

DISSERTATION  
submitted  
to the  
Combined Faculty for the Natural Sciences and Mathematics  
of  
Heidelberg University, Germany  
for the degree of  
Doctor of Natural Sciences

Put forward by

M.Sc. Tran Van Canh

Born in: Ky Anh, Ha Tinh, Viet Nam

Oral examination:.....

LEARNING SOCIAL LINKS AND COMMUNITIES FROM  
INTERACTION, TOPICAL, AND SPATIO-TEMPORAL  
INFORMATION

Advisor: Prof. Dr. Michael Gertz

## Abstract

The immense popularity of today’s social networks has led to the availability and accessibility of vast amounts of data created by users on a daily basis. Various types of information can be extracted from such data, for example, interactions among users, topics of user postings, and geographic locations of users. While most of the existing works on social network analysis, in particular those focusing on social links and communities, rely on explicit and static link structures among users, extracting knowledge from exploiting more features embedded in user-generated data is another important direction that only recently has gained more attention. Initial studies employing this approach show good results in terms of a better understanding latent interactions among users.

In the context of this dissertation, multiple features embedded in user-generated data are investigated to develop new models and algorithms for (1) revealing *hidden social links* between users and (2) extracting and analyzing *dynamic feature-based communities* in social networks. We introduce two approaches for extracting and measuring interpretable and meaningful social links between users. One is based on the participation of users in threads of discussions. The other one relies on the social characteristics of users as reflected in their postings. A novel probabilistic model called *rLinkTopic* is developed to address the problem of extracting a new type of feature-based community called regional *LinkTopic*: a community of users that are geographically close to each other over time, have common interests indicated by the topical similarity of their postings, and are contextually linked to each other. Based on the *rLinkTopic* model, a comprehensive framework called *ErLinkTopic* is developed that allows to extract and capture complex changes in the features describing regional *LinkTopic* communities, for example, the community membership of users and topics of communities. Our framework provides a novel basis for important studies such as exploring social characteristics of users in geographic regions and predicting the evolution of user communities.

For each approach developed in this dissertation, extensive comparative experiments are conducted using data from real-world social networks to validate the proposed models and algorithms in terms of effectiveness and efficiency. The experimental results are further discussed in detail to show improvements over existing approaches and the applicability and advantages of our models in terms of learning social links and communities from user-generated data.

## Zusammenfassung

Die immense Popularität heutiger sozialer Netzwerke hat zur Verfügbarkeit enormer Mengen an kontinuierlich aktualisierten nutzergenerierten Daten geführt. Aus diesen Daten können eine Vielzahl von Informationen extrahiert werden, beispielsweise Interaktionen zwischen Nutzern, Themen zu Postings von Nutzern sowie Standorte von Nutzern. Bisherige Arbeiten zur Analyse von sozialen Netzwerken, insbesondere aber Arbeiten zur Erkennung von sozialen Verbindungen und Nutzergruppen (Communities), beruhen ausschließlich auf der Verwendung expliziter und statischer Strukturen von Verbindungen zwischen Nutzern; Methoden zur Verwendung weiterer in Nutzerdaten und Postings eingebetteter Features haben erst in letzter Zeit mehr Beachtung gefunden. Erste Ansätze, die diese weiterführende Methodik verwenden, zeigen gute Resultate bzgl. eines besseren Verständnisses latenter Interaktionen zwischen Nutzern.

In dieser Arbeit werden verschiedene Eigenschaften von nutzergenerierten Daten mit dem Ziel untersucht, neue Modelle und Algorithmen zu entwickeln, um (1) latente soziale Verbindungen zwischen Nutzern aufzudecken und (2) dynamische Communities zu extrahieren und zu analysieren. Hierzu stellen wir zwei neue Ansätze vor, um aussagekräftige und interpretierbare Informationen zu sozialen Verbindungen zwischen Nutzern zu extrahieren. Ein Ansatz basiert auf der Interaktion von Nutzern in Diskussionsforen. Der andere Ansatz basiert auf den sich in den Postings widerspiegelnden sozialen Charakteristiken von Nutzern.

Hierzu wird in dieser Arbeit ein neues probabilistisches Modell (*rLinkTopic*) entwickelt, das es erlaubt, einen neuen Typ merkmalsbasierter Communities (sogenannte regionale *LinkTopics*) zu extrahieren. Hierbei handelt es sich um eine Community, bei der sich die Nutzer über einen Zeitraum hinweg in geographischer Nachbarschaft befinden, gemeinsame Interessen haben bzgl. der Themen in ihren Postings und untereinander über die Postings implizit verlinkt sind. Basierend auf dem *rLinkTopic* Modell wird ein Rahmenwerk entwickelt (*ErLinkTopic*) welches es erlaubt, komplexe Veränderungen der Eigenschaften von regionalen *LinkTopics* zu modellieren und zu extrahieren, wie die Zugehörigkeit von Nutzern zu Communities und die Themen einer Community über Zeit und Raum. Das Rahmenwerk bildet die Basis für neuartige Studien, wie beispielsweise die Exploration von sozialen Charakteristiken von Nutzern in geographischen Regionen und Vorhersagen zur Entwicklung von Communities.

Alle in dieser Arbeit vorgestellten Methoden werden in umfangreichen, vergleichenden Experimenten hinsichtlich ihrer Effektivität und Effizienz evaluiert. Hierzu werden Daten aus realen sozialen Netzwerken verwendet. Die Ergebnisse der Evaluation werden im Detail diskutiert und die Vorteile dieser neuen Ansätze gegenüber existierenden Ansätzen herausgestellt. Zudem werden die Eignung und die Vorteile der Modelle in Bezug auf die Vorhersagbarkeit von sozialen Verbindungen und Communities basierend auf nutzergenerierten Daten diskutiert.



## Acknowledgements

This dissertation would not be possible without the support of many people whose help is either visible or hidden to me in the years of my Ph.D study. First and foremost, I would like to express my deep gratitude to my advisor, professor Michael Gertz, for the invaluable encourage, advice, and support I received from him, and for his patience and sympathies to my research progress and personal issues over the years. For these and much more I learned from him, I feel myself really lucky to be one of his students and I am forever in his debt. I am also grateful to the reviewers of this dissertation for the comments I received.

During my Ph.D study I had the pleasure to work with fantastic and talented colleagues at the Database Systems Research Group, Heidelberg University. Thank you Jan-nik Strötgen for inviting me to join the group of HeidelTime and for the papers we worked together on. My thanks to Florian Flatow, Hamed Abdelhaq, and Katarina Gavrić for all we had in our office. You are ideal office mates and I am glad to sit next to you. Thank you very much Florian for the many times of interpretation and communication through phones that helped me a lot to deal with German paperworks. Thanks Hamed for sharing many similar difficulties that we both have experienced as doing Ph.D study while having children at home. Thanks Katarina (Keti) for a lot of fun you have brought to our office since you came. My thanks also to Christian Sengstock, Natalia Ulrich, Le Van Quoc Anh, Ayser Armiti, Hui Li, and Thomas Bögel for your friendship and support over the years.

I am thankful to the Ministry of Education and Training of Vietnam (MOET) and the Deutscher Akademischer Austausch Dienst (DAAD) for the financial support so that this dissertation has become possible. Thank you very much Dr. Kristen Nawrotzki at Pädagogische Hochschule Heidelberg and Dr. Denis Vogel at the Faculty of Mathematics and Computer Science for the projects that made my study more busy but helpful.

When these lines are written, it has been 1455 days since I first entered Germany. I left my son when he started learning to walk and now he is ready for the first class at school in fall. My daughter is almost two years old and learning to communicate with papa through Webcam every morning. During these days, all works at home that were supposed to be mine were managed and carried out by my wife. Saying thank you would not be enough to express all gratefulness towards them. However, I am sure that living far from them makes me realize more than ever how much they mean to me. They are the underlying reason and semantics of all that I do every day.

Finally, this dissertation marks the end of a long journey that began in a rural area located in the North-Central of Vietnam. There, not many friends of my age had an opportunity to attend university or even finished high school because of poverty. However, my parents made tremendous sacrifices to ensure that I had a good education. Their hard working and motivation gave me a chance to meet many people, to know how much I need to learn and hopefully how to motivate my children to work hard and success in their life. For these and much more, I am grateful beyond words to my parents.

To my parents, my wife, and children.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Research Problems and Goals . . . . .	2
1.2	Motivation and Challenges . . . . .	2
1.3	Contributions . . . . .	5
1.4	Thesis Outline . . . . .	6
<b>2</b>	<b>Background and Related Work</b>	<b>7</b>
2.1	Overview and Objectives . . . . .	7
2.2	Social Networks . . . . .	7
2.2.1	A Brief History of Social Networks . . . . .	7
2.2.2	Examples of Social Networks . . . . .	9
2.3	Graph Principles for Social Network Studies . . . . .	10
2.3.1	Basics of Graph . . . . .	10
2.3.2	Centrality Measures . . . . .	11
2.3.3	Other Measures and Definitions . . . . .	13
2.4	Finding Communities in Graph . . . . .	14
2.4.1	Community Structures . . . . .	14
2.4.2	Graph Clustering Approaches . . . . .	17
2.5	Probabilistic Models for Discovering Communities . . . . .	21
2.5.1	Random Variables and Bayes' Theorem . . . . .	21
2.5.2	Exchangeability . . . . .	24
2.5.3	Conjugate Prior . . . . .	25
2.5.4	Graphical Model . . . . .	28
2.5.5	Gibbs Sampling for Posterior Estimation . . . . .	30
2.5.6	Related Work Employing Probabilistic Models . . . . .	31
2.6	Evolving Social Networks and Communities . . . . .	32
2.6.1	Dynamics of Social Networks . . . . .	32
2.6.2	Evolution of Communities . . . . .	32

<b>3</b>	<b>Extraction and Measurements of Social Links</b>	<b>35</b>
3.1	Overview and Objectives . . . . .	35
3.2	Social Network Data . . . . .	36
3.2.1	User-Centric Model . . . . .	36
3.2.2	Assumptions and Conventions . . . . .	37
3.2.3	Social Links . . . . .	38
3.3	Measuring Social Links from Interactions . . . . .	39
3.3.1	User <sup>2</sup> -Thread Network . . . . .	40
3.3.2	User-Thread Association . . . . .	42
3.3.3	Thread Association-Based Link Measure . . . . .	44
3.3.4	Interaction-Based Link Measure . . . . .	45
3.4	Semantic Analysis for Measuring Social Links . . . . .	45
3.4.1	Term Significance for Users . . . . .	46
3.4.2	Semantic-based Social Link . . . . .	47
3.5	Recommendation Applications . . . . .	48
3.5.1	Collaborative Filtering Paradigm . . . . .	48
3.5.2	Friend Recommendation . . . . .	49
3.5.3	Thread Recommendation . . . . .	50
3.6	Experiments . . . . .	51
3.6.1	Dataset for Experiments . . . . .	51
3.6.2	Interaction-based Link Network . . . . .	54
3.6.3	Latent Semantic-based Network . . . . .	63
3.7	Summary and Discussion . . . . .	66
3.7.1	Summary . . . . .	66
3.7.2	Outlook for LBSN Data . . . . .	66
<b>4</b>	<b>Regional LinkTopic Community Extraction</b>	<b>69</b>
4.1	Overview and Objectives . . . . .	69
4.2	Topic Models . . . . .	70
4.2.1	Latent Dirichlet Allocation . . . . .	70
4.2.2	Gibbs Sampling for LDA . . . . .	73
4.3	Regional LinkTopic Communities . . . . .	79
4.3.1	Introduction . . . . .	79
4.3.2	Preliminaries . . . . .	80
4.4	rLinkTopic Probabilistic Model . . . . .	82
4.4.1	Joint Contextual Links and Topics . . . . .	82
4.4.2	Geographic Region Model . . . . .	83
4.4.3	Generative Process . . . . .	85
4.4.4	Posterior Estimation for rLinkTopic . . . . .	87
4.4.5	Gibbs Sampling Algorithm . . . . .	99

4.5	Evaluation Measures . . . . .	101
4.5.1	Spatial Entropy Measure . . . . .	101
4.5.2	Perplexity Measure . . . . .	102
4.6	Experiments . . . . .	103
4.6.1	Twitter Datasets . . . . .	103
4.6.2	Link Structure and Spatial Characteristics of Datasets . . . . .	104
4.6.3	Experimental Setup . . . . .	105
4.6.4	Regional LinkTopic Communities . . . . .	107
4.6.5	Quantitative Evaluation . . . . .	111
4.7	Summary and Discussion . . . . .	118
<b>5</b>	<b>Analysis of Community Evolution</b>	<b>119</b>
5.1	Overview and Objectives . . . . .	119
5.2	Data Model and Notations . . . . .	120
5.3	ErLinkTopic Probabilistic Model . . . . .	122
5.3.1	rLinkTopic to ErLinkTopic . . . . .	122
5.3.2	Posterior Estimation for ErLinkTopic Model . . . . .	124
5.4	Evolution of Communities . . . . .	129
5.4.1	Changes in Community Members . . . . .	131
5.4.2	Changes in Topics of Communities . . . . .	133
5.5	Experiments . . . . .	134
5.5.1	Twitter Datasets . . . . .	134
5.5.2	Experimental Setup . . . . .	135
5.5.3	Dynamic Measure Analysis . . . . .	135
5.5.4	Selected Evolving Communities . . . . .	143
5.5.5	Evolution of Topics Associated with Communities . . . . .	149
5.5.6	Evaluation of Runtime . . . . .	152
5.6	Summary and Discussion . . . . .	154
<b>6</b>	<b>Conclusions and Future Work</b>	<b>157</b>
6.1	Summary . . . . .	157
6.2	Future Work . . . . .	158
	<b>Bibliography</b>	<b>160</b>



# Chapter 1

## Introduction

Social network analysis (SNA) has become a rapidly emerging research discipline in the last decade. The methods and techniques of SNA involve a variety of areas including mathematics, statistics, and computer science [121]. Due to its relevance to various processes taking place in society, SNA finds significant applications in several fields such as sociology, biology, communication, geography, social computing, and business [14, 15, 107]. In the context of data mining and towards applications, results of SNA are used extensively in data aggregation, modeling of information propagation, advertisement, and recommendation, to name but a few [110]. Recently, the emergence of online social networks provides huge amounts of rich-feature data created by hundreds of millions of users on a daily basis. On one hand, this gives much better opportunities than ever before for researchers to study many other problems and evaluate the models developed. On the other hand, one has to deal with more challenges due to the sparsity and noise of data, besides the need for the flexibility, complexity, and scalability of the models introduced.

Among many other research issues in SNA, the relationships between and communities of users have gained significant attention and lots of work has been conducted on these topics. This is because information obtained from studying social links and social communities is useful for many applications built on top of social networking services. Examples include targeted advertising, content delivery, and personalized recommendation. A deep understanding of social links and communities can also provide important insights into questions of human social behavior, as well as designing new services for social platforms. Initial studies in SNA focus on the topological characteristics of the social graph that capture the explicit relationships indicated by link structures among users [4, 82, 124]. Recently, researchers have shifted the attention to the observable activities of users to create more accurate predictive models for social behavior [10, 27, 29, 58, 117]. The goal is to better understand the true nature of relationships between users. For example, in [10] the authors determine that there are actually more users reading the content one posts to a network than those observed from friendship links. Such users are called *invisible audience*, who might share some interest with the author of the posting. Similarly, studies in [27, 117] report that results obtained from analyzing networks built upon the activity of users, called

*activity networks*, are more informative compared to information derived from friendship networks to understand the social behavior of users. In [58, 81], by analyzing data collected from different social networks, the authors discover that latent interactions are much more prevalent and frequent than observed ones. These imply the existence of so-called *hidden social links* between users. Thus, it is reasonable to expect that social links and communities can be extracted as latent structures from different features associated with users, instead of only relying on explicit and static link structures. This initially shapes the ideas for the study presented in this dissertation, which are the measurements of hidden social links between users, and the extraction and analysis of (dynamic) *feature-based* communities in social networks.

In the following, the main problems that will be addressed in this dissertation are first described. We then give the motivation and challenges of our study in Section 1.2. The main contributions of the dissertation are summarized in Section 1.3 before we conclude this chapter with the thesis outline in Section 1.4.

## 1.1 Research Problems and Goals

Broadly speaking, this thesis aims at developing models and algorithms that rely on different features of user-generated data in social networks for answering questions related to the existence of hidden social links and communities. Particular concerns related to the investigation include, for example, *instead of relying on link structures, can one determine more useful and interpretable social links between users based on their activities and associated contents in a social network ?*, *given that user-generated data in social networks contain rich features, which ones should be considered and how to employ them to develop a model for meaningful community extraction ?*, and *given that a community evolves over time regarding changes in the features describing it, how to extract and capture such complex evolutions of communities ?* In this dissertation, various techniques from graph theory, latent semantic analysis, and Bayesian statistics are employed to address such questions. Particularly, for the first goal, two novel models for measuring *interaction-based* and *latent semantic-based* social links are introduced. For the goal of extracting and analyzing feature-based communities, a complex probabilistic framework is presented and the corresponding Gibbs sampling algorithms are developed. Information about geographic locations, topics of interest, and contextual links of users over time are taken into account to address the problems. To validate the proposed models in terms of effectiveness and efficiency, different real-world social network datasets are used and the obtained results are discussed as well.

## 1.2 Motivation and Challenges

In the last few years, one has witnessed a dramatic popularization of social networks. The number of users in social networks is now approaching 20% of the world population and is more than 50% of the people using the Internet [42]. Thus, social networking becomes a real



demand and plays a significant role in the daily life of people all over the world. Nowadays, people can access social networks using both computers and smartphones. According to a study conducted by the Nielsen company<sup>1</sup> in 2010, people worldwide spent over 110 billion minutes in social networks per month, which accounts for 22% of all time people spent on the Internet. A recent report from the Pew Internet and American Life Project<sup>2</sup> shows that smartphone ownership among American adults has increased from 35% in 2011 to 46% in 2012. Among these smartphone owners about 74% of people use location-based applications on their phone to get directions and recommendations, and about 12% in 2011 and 18% in 2012 of people use services like *Foursquare*, *Gowalla*, and *Facebook Places* to *check-in* at certain locations and share contents to the public. Such an emergence of social networks opens a lot of new challenging research problems, which are inspired by the fact that activities of users in social networks exhibit a mirror of their real-life. In this dissertation, we particularly draw our attention to the measurements of hidden social links between users, and the extraction and analysis of feature-based communities. Observations and challenges motivating our study are summarized as follows.

**Application perspectives.** Generally, the input of SNA is assumed to be a social graph of users. By this, one often abstracts from the social link connecting users or normally considers the explicit link structures among users as an evidence of their relationships. Nevertheless, it is important to note that almost all SNA tasks involve some social link measure, even though the extent it is employed varies. For the goal of understanding general properties of a network, one might not pay much attention to the link measure. On the contrary, in cases of conducting an analysis for specific purposes, the relationships between users are often needed to be semantically and quantitatively identified. One might think of what kind of relationships connects two users and how strong a relationship is between them. As mentioned, the observed link structures are typically employed to measure relationships between users, e.g., [85, 97, 123]. However, scholars in psychology and sociology have cast doubts on the practice of detecting meaningful relationships from link structures alone, given how easy it is for a user to create a link to other users in a social network [58, 81]. In this work, we investigate two features, namely the participation of users in discussion topics and the semantics of users' postings to extract and measure hidden social links between them. Concrete applications of social link measures include, among others, community detection, friend recommendation, and content delivery.

