Naaman, “Generating diverse and representative image search results for landmarks,” in *Proceedings of the International Conference on World Wide Web (WWW)*, 2008, pp. 297–306.

[56] M. Kimura and K. Saito, “Tractable models for information diffusion in social networks,” in *Proceedings of the European Conference on Principles of Data Mining and Knowledge Discovery (PKDD)*, 2006, pp. 259–271.

[57] Y. Koren, “Factorization meets the neighborhood: A multifaceted collaborative filtering model,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2008, pp. 426–434.

[58] Y. Koren, R. Bell, C. Volinsky *et al.*, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[59] T. Kurashima, T. Iwata, G. Irie, and K. Fujimura, “Travel route recommendation using geotags in photo sharing sites,” in *Proceedings of the 19th International Conference on Information and Knowledge Management (CIKM)*, 2010, pp. 579–588.

[60] C.-C. Lai, “An empirical study of three machine learning methods for spam filtering,” *Knowledge-Based Systems*, pp. 355–362, 2008.

[61] K. Lee, J. Caverlee, and S. Webb, “Uncovering social spammers: Social honeypots + machine learning,” in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2010, pp. 435–442.

[62] F. Li, M. Huang, Y. Yang, and X. Zhu, “Learning to identify review spam,” in *Proceedings of International Joint Conferences on Artificial Intelligence (IJCAI)*, 2011, pp. 2488–2493.

[63] G. Linden, B. Smith, and J. York, “Amazon. com recommendations: Item-to-item collaborative filtering,” *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[64] J. Lines, L. M. Davis, J. Hills, and A. Bagnall, “A shapelet transform for time series classification,” in *Proceedings of the ACM international conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2012, pp. 289–297.

[65] F. Liu and H. J. Lee, “Use of social network information to enhance collaborative filtering performance,” *Expert Systems with Applications*, vol. 37, no. 7, pp. 4772–4778, 2010.

[66] Y. Liu, P. Zhao, X. Liu, M. Wu, and X.-L. Li, “Learning optimal social dependency for recommendation,” *Information Retrieval Journal*, 2016.

[67] X. Lu, C. Wang, J.-M. Yang, Y. Pang, and L. Zhang, “Photo2trip: Generating travel routes from geo-tagged photos for trip planning,” in *Proceedings of the 18th ACM International Conference on Multimedia (ACM MM)*, 2010, pp. 143–152.

[68] Y. Lu, P. Tsaparas, A. Ntoulas, and L. Polanyi, “Exploiting social context for review quality prediction,” in *Proceedings of the ACM International Conference on World Wide Web (WWW)*, 2010, pp. 691–700.

[69] J. Luo, D. Joshi, J. Yu, and A. Gallagher, “Geotagging in multimedia and computer visiona survey,” *Multimedia Tools and Applications*, pp. 187–211, 2011.

[70] H. Ma, “An experimental study on implicit social recommendation,” in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2013, pp. 73–82.

[71] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, “Recommender systems with social regularization,” in *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, 2011.

[72] B. Markines, C. Cattuto, and F. Menczer, “Social spam detection,” in *Proceedings of the ACM International Workshop on Adversarial Information Retrieval on the Web*, 2009, pp. 41–48.

[73] M. McPherson, L. Smith-Lovin, and J. M. Cook, “Birds of a feather: Homophily in social networks,” *Annual Review of Sociology*, pp. 415–444, 2001.

[74] Q. Mei, D. Cai, D. Zhang, and C. Zhai, “Topic modeling with network regularization,” in *Proceedings of the ACM International Conference on World Wide Web (WWW)*, 2008, pp. 101–110.

[75] R. J. Mooney and L. Roy, “Content-based book recommending using learning for text categorization,” in *Proceedings of the ACM International Conference on Digital libraries (DL)*. ACM, 2000, pp. 195–204.

[76] T. Moore, R. Clayton, and H. Stern, “Temporal correlations between spam and phishing websites.” in *Proceedings of the 2nd USENIX Conference on Large-scale Exploits and Emergent Threats (LEET)*, 2009, pp. 5–5.

[77] A. Mueen, E. Keogh, and N. Young, “Logical-shapelets: An expressive primitive for time series classification,” in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2011, pp. 1154–1162.

[78] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, “An analysis of approximations for maximizing submodular set functions,” *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[79] Z. Obradovic, K. Peng, S. Vucetic, P. Radivojac, and A. K. Dunker, "Exploiting heterogeneous sequence properties improves prediction of protein disorder," *Proteins: Structure, Function, and Bioinformatics*, vol. 61, no. S7, pp. 176–182, 2005.

[80] T. Rakthanmanon and E. Keogh, "Fast shapelets: A scalable algorithm for discovering time series shapelets," in *Proceedings of the 13th SIAM International Conference on Data Mining*, 2013, pp. 668–676.

[81] T. Rattenbury, N. Good, and M. Naaman, "Towards automatic extraction of event and place semantics from flickr tags," in *Proceedings of the International Conference on Special Interest Group on Information Retrieval (ACM SIGIR)*, 2007, pp. 103–110.

[82] M. J. Salganik, P. S. Dodds, and D. J. Watts, "Experimental study of inequality and unpredictability in an artificial cultural market," *Science*, vol. 311, no. 5762, pp. 854–856, 2006.

[83] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender system-a case study," DTIC Document, Tech. Rep., 2000.

[84] ——, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the ACM International Conference on World Wide Web (WWW)*, 2001, pp. 285–295.

[85] B. Shapira, F. Ricci, P. B. Kantor, and L. R. (Eds.), "Recommender systems handbook." 2011.

[86] G. Shen, B. Gao, T.-Y. Liu, G. Feng, S. Song, and H. Li, "Detecting link spam using temporal information," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2006, pp. 1049–1053.