One of the important implications derived from social behavior of users is known as community. Semantically, a community can be generally defined as a group of users who exhibit more similar behavior to each other than to those not in that group. In this dissertation, we aim at extracting communities where users in a community are related to each

---

<sup>1</sup><http://www.nielsen.com/us/en/newswire/2010/social-media-accounts-for-22-percent-of-time-online.html>[Accessed April 2014]

<sup>2</sup>Three-quarters of smartphone owners use location-based services, Pew Internet and American Life Project: <http://pewinternet.org/Reports/2012/Location-based-services.aspx>[Accessed April 2014]

other in the sense that (1) they are spatially located close to each other over time; (2) they have common interests indicated by the topical similarity of their postings; and (3) they are contextually linked to each other in the messages sent to a social network. Even though several approaches were proposed, none of the existing models takes all these features into account for discovering and analyzing communities. In terms of applications, extracting such feature-based communities and capturing their evolution provide useful insights into the behavior of users and communities especially when geographic and regional information is considered. Some specific applications that might benefit from such information include the targeted community recommendation, geographically focused social studies, and evolution and trend prediction such as disease propagation and political trends in local areas.

**Feasibility perspective.** One of the main difficulties that researchers face in analyzing social behavior and relationships between people in the past is the lack of relevant data for evaluating the models and algorithms developed. Initial works were conducted on the data collected from using questionnaires, interviews, and other labor-intensive methods, which are only appropriate for studying some social phenomena in particular social settings [14]. This, however, is not a big problem nowadays thanks to the emergence of online social networks. Many sophisticated features have been added to such services in recent years, which provide users various tools to share their real-life to virtual societies. Users can post several types of media (e.g., text messages, pictures, movies,...) and create not only static links but also contextual links to each other. Most social networks provide methods allowing people to collect such rich-feature data generated by users on a daily basis.

In addition, there has been a significant change regarding the way people connect to the Internet in the last few years. People nowadays can access the Internet using their smartphones from almost everywhere. Most smartphones are also equipped with a GPS sensor that allows to develop applications to retrieve the geographic location of users. Social network providers have quickly adopted such location-sensing features. Client services have been developed so that a geographic location can be explicitly or implicitly associated with the content posted by users. For example, a user can *check-in* to tell friends her whereabouts or tag a geographic location with a picture she posts to a social network. Having witnessed the strong adoption of users for location sharing features, the most popular social networks including *Facebook* and *Twitter* have recently launched location embedding features that allow users to tag a geographic location to the media posted to such networks. Indeed, almost all social networks nowadays are becoming location-aware and, thus, there is no clear distinction between purely location-based social networks and general social network platforms anymore [105].

As a consequence, data collected from social networks often contain spatio-temporal information, contextual links exhibiting social connections, and textual descriptions reflecting the real-life of users. An example is a user posting a picture enriched with a textual tag describing an event, a geographic location telling where the picture was taken, and some

contextual links connecting to her friends. Thus, with the availability and accessibility of such heterogeneous and rich-feature data, there is a great opportunity to develop and evaluate complex models for investigating hidden social links between users and feature-based communities.

**Challenging perspective.** Extracting latent structures and patterns from the data is generally a challenging task. In the context of this dissertation, our first aim is the measurements of hidden social links between users. This is not trivial because one first needs to investigate the features offered by the social network under consideration, and, consequently, study how users exhibit their real-life in the network as well. The second problem that we deal with is the discovery and analysis of feature-based communities. The main questions are, for example, *how to use available information obtained from user-generated data to extract meaningful communities ?* and *how to accurately and efficiently capture changes in the features describing communities over time ?* Finding solutions for such questions is clearly not a simple task.

To this end, given that social network data are noisy and sparse in nature, developing complex models that take different features embedded in such data into account to achieve the goals of this dissertation is challenging. However, under the application perspective, this is a helpful task.

### 1.3 Contributions

This dissertation makes the following main contributions to the research topics related to the extraction and analysis of social relationships and communities of users in social networks.

- We introduce a data model for analyzing social networks, particularly for measuring hidden social links, and for extracting feature-based communities and analyzing their evolution.
- We introduce two models for measuring hidden social links between users, which are derived (1) from the participation of users in discussion threads and (2) from the social characteristics of users obtained as the result of applying latent semantic analysis to their postings, respectively.
- We develop a complex probabilistic framework and derive Gibbs sampling algorithms for extracting and analyzing a new type of feature-based community called regional *LinkTopic*. A community of this type is identified on the basic of geographic locations, topics of interest, and contextual links of users over time.
- We conduct extensive comparative experiments to evaluate the effectiveness and efficiency of the proposed models using data collected from different real-world social networks. The results obtained from each model are further discussed as well to show the applicability and advantages of the approach introduced.

## 1.4 Thesis Outline

The rest of the thesis is structured as follows.

- **Chapter 2.** In this chapter, we present the background and related work relevant to the problems studied in this dissertation. An overview of social networks is first introduced and then some fundamental concepts in graph theory are given. Community structures embedded in graphs and the two main approaches, namely graph clustering-based and probabilistic-based, for extracting and analyzing communities are discussed in detail in this chapter as well.
- **Chapter 3.** A data model used throughout the dissertation is first formalized and two approaches for extracting and measuring social links are then developed. In the first model designed for blog and forum networks, a *hyper-bipartite* graph is proposed to represent interactions among users. Based on this graph, a Markov Random Walk strategy is employed to derive hidden social link weights. In the second model, a refined term frequency-inverse document frequency schema is introduced on which social link scores are derived using latent semantic analysis. A dataset collected from the *BBC Message Boards* network is used to evaluate the proposed models.
- **Chapter 4.** In this chapter, we introduce a new type of feature-based community called regional *LinkTopic*. A regional *LinkTopic* community is formed by users that are located in spatio-temporal proximity, have common interests indicated by the topical similarity of their postings, and are contextually linked to each other, e.g., by tagging or mentioning each other in their postings. Thus, a community of this type is characterized by not only the identity of users but also by the topics of interest and regional aspects. We develop a novel probabilistic model called *rLinkTopic* for extracting such meaningful communities. Extensive experiments are conducted using *Twitter* data, and the obtained results are evaluated to show the utility and advantages of the model compared to others.
- **Chapter 5.** Inspired by the fact that communities evolve over time, in this chapter the *rLinkTopic* model developed in Chapter 4 is extended to build a comprehensive framework called *ErLinkTopic*. The model is not only able to extract regional *LinkTopic* communities but also, at the same time, to capture the evolution of the features describing each community. By this, complex evolutions of communities are determined and analyzed. The results obtained from experimental evaluations using *Twitter* data are discussed to show the effectiveness, efficiency, and applicability of the approach.
- **Chapter 6.** This final chapter gives a summary of our work presented in this dissertation and describes open issues for further studies.

## Chapter 2

# Background and Related Work

### 2.1 Overview and Objectives

As stated in the previous chapter, this thesis is mainly about measuring social links between users, and detecting and analyzing the evolution of communities in social networks. Different from existing studies, which mainly rely on link structures, in this framework we aim at analyzing more features describing users to achieve our goals. For this purpose, techniques for data analysis using graphs, latent semantic extraction, and spatio-temporal and topical analysis are employed to develop new models and algorithms. This chapter presents the background and discusses related work that are most relevant to our study. We begin in Section 2.2 with a brief overview of social networks. In Section 2.3, we review basic concepts and statistical measures for graphs that are useful for studying social networks. Community structures and graph clustering-based methods for detecting communities are presented in Section 2.4. A recent approach that employs probabilistic models for extracting communities is discussed in Section 2.5. In Section 2.6, we briefly recap works on the dynamics of social networks and approaches to analyzing the evolution of communities.

### 2.2 Social Networks

#### 2.2.1 A Brief History of Social Networks

The concept of a *social network* exists since humans began socializing. It simply describes interactions between people in any kind of communication. This means that the theory and techniques of social network analysis have a long history [39]. Since the last decades, however, there has been a shift in the usage of the term *social networks*. Nowadays, it is used to denote online services on the Internet that allow registered users to connect to each other, to exchange information, and to share information. In this work, models and algorithms are developed to analyze data collected from such Internet-generation social networking platforms. Therefore, we adopt this new meaning respect of social networks throughout the

dissertation. Indeed, there are several online applications supporting users to create social interactions and to exchange information. This implies no proper classification of which applications are the real social networks. Nevertheless, a brief history of the development of online services that are often adopted as social networks is summarized as follows.

In 1994, the first web-based social networking application, *Geocities*, was developed. *Geocities* allows users to create their own websites like today's blog-sites. One year later, the *theglobe.com* was built. It allows users to publish contents and to interact with other users who share similar interests. *AOL instant messenger* emerged in 1997. This service offers a new concept called *instant messaging* that becomes a very popular feature in social networks nowadays. In the same year, *sixdegrees.com* was launched, which allows users to create their individual profile and to search for friends. Launched in 2002, *Friendster* was a real breakthrough in the field of social networking services. *Friendster* is the pioneer in using the concept of online networking between real-world friends. In 2003, *Myspace*, at first as a clone of *Friendster*, and many other social networks were launched. Among those networks, *LinkedIn*<sup>1</sup>, which was designed for professional users to connect and collaborate, is widely adopted until today. In 2004, *Facebook*<sup>2</sup> was launched at Harvard University. The first version of *Facebook* was designed as a service for connecting U.S. college students. Two years later, in 2006, *Twitter*<sup>3</sup> was launched as a social networking service that allows users to create micro-blogging sites, and to send and receive 140-character messages called *tweets*. In 2008, *Facebook* overtook *MySpace* to become the leader among social networking sites. *Google plus* joined the world of social networks in 2011. A detailed history of social networks can be found in a report by the University of North Carolina<sup>4</sup>.

An important feature provided in today's social networks is the support for the association of information about geographic locations of users with the content they post to the network. This feature leads to a new concept called Location-Based Social Networks (LBSNs). A first large scale commercial LBSN was *Dodgeball*, which was created in 2002 and then bought by *Google* in 2005. *Dodgeball* introduced a *check-in* concept in a form of a SMS text message with a geographic location. Users employ this form to send messages together with their location to a central server and the server then delivers such information to their friends. After appearing in *Dodgeball*, *check-in* becomes a prominent feature in today's LBSNs thanks to the development of GPS equipped mobile devices. In 2007, *Brightkite*<sup>5</sup> was founded as a social networking service that allows users to share their location with friends. The original authors of *Dodgeball* launched *Foursquare*<sup>6</sup> in 2009. The service supports a game feature to the traditional *check-in* so that the user having the highest number of *check-in* in the last 60 days is deemed as a *mayor* of a place. This feature encourages

---

<sup>1</sup><http://www.linkedin.com>

<sup>2</sup><http://www.facebook.com>

<sup>3</sup><http://www.twitter.com>

<sup>4</sup><http://www.uncp.edu/home/acurtis/NewMedia/SocialMedia/SocialMediaHistory.html>

<sup>5</sup><http://www.brightkite.com>

<sup>6</sup><http://www.foursquare.com>

users sharing their location in order to win the competition to become a mayor. It was also in 2009 that *Gowalla*<sup>7</sup> was launched, which then was bought by *Facebook* in 2011.

### 2.2.2 Examples of Social Networks

To give the reader a better intuition of key features of social networks, we briefly describe here three networks that are currently the most prominent social networking platforms.

**Facebook:** *Facebook* is one of the largest and best-known social networks today. The number of active users on *Facebook* increases from more than 500 million in October 2010 to 1.01 billion in October 2012 [42, 59]. Being a member of *Facebook*, a user has a profile that contains basic information such as name, date of birth, marital status, and personal interests [124]. Each user profile has a so-called *Wall* where the owner and friends can post messages and reply to messages posted. *Facebook* provides users with different features to interact with other *Facebook* users, some of which are summarized below.

- Connect to other *Facebook* users and request to make friends with them.
- Create contextual links by tagging other users in postings.
- Post messages on friends' *Wall* and send private messages to friends.
- Create a page for some event such as a birthday party or a workshop and invite friends to join the event.

**Twitter:** *Twitter* is an online service that allows registered users to post short messages, called *tweets*, of up to 140 characters. The main feature that distinguishes *Twitter* from other social networks is that *Twitter* users mainly post messages talking about what is currently happening around them. A *Twitter* user can follow other users, meaning that she decides to be a follower or a *friend* of those users. Such a friendship on *Twitter* is not necessarily reciprocal. A *follower* is able to see all tweet chains posted or retweeted by users she follows. A *Twitter* user interacts with other users or shares information by creating a new tweet or replying to tweets created by other users. In addition, *Twitter* users are able to specify contextual links by mentioning other users in their tweets. *Twitter* has been strongly adopted by people worldwide as there are more than 200 million daily tweets posted by users in 1<sup>st</sup> August, 2011 [71]. In April 2012, *Twitter* has more than 500 million active users [59].

**LinkedIn:** *LinkedIn* is designed as a credible professional social network. Being a member of *LinkedIn*, a user can set up a profile containing both personal and professional information. One of the reasons why *LinkedIn* is a useful tool for business and research is that it organizes users based on professional life in standardized categories. For example, one can query for users based on the university they attended, what their qualifications are, or which companies they have worked with. The number of active *LinkedIn* users increased from more than 90 million in January 2011 to 175 million in June 2012 [59, 76].

---

<sup>7</sup><http://www.gowalla.com>

There are, however, other services such as forums, blog sites, and email systems that can also be considered social networks. For example, users in email networks interact by sending and receiving mails, while users in a forum post messages to specific threads for discussions. Thus, these networks exhibit similar features as a social networking service.

## 2.3 Graph Principles for Social Network Studies

The main feature that differentiates social networks from other applications is that social networks allow users not only to post different types of information to the network but also to create explicit or exhibit implicit *social links* to each others. The latter aspect becomes a subject of major research topics that are driven by various questions raised in different application domains. Examples include the study of social behavior and ties of users, and the extraction of communities. These often take into account both users and relationships among them as input, which are typically represented as a graph structure. Each node of the graph corresponds to a user, and each edge of the graph encodes the relationship between two users. The graph is referred to as a *social graph* or a *link graph*. Relationships among users are extracted from their data, which can be any kind of *social connection* of interest including explicit link structures, and common interests or common behavior of users. The latter two features indicate implicit links between users, which are extracted from user-generated data by employing various techniques such as topical and spatio-temporal mobility analysis, e.g., [29, 135].

Given that graph structures play an important role in the development of models and algorithms for the analysis of social networks, this section briefly presents the basic concepts and statistical measures for graphs that are necessary for discussing the background and related work of our study. We adopt common notations used in the literature, for example, the definitions presented in [121], for graph formalization.

### 2.3.1 Basics of Graph

Graph theory has a long history that might date back to Euler’s solution for the puzzle of Königsberg’s bridge in 1736, or even earlier [37]. Broadly speaking, graphs are a mathematical means for representing systems that have objects interacting or connected to each others. Examples include the protein interaction networks, computer networks, the WWW, and the connections among users in social networks known as *social graphs*, to name but a few. One often finds a formalization of graphs as follows.

**Definition 2.1 (Graph)** *A graph is an abstract representation of a set of objects where some pairs of the objects are connected by some type of link. Objects are called vertices or nodes and links are called edges. In the most common sense of term, a graph is mathematically represented as  $\mathcal{G} = \langle V, E \rangle$  where  $V$  is a set of vertices and  $E \subseteq V \times V$  is a set of edges. The notions  $|V|$  and  $|E|$  denote the number of vertices and the number of edges, respectively.*



Graphs can be classified into undirected and directed graphs. In an undirected graph, there is no direction existing on the endpoints of edges. In other words, if  $(u, v)$  is an edge of the graph then so is  $(v, u)$ . In contrast, there exists an order between the two vertices of an edge in a directed graph meaning that there might be no edge from  $v$  to  $u$  even though there exists an edge from  $u$  to  $v$ . Edges in directed graphs are often called arcs. For convenience, in the rest of this study, we will use the terms *undirected graph*, *graph*, and *network* as synonyms. The terms *social link* and *social relationship* are also used interchangeably. For example, the friendship network of *Facebook* users is an undirected graph because making friends on *Facebook* requires an agreement of both users involved. On the other hand, the *following* relationship between *Twitter* users derives a directed graph because a *Twitter* user can follow any other user and such a relationship is not necessarily reciprocal. Edges of a graph might be weighted and the graph is then called a weighted graph. In a social graph, the weight of an edge indicates the strength of the interaction or the relationship between two users connected by the edge. For example, the number of messages users exchanged or the semantic similarity score derived from the messages of users can be used to weight edges of a social network. In social networks, an interesting implication derived from link structures associated with users is community structures. In graph terms, a community is formalized based on the concept of subgraph, defined as follows.

**Definition 2.2 (Subgraph)** A graph  $\mathcal{G}_s = \langle V_s, E_s \rangle$  is a subgraph of a graph  $\mathcal{G} = \langle V, E \rangle$  if  $V_s \subseteq V$ , and  $E_s \subseteq E$  restricted to vertices in  $V_s$ .

### 2.3.2 Centrality Measures

In social network analysis, an important task is to identify the most important users in the network such as finding users who have strong influence on others, or users playing some *central role* in communication for the whole network or within a community. In graph theory, centrality measures are regarded as a conceptual means used to explore the relative importance of nodes and edges in a graph. Therefore, the idea of centrality measures has been adopted to assess social roles of an individual user or a group of users in a social network [17, 35, 91]. There are four instances of centrality measures mainly used, which are the degree centrality, betweenness centrality, closeness centrality, and eigenvector centrality. Details of these measures are presented in the following paragraphs.

**Degree centrality.** The degree centrality for a node  $v$  is the number of edges that are incident on  $v$ . In a directed graph, each node  $v$  has two measures of degree, namely *indegree* and *outdegree*. Indegree is the number of edges that direct to  $v$ , while outdegree is the number of edges that the node  $v$  directs to other nodes. The degree centrality is used to measure how important a node is in the sense that nodes having the most directed ties to other nodes will be the most important nodes in the network. This is because such nodes play an active role in communicating with other nodes. Degree centrality is a local

measure because only the edges formed by  $v$  with its adjacent nodes are taken into account to evaluate the importance of  $v$ .

**Closeness centrality.** The closeness centrality is a measure aimed at evaluating how *close* a node is to other nodes in the graph. The idea is that a node is central if it can quickly interact with other nodes. In other words, nodes that have shorter geodesic distances to other nodes should have a higher closeness measure. The closeness centrality for a node  $v$  in a connected graph is derived from the mean of lengths of all shortest paths from  $v$  to other nodes in the graph, which is formalized as

$$Close_c(v) \triangleq \frac{|V| - 1}{\sum_{u \neq v} length(u, v)}, \quad (2.1)$$

where  $length(u, v)$  is the length, computed based on some distance measure, of the shortest path between  $u$  and  $v$ . A proposal for measuring the closeness of nodes in a disconnected graph can be found in [30].

**Betweenness centrality for nodes.** This measure considers nodes appearing in more shortest paths between other nodes to have a higher betweenness score in the graph. Particularly, the betweenness centrality for a node  $v$  is the fraction of the number of shortest paths between pairs of nodes that  $v$  appears in. Let  $\sigma(u, q)$  be the number of shortest paths between nodes  $u$  and  $q$ , and  $\sigma(u, q, v)$  be the number of shortest paths between  $u$  and  $q$  that contain  $v$ . Then, the betweenness centrality for  $v$  is measured as follows.

$$Between_c(v) \triangleq \sum_{u \neq q \neq v} \frac{\sigma(u, q, v)}{\sigma(u, q)} \quad (2.2)$$

The value of  $Between_c(v)$  ranges from 0 to the number of pairs of nodes in the graph excluding  $v$ , i.e.,  $(|V| - 1) \times (|V| - 2)/2$ . Therefore, one can normalize the betweenness centrality measure for a node as follows.

$$Between_c(v) \triangleq \sum_{u \neq q \neq v} \frac{\sigma(u, q, v)}{\sigma(u, q)} \times \frac{2}{(|V| - 1) \times (|V| - 2)} \in [0, 1] \quad (2.3)$$

**Eigenvector centrality.** The eigenvector centrality measure assesses the importance of a node in a graph by putting it in the context of social influence. This measure gives a relative score to each node in the graph based on the principle that links to high-scoring nodes contribute more to the score of the node. Specifically, let  $\mathbf{A}$  be the adjacency matrix of the graph, i.e.,  $a_{u,v} = 1$  if nodes  $u$  and  $v$  are adjacent and  $a_{u,v} = 0$  otherwise. We want to assign scores to nodes based on the idea that the score of node  $v$  should be proportional to the sum of the scores of other nodes that are adjacent to  $v$ . This can be formalized as

$$Eigen_c(v) \triangleq \beta \times \sum_{(v,u) \in E} Eigen_c(u) = \beta \times \sum_{u \in V} a_{u,v} \times Eigen_c(u), \quad (2.4)$$

where  $\beta$  is some constant. By using vector representation, one can rewrite the above equation as

$$Eigen_c = \beta \times \mathbf{A} \times Eigen_c \quad \text{or} \quad \mathbf{A} \times Eigen_c = \lambda \times Eigen_c, \quad (2.5)$$

where  $\lambda = \frac{1}{\beta}$ . It turns out that to compute scores for nodes (i.e., to find the vector  $Eigen_c$ ), one needs to find the eigenvalues,  $\lambda$ . A study by Newman [87] already proved that only the greatest eigenvalue results in the desired centrality measure. Eigenvector centrality measure has been applied in different applications to assess the prominence of objects in the corresponding setting. The PageRank algorithm [92] of *Google* is an example of a successful application of the Eigenvector centrality measure.

**Betweenness measure for edges.** This is a measure of how important an edge  $e$  is in the graph according to the participation of  $e$  to some process running on the graph. The measure was proposed by Girman and Newman [88] and has become a well-known method to detect communities in social networks. The simplest definition of the measure is based on the number of shortest paths between any pair of nodes that go through a particular edge. Using the notations defined for the betweenness centrality measure for nodes, one can formalize the betweenness centrality for an edge  $e$  as

$$Between_c(e) \triangleq \sum_{u \neq v} \frac{\sigma(u, v, e)}{\sigma(u, v)} \times \frac{2}{(|V|) \times (|V| - 1)} \in [0, 1], \quad (2.6)$$

where  $\sigma(u, v, e)$  is the number of shortest paths between nodes  $u$  and  $v$  that go through  $e$ .

### 2.3.3 Other Measures and Definitions

**Clustering coefficient.** This is a measure to assess how likely nodes in a graph tend to connect to each other. It is defined based on the number of triangles and the number of triples formed by nodes and edges in the graph. A triangle is a complete subgraph of three nodes all connected to each other whereas a triple is a connected subgraph of three nodes. This measure can be employed either for a node or for the whole graph. The clustering coefficient for a node  $v$ , referred to as *local clustering coefficient*, is the likelihood that two adjacent nodes of  $v$  are adjacent as well [122]. It is computed as the number of edges connecting adjacent nodes of  $v$  divided by the number of possible edges between such nodes. Assume that  $v$  has degree of  $d$  and there exist  $n$  edges among these  $d$  nodes, then the clustering coefficient for  $v$  is  $2 \times n / (d \times (d - 1))$ . Using the triangle and triple notations, the local clustering coefficient is computed as

$$cluster_{coeff}(v) \triangleq \frac{triangle(v)}{triple(v)} \in [0, 1], \quad (2.7)$$

where  $triangle(v)$  and  $triple(v)$  are the number of triangles and the number of triples formed by  $v$  and its adjacent nodes, respectively. When applied to the whole graph, the measure is called *global clustering coefficient* and is computed as follows.

$$cluster_{coef}(\mathcal{G}) \triangleq \frac{3 \times \text{number of triangles}}{\text{number of triples}} \in [0, 1] \quad (2.8)$$

**Graph density.** The density of a graph  $\mathcal{G} = \langle V, E \rangle$  is measured as the proportion of the number of edges in  $\mathcal{G}$  to the maximum possible number of edges. For an undirected graph, the maximum number of edges is  $|V| \times (|V| - 1)/2$ . Therefore, the density of  $\mathcal{G}$  is computed as follows.

$$\delta(\mathcal{G}) \triangleq \frac{2 \times |E|}{|V| \times (|V| - 1)} \quad (2.9)$$

The value of  $\delta(\mathcal{G})$  ranges from 0 to 1. A larger value of  $\delta(\mathcal{G})$  indicates that the graph is more cohesive.  $\delta(\mathcal{G}) = 0$  if there is no edge in the graph, and  $\delta(\mathcal{G}) = 1$  if every node in the graph is adjacent to all other nodes. Density measure is the basic guideline for the formalization of community structures in graphs, which will be discussed in Section 2.4.

**Path and Diameter.** A path connecting node  $u$  to node  $v$  in a graph is a sequence of distinct nodes ( $u = v_1, v_2, \dots, v_k = v$ ) such that from each node in the sequence there is an edge to the next node. The length of a path is the number of edges along the path. If there exists a path between two nodes then these two nodes are *reachable* from each other. The shortest path between two nodes is called the *geodesic* between them. The diameter of a graph is the length of the longest geodesic between any two nodes in the graph. One can also measure the diameter of a subgraph as the longest geodesic between any pair of nodes within the subgraph.

**Connectivity of graph.** A graph is connected if there exists a path between any pair of nodes in the graph, otherwise it is disconnected. A disconnected graph is formed by different *components* where each component is a maximal subgraph whose nodes are reachable.

## 2.4 Finding Communities in Graph

This section discusses important models and algorithms developed for extracting community structures from a graph. We first present approaches to defining community structures in Section 2.4.1 and then summarize the methods that rely on graph clustering algorithms to detect communities in Section 2.4.2.

### 2.4.1 Community Structures

Even though several approaches have been developed for detecting community structures in a graph, no universal definition of communities is accepted. As a matter of fact, the definition of community is subjective to the application under consideration and thus rather

depends on the algorithm(s) employed. Nevertheless, the underlying idea of any model for extracting communities from a graph is that edges connecting nodes within a community are sufficiently denser than those connecting nodes in the community to other nodes of the graph [5, 19, 121]. The idea can be mathematically described as follows.

Given a graph  $\mathcal{G} = \langle V, E \rangle$  and a subgraph  $\mathcal{G}_s = \langle V_s, E_s \rangle$  of  $\mathcal{G}$ , one can measure the *internal* degree  $d_{int}(v)$  and *external* degree  $d_{ext}(v)$  of a node  $v \in \mathcal{G}_s$  as the number of edges connecting  $v$  to nodes in  $\mathcal{G}_s$  and to other nodes in the rest of the graph, respectively. The internal degree  $d_{int}(\mathcal{G}_s)$  and external degree  $d_{ext}(\mathcal{G}_s)$  of subgraph  $\mathcal{G}_s$  are then computed as the sum of internal degrees and external degrees of all nodes in  $\mathcal{G}_s$ , respectively. Based on such measures, the *internal density*  $\delta_{int}(\mathcal{G}_s)$  of subgraph  $\mathcal{G}_s$  is derived as half of the internal degree of  $\mathcal{G}_s$  normalized by the number of all possible internal edges, determined as follows.

$$\delta_{int}(\mathcal{G}_s) \triangleq \frac{d_{int}(\mathcal{G}_s)}{2} \times \frac{2}{|V_s| \times (|V_s| - 1)} = \frac{d_{int}(\mathcal{G}_s)}{|V_s| \times (|V_s| - 1)} \quad (2.10)$$

Similarly, the external density of subgraph  $\mathcal{G}_s$  is measured as the external degree of  $\mathcal{G}_s$  normalized by the maximum number of external edges of  $\mathcal{G}_s$ , computed as follows.

$$\delta_{ext}(\mathcal{G}_s) \triangleq \frac{d_{ext}(\mathcal{G}_s)}{|V_s| \times (|V| - |V_s|)} \quad (2.11)$$

As presented in the previous section, the density  $\delta(\mathcal{G})$  of graph  $\mathcal{G}$  is defined as the number of edges in  $\mathcal{G}$  normalized by the number of all possible edges created from nodes of  $\mathcal{G}$ , i.e.,  $\delta(\mathcal{G}) = \frac{2 \times |E|}{|V| \times (|V| - 1)}$ . Assume that subgraph  $\mathcal{G}_s$  exhibits a community structure in graph  $\mathcal{G}$ , then one expects that the internal density of  $\mathcal{G}_s$  is reasonably larger than both the density of  $\mathcal{G}$  and the external density of  $\mathcal{G}_s$  itself. In addition,  $\mathcal{G}_s$  must be a connected subgraph because such a connectivity reflects the relationships between entities in a community. That is, any member in a community must be reachable from other members. Graph clustering algorithms try to partition a graph into subgraphs or communities to achieve a *best trade-off* between a large internal density and a small external density of subgraphs discovered [37]. Density measures (i.e., internal density and external density) are the principles of any approach for detecting communities in graphs.

A community structure in a graph can be generally defined as a maximal subgraph whose nodes are, to some extent, *strongly connected*. The maximal subgraph is in the sense that no more nodes and incident edges can be added to the subgraph so that it still has the strongly connected property defined. There are different definitions of a community structure realizing this general guideline. The most cohesive community structure is a maximal clique where all nodes are connected to each other. The simplest instance of a clique is a triangle structure, which often appears in graphs. Larger cliques, however, are not very likely in real-world applications due to the strict constraint employed. For example, any person in a friendship community has to have a friend relationship to all other persons in the community. Moreover, the degree of nodes in a clique increases as the size of the clique

increases. Therefore, by defining a community as a clique structure, methods to collect data might affect the result of analysis, too. For example, one cannot find any clique community that has more than 3 persons in a friendship network if during the step of collecting data each person is asked to list a maximum of 3 closest friends [121].

**Definition 2.3 (Clique structure)** *Given a graph  $\mathcal{G} = \langle V, E \rangle$ , a clique is a complete subgraph  $\mathcal{G}_c = \langle V_c, E_c \rangle$  of  $\mathcal{G}$ .  $\mathcal{G}_c$  is called a maximal clique if there exists no complete subgraph  $\mathcal{G}'_c = \langle V'_c, E'_c \rangle$  of  $\mathcal{G}$  such that  $V_c \subset V'_c$ .*

In the context of social network analysis, a clique community is a group of users where each user has relationships to all other users in the community. A clique structure forms a perfect community in terms of social links among users. However, as stated above, it is unlikely to observe large clique communities in social networks. Finding cliques in a graph is an **NP**-complete problem [13, 18].

It is possible to give some exceptions to relax the clique constraint so that community structures derived are clique alike. A typical method is to rely on the concept of reachability of nodes in a community structure. In particular, a predefined constraint is employed so that there exists a path with a *limited length* between any pair of nodes in a community. Examples of clique alike community structures include  $n$ -clique [5, 77],  $n$ -clan, and  $n$ -club [84]. Recently, Palla et al. introduced a concept called  $k$ -clique community or *clique chain* community [93]. The main advantage of this model is that it allows to relax the clique constraint and to find overlapping communities as well.

**Definition 2.4 ( $k$ -clique community)** *Given a graph  $\mathcal{G} = \langle V, E \rangle$ , a  $k$ -clique community structure is defined as a union of all adjacent size- $k$  cliques where the adjacency means that two size- $k$  cliques share  $k - 1$  nodes.*

It is noted that nodes in a  $k$ -clique community form local cliques. This feature is more likely in social networks, where a community is formed by many users among which there are subgroups whose members are completely linked. Details of the algorithm to detect  $k$ -clique communities will be discussed in Section 2.4.2. Some other local connectivity features in graphs that can be used to identify communities are the density-connected structure and the star structure.

**Definition 2.5 (Density-connected structure)** *Given a graph  $\mathcal{G} = \langle V, E \rangle$ , a density-connected structure is a subgraph  $\mathcal{G}_d = \langle V_d, E_d \rangle$  of  $\mathcal{G}$  where nodes in  $V_d$  are linked by edges in  $E_d$  to form a density-connected cluster with respect to the neighbor relationship of nodes.*

The density-connected cluster defined by Ester et al. [34] is understood, in graph terms, as a subgraph constituted by some dense subgraphs linked through some sparse ones. One might consider a density-connected structure as a general model for relaxing the clique constraint. The model is used to detect communities where links between users in a community do not necessarily form a *spherical shape*. In other words, it allows many users in

a community not to have a direct link to each other. The DBSCAN algorithm used to find density-connected communities will be discussed in Section 2.4.2. The star structure defined as following is another specific structure of interest in detecting communities and finding social influence users in social networks.

**Definition 2.6 (Star structure)** *Given a graph  $\mathcal{G} = \langle V, E \rangle$ , a star structure is a subgraph  $\mathcal{G}_s = \langle V_s, E_s \rangle$  of  $\mathcal{G}$  such that there is a node in  $V_s$  called “center node” that has neighbor relationships to all other nodes.*

Star structures are found in many applications. Examples include the communities observed under the advisor-advisee relationship where the advisor knows all his/her students who might not know each other, or a community formed by *Facebook* users where a user has many friends. Extracting star communities is based on the degree centrality of nodes in the graph. Given a neighbor threshold  $k$ , a node that has at least  $k$  adjacent nodes in the graph together with its neighbors form a star community.

## 2.4.2 Graph Clustering Approaches

Detecting community structures in a graph can be generally considered a clustering problem. It is to arrange (data) points in a dataset into different groups where points within a group are more *similar* or *closer* to each other than those in different groups [47]. The similarity or closeness between points are identified based upon the application under consideration.

**Partitioning approach.** Partitioning approach is the simplest and most fundamental way in clustering data. The idea is to assign points to a given number of  $K$  clusters,  $C = \{c_1, c_2, \dots, c_K\}$ , so that an objective function computed as the sum of the distances from points to the *centroid* of the corresponding cluster is minimal. In graph terms, the algorithm is initialized by selecting  $K$  nodes to be the centroids of clusters and then it performs a number of iterations to refine the solution. At each step of iterations, each node is assigned to the cluster whose centroid is the closest one to that node compared to other centroids; the centroid of each cluster is then recomputed based on nodes assigned in the cluster. After a number of iterations, the structures of clusters become stable and no new assignment for nodes is needed. The most popular implementation of partitioning methods is  $k$ -means clustering [78]. The algorithm employs a *squared error (SE)* as an objective function to identify the convergence for a clustering solution. The  $SE$  function is computed as the total of *intra-cluster* distances, determined as follows.

$$SE \triangleq \sum_{i=1}^K \sum_{v \in c_i} dist(v, centroid_i)^2 \quad (2.12)$$

The result of the  $k$ -means algorithm is not a global optimum because the algorithm often terminates at a local optimum solution. In addition, the initialization of centroids strongly affects the result. A typical method to improve the result of  $k$ -means clustering

is to select initial centroids such that they are as far as possible from each other, and to run the algorithm multiple times and choose the best solution based on a quality measure, for example, the modularity discussed below. There are variants of  $k$ -means for clustering graphs such as the algorithms developed in [53, 101]. Another method following the partitioning strategy for clustering graphs is to minimize the number of edges connecting nodes from different clusters. A set of edges that connect nodes of two clusters are called *cut size*. Kernighan and Lin follow this partitioning direction and propose a graph clustering algorithm in [63]. A label propagation-based method introduced by Raghavan et al. [99] is another algorithm to partition a graph into a number of community structures.

**Hierarchical clustering.** Hierarchical clustering methods work by creating a tree structure representing a clustering solution. In other words, clusters are formed in a hierarchical manner. There are two categories of hierarchical clustering algorithms, namely agglomerative and divisive.

**Agglomerative approach.** This approach uses a bottom-up strategy to build clusters. At the beginning, each node in the graph is considered a cluster. The clustering process works through a number of merging two *closest* clusters. There are different strategies to measure how close two clusters are such as *single linkage*, *complete linkage*, and *average linkage* measures. CHAMELEON is an agglomerative hierarchical clustering algorithm that is widely used [62].

**Divisive approach.** Divisive clustering methods employ a top-down strategy to divide nodes of a graph into clusters. At the beginning, the whole graph is considered a *root* cluster. The algorithm works through a number of iterations. At each step, a search for a best cut size in the clusters is applied and the cluster that contains the identified cut size is split by removing the cut size edges. The divisive algorithm proposed by Newman and Girvan [88], which employs the edge betweenness measure to split a graph, is one of the most successful algorithms applied to detect community structures in a graph.

**Density-based clustering.** Density-based clustering methods aim at finding clusters that do not have spherical-shapes as the clusters discovered by partitioning approaches. Strategically, a cluster is generally considered a *dense region* that is surrounded by areas having a lower density of objects. Here, the density is defined as the number of *neighbors* or, in graph terms, the number of adjacent nodes. DBSCAN [34] is a well-known algorithm for detecting density-connected clusters.

The underlying idea of the DBSCAN algorithm is that a cluster is derived from extending small dense regions, where the density is measured based on the number of neighbors of an object, given a neighbor relation  $R$ . The basic dense unit is determined by the concept of core objects, those having a number of neighbors larger than some threshold *minPts*. Particularly,  $o$  is a core object if  $|\mathcal{C}(o, R)| > \text{minPts}$ , where  $\mathcal{C}(o, R)$  is the set of objects in the neighborhood of object  $o$  under the neighbor relation  $R$ . Objects in  $\mathcal{C}(o, R)$  of a core object  $o$  are called *directly density-reachable* from  $o$ . DBSCAN creates a new cluster by



adding an unvisited core object and its neighbors to initialize the cluster. It then iterates to add unclustered objects that are directly density-reachable from some (core) object in the cluster until no more objects can be added. The clustering process stops when all objects are visited. It is easy to apply density-based clustering methods to find community structures in a graph. This is basically done by considering the adjacency of nodes as the neighbor relation defined in DBSCAN. A node  $v_i$  in a graph is a core node if it has more than  $minPt$  adjacent nodes.

**Overlapping community detection.** Partitioning, hierarchical, and density-based clustering approaches find exclusive communities in a graph, meaning that one node in the graph can only belong to one community. However, in real-world applications, especially in social works, one user might be a member of different communities at the same time. Therefore, detecting communities that share members is necessary. Palla et al. [93] introduce a *Clique Percolation Method* to find overlapping community structures in a graph. The model is based on the idea that nodes within a community might not be necessary to form a clique rather they form local cliques. The authors propose two concepts called *k-clique* (i.e., a clique of  $k$  nodes) and *k-clique chain* that contains a chain of adjacent  $k$ -cliques. Here, the adjacency indicates two  $k$ -cliques sharing  $k-1$  nodes. Two  $k$ -cliques are connected if they are part of a  $k$ -clique chain. Having these two concepts defined, a so called *k-clique community* in a graph is formalized as a connected subgraph formed by the union of all  $k$ -cliques that are connected.

The first step to find  $k$ -clique communities in a graph is to extract a set of all maximal cliques,  $CL = \{cl_1, cl_2, \dots, cl_N\}$ . The second step is to build a matrix  $\mathbf{A}$  of size  $N \times N$  representing the overlap between maximal cliques. Each entry of matrix  $\mathbf{A}$  records the number of common nodes shared by the two cliques indicated by the corresponding row and column of the entry. The next step is to extract  $k$ -clique communities from matrix  $\mathbf{A}$ , which is performed by two sub-steps as follows: 1) erasing every off-diagonal entry that is smaller than  $k-1$  and every diagonal entry smaller than  $k$  and replacing the remaining entries by 1; 2) finding connected components of the graph represented by such an adjacency matrix  $\mathbf{A}$ . As a result, each derived component is a  $k$ -clique community. Other methods for detecting overlapping communities in a graph are discussed in a comprehensive survey by Xie et al. [125].