[87] B. Sigurbjörnsson and R. Van Zwol, "Flickr tag recommendation based on collective knowledge," in *Proceedings of the ACM International Conference on World Wide Web (WWW)*, 2008, pp. 327–336.

[88] P. Slovic, “The construction of preference.” *American Psychologist*, vol. 50, no. 5, p. 364, 1995.

[89] J. Song, S. Lee, and J. Kim, “Spam filtering in twitter using sender-receiver relationship,” in *International Workshop on Recent Advances in Intrusion Detection*, 2011, pp. 301–317.

[90] T. Stein, E. Chen, and K. Mangla, “Facebook immune system,” in *Proceedings of the ACM Workshop on Social Network Systems*, 2011, pp. 8–16.

[91] G. Stringhini, C. Kruegel, and G. Vigna, “Detecting spammers on social networks,” in *Proceedings of the 26th Annual Computer Security Applications Conference*, 2010, pp. 1–9.

[92] J. Tang, X. Hu, H. Gao, and H. Liu, “Exploiting local and global social context for recommendation.” in *Proceedings of the ACM International Joint Conference on Artificial Intelligence (IJCAI)*, 2013, pp. 264–269.

[93] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, “Design and evaluation of a real-time url spam filtering service,” in *2011 IEEE Symposium on Security and Privacy*, 2011, pp. 447–462.

[94] Y. Wang, T. Mei, J. Wang, H. Li, and S. Li, “Jigsaw: Interactive mobile visual search with multimodal queries,” in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2011, pp. 73–82.

[95] Y. Wang, G. Cong, G. Song, and K. Xie, “Community-based greedy algorithm for mining top-k influential nodes in mobile social networks,” in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2010, pp. 1039–1048.

[96] J. Weng, E.-P. Lim, J. Jiang, and Q. He, “Twitterrank: Finding topic-sensitive influential twitterers,” in *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*, 2010, pp. 261–270.

[97] S. Xie, G. Wang, S. Lin, and P. S. Yu, “Review spam detection via temporal pattern discovery,” in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2012, pp. 823–831.

[98] X. Xin, I. King, H. Deng, and M. R. Lyu, “A social recommendation framework based on multi-scale continuous conditional random fields,” in *Proceedings of the ACM International Conference on Information & Knowledge Management (CIKM)*, 2009, pp. 1247–1256.

[99] C. Xu, B. Su, Y. Cheng, W. Pan, and L. Chen, “An adaptive fusion algorithm for spam detection,” *IEEE Intelligent Systems*, vol. 29, no. 4, pp. 2–8, 2014.

[100] Q. Xu, E. W. Xiang, Q. Yang, J. Du, and J. Zhong, “Sms spam detection using noncontent features,” *IEEE Intelligent Systems*, vol. 27, no. 6, pp. 44–51, 2012.

[101] K. Yanai, H. Kawakubo, and B. Qiu, “A visual analysis of the relationship between word concepts and geographical locations,” in *Proceedings of the ACM International Conference on Image and Video Retrieval*, 2009, pp. 13–23.

[102] C. Yang, R. C. Harkreader, and G. Gu, “Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers,” in *International Workshop on Recent Advances in Intrusion Detection*, 2011, pp. 318–337.

[103] W. Yao, J. He, G. Huang, J. Cao, and Y. Zhang, “A graph-based model for context-aware recommendation using implicit feedback data,” *World Wide Web*, vol. 18, no. 5, pp. 1351–1371, 2015.

[104] S. Yardi, D. Romero, G. Schoenebeck *et al.*, “Detecting spam in a twitter network,” *First Monday*, vol. 15, no. 1, 2009.

[105] L. Ye and E. Keogh, “Time series shapelets: A new primitive for data mining,” in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2009, pp. 947–956.

[106] B. Yu and Z.-b. Xu, "A comparative study for content-based dynamic spam classification using four machine learning algorithms," *Knowledge-Based Systems*, pp. 355–362, 2008.

[107] D. Zeilberger, "Garsia and milne's bijective proof of the inclusion-exclusion principle," *Discrete Mathematics*, vol. 51, no. 1, pp. 109–110, 1984.

[108] Z.-J. Zha, L. Yang, T. Mei, M. Wang, Z. Wang, T.-S. Chua, and X.-S. Hua, "Visual query suggestion: Towards capturing user intent in internet image search," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, pp. 13–25, 2010.

[109] Q. Zhang, Q. Zhang, G. Long, P. Zhang, and C. Zhang, "Exploring heterogeneous product networks for discovering collective marketing hyping behavior," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD).* Springer, 2016, pp. 40–51.

[110] Y. Zhang and J. Koren, "Efficient bayesian hierarchical user modeling for recommendation system," in *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, 2007, pp. 47–54.

[111] Y. Zheng and X. Xie, "Learning travel recommendations from user-generated gps traces," *ACM Transactions on Intelligent Systems and Technology (TIST)*, pp. 2–32, 2011.

[112] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web (TWEB).*

[113] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from gps trajectories," in *Proceedings of the International Conference on World Wide Web (WWW)*, 2009, pp. 791–800.

[114] C. Zhou, N. Bhatnagar, S. Shekhar, and L. Terveen, “Mining personally important places from gps tracks,” in *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, 2007, pp. 517–526.

[115] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, and Y.-C. Zhang, “Solving the apparent diversity-accuracy dilemma of recommender systems,” *National Academy of Sciences*, vol. 107, no. 10, pp. 4511–4515, 2010.

[116] F. Zhu and X. Zhang, “Impact of online consumer reviews on sales: The moderating role of product and consumer characteristics,” *Journal of Marketing*, pp. 133–148, 2010.