**Quality functions.** A quality function is a quantitative measure to assess how good a clustering solution discovered by a clustering algorithm is. Finding a clustering solution that maximizes the quality function is the final goal of clustering data, in general. In the context of extracting community structures from a graph, an algorithm, e.g., the hierarchical clustering, might return a number of clustering solutions, and one needs a quality measure to identify which is the best one. Some popular quality functions employed in graph clustering are summarized below.

**Performance.** The performance [115] is derived from the number of pairs of nodes that seem to be *correctly* clustered into communities. That is, the number of *pairs of nodes* that are connected by an edge and are clustered into the same communities, and the number of *pairs of nodes* that are not connected by an edge and are clustered into different communities. Assume  $C_G = \{c_1, c_2, \dots, c_K\}$  is a clustering solution and  $E(c_i)$  denotes the number of edges within a community  $c_i$ . The performance function  $f(C_G)$  is then formalized as follows.

$$f(C_G) \triangleq \frac{\sum_{i=1}^K (E(c_i) + \sum_{j>i} |(u,v) \notin E| u \in c_i, u \in c_j)}{|V| \times (|V| - 1)/2} \quad (2.13)$$

The defined measure assumes an unweighted graph, but there are also variants for weighted graphs introduced in [16]. Values of  $f(C_G)$  range from 0 to 1, and a higher value indicates that detected communities are both internally dense and externally sparse and, therefore, a better clustering solution. However, when the performance measure is applied to complex networks, which tend to be sparse in nature, it is possible that the second term in the numerator of  $f(C_G)$  becomes so large. As a result, it will dominate all other factors in the formula and gives a high score indiscriminately [6].

**Modularity.** The modularity concept proposed by Newman and Girman [87, 88] is known the best quality function to date. Given a clustering solution  $C_G = \{c_1, c_2, \dots, c_K\}$ , the modularity function  $Q(C_G)$  is defined as

$$Q(C_G) \triangleq \sum_{k=1}^K \left( \frac{E(c_k)}{|E|} - \left( \frac{D(c_k)}{2 \times |E|} \right)^2 \right), \quad (2.14)$$

where  $E(c_i)$  is the number of edges connecting nodes within community  $c_i$  and  $D(c_i)$  is the sum of the degrees of nodes in  $c_i$ . The first term in  $Q(C_G)$  indicates the internal density of a detected community  $c_i$  while the second term is the expected internal density of  $c_i$  obtained from a random graph having the same node degrees as graph  $\mathcal{G}$ . The idea is that a random graph exhibits no community structures. Therefore, the first term is often greater than the second term, and one expects a clustering solution  $C_G$  that has the highest modularity measure compared to other clustering solutions. Note that by definition there are cases where the modularity measure has a negative value [37]. Even though the modularity measure is widely adopted, it suffers from a resolution limit meaning that it might merge two connected communities  $c_1$  and  $c_2$  in case  $D(c_1) \times D(c_2) < 2 \times |E|$  [36, 38]. In addition to the performance and modularity, there are other measures proposed for assessing the quality of a clustering solution. Examples include the *coverage* and *conductance*. For a more detailed discussion of quality functions, we refer the reader to [40].

To close this section, it is noted that community structures and techniques for discovering such structures discussed so far are defined and developed solely based on link structures in a graph. Employing such approaches to detecting communities of users in a social network therefore returns so-called *link-based* communities.

## 2.5 Probabilistic Models for Discovering Communities

Another strategy for extracting communities from social networks is to apply Bayesian statistics methods to learn (hidden) communities from not only link structures but also from other features describing users. There are several probabilistic models introduced to explain a social network in a way that the observed data are generated by users belonging to some communities. Generally, a probabilistic model consists of a number of random variables including both observed and hidden ones, among them there are variables depending on each other. The value of a particular variable in the model is assumed to be drawn from a specified probability distribution. The dependency defined by the conditional probability distributions between random variables forms a joint probability distribution of the model, which is normally represented by a graphical model. In the context of detecting communities in a link graph, one can think of having a model in which observed variables represent links of users and hidden variables are the assignments of users to communities [106].

There are two main advantages of the probabilistic modeling approaches for extracting communities of users from a social network. First, one can add different types of observed information associated with users (e.g., links, messages) to the model so that the communities detected become more meaningful. Such communities, therefore, are often referred to as *feature-based* communities. Second, the membership of a user in a community is modeled as a probability measure. This means a user can be a member of multiple communities, which is more realistic in practical applications.

This section presents the basic concepts and background for the development of a probabilistic model and summarizes recent studies that employ probabilistic modeling approaches for extracting communities. In Section 2.5.1, we briefly give an overview of probability theory and Bayesian statistics as far as necessary for later presenting the models developed in this dissertation. The two important concepts, namely the exchangeability and conjugacy prior are discussed in Section 2.5.2 and Section 2.5.3, respectively. We then describe the graphical model and Gibbs sampling method in Sections 2.5.4 and 2.5.5. The related work applying probabilistic models for community extraction is summarized in Section 2.5.6.

### 2.5.1 Random Variables and Bayes' Theorem

We review in this section some definitions of probability theory, which are the underlying fundamentals for building a probabilistic model.

**Definition 2.7 (Sample Space)** *A set  $\Omega$  of all possible outcomes of a probabilistic experiment is called the sample space of the experiment. Each element  $\omega \in \Omega$  is called a sample outcome.*

A typical example of a sample space is the set of all possible outcomes if two coins are tossed, which gives  $\Omega = \{HH, HT, TH, TT\}$ , where  $H$  indicates a head and  $T$  indicates a tail of a coin.

**Definition 2.8 (Random Variable)** A random variable is a mapping  $X : \Omega \rightarrow \mathbb{R}$  that assigns a real number to each outcome  $\omega$  of a sample space  $\Omega$ .

An example random variable defined on  $\Omega = \{HH, HT, TH, TT\}$  is “number of heads observed”, which can be presented  $X(HH) = 2, X(HT) = X(TH) = 1$ , and  $X(TT) = 0$ . The probability that a random variable  $X$  has a value  $x$  is denoted  $P(X = x)$  or  $P(x)$ . If  $X$  is a discrete random variable then  $\sum_{x \in X} P(x) = 1$  whereas the summation is replaced by the integral if  $X$  is a continuous random variable, i.e.,  $\int_{x \in X} P(x) dx = 1$ .

**Probability density/mass function.**

A function  $f(x)$  that assigns probabilities for all outcomes of a sample space  $\Omega$  or consequently the values of a random variable  $X$  defined on  $\Omega$  is called a *probability density function* if  $X$  is a continuous and is called a *probability mass function* if  $X$  is discrete. In both cases,  $f(x)$  is denoted *pdf* and it must satisfy the following properties.

$$\begin{cases} \sum_{x \in X} f(x) = 1 & \text{if } X \text{ is discrete} \\ \int_{x \in X} f(x) dx = 1 & \text{if } X \text{ is continuous} \\ f(x) \geq 0 \end{cases} \quad (2.15)$$

If a random variable  $X$  is presented by a *pdf*  $f(x)$  then one normally says that values of  $X$  are generated from a probability distribution defined by the *pdf*  $f(x)$  or  $X$  is distributed under the distribution defined by the *pdf*  $f(x)$ . The notation  $P(X)$  denotes the probability distribution over the random variable  $X$ .

There are two important statistics for summarizing a probability distribution, the expectation and the variance. Given a random variable  $X$  whose values are distributed under a probability distribution defined by  $f(x)$ , then the expectation of  $X$ , denoted  $\mathbb{E}[X]$ , is the *weighted* average of the values of  $X$  drawn from  $f(x)$ .

$$\mathbb{E}[X] = \begin{cases} \sum_{x \in X} x f(x) & \text{if } X \text{ is discrete} \\ \int_{x \in X} x f(x) dx & \text{if } X \text{ is continuous} \end{cases} \quad (2.16)$$

The variance of a random variable  $X$ , denoted  $Var(X)$ , is a measure of the dispersion of the values of  $X$  around the expectation.

$$Var(X) = \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \quad (2.17)$$

Two random variables  $X$  and  $Y$  are independent if for every  $x \in X$  and  $y \in Y$  we have  $P(x, y) = P(x)P(y)$ . Otherwise, the two variables are known as dependent. The notation  $P(X, Y)$  is called the joint probability distribution of  $X$  and  $Y$ . In case  $X$  and  $Y$  are dependent, the notation  $P(x|y)$  is used to denote the probability that  $X$  has a value  $x$  given that  $Y$  has a value  $y$ . The joint probability of  $X = x$  and  $Y = y$  is then determined by the product rule  $P(x, y) = P(x|y)P(y) = P(y|x)P(x)$ . The notation  $P(X|Y)$  is called conditional probability distribution of  $X$  given  $Y$  and, thus, we represent the joint

probability distribution of two dependent random variables  $X$  and  $Y$  as follows.

$$P(X, Y) = P(X|Y)P(Y) = P(Y|X)P(X) \quad (2.18)$$

Given three random variables  $X$ ,  $Y$  and  $Z$ , the two variables  $X$  and  $Y$  are said conditionally independent given  $Z$  if  $P(X, Y|Z) = P(X|Z)P(Y|Z)$ . This means if one has information about  $Z$  that both  $X$  and  $Y$  depend on, knowing value of  $X$  or  $Y$  does not change her knowledge about the another variable [11].

**Bayes' theorem.** Having the above basic notations defined, the *Bayes' theorem*, which is the foundation of Bayesian statistics, stated for two random variables  $X$  and  $Y$  is as follows.

$$\underbrace{P(X|Y)}_{\text{Posterior}} = \frac{\overbrace{P(Y|X)}^{\text{Likelihood}} \overbrace{P(X)}^{\text{Prior}}}{\underbrace{P(Y)}_{\text{Evidence}}} \quad (2.19)$$

In Bayesian perspective, the above equation is interpreted as a process to update knowledge about  $X$  using some prior information together with some evidence related to  $X$ . Specifically, the posterior distribution  $P(X|Y)$  (e.g., the probability that  $X$  has a value  $x$  if we see that  $Y$  has a value  $y$ ) is computed from the likelihood function  $P(Y|X)$  (e.g., how likely that  $Y$  has a value  $y$  given that  $X$  has a value  $x$ ) and the prior distribution  $P(X)$  (e.g., the probability that  $X$  has a value  $x$ ). The denominator  $P(Y)$  called the marginal distribution of  $Y$  (e.g., the total probability that  $Y$  has a value  $y$ ) is the normalizing constant to ensure that  $P(X|Y)$  is a probability density function.

By employing a statistical modeling approach to analyzing data, a given dataset consisting of *data points* (also called observations)  $\mathcal{D} = \{x_1, x_2, \dots, x_N\}$  is assumed to be generated from some probability distribution having (unknown) parameter(s)  $\theta$ . Such an assumption is represented by a likelihood function  $P(\mathcal{D}|\theta)$ . Even though  $\theta$  is unknown, one can give some prior knowledge to the model by considering that the values of  $\theta$  are generated by some distribution  $P(\theta; \alpha)$ , where  $\alpha$  is known-value parameter(s) called hyperparameter<sup>8</sup>. This is the underlying key idea of Bayesian statistics approach compared to classical statistics where the parameter  $\theta$  is assumed to have a fixed value. The joint distribution of the observed data and the parameters defines a probabilistic model.

$$P(\mathcal{D}, \theta; \alpha) = P(\mathcal{D}|\theta)P(\theta|\alpha) \quad (2.20)$$

Thus, under Bayesian statistics point of view, both the dataset  $\mathcal{D}$  and the parameter  $\theta$  are considered random variables. One can, therefore, apply *Bayes' theorem* to compute the posterior distribution of the parameter  $\theta$  as follows.

---

<sup>8</sup>In this thesis, the semicolon (;) is used to separate unknown parameters and hyperparameters and, therefore,  $P(\theta; \alpha)$  is understood  $P(\theta|\alpha)$  when  $\alpha$  is a hyperparameter.

$$P(\theta|\mathcal{D}; \alpha) = \frac{P(\mathcal{D}|\theta)P(\theta|\alpha)}{P(\mathcal{D}|\alpha)} \quad (2.21)$$

It is intuitive that one can again model  $\alpha$  as to be generated by some distribution having possibly unknown parameters. This leads to a hierarchical Bayesian model representing the underlying generative process of how the dataset  $\mathcal{D}$  has been produced under the defined distributions of the variables in the model. All parameters in a probabilistic model except hyperparameters and variables representing observed data are called *hidden variables*.

By integrating both sides of Eq. 2.21 with respect to  $\theta$ , the marginal distribution  $P(\mathcal{D}|\alpha)$  of the dataset  $\mathcal{D}$  can be represented in terms of the likelihood function  $P(\mathcal{D}|\theta)$  and the prior distribution  $P(\theta|\alpha)$ .

$$P(\mathcal{D}|\alpha) = \int_{\theta} P(\mathcal{D}|\theta)P(\theta|\alpha)d\theta \quad (2.22)$$

In addition to the computation of the posterior distribution of the parameters in the model for explaining the observed data in the dataset  $\mathcal{D}$ , one can also derive a prediction for a new coming observation. Specifically, the joint probability of a new observation  $x_{new}$  and the parameter  $\theta$  given the observed data in the dataset  $\mathcal{D}$  is computed as follows.

$$P(x_{new}, \theta|\mathcal{D}; \alpha) = P(x_{new}|\theta)P(\theta|\mathcal{D}; \alpha) \quad (2.23)$$

By integrating over the parameter  $\theta$ , the probability of a new data point given the previous ones is computed.

$$P(x_{new}|\mathcal{D}; \alpha) = \int_{\theta} P(x_{new}|\theta)P(\theta|\mathcal{D}; \alpha)d\theta \quad (2.24)$$

The underlying principle allowing to build a probabilistic model for learning hidden structures in an observed dataset comes from the *De Finetti's theorem*, which is derived from a concept called *exchangeability* presented in the next section.

### 2.5.2 Exchangeability

The exchangeability concept is used to indicate the invariant of the joint probability distribution of a number of random variables with respect to the order of the variables. That is,  $N$  random variables  $X_1, X_2, \dots, X_N$  are said to be *exchangeable* if every permutation, or reordering, of their indices does not change the joint probability distribution. This is represented as

$$P(X_1, X_2, \dots, X_N) = P(X_{\pi(1)}, X_{\pi(2)}, \dots, X_{\pi(N)}) \quad (2.25)$$

for every permutation  $\pi$  on  $\{1, \dots, N\}$ .

This definition is also extended to an infinite number of random variables, stating that  $X_1, X_2, \dots, X_N, \dots$  are infinitely exchangeable if any finite subsequence of such variables is

exchangeable [9]. One important result of the exchangeable property, as shown in the following *De Finetti's theorem*, is that exchangeable observations (i.e., observed data points in a dataset) can always be represented by a probabilistic model where the observations are generated from a distribution having some parameter  $\theta$  whose values are again distributed under some prior distribution.

**Theorem 2.1 (De Finetti's theorem)** *For any infinitely exchangeable sequence of random variables  $X_1, X_2, \dots, X_N, \dots$ ,  $X_i \in \mathcal{X}$ , there exists some space  $\Theta$ , and a corresponding density function  $P(\theta)$ ,  $\theta \in \Theta$ , such that the joint probability of any  $N$  observations has a mixture representation:*

$$P(X_1, X_2, \dots, X_N) = \int_{\Theta} P(\theta) \prod_{i=1}^N P(X_i|\theta) d\theta \quad (2.26)$$

The original proof of the *De Finetti's theorem* for infinite binary-value exchangeable random variables dates back to the 1930's, see [50] for a proof of that case and [9, Section 4.5] for generalizations and further references.

As an example to demonstrate the *De Finetti's theorem*, we assume that each  $X_i$  can take one of  $K$  discrete values, i.e.,  $\mathcal{X}$  is a  $K$ -dimensional discrete space, then one can choose  $\Theta$  as a  $K - 1$  simplex space, i.e., for any  $\theta = \langle \theta_1, \dots, \theta_K \rangle \in \Theta$  then  $\sum_{i=1}^K \theta_i = 1$ , and the *Dirichlet* distribution is chosen as the prior distribution of  $\theta$  [112].

### 2.5.3 Conjugate Prior

There are two leaning problems regarding a probabilistic model presented in Eq. 2.20 for an observed dataset. These include the estimation of the parameter(s)  $\theta$  to best explain the underlying patterns in the dataset (Eq. 2.21), and the prediction for a new observation (Eq. 2.24). As Bayesian approach computes the posterior distribution of parameters and uses some statistics (e.g., the expectation and variance) of the derived distribution as the estimation quality or confidence of the parameters, it is required to marginalize (i.e., to compute the summation or the integral) over the whole of parameter space, which often becomes quite difficult. The common strategy to get the computation tractable and also to build a framework for prediction is to employ *conjugate prior* distributions. A probability distribution  $P(\theta|\alpha)$  is called conjugate prior of a likelihood function  $P(\mathcal{D}|\theta)$  if the posterior distribution  $P(\theta|\mathcal{D}; \alpha)$  has the same functional form as the prior. A detailed discussion of the existence of a prior distribution for a likelihood function built from a probability density in *exponential family* probability distributions is presented in [11, Section 2.4].

In a probabilistic model, the likelihood function represents our view about the observed dataset (i.e., from which distribution the dataset is generated), which is fixed under the application. Therefore, one tries to seek a prior distribution that is conjugate to the defined likelihood. For a further explanation, we represent the posterior distribution of the

probabilistic model in Eq. 2.20 as follows.

$$P(\theta|\mathcal{D}; \alpha) = \frac{P(\mathcal{D}|\theta)P(\theta|\alpha)}{\int_{\theta} P(\mathcal{D}|\theta)P(\theta|\alpha)d\theta} \quad (2.27)$$

The underlying principle of using a conjugate prior to the likelihood is that it makes the calculation of the integral in the denominator (i.e., the marginal distribution of the dataset) become simple. In particular, each product  $P(\mathcal{D}|\theta)P(\theta|\alpha)$  returns an expression of the same form as of the prior distribution with the information from the dataset  $\mathcal{D}$  added to the hyperparameter  $\alpha$ . Therefore, the denominator is thus the integral of the unnormalized density function of the updated prior distribution over the parameter space. Consequently, this integral results in an inversion of the normalizing constant of the updated prior distribution with respect to the information from the dataset added to the hyperparameters  $\alpha$ . As an example, we consider in the following the conjugacy between the *Dirichlet* distribution and the *Multinomial* distribution, which is used later in this dissertation for extracting feature-based communities from social networks.

**Multinomial variable.** A random variable  $X$  that can take one of  $K$  categorical values, so that  $\mathcal{X} = \{1, \dots, K\}$ , is called a multinomial variable. If we denote the probability that “ $X$  has the value  $k$ ” by a parameter  $\theta_k$  ( $\theta_k \geq 0$  and  $\sum_{k=1}^K \theta_k = 1$ ), then the probability distribution of  $X$  is given as follows [112].

$$P(X|\theta_1, \theta_2, \dots, \theta_K) = \prod_{k=1}^K \theta_k^{\delta(X,k)} \quad \text{where} \quad \delta(X, k) = \begin{cases} 1 & \text{if } X = k \\ 0 & \text{if } X \neq k \end{cases} \quad (2.28)$$

Consider a dataset  $\mathcal{D} = \{x_1, x_2, \dots, x_N\}$ ,  $x_i \in \mathcal{X}$ , generated by taking  $N$  independent trials on the multinomial variable  $X$  defined by  $\theta = \langle \theta_1, \theta_2, \dots, \theta_K \rangle$ , then the likelihood function of the dataset is

$$P(\mathcal{D}|\theta) = \prod_{i=1}^N P(x_i|\theta) = \prod_{i=1}^N \prod_{k=1}^K \theta_k^{\delta(x_i,k)} = \prod_{k=1}^K \theta_k^{\sum_{i=1}^N \delta(x_i,k)} = \prod_{k=1}^K \theta_k^{c_k}, \quad (2.29)$$

where  $c_k$  is the number of data points in the dataset that has the value  $k$ . The likelihood function of a dataset generated as described is the unnormalized *Multinomial* probability distribution [9, 11].

**Dirichlet distribution.** To complete a probabilistic model for the *Multinomial* dataset  $\mathcal{D}$ , we need to specify a prior distribution for the multinomial parameter  $\theta$ . The *Dirichlet* probability distribution is selected because it is conjugate prior to the *Multinomial* distribution. The *Dirichlet* distribution is defined as

$$Dirichlet(\theta|\alpha) = \frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K \theta_k^{\alpha_k - 1}, \quad (2.30)$$



where  $\alpha = \langle \alpha_1, \alpha_2, \dots, \alpha_K \rangle$  is a  $K$ -dimensional hyperparameter and each  $\alpha_k$  is a positive real number indicating the prior belief that one puts on the corresponding component  $\theta_k$  of the multinomial parameter  $\theta$ .

*Dirichlet* distribution has a number of interesting properties. For example, if all components of the hyperparameter  $\alpha$  are assigned a small value (i.e.,  $\sum_{k=1}^K \alpha_k \rightarrow 0$ ) then the distribution can be simplified as in Eq. 2.31, which leads to a phenomenon that the  $\theta$ s with many *zero-components* are heavily favored.

$$Dirichlet(\theta|\alpha) \propto \prod_{k=1}^K \theta_k^{\alpha_k-1} \propto \prod_{k=1}^K \frac{1}{\theta_k} \quad (2.31)$$

The expectation of the *Dirichlet* distribution, i.e., the expectation of a component  $\theta_k$  in  $\theta$ , is computed as follows.

$$\mathbb{E}[\theta_k|\alpha] = \frac{\alpha_k}{\alpha_0} \quad \text{where} \quad \alpha_0 = \sum_{k=1}^K \alpha_k \quad (2.32)$$

**Posterior distribution.** Having the likelihood function (Eq. 2.29) and the *Dirichlet* prior distribution (Eq. 2.31) described, the posterior distribution of the parameter  $\theta$  (Eq. 2.27) is now computed by

$$P(\theta|\mathcal{D}; \alpha) = \frac{\prod_{k=1}^K \theta_k^{c_k} \prod_{k=1}^K \theta_k^{\alpha_k-1}}{\int_{\theta} \prod_{k=1}^K \theta_k^{c_k} \prod_{k=1}^K \theta_k^{\alpha_k-1} d\theta} = \frac{\prod_{k=1}^K \theta_k^{c_k+\alpha_k-1}}{\int_{\theta} \prod_{k=1}^K \theta_k^{c_k+\alpha_k-1} d\theta}. \quad (2.33)$$

By multiplying the denominator of the above equation with 1 represented by

$$\frac{\prod_{k=1}^K \Gamma(c_k + \alpha_k) \Gamma(\sum_{k=1}^K c_k + \alpha_k)}{\Gamma(\sum_{k=1}^K c_k + \alpha_k) \prod_{k=1}^K \Gamma(c_k + \alpha_k)}$$

the denominator becomes

$$\begin{aligned} \int_{\theta} \prod_{k=1}^K \theta_k^{c_k+\alpha_k-1} d\theta &= \frac{\prod_{k=1}^K \Gamma(c_k + \alpha_k) \Gamma(\sum_{k=1}^K c_k + \alpha_k)}{\Gamma(\sum_{k=1}^K c_k + \alpha_k) \prod_{k=1}^K \Gamma(c_k + \alpha_k)} \int_{\theta} \prod_{k=1}^K \theta_k^{c_k+\alpha_k-1} d\theta \\ &= \frac{\prod_{k=1}^K \Gamma(c_k + \alpha_k)}{\Gamma(\sum_{k=1}^K c_k + \alpha_k)} \int_{\theta} \frac{\Gamma(\sum_{k=1}^K c_k + \alpha_k)}{\prod_{k=1}^K \Gamma(c_k + \alpha_k)} \prod_{k=1}^K \theta_k^{c_k+\alpha_k-1} d\theta \\ &= \frac{\prod_{k=1}^K \Gamma(c_k + \alpha_k)}{\Gamma(\sum_{k=1}^K c_k + \alpha_k)}. \end{aligned} \quad (2.34)$$

Finally, the posterior distribution of  $\theta$  is

$$P(\theta|\mathcal{D}; \alpha) = \frac{\Gamma(\sum_{k=1}^K c_k + \alpha_k)}{\prod_{k=1}^K \Gamma(c_k + \alpha_k)} \prod_{k=1}^K \theta_k^{c_k+\alpha_k-1} = Dir(\theta|c + \alpha) \quad (2.35)$$

where  $c = \langle c_1, c_2, \dots, c_K \rangle$ . Thus, the posterior distribution of the parameter  $\theta$  is the *Dirichlet* distribution where the information from the dataset (i.e., the count of the number of data points for each category) is added to the hyperparameter  $\alpha$ . One can now, for example, estimate each component of  $\theta$  using the expectation of the *Dirichlet* distribution.

$$\mathbb{E}[\theta_k | c + \alpha] = \frac{c_k + \alpha_k}{\alpha_0} \quad \text{where} \quad \alpha_0 = \sum_{k=1}^K c_k + \alpha_k \quad (2.36)$$

#### 2.5.4 Graphical Model

One of the challenges in presenting a probabilistic model is that it is hard to explain the joint distribution of all random variables in the model. This is because of a huge number of combinations of the values of variables in the model. Even in the simplest case where the model has  $N$  binary-valued random variables, the joint distribution requires a specification of  $2^N$  numbers - the probabilities of  $2^N$  different assignments of the values of  $X_1, \dots, X_N$ . Graphical model is a language that uses graph notations for intuitively representing a probabilistic model in a compact way and for interpreting the underlying generative process of how the observations in dataset  $\mathcal{D}$  are generated from the model. The main idea of graphical model is to exploit the independent of variables to factor the representation of the model into modular components [64].

There are two main classes of graphical models, which are called Bayesian networks and Markov networks. A Bayesian network is represented by a directed graph and hence it is also called directed graphical model. A Markov network is represented by an undirected graph and is called Markov random fields (MRFs) or undirected graphical model. In the following paragraph, we briefly give some basics of a Bayesian network that will be employed to develop the models in this dissertation. For detailed discussions of graphical models, we refer the reader to [11, 60, 64, 118].

A graphical model for a Bayesian network representing the joint distribution  $P(X_1, X_2, \dots, X_N)$  of random variables  $X_1, \dots, X_N$  is a directed acyclic graph  $\mathcal{G}$ . Nodes of the graph are random variables in the model and each directed edge is created to connect two variables having a conditional (probability) distribution relationship in the factorization of the joint distribution. Specifically, if there is a conditional distribution  $P(X_k | \mathbf{Pa}_{X_k})$  in the factorization of the joint distribution  $P(X_1, X_2, \dots, X_N)$  then for each variable  $X_i \in \mathbf{Pa}_{X_k}$  there is a directed edge connecting  $X_i$  to  $X_k$ . Variables in  $\mathbf{Pa}_{X_k}$  are called parent variables of  $X_k$ . Intuitively, each node  $X_k$  in a graphical model represents the conditional distribution of  $X_k$  given its parent variables. An important property of a graphical model is that it encodes the *local Markov assumption* for random variables in the graph meaning that each variable  $X_k$  in the graph is conditionally independent of its non-descendants given its parent variables [64]. Figure 2.1 shows a graphical model

presenting a probabilistic model consisting of four random variables  $X, Y, Z$  and  $\theta$  where  $X$  and  $Y$  are conditionally independent given  $Z$ , and  $Z$  depends on  $\theta$ .

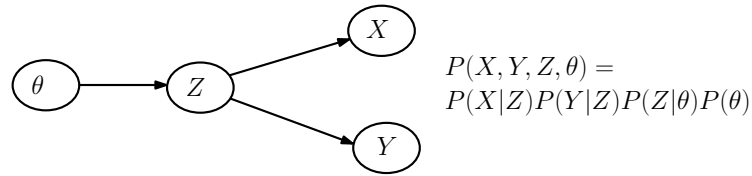


Figure 2.1: A graphical model representing the joint distribution of  $X, Y, Z$ , and  $\theta$  factorized based on the (assumption) dependency between variables:  $P(X, Y, Z, \theta) = P(X|Z)P(Y|Z)P(Z|\theta)P(\theta)$

A graphical model can be represented in a more compact way by using *plate* notations in which several random variables of *the same kind* are shown in the graph by only one representative node with an index and that node is covered by a box labeled with a number indicating the cardinality of such variables [11, Chapter 8]. Another notation used in graphical model is that nodes represented for observed random variables (i.e., variables encode the observed features of data points in a dataset) are shaded. As an example, we consider the joint distribution shown in Eq. 2.37 of the probabilistic model described by *De Finetti's theorem* (Theorem. 2.1) for a finite number of observations  $X_1, X_2, \dots, X_N$ , assuming that the prior distribution for parameter  $\theta$  has some hyperparameter  $\alpha$ . The corresponding graphical models are shown in Figure 2.2.

$$P(X_1, X_2, \dots, X_N, \theta) = P(\theta; \alpha) \prod_{i=1}^N P(X_i|\theta) \tag{2.37}$$

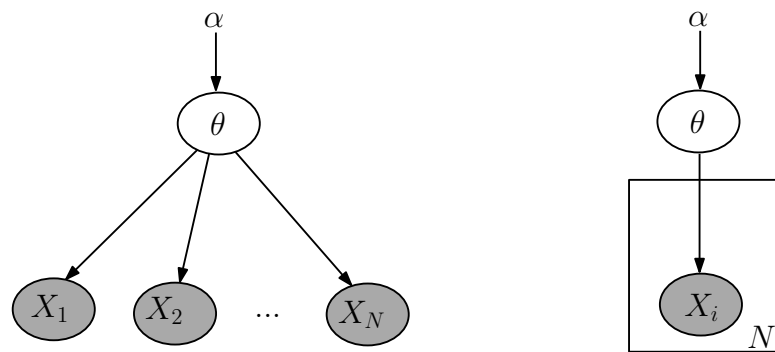


Figure 2.2: Two graphical models representing the same probabilistic model described by *De Finetti's theorem* (Theorem. 2.1) for a finite number of observations  $X_1, X_2, \dots, X_N$ . The graphical model on the right is presented using plate notations.

### 2.5.5 Gibbs Sampling for Posterior Estimation

Computing the posterior distribution of hidden variables, given a dataset and the hyperparameters of the prior distribution of hidden parameters, in a probabilistic model is the main goal for explaining the observed data in the context described by the model. Such a computation is often intractable because of the marginalization, as described above. Note that the integral or summation appears not only in the denominator of Eq. 2.27 but also in the likelihood function  $P(\mathcal{D}|\theta)$  if one is interested in only some hidden variables and, therefore, needs to integrate out the others.

There are three popular strategies to approximate the posterior distribution in a probabilistic model. These include the sampling based on Markov Chain Monte Carlo [80], Expectation Maximization (EM), and variational parameter methods (optimization-based). Gibbs sampling [41], a special form of the Metropolis-Hastings algorithm [48], is discussed in this section as we will employ Gibbs sampling in this dissertation. For further details of the EM and variational parameter methods, we refer the reader to [32, 61].

**Monte Carlo method.** The underlying idea for deriving the posterior distribution of hidden variables is that if such a probability distribution is computed (or is approximated in most of the cases) then one can use typical statistics such as the expectation and the variance of the distribution to summarize the values of hidden variables. Monte Carlo method is based on the idea that one can learn a complex distribution by repeatedly drawing samples from it and empirically summarizing those samples. For example, the expectation of the posterior distribution defined in Eq. 2.27 is analytically derived from

$$\mathbb{E}[\theta|\mathcal{D}; \alpha] = \int_{\theta} \theta P(\theta|\mathcal{D}; \alpha) d\theta \quad (2.38)$$

However, if it is able to produce a *large enough* number of samples  $\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(N)}$  from  $P(\theta|\mathcal{D}; \alpha)$  then one can approximate the expectation of  $\theta$  with respect to the given dataset  $\mathcal{D}$  and the hyperparameter  $\alpha$  by computing the average of such samples.

$$\mathbb{E}[\theta|\mathcal{D}; \alpha] = \int_{\theta} \theta P(\theta|\mathcal{D}; \alpha) d\theta \approx \frac{1}{N} \sum_{i=1}^N \theta^{(i)} \quad (2.39)$$

The variance of  $\theta$  is therefore derived from the approximated expectation.

$$\text{Var}(\theta|\mathcal{D}; \alpha) = \mathbb{E}[\theta^2|\mathcal{D}; \alpha] - \mathbb{E}[\theta|\mathcal{D}; \alpha]^2 \quad (2.40)$$

**Gibbs Sampling.** It is clear that in order to employ the Monte Carlo strategy to summarize a probability distribution one needs to find a method to *correctly* draw samples from that distribution. In our scenario of approximating the posterior distribution  $P(\theta|\mathcal{D}; \alpha)$  of hidden variables, we need to draw  $\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(N)}$  from  $P(\theta|\mathcal{D}; \alpha)$ . Gibbs sampling is one of the algorithms designed to do so. The basic idea of Gibbs sampling is that it produces

a Markov chain of states of hidden variables. The value of a variable at each state is drawn conditionally on the values of other variables.

Assume that we need to draw samples from a distribution  $P(\theta|\mathcal{D};\alpha)$  where  $\theta$  consists of  $K$  hidden variables  $\theta = \{\theta_1, \theta_2, \dots, \theta_K\}$ , then the general schema of a Gibbs sampling for that model is as follows.

---

**Algorithm 1:** A general Gibbs sampling algorithm

---

```

1 /* State initialization */
2  $\theta^{(0)} \leftarrow \theta_1^{(0)}, \theta_2^{(0)}, \dots, \theta_K^{(0)}$ ;
3 /* Markov chain */
4 foreach  $t = 1 \dots T$  do
5   foreach  $i = 1 \dots K$  do
6      $\theta_i^{(t+1)} \sim P(\theta_i | \theta_1^{(t+1)}, \theta_2^{(t+1)}, \dots, \theta_{i-1}^{(t+1)}, \theta_{i+1}^{(t)}, \dots, \theta_K^{(t)}, \mathcal{D}; \alpha)$ 

```

---

It is important to note that samples drawn from a Gibbs sampling algorithm only get to a steady state or converge to the real distribution after a number of iterations called *Burn-in* stage [41]. Therefore, one needs to discard the results obtained from the first *Burn-in* steps before collecting samples for summarizing the distribution.

## 2.5.6 Related Work Employing Probabilistic Models

Several probabilistic models have been introduced to simulate the generation of an observed link graph from which to extract link-based communities, e.g., [52, 133, 134]. Recently, researchers have shifted the attention to not only link structures but also to topical information describing users to extract different types of communities. The main goal is to identify communities where users in a community are related to each other regarding both link structures and common interests. The latter aspect is obtained from analyzing the postings of users. Latent Dirichlet Allocation (LDA) [12] proposed for extracting topics in documents has become a breakthrough for the development of probabilistic models for detecting communities based on this guideline [28, 86, 131, 135, 136]. For example, Wang et al. [120] studied a model of entity relationships and textual attributes to simultaneously discover groups among the entities and topics among corresponding texts. Zhou et al. [136] introduced two Community-User-Topic (CUT) models to extract *e-communities* from an email corpus. In the first model, topics are generated conditionally on a community while in the second model communities are generated conditionally on a topic. In both models, only the recipients of emails are considered to form communities. The Community-Author-Recipient-Topic (CART) [96] is an extension to the Author-Recipient-Topic (ART) [79] model. While ART was developed for extracting topics related to pairs of “a sender and a receiver”, CART adds a community variable for extracting topic-based communities. Similar to CUT, CART was designed to work on an email network where an email is modeled

as a mixture of topics. A recent study by Yin et al. [129] introduced a model to work on text-associated graphs. Their model integrates the generation of links and messages of users. By this, the model is able to extract communities based on both the link structures and topics of users. Sachan et al. [102] proposed a Topic-User-Recipient-Community Model (TURCM) where users are selected into a community based on their so-called type of interactions and topics of interests. In [21], we developed a two-step generative model for discovering *regional* communities. The model first employs the co-occurrences of users in spatio-temporal proximity and then applies topic modeling to the postings of users to extract communities.

## 2.6 Evolving Social Networks and Communities

This section outlines relevant studies that analyze the evolution aspect of social networks and communities. We first give a short discussion about the dynamics of social networks in Section 2.6.1. Approaches developed for analyzing the evolution of communities are then described in Section 2.6.2.

### 2.6.1 Dynamics of Social Networks

Users participating in a social network and interactions among them change over time. These lead to the temporal dynamics of the network. Studies of this aspect of social networks attempt to build models to understand the evolution in terms of network structures including links and interactions among users. The first approach is based on the idea that the development of such structural properties of social networks follows some phenomena. Examples include: 1) the preference attachment [8, 116] stating that users having more connections with other users are more likely to create new links; 2) the random walk mechanism [104], i.e., a user creates a new link to another user by taking a random walk on the network; and 3) the common neighbors [89], i.e., users make friends with whom they share many friends. The second approach explores data of a particular social network from which a model for the evolution of the network is derived. Models based on this approach are therefore considered *explanatory models*. Kumar et al. [66] studied the evolution of Flickr and Yahoo!360 networks. Leskovec et al. [69] analyzed the evolution of evolving graphs built from collaboration networks and recommendation networks. Mislove et al. [82] explored different measures for graphs of Flickr, LiveJournal, Orkut, and Youtube. Recently, Gong et al. [43] proposed a similar study for the Google+ network.

### 2.6.2 Evolution of Communities

In addition to extracting static communities, several models have been introduced to study the evolution of communities regarding changes in the community members over time. Three main approaches have been applied, namely snapshot community matching, evolutionary clustering, and probabilistic models.

The MONIC framework for finding and monitoring cluster transactions was proposed in [109]. The authors consider the number of common objects (users) between two clusters (community structures) at two consecutive snapshots as a measure to decide whether a cluster has transitioned to or evolved from another. Based on this measure, five events called *becomes*, *splits*, *merges*, *disappears*, and *appears* that might happen to a community during two consecutive snapshots are defined. Sitaram Asur et al. [7] developed a similar framework to study community evolution. By matching snapshot communities, the authors formalized five temporal events that are identically interpreted as those in MONIC. Other measures called *stability*, *sociability*, *popularity*, and *influence* to study the behavior of users in a network were defined in this framework also. Palla et al. [93, 94] introduced a *Clique Percolation Model* and proposed a method to capture the evolution of communities between two consecutive snapshots by creating a union graph and matching community structures found in this graph with community structures found at the two snapshots.

Studies based on the evolutionary clustering approach build *unified* models to find *temporal smooth* evolving communities. The main idea of this approach is that the objective function employed in graph partitioning algorithms consists of two components, the *history quality* and the *snapshot quality*. The snapshot quality measures how accurate the resulting clusters capture the structure of the network at the current snapshot, while the history quality measures how consistent the resulting clusters are, with respect to the clusters discovered at the previous snapshot. Algorithms are designed to find a partition that is trade-off to these two quality components. The first study in this direction was introduced by Chakrabarti et al. [22]. In their work, the  $k$ -means and hierarchical clustering algorithms were extended to produce evolving clusters. Lin et al. [72, 73] developed a FacetNet framework, which is based on non-negative matrix factorization [33] to approximate the structure of a snapshot. The snapshot quality and history quality are computed using Kullback Leibler divergence distance. Evolving communities are identified by optimizing the clustering solution with respect to both the snapshot quality and the history quality. The authors of FacetNet also introduced a similar framework called MetaFac that employs metagraph factorization to extract communities in dynamic and rich media networks [74]. Other studies on the evolutionary clustering approach employed spectral clustering methods. Examples include the studies by Chi et al. [24, 25].

The probabilistic modeling approaches extract communities from each snapshot and make prediction about the evolution of communities using Bayesian prediction strategy. A probabilistic model is developed to discover communities in each snapshot, which is basically similar to the idea applied to extract static communities discussed in Section 2.5. However, to capture the evolution of communities, the community membership of users at the previous snapshot is used as a prior knowledge for computing such a membership at the current snapshot. Communities gradually evolve over time, which is indicated by changes in the membership of users in communities discovered over snapshots [54, 128].





## Chapter 3

# Extraction and Measurements of Social Links

### 3.1 Overview and Objectives

In this chapter, we develop solutions for the extraction and measurements of social links between users in different types of social networks. Social link measures are employed in applications where social relationships are used as input or evidence for building a model to achieve some application specific goals. Community extraction and feature-based recommendation are typical examples. So far, techniques and models proposed for such applications mainly rely on link structures associated with users. However, adopting only such explicit links might not be sufficient to give good results. This is because an explicit link basically presents a static connection between two users, which provides less information about the meaning of the relationship between them. Developing models for measuring social links based on more features describing users is therefore necessary and probably leads to more meaningful and practical results when the models are applied to such aforementioned applications.

Motivated by these observations, in this chapter we first present a data model of social networks and then introduce two social link measures, namely *interaction-based model* and *semantic-based model*. The first model computes relationships between users based on their association in discussion topics and direct interactions. The second model employs a technique based on latent semantic analysis to extract social characteristics of users from which social links are derived. To demonstrate the applications of the proposed measures, collaborative filtering algorithms for suggesting friends and topics of discussions to users are introduced.

This chapter is organized as follows. Section 3.2 presents a user-centric model of social network data and discusses assumptions and conventions underlying the analysis of a social network. An overview of social link identification is then introduced. Section 3.3 develops

a model for measuring social links based on the association of users in discussion topics and their direct interactions. Section 3.4 applies a semantic analysis method to assess latent social links between users. Two algorithms for feature-based collaborative filtering recommendations are discussed in Section 3.5 as example applications of the developed social link measures. We use a dataset from the *BBC Message Boards* network for conducting the experimental evaluations. Details of the dataset and the results of our experiments are presented in Section 3.6. We summarize this chapter and give an outlook for extracting social links from the spatio-temporal mobility history of users in Section 3.7.

## 3.2 Social Network Data

### 3.2.1 User-Centric Model

We consider a social network of a set of users  $U$ . Each user  $u \in U$  is described by complex features. There are several social networks, each of which was designed to provide users with different features. Blog and forum networks are structured in categorical topics called *threads*. Users in these networks interact with each others by posting messages in threads (e.g., religious or political topics) to discuss or to share ideas about specific topics. Typically, blog and forum users do not have the “*link users*” feature known as, for example, *friends* and *followers*. Recent emerging networks, however, provide mechanisms allowing a user to explicitly connect to other users. Protocols for a user to create a connection to another user vary from network to network and the notations used are different as well. For example, the *friend* feature on *Facebook* requires the agreement of both users while the *follow* feature introduced by *Twitter* allows a user to follow any other users. In this dissertation, such explicit connections are generally called “*link users*” feature.

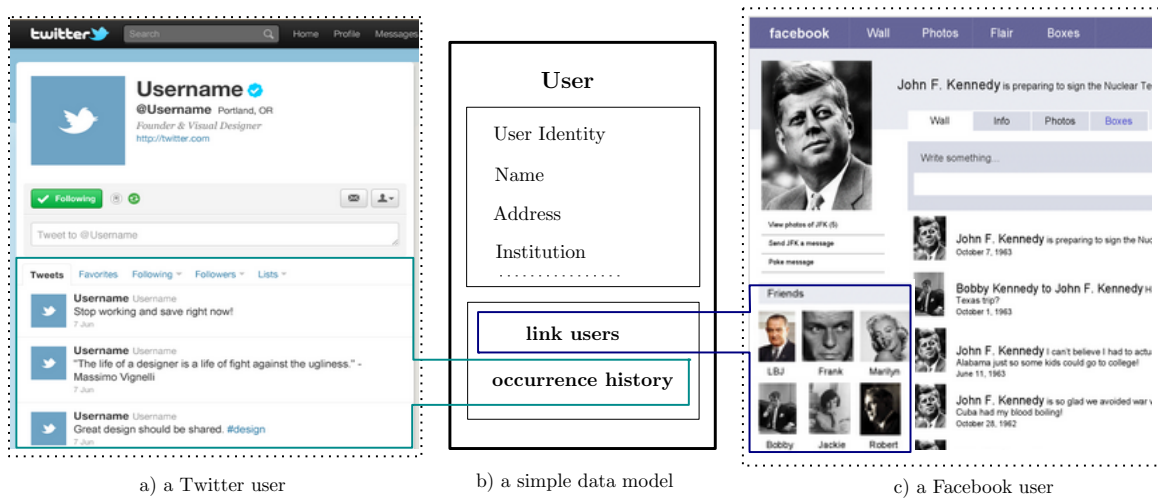


Figure 3.1: (a) and (c) are profile templates of a *Twitter* user and a *Facebook* user, respectively. (b) is an illustration of the underlying data model for a user in a social network.

The model of a user in a social network can be as simple as illustrated in Figure 3.1. It has a profile consisting of, for example, some descriptive attributes (e.g., user identity, name, and address), an explicit list of link users (e.g., friends on *Facebook* and followers on *Twitter*), and a collection of *occurrences* of the user in the network. The term occurrence is used to indicate an activity of the user such as posting a message, uploading a picture, clicking on a *like/dislike* feature, etc. For our study, each occurrence of a user is assumed to be a text message associated with other features, formalized as follows.

**Definition 3.1 (User Occurrence)** *An occurrence  $o = \langle u, loc, msg, f, thread, t \rangle$  of a user  $u \in U$  in a social network consists of a message  $msg$  posted by  $u$ , optionally in a categorical thread at a geographic location  $loc$  and at time point  $t$  with an optional set of contextual link users  $f \subseteq U$ . The message  $msg$  contains a set of words from a vocabulary  $V$ .*

Note that not all features presented in the above definition, i.e., the user occurrence, are required to be available in the data of users or are used in all models developed in this dissertation. For example, the geographic location indicating where the user occurs is only available in recent social networks, while the thread feature is mainly adopted in blog and forum networks to organize network structures in categorical topics. The above definition will be reformalized in each model later on where unnecessary features are removed. Two examples of user occurrences on *Foursquare* and *Twitter* are shown in Figure 3.2.

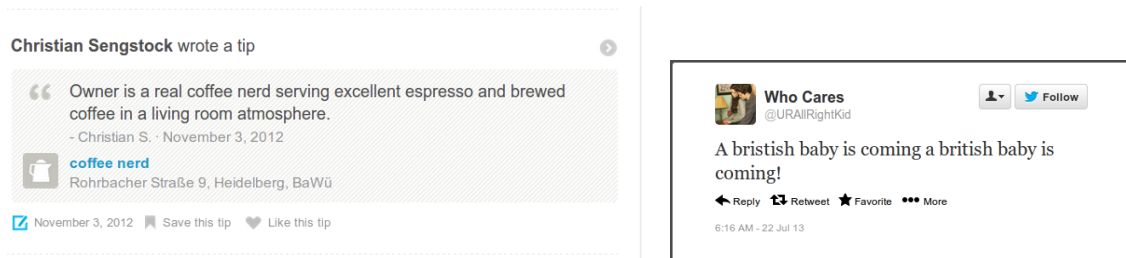


Figure 3.2: Examples of postings of a user on *Foursquare* (left) and on *Twitter* (right). In this dissertation, features of interest associated with a user posting are formally called a user occurrence.

### 3.2.2 Assumptions and Conventions

The data used to study social networks are normally collected by some crawling approaches using Application Programming Interfaces (APIs) provided by network services to access user-generated data. Since such a data acquisition process is typically time-consuming, as API calls are often rate-limited and monitored by the service provider, most of the social network datasets people have collected are some kind of samplings of the target network. Furthermore, social network data exhibit a certain heterogeneity and noise in nature. Because of this, data preprocessing is an important step needed to be done before any analysis task is applied. It is therefore important to note that when analyzing social

network data one either explicitly or implicitly assumes that network sampling techniques and data preprocessing tasks do not affect much the statistical properties of interests of the whole network.

A social network is dynamic with respect to the number of participating users and their activities or interactions. To capture such evolutions we adopt the *snapshotting idea* to model a social network over time. In particular, we consider a social network as a sequence of snapshots. Each snapshot consists of occurrences of users within a certain period of time, defined as follows.

**Definition 3.2 (Snapshot)** *Given a set  $U$  of users in a social network, the set of occurrences of users in  $U$  during a time interval  $\Delta t = [t_s, t_e]$  is called a snapshot of the network, denoted  $sn_t = \{\langle u, loc, msg, f, thread, t \rangle\}$ , where  $u \in U$  and  $t \in \Delta t$ .*

The time granularity of snapshots is application specific and can be, for example, daily or weekly interval snapshots. Having the concept of a snapshot formalized, a social network  $SN$  is then modeled as a sequence of snapshots  $SN = \{sn_1, sn_2, \dots, sn_T\}$ .

### 3.2.3 Social Links

There are different perspectives to identify *social links* between users in a social network. Determining whether two users in a social network are socially linked is a subjective task, which is identified based on (1) the application at study and (2) the particular social network under consideration. The former indicates what social characteristics of users are of interest to measure the relationships between them. The latter means that one needs to study features in a particular network to figure out what information can be employed to extract social links. In general, there are two types of social links one can observe or extract from social network data. The first type adopted by almost of all existing studies is the explicit link structure derived from link users. The second type known as *hidden social links* can be extracted from analyzing data features associated with the occurrences of users.

Identifying social links between users is the basic step for conducting further tasks of social network analysis and for building related applications. Generally, for analyzing a social network, a link graph is built representing connections among users in the context of a defined social link measure. Analysis tasks are then performed on the link graph. For example, as presented in Section 2.4, many algorithms for detecting social communities work on link graphs.

Most of the approaches addressing social network analysis adopt explicit link structures of users as an evidence to assess social links. Indeed, link structures are mainly suitable for getting an idea about the statistical properties of a network, and often fail when used to get more insights into the underlying relationships between users. Such information, however, is more interesting for applications built on top of social networks. In addition, explicit links associated with users might not reflect real social interactions. For example, there

are many users being added as friends of a particular user  $u$  on *Facebook*, but there exist very few interactions among those friends with  $u$  [124]. This is similar to *Twitter* where a large proportion of users who decide to follow someone just to make use of *Twitter* as an information providing resource [67]. Also, as mentioned in the previous section, there is no “*link users*” feature in blogs and forums meaning that one needs to study other features to derive social links between users in such networks.

This chapter aims at introducing new methods to extract and measure more meaningful social links between users based on their occurrences instead of depending on their explicit link structures. In particular, we study two features, namely the association of users in the same threads of discussions together with their direct interactions, and the similarity of social characteristics of users extracted from their postings to measure social links between them. The observations motivating our models can be summarized as follows. First, it is obvious that two users exchanging messages with each other or posting messages on the same topic of discussions are socially related. The frequency of such interactions between two users indicates the strength of the relationship between them. On the other hand, the semantic similarity extracted from what the users post to a social network indicates social links as well. Users posting messages about a specific topic, e.g., politics, are more likely to have common interests in that topic than those posting messages about general or broad topics. Such common interests intuitively imply a social link.

In addition to these features, the similarity of the spatio-temporal mobility history of users is also a hint of having a relationship between them [26]. This feature was hard to explore in the past due to the lack of information about geographic locations related to the activities of users. However, with the prominence of location-based features in today’s social networks, one can take location information associated with the content posted by users into account to study social links. This feature is presented in the last section of this chapter, where an outlook for the extraction of social links based on spatio-temporal mobility history of users is discussed.

Since a social network evolves over time, the models proposed in this chapter are applied to measure social links between users within a time interval. In terms of our data model, social links are extracted from the occurrences of users in a snapshot of the network. Details of the models are presented in the following sections.

### 3.3 Measuring Social Links from Interactions

This section introduces an interaction-based social link measure for assessing relationships between users. We aim at proposing a model applied to measure social links between users in a blog or a forum network. In such applications, categorical topics called *threads* are created and users post messages to specific threads of interest as a way to exchange information and to interact with each other. Messages in a thread can be classified into two types,

those containing interaction information called *contextual links* (e.g., a message created to reply to a message of another user, or a message mentioning other users) and those that do not have contextual links included. In other words, there are two types of interactions among users, namely the participations in the same threads and the direct interactions. Examples of blog and forum networks include the *Volconvo*<sup>1</sup> forum, *BBC Message Boards*<sup>2</sup> network, and *Digg*<sup>3</sup> network. A screenshot of the *BBC Message Boards* network is shown in Figure 3.3 to give the readers an intuition about interactions of users through posting messages in categorical topics.

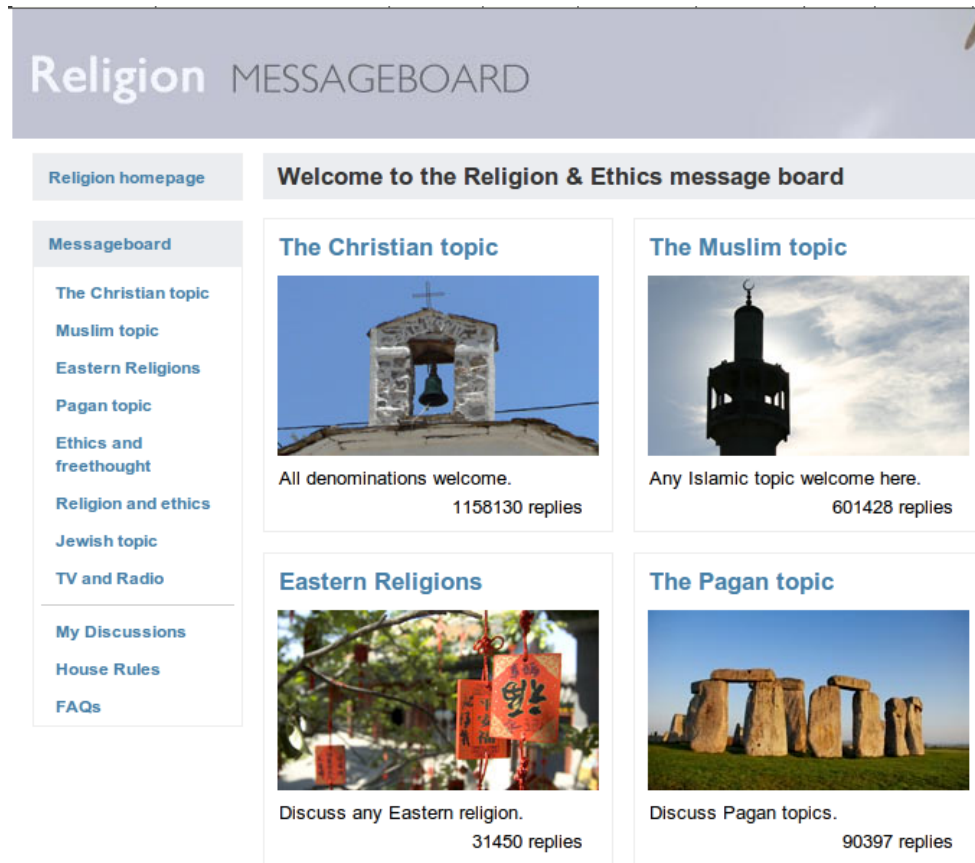


Figure 3.3: A screenshot of the *BBC Message Boards* network showing some religious topics of discussions. **Source:** <http://www.bbc.uk/messageboards/religion>

### 3.3.1 User<sup>2</sup>-Thread Network

Due to the thread-oriented characteristic of blogs and forums, as presented, a network of this type can be modeled as a *hyper-bipartite* graph. It consists of two disjoint sets of nodes, namely *users* and *threads*, and two sets of edges. The first set of edges indicates the participations of users in threads (i.e., the interest of a user in a thread is indicated by

<sup>1</sup><http://www.volconvo.com>

<sup>2</sup><http://www.bbc.co.uk/messageboards/>

<sup>3</sup><http://www.digg.com/>

the number of messages she posts in that thread). The second set of edges represents the direct interactions between users (i.e., a user interacts with another user by replying to her message or mentioning her in a message). Specifically, the graph is formalized as follows.

**Definition 3.3 (User<sup>2</sup>-thread network)** *A user<sup>2</sup>-thread network is a hyper-bipartite graph  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$ .  $U$  and  $Z$  are two disjoint sets of nodes, where  $U$  is a set of users and  $Z$  is a set of threads in the network.  $E_{U,Z} \subseteq U \times Z$  and  $E_{U,U} \subseteq U \times U$  are two sets of edges representing the participations of users in threads and the direct interactions between users, respectively.*

Figure 3.4 illustrates a user<sup>2</sup>-thread network consisting of six users and three threads. The participations of users in threads are indicated by blue edges, and the direct interactions between users are represented by grey edges. The figure is used as a running example to explain the measure being developed. In particular, we consider two factors, the likelihood of two users posting messages in the same thread and the direct interactions indicated by their contextual links to measure the strength of the social link between them. The term *interaction-based* social link is used to indicate the measure.

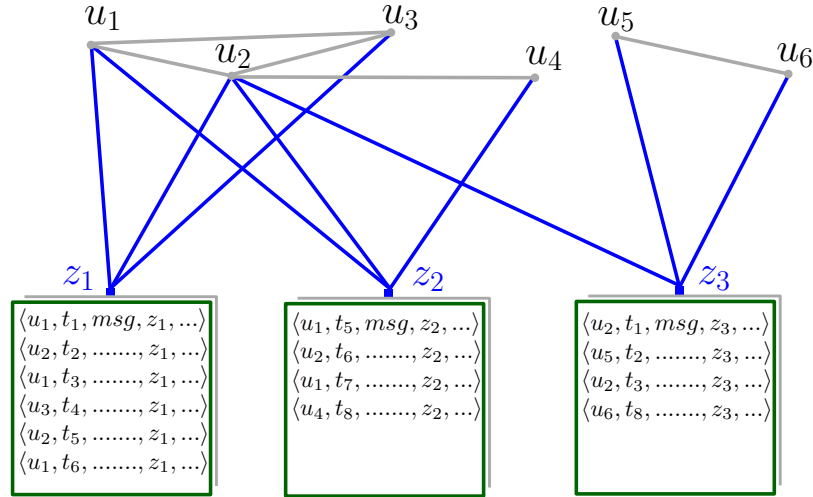


Figure 3.4: An illustration of a user<sup>2</sup>-thread network representing the participations of six users  $u_1, \dots, u_6$  in three threads  $z_1, \dots, z_3$ , and the direct interactions between these users.

Notations used for formalizing the measure are shown in Table 3.1. As presented in the previous section, it is noted that the social link measures introduced in this chapter are applied to a snapshot of a social network. By this, a snapshot  $sn_t$  is considered an instance of the social network; and we therefore omit the subscript  $t$  in our notations for simplicity.

Referring to the user occurrence formalized in Def. 3.1, four features, i.e., the user identity, message, thread, and contextual links, are used in this model. An occurrence  $o$  of a user  $u$  is therefore represented as  $o = \langle u, msg, thread, f, t \rangle$ . As input to the model, a snapshot of a social network is given, and the following components are extracted.

Table 3.1: Notations and their definition used to present models for measuring social links between users in a blog or a forum network.

Notation	Description
$post(*, z)$	set of messages posted by all users in a thread $z$ is denoted $post(*, z)$ . For a given thread $z$ , we count all messages created by users, either for replying to another message or a new message, as the number of messages in $z$ .
$post(u, *)$	set of all messages posted by user $u$ in all threads is called messages of $u$ and is denoted $post(u, *)$ .
$post(u, z)$	set of messages posted by user $u$ in thread $z$ is denoted $post(u, z)$ .
$post(u_i, u_j)$	set of messages posted by user $u_i$ in which each message has a contextual link to user $u_j$ . For example, user $u_i$ replies to a message of user $u_j$ , or user $u_i$ adds user $u_j$ in a message.

1. Set of users  $U = \{u_1, u_2, \dots, u_M\}$
2. Set of threads  $Z = \{z_1, z_2, \dots, z_K\}$
3. Set of messages created by users  $POST = \{msg_1, msg_2, \dots, msg_N\}$
4. The associations of users in threads (i.e, the threads a user  $u$  participates in)
5. The direct interactions between users indicated by the contextual links  $f$  in each user occurrence

In other words, a user<sup>2</sup>-thread network  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$  is derived from the data. This is done by (1) connecting a user  $u \in U$  to a thread  $z \in Z$  if user  $u$  has created a message in thread  $z$ , and (2) connecting users having a direct interaction to each other. In the following section, we first present a model for measuring the weight of edges  $(u, z) \in E_{U,Z}$  of the graph  $\mathcal{B}$ , which indicates the interest of a user  $u$  in a thread  $z$  of the network.

### 3.3.2 User-Thread Association

Given a user<sup>2</sup>-thread network  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$ , we define a user-thread association matrix  $\mathbf{B}$  of size  $M \times K$  where  $M$  is the number of users and  $K$  is the number of threads in the network. The value of  $\mathbf{B}[u, z]$  is the weight for the edge  $(u, z) \in E_{U,Z}$ , representing the interest of a user  $u \in U$  in a thread  $z \in Z$ . To derive the value of each  $\mathbf{B}[u, z]$ , we consider two factors  $f_1(u, z)$  and  $f_2(u, z)$  specified as follows:

- $f_1(u, z)$ : the first factor is the participation of user  $u$  in thread  $z$  in consideration of other users in that thread. This factor  $f_1(u, z)$  is computed as the likelihood of finding a message of user  $u$  in thread  $z$ .

$$f_1(u, z) \triangleq P(u|z) = \frac{|post(u, z)|}{|post(*, z)|} \quad (3.1)$$



- $f_2(u, z)$ : the second factor is the contribution of user  $u$  in thread  $z$  in comparison to the participation of  $u$  in other threads. Similar to the first factor,  $f_2(u, z)$  is evaluated as the likelihood that user  $u$  posts a message in thread  $z$ .

$$f_2(u, z) \triangleq P(z|u) = \frac{|post(u, z)|}{|post(u, *)|} \quad (3.2)$$

Having these two factors identified, the weight for an edge  $(u, z) \in E_{U,Z}$  in the user<sup>2</sup>-thread network  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$  is obtained as follows.

**Definition 3.4 (User-thread association weight)** *The interest of a user  $u$  in a thread  $z$  of a network  $\mathcal{B}$  is measured as the product of two factors  $f_1(u, z)$  and  $f_2(u, z)$ .*

$$\mathbf{B}[u, z] = f_1(u, z) \times f_2(u, z) = \frac{|post(u, z)|^2}{|post(*, z)| \times |post(u, *)|} \in [0, 1] \quad (3.3)$$

By applying the above definition of the user-thread association weight, we compute values for all entries in the matrix  $\mathbf{B}$ . Each entry represents the weight of the edge connecting a user to a thread in the social network. The value of each entry  $\mathbf{B}[u, z]$  falls in the range of  $[0,1]$  where 0 means that a user  $u$  does not have any message posted in thread  $z$ , while 1 means that  $u$  participates only in  $z$  and  $z$  contains only messages of  $u$ . Figure 3.5 (a) shows a matrix whose entries are the number of messages posted by users in threads of the running example illustrated in Figure 3.4. Figure 3.5 (b) shows the corresponding user-thread association matrix  $\mathbf{B}$  computed using the defined user-thread association weight measure. The normalized version of the matrix  $\mathbf{B}$  such that values of the entries in the matrix are summed up to 1 is shown in 3.5 (c).

	$z_1$	$z_2$	$z_3$	$ post(u, *) $
$u_1$	3	2	0	5
$u_2$	2	1	2	5
$u_3$	1	0	0	1
$u_4$	0	1	0	1
$u_5$	0	0	1	1
$u_6$	0	0	1	1
$ post(*, z) $	6	4	4	

a)

	$z_1$	$z_2$	$z_3$
$u_1$	$\frac{9}{30}$	$\frac{4}{20}$	0
$u_2$	$\frac{4}{30}$	$\frac{1}{20}$	$\frac{4}{20}$
$u_3$	$\frac{1}{6}$	0	0
$u_4$	0	$\frac{1}{4}$	0
$u_5$	0	0	$\frac{1}{4}$
$u_6$	0	0	$\frac{1}{4}$

b)

	$z_1$	$z_2$	$z_3$
$u_1$	0.167	0.111	0
$u_2$	0.074	0.028	0.111
$u_3$	0.093	0	0
$u_4$	0	0.139	0
$u_5$	0	0	0.139
$u_6$	0	0	0.139

c)

Figure 3.5: The user-thread association matrix built from the user<sup>2</sup>-thread network in Figure 3.4. (a) shows the matrix representing the number of messages posted by users in threads; (b) shows the matrix  $\mathbf{B}$  computed from applying the user-thread association weight measure on the matrix in (a); (c) is the matrix derived from  $\mathbf{B}$  by normalizing the values of entries to sum up to 1.

### 3.3.3 Thread Association-Based Link Measure

We now have the user<sup>2</sup>-thread network  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$  with weights attached to edges  $(u, z)$  in  $E_{U,Z}$ . The weight of an edge  $(u, z)$  implies the interest of user  $u$  in thread  $z$ . Matrix  $\mathbf{B}$  represents all the information about the structure of the network  $\mathcal{B}$  regarding the participations of users in threads. Nevertheless, what we want to measure is to what extent two users are socially linked with respect to the threads they are involved in. Particularly, we need a social link measure for users based on their association in threads, which is implied by the weight of edges in  $E_{U,Z}$  of the network  $\mathcal{B}$ . For this purpose, we embed the users in a  $K$ -dimensional Euclidean space where  $K$  is the number of threads in the network, and employ the method developed in [130] to assess social links between users. Specifically, the weight of a link between two users  $u_i$  and  $u_j$  is computed based on a Markov Random Walk applied to the bipartite graph  $\mathcal{G}_{U,Z} = \langle U, Z, E_{U,Z} \rangle$  derived from the user<sup>2</sup>-thread network  $\mathcal{B}$ . That is, the more likely it is to obtain a *walk* from a node representing user  $u_i$  to a node representing user  $u_j$ , or vice versa, through the threads they are involved in, the stronger the link they have. Note that the bipartite graph  $\mathcal{G}_{U,Z}$  is represented by the matrix  $\mathbf{B}$  computed above. In order to apply the Markov Random Walk strategy, the matrix  $\mathbf{B}$  is first normalized so that  $\sum_{u \in U, z \in Z} \mathbf{B}[u, z] = 1$  as illustrated in Figure 3.5 (c). The probability to get a walk from node  $u_i$  to node  $u_j$ , denoted  $P(u_i, u_j)$ , is computed as

$$P(u_i, u_j) = P(u_i)P(u_j|u_i), \quad (3.4)$$

where  $P(u_i)$  is the probability that user  $u_i$  is sampled from the network<sup>4</sup>. Because users are linked through threads, the conditional probability  $P(u_j|u_i)$  is derived from

$$P(u_j|u_i) = \sum_{z \in Z} P(z|u_i)P(u_j|z) = \sum_{z \in Z} \frac{P(u_i, z)}{P(u_i)} \frac{P(u_j, z)}{P(z)}, \quad (3.5)$$

where  $P(u_i, z)$  is the weight of the edge  $(u_i, z)$  in the graph  $\mathcal{B}$ , which is encoded by the value of the entry  $\mathbf{B}[u_i, z]$ . Therefore, we have

$$P(u_i, u_j) = \sum_{z \in Z} \frac{P(u_i, z)P(u_j, z)}{P(z)} = \sum_{z \in Z} \frac{P(u_i, z)P(u_j, z)}{\sum_{u \in U} P(u, z)}. \quad (3.6)$$

The joint probability  $P(u_i, u_j)$  is used as the measure of the social link between users  $u_i$  and  $u_j$  based on their participations in threads. Formally, we define a so-called *link<sub>thread</sub>* measure as follows.

$$\text{link}_{\text{thread}}(u_i, u_j) \triangleq \sum_{z \in Z} \frac{\mathbf{B}[u_i, z] \times \mathbf{B}[u_j, z]}{\sum_{u \in U} \mathbf{B}[u, z]} \in [0, 1] \quad (3.7)$$

---

<sup>4</sup> $P(u_i)$  is reduced in the next step, otherwise it is computed as the sum of the weights of edges connecting  $u_i$  to threads.

Intuitively, each component  $\frac{\mathbf{B}[u_i, z] \times \mathbf{B}[u_j, z]}{\sum_{u \in U} \mathbf{B}[u, z]}$  in the definition of  $link_{thread}(u_i, u_j)$  measures the social link between users  $u_i$  and  $u_j$  regarding their interest in a thread  $z$ . If either  $u_i$  or  $u_j$  does not participate in thread  $z$ , then these two users have no common interest with respect to  $z$ . In this case, the social link between  $u_i$  and  $u_j$  derived from  $z$  will be 0 because either  $\mathbf{B}[u_i, z] = 0$  or  $\mathbf{B}[u_j, z] = 0$ .

### 3.3.4 Interaction-Based Link Measure

To complete the measure for assessing social links between users in the user<sup>2</sup>-thread association network  $\mathcal{B} = \langle U, Z, E_{U,Z}, E_{U,U} \rangle$ , we now add the direct interactions between users to the model. This component is represented by the graph  $\mathcal{G}_{U^2} = \langle U, E_{U,U} \rangle$  derived from the network  $\mathcal{B}$ . To measure the social link between two users based on their direct interactions, we employ a typical method by counting the number of times they interact with each other. Specifically, a so-called  $link_{direct}$  measure defined as follows is applied to compute a normalized value indicating how often two users  $u_i$  and  $u_j$  directly interact with each other.

$$link_{direct}(u_i, u_j) \triangleq \frac{|post(u_i, u_j)| + |post(u_j, u_i)|}{\sum_{u_i, u_j \in U} |post(u_i, u_j)| + |post(u_j, u_i)|} \in [0, 1] \quad (3.8)$$

Finally, by employing  $link_{thread}$  (Eq. 3.7) and  $link_{direct}$  (Eq. 3.8) together, we obtain a measure that considers both the participations of users in threads and the direct interactions between users to assess social links between them. The measure is formalized as follows.

**Definition 3.5 (Interaction-based social link)** *The interaction-based social link between two users  $u_i$  and  $u_j$  is derived as a combination of the thread association and the direct interaction link measures, computed as follows.*

$$link_{it}(u_i, u_j) \triangleq \alpha \times link_{thread}(u_i, u_j) + (1 - \alpha) \times link_{direct}(u_i, u_j) \in [0, 1] \quad (3.9)$$

In Eq. 3.9,  $\alpha \in [0, 1]$  is a constant used to specify which component is more important for evaluating relationships between users. Because direct interactions are observable, one is therefore often interested in extracting hidden links between users, i.e., one wants to put more interest on the social links derived from the participations of users in the same threads. To do this,  $\alpha$  should be assigned a value greater than 0.5.

## 3.4 Semantic Analysis for Measuring Social Links

Identifying social links based on the implicit and explicit interactions between users in threads as presented in the previous section is effective when applied to blog and forum networks. Such an approach, however, might fail in measuring relationships between users in a social network where posting messages to specific threads for discussions is not the main feature provided. In other words, users are generally free to post messages sharing their

ideas and thoughts to the network without any thread-oriented guideline. In such a network, one is more interested in a social link measure that relies on the semantics extracted from the users’ postings. Two users might neither interact directly nor participate in the same threads but still exhibit social links as they share something latent in the content of their postings. This section introduces a model for measuring such latent semantic social links.

So far, the three best known models for extracting latent semantics from documents are Latent Semantic Indexing (LSI) [31], Probabilistic Latent Semantic Indexing (pLSI) [55], and Latent Dirichlet Allocation (LDA) [12]. Our semantic-based social link measure proposed in this section is an extension of the LSI model. The main reasons for the adoption of the LSI model instead of pLSI or LDA come from both the application perspective of these models and the appropriation of the LSI model for being extended to measure social links. Particularly, the main application of pLSI and LDA is to extract semantic topics from documents, while LSI is used to semantically compare documents. Also, the term frequency-inverted document frequency weighting schema (*TF.IDF*) employed in LSI can be adjusted to fit the setting of comparing users based on the semantics of their postings in a social network as presented in the following sections.

### 3.4.1 Term Significance for Users

We again assume a snapshot of a social network is given from which a set of users  $U = \{u_1, u_2, \dots, u_M\}$ , and a set of messages  $POST = \{msg_1, msg_2, \dots, msg_N\}$  called a message corpus are extracted. It is noted that in this model we assume no categorical threads and interaction information associated with messages of users. These features can be taken into account by combining this model with the interaction-based model developed in the previous section. More details about this will be given in Section 3.7. The user occurrence (Def. 3.1) applied to this model is therefore simplified to  $o = \langle u, msg, t \rangle$ . The message corpus being the input to the model is first processed so that stop words are removed and then stemming is applied to refine the corpus.

To employ the semantic analysis for measuring relationships between users, we introduce a variation of the *TF.IDF* weighting schema so that it is more applicable to apply to the messages created by users in a social network. The goal is to derive a significant score of a term  $w$  for a user  $u$  based on her messages. Basically, the *TF.IDF* weighting schema is used to compare documents in a corpus represented by a vector space model [103]. It considers two factors to weight the significance of a term  $w$  in a document  $d$  regarding how often that term appears in  $d$  and also in other documents. The first factor called *local weight* is the significance of  $w$  measured locally in the document  $d$ . The local weight increases proportionally to the number of times  $w$  appears in that document. The second factor called *global weight* takes the frequency of  $w$  in other documents into account. The more occurrences of  $w$  in the whole corpus, the less significant it is in the document  $d$ . In our setting for extracting semantic-based social links, a user posts messages to the network

discussing with other users or sharing her ideas about specific topics or whatever she is interested in. Over time, the social characteristics of a user are likely to show up in her postings. For example, what topics she is more interested in and which words she often uses to express her ideas. Therefore, the significance of a term  $w$  for a user  $u$  increases proportionally to both the number of messages posted by  $u$  that contain  $w$  and the frequency of  $w$  within each message as well. On the other hand, the significance of  $w$  decreases with the frequency of  $w$  in messages posted by other users in the network. To take such observations into account, we introduce a variation of the local weight and the global weight of a term  $w$  in the message corpus  $POST$  as follows.

$$localWeight(w, u) \triangleq \frac{\sum_{msg \in post(u, *)} frequency\ of\ w\ in\ msg}{|post(u, *)|} \quad (3.10)$$

$$globalWeight(w) \triangleq \log \left( \frac{|POST|}{\sum_{msg \in POST} frequency\ of\ w\ in\ msg} \right) \quad (3.11)$$

Finally, the significance of a term  $w$  for a user  $u$ , denoted  $sig(w, u)$ , is derived using the same formulas as for the *TF.IDF* weighting schema. Particularly, it is the product of the local and global weights, which is formalized as follows.

$$sig(w, u) \triangleq localWeight(w, u) \times globalWeight(w) \quad (3.12)$$

The value of  $sig(w, u)$  can be interpreted as a *social sensor* for distinguishing user  $u$  from other users regarding how she is characterized by the frequency of choosing term  $w$  in her postings compared to others. The defined model is employed to compute a *term-user* matrix  $\mathbf{W}$  of size  $V \times M$  where  $V$  is the number of terms (the vocabulary size after cleaning data) extracted from the message corpus, and  $M$  is the number of users in the network.

To summarize, in our model, as an extension of the *TF.IDF* model, the local significance of a term  $w$  for a user  $u$  is not computed from the frequency of  $w$  in the composed document of  $u$  as would be done in *TF.IDF*. It is rather derived from the frequency of  $w$  in each message of  $u$  and the frequency of the messages of  $u$  that contain  $w$  as well. Similarly, the frequency of  $w$  in each message is taken into account to derive the global weight for  $w$ . Nevertheless, once the term-user matrix  $\mathbf{W}$  is computed, it plays a similar role as the *term-document* matrix in the LSI model applied to information retrieval applications.

### 3.4.2 Semantic-based Social Link

Since users in a social network can post messages expressing diverse topics, the term-user matrix  $\mathbf{W}$  is often very sparse. We apply Singular Value Decomposition (SVD) technique to reduce the matrix  $\mathbf{W}$  to a lower dimensional space of terms from which the similarity between users is computed. SVD is based on a linear algebra theorem stating that a rectangular matrix  $\mathbf{W}$  of size  $V \times M$  can be decomposed into the product of three matrices:

an orthogonal matrix  $\mathbf{Y}$  of size  $V \times V$ , a diagonal matrix  $\mathbf{\Lambda}$  of size  $V \times M$ , and the transpose of an orthogonal matrix  $\mathbf{X}$  of size  $M \times M$ . Particularly, the theorem is written as

$$\mathbf{W}_{V \times M} = \mathbf{Y}_{V \times V} \times \mathbf{\Lambda}_{V \times M} \times \mathbf{X}_{M \times M}^T, \quad (3.13)$$

where  $\mathbf{Y}^T \mathbf{Y} = \mathbf{I}$  and  $\mathbf{X}^T \mathbf{X} = \mathbf{I}$ ; the columns of  $\mathbf{Y}$  are the orthonormal Eigenvectors of the matrix  $\mathbf{W} \mathbf{W}_{V \times V}^T$ , and the columns of  $\mathbf{X}$  are the orthonormal Eigenvectors of the matrix  $\mathbf{W}^T \mathbf{W}_{M \times M}$ ;  $\mathbf{\Lambda}_{V \times M}$  is a diagonal matrix containing the square roots of Eigenvalues (in a descending order) corresponding to the Eigenvectors forming the columns of  $\mathbf{Y}$  or  $\mathbf{X}$ .

Note that in our setting we are interested in the matrix  $\mathbf{W}^T \mathbf{W}$  since each entry of this matrix represents a correlation between two users, which is computed as the dot product of the weights of terms used by them. Thus, we reduce the term dimensions of the matrix  $\mathbf{W}^T \mathbf{W}$  by computing matrix  $\mathbf{\Gamma} = \mathbf{\Lambda}_{V' \times M} \times \mathbf{X}_{M \times M}^T$  (having size of  $V' \times M$ ), where  $V'$  is the number of term dimensions reduced from  $V$  applied to matrix  $\mathbf{\Lambda}$ . Each column of matrix  $\mathbf{\Gamma}$  is now representing the significances of terms for a user in the new dimensional space of terms. Finally, the cosine similarity between columns of  $\mathbf{\Gamma}$  is employed to derive the similarity between the corresponding users. Given the model as introduced, each user  $u_i$  is now represented by a vector  $u_i(uw_{i,1}, uw_{i,2}, \dots, uw_{i,V'})$  where  $uw_{i,k}$  is the significance of term  $w_k$  in the new term space for user  $u_i$ . A so-called *link<sub>latent</sub>* computed as the cosine similarity between two users  $u_i$  and  $u_j$  is formalized as follows.

**Definition 3.6 (Latent semantic-based social link)** *The latent semantic-based social link between two users  $u_i$  and  $u_j$  is computed as the cosine similarity between the two vectors representing the two users in the latent semantic-based model.*

$$link_{latent}(u_i, u_j) \triangleq \frac{u_i \cdot u_j}{|u_i| \times |u_j|} = \frac{\sum_{k=1}^{V'} uw_{i,k} \times uw_{j,k}}{\sqrt{\sum_{k=1}^{V'} uw_{i,k}^2} \times \sqrt{\sum_{k=1}^{V'} uw_{j,k}^2}} \in [0, 1] \quad (3.14)$$

The cosine similarity is normally in the range of  $[-1, 1]$ . However, in our setting, the similarity of the two users will not be negative because the significance of a term for a user as defined is not less than 0. Our model is thus an application of dimensionality reduction in extracting social characteristics of users from their postings.

## 3.5 Recommendation Applications

This section introduces example applications of the proposed social link measures. Two algorithms for suggesting friends and topics of discussions to users are presented.

### 3.5.1 Collaborative Filtering Paradigm

Typically, e-commerce applications running businesses on the Internet environment are employed to illustrate the concept of recommendation systems. In such applications, the two key components are “items” and “customers”. A recommender is developed to rely

on explicit or implicit evidences of the shopping trend of a particular customer and other related information to guide the customer to the items that are most likely of interest to her. In this respect, the goal of the recommendation is to increase the chance that more items will be bought by customers. Recommendation mechanisms can be applied, however, to many other applications as well. Friend suggestion in a social network is one example. Indeed, a recommender can be generally described as a module designed to find appropriate “items” for a particular “object” (of the same or different type as of such items) in an application running on the network environment.

Collaborative filtering is one of the main strategies employed to develop recommendation systems [111]. The basic idea of the collaborative filtering is “if objects  $A$  and  $B$  are similar and  $A$  is related to an item  $X$  then it is likely that  $B$  is also related to  $X$ ”. By considering the scenario of recommending sale items ( $\mathcal{I}$ ) to customers ( $\mathcal{U}$ ), as an example, each customer  $u \in \mathcal{U}$  has a rating profile about her item preferences,  $\mathcal{R}_u = \{I_{1,u}, I_{2,u}, \dots, I_{|\mathcal{I}|,u}\}$ ,  $I_i \in \mathcal{I}$ . The rating can be realized using different methods, e.g., a range of numbers, or a list of selected options. Such profiles of all customers form a rating space  $\mathcal{R}$ . To suggest items to a customer  $u$ , the system first finds the most similar customers called *neighbors* of  $u$  based on some distance measure defined on the item preference rating space  $\mathcal{R}$ . Once a neighborhood of  $u$  is formed, an algorithm is employed to combine the preferences of the neighbors to produce a prediction item or top- $N$  most likely items as suggestions to  $u$ .

### 3.5.2 Friend Recommendation

In social networking services, a general method for suggesting friends to a user  $u$  is to find users who are most similar to  $u$  regarding some social aspects. Following this guideline, we propose a friend suggestion algorithm based on the social link measures introduced in this chapter. Given a social network consisting of users  $U$ , the overall steps for suggesting friends to users are as follows. First, a social link measure is employed to compute a matrix  $\mathbf{S}$  whose elements are the link scores between users. Based on matrix  $\mathbf{S}$  and a given threshold  $M'$ , a set of users who have the highest link scores with a user  $u$ , excluding the users already in the friend list of  $u$ , is selected as the candidates, denoted  $u.suggestList$ , for being suggested to  $u$ . The collaborative filtering guideline is then applied to suggest “friends of friends” to  $u$ . This step is taken place by finding a set *moreCandidates* of users who have the highest link scores with the friends of  $u$ . Each user  $u'$  in *moreCandidates* is associated with a *count*, denoted  $u'.count$ , representing the number of friends of  $u$  whose *suggestList* contains  $u'$ . The users in *moreCandidates* are then ranked based on their *count*. The larger the *count* a user has, the higher the score assigned to that user. Finally, selected users having the highest ranking scores are added to the suggestion list for  $u$ . Algorithm 2 shows the pseudo-code illustrating the main steps of suggesting friends to a user  $u$ , given a social link matrix  $\mathbf{S}$  and a threshold  $M'$  used to compute the nearest neighbors to be suggested to  $u$ .

---

**Algorithm 2:** Application of the collaborative filtering approach for suggesting friend candidates to a user: **suggestFriends**( $u, M', \mathbf{S}$ )

---

**Input:**  
 $u$ : the user to be suggested friend candidates  
 $\mathbf{S}$ : social link matrix  
 $M'$ : a threshold of number of neighbors will be considered

**Output:**  
 $u.suggestList$ : list of users being suggested to  $u$

- 1  $u.suggestList \leftarrow getTopLinkUsers(u, \mathbf{S}, M')$ ;
- 2  $u.suggestList \leftarrow u.suggestList \setminus u.friends$ ;
- 3  $moreCandidates \leftarrow \emptyset$ ;
- 4  $neighbors \leftarrow u.friends$ ;
- 5 **foreach**  $f \in neighbors$  **do**
- 6      $f.suggestList \leftarrow getTopLinkUsers(f, \mathbf{S}, M')$ ;
- 7      $moreCandidates.update(f.suggestList)$ ;
- 8  $moreCandidates \leftarrow getTopRankUsers(moreCandidates, M')$ ;
- 9  $u.suggestList \leftarrow u.suggestList \cup moreCandidates$ ;
- 10 **return**  $u.suggestList$ ;

---

**Complexity Analysis.** The major computation of the algorithm **suggestFriends**( $u, M', \mathbf{S}$ ) comes from the two ranking procedures,  $getTopLinkUsers(\dots)$  and  $getTopRankUsers(\dots)$ . Each call to  $getTopLinkUsers(\dots)$  has the time complexity  $O(M \log(M))$  for sorting  $M - 1$  users based on their social links to a user  $u$ . However, an index structure can be employed while the matrix  $\mathbf{S}$  is being computed so that such sorting steps can be reduced. The number of users in the set  $moreCandidates$  of candidates derived from applying the collaborative filtering mechanism varies for each user  $u$  and for each neighbor threshold  $M'$  employed. Let the number of users in the candidate set  $moreCandidates$  be  $N$ . The time complexity of  $getTopRankUsers(\dots)$  is  $O(N \log(N))$ , which is obviously scalable because  $N$  is much smaller than the number of users  $M$  in the network, given that the neighbor threshold  $M'$  is often assigned a small number.

### 3.5.3 Thread Recommendation

Another recommendation application of our social link measures is for suggesting threads of discussions to appropriate users in a forum or blog network. This is done as a collaborative filtering process by relying on the threads associated with the users that are socially linked with user  $u$  regarding the thread association-based link measure  $link_{thread}$  (Eq. 3.7). It is noticed that there is no explicit concept of *item rating* specified in our recommendation setting, which is usually used to indicate the degree of interest of a user to an item in a recommendation system. For suggesting threads to a user, we consider the user-thread association measure (Eq. 3.3) as the *user-thread rating* score.



Similar to the friend recommendation discussed in the previous section, a link matrix  $\mathbf{S}_{thread}$  is computed using  $link_{thread}$  social link measure. The scenario for suggesting threads to a user  $u$  is as follows. First, a set of users who have the highest link scores with user  $u$  is determined. To be more precise, such a set of users is denoted  $linkUsers = \{u_1, u_2, \dots, u_{M'}\}$ . In the next step, a set of threads associated with users in  $linkUsers$  is extracted, which is denoted  $linkThreads = \{z_1, z_2, \dots, z_{K_u}\}$ . Threads in  $linkThreads$  are then ordered based on the number of users in  $linkUsers$  associated with each thread. That is, the more users in  $linkUsers$  sending messages to thread  $z$ , the higher the score assigned to  $z$  is. Finally, top  $K'$  threads that have the highest ranking scores will be suggested to  $u$ . The pseudo-code illustrating the main steps for suggesting threads to a user  $u$  is shown in Algorithm 3.

---

**Algorithm 3:** Application of the collaborative filtering approach for suggesting threads to users: **suggestThreads**( $u, K', M', \mathbf{S}_{thread}$ )

---

**Input:**

$u$ : the user to be suggested thread candidates

$\mathbf{S}_{thread}$ : social link matrix derived from  $link_{thread}$  measure

$M'$ : a threshold of number of neighbors will be considered

$K'$ : a threshold of top rank threads will be considered

**Output:**  $u.suggestThreads$ : list of threads to be suggested to  $u$

- 1  $linkUsers \leftarrow getTopLinkUsers(u, \mathbf{S}_{thread}, M')$ ;
  - 2  $linkThreads \leftarrow \bigcup_{u' \in linkUsers} u'.threads \setminus u.threads$ ;
  - 3  $u.suggestThreads \leftarrow getTopRankThreads(linkThreads, K')$ ;
  - 4 **return**  $u.suggestThreads$ ;
- 

Similar to the friend suggestion (Algorithm 2), the time complexity of the thread suggestion algorithm **suggestThreads**( $u, K', M', \mathbf{S}_{thread}$ ) is  $O(N \log(N))$ , where  $N$  is the number of thread candidates in the set  $linkThreads$  derived using the threshold  $M'$ .

## 3.6 Experiments

### 3.6.1 Dataset for Experiments

**BBC Message Boards dataset.** The BBC (British Broadcasting Corporation) website provides a forum network called *Message Boards*<sup>5</sup>. The service allows registered users to post messages discussing different topics they are interested in. We select this network to conduct experimental evaluations for the models presented in this chapter. The dataset used for our evaluations is a subset of the *BBC Message Boards* network spanning from June 20, 2005 to June 16, 2009. This dataset is published for research purposes at the *CyberEmotion* Website<sup>6</sup> [95]. It consists of 2,474,781 messages posted by 18,249 users in 97,946 discussion threads mainly about ethical, religious, and news-related topics.

<sup>5</sup><http://www.bbc.co.uk/messageboards/>

<sup>6</sup><http://www.cyberemotions.eu>

Table 3.2: Main topics discussed by users in the *BBC Message Boards* network. These topics are used as ground truth for experimental evaluations.

Topic Id	Categorization	#Messages
01	Eastern Religions	21.402
02	Christian topic	715.792
03	Ethics and free thought	86.088
04	Jewish topic	65.141
05	TV and Radio	20.518
06	UK News	1.063.136
07	World News	489.247

After running a data cleaning step and removing empty messages, the dataset finally contains 2.461.324 messages, 18.031 users, and 97.942 threads. The list of main topics and the number of messages categorized in each topic are presented in Table 3.2, which are used as ground truth for our evaluations. We further organize the dataset in monthly interval snapshots for conducting the experiments. The number of users, number of threads, and number of messages posted by users in 49 monthly interval snapshots are shown in Figure 3.6.

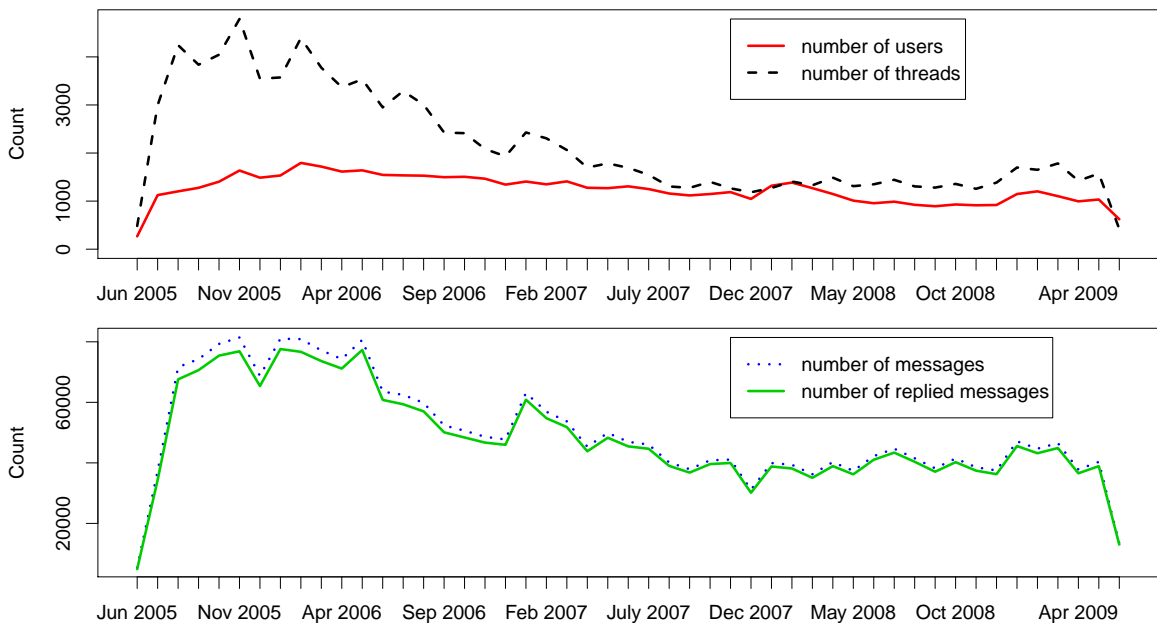


Figure 3.6: Statistics of the number of users, number of threads, and number of messages posted by users in monthly interval snapshots of the *BBC Message Boards* dataset. The first and the last snapshot contain only the last 10 days of June 2005 and the first 16 days of June 2009, respectively.

**Dynamics of Users and Threads.** To get an idea about the dynamics of the *BBC Message Boards* network over time, we retrieve users in each snapshot and compute the union and the intersection of the user identities in snapshots. The results show that about 42.50% of users are stable over consecutive snapshots. The stable measure here is derived using the Jaccard coefficient, i.e., the fraction of the intersection and the union of the user identities in two consecutive snapshots. The same method is employed to measure the dynamics of threads, and we find that only 5.88% of threads remain over consecutive snapshots. Statistical results of the dynamics of users and threads are shown in Figure 3.7.

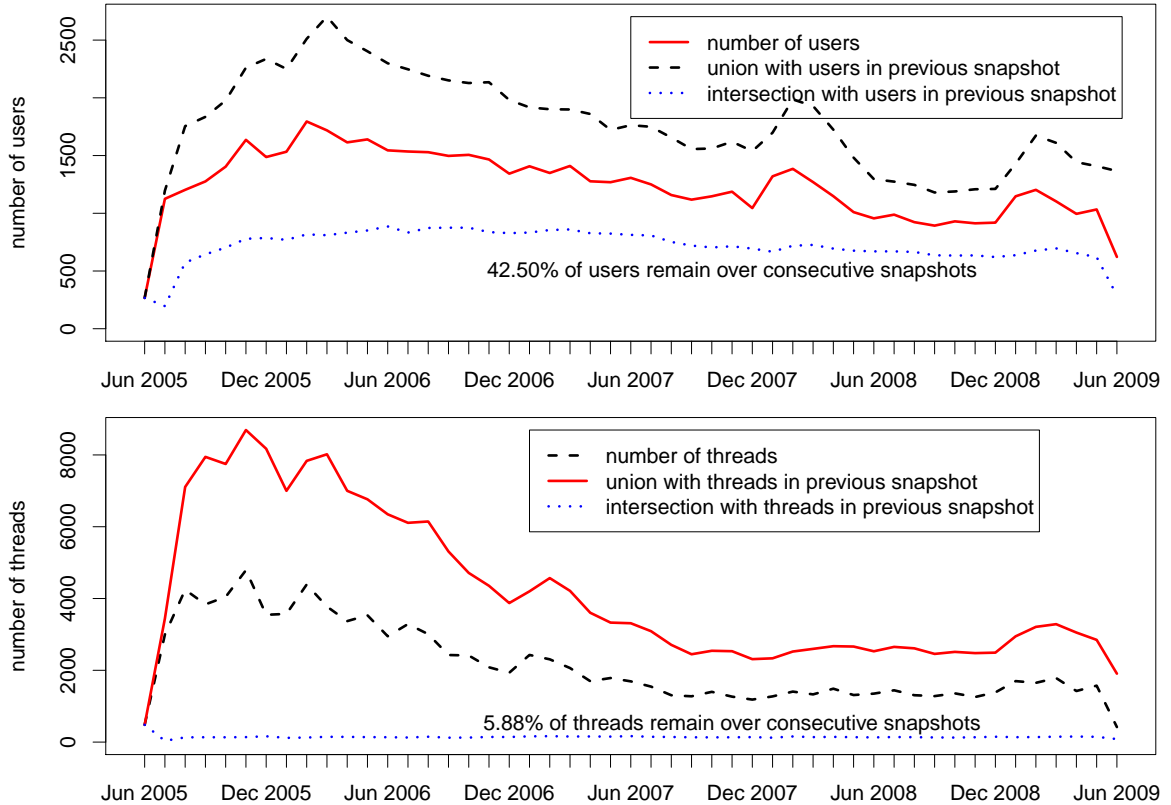


Figure 3.7: The dynamics of users and threads in the *BBC Message Boards* network over time.

**Data Filtering.** For the purpose of efficiency and reliability in conducting evaluations, we first apply two steps of data filtering to remove users and threads that are not of importance in the network. These are users and threads that post and contain, respectively, a very few messages compared to others. Figure 3.8 shows the histograms of the number of messages posted by users, and of the number of messages in threads of the network. Both exhibit power law distributions. Based on this, a threshold is applied to filter users and threads. The results of the following experiments are computed after filtering users who post less than 10 messages and threads that contain less than 10 messages in a snapshot. After filtering the data as described, the stable measure computed for users increases to

58.30% while the number of stable threads decreases to 3.01%. The details of such measures computed for six snapshots from February 2008 to July 2008 after filtering the data are summarized in Table 3.3. These six snapshots are selected to conduct evaluations for social link measures, which are discussed in the following sections.

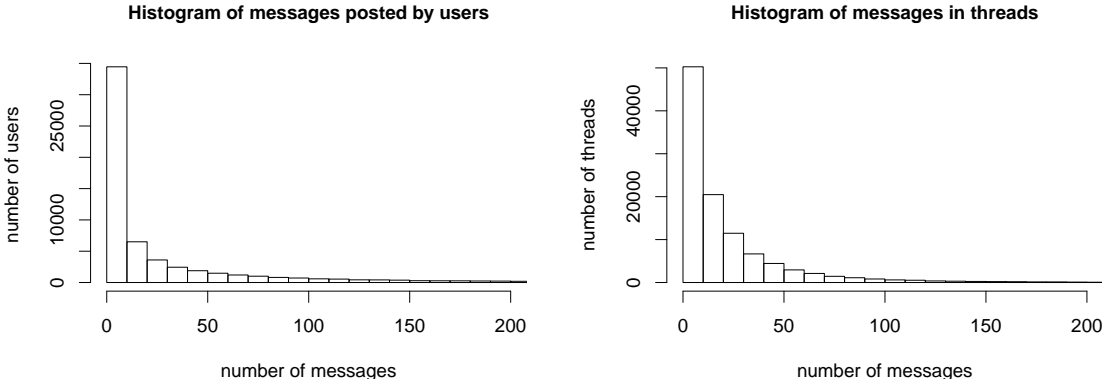


Figure 3.8: Histograms of the number of messages posted by users (left) and the number of messages in threads (right). Data for this plot are aggregated over all 49 monthly snapshots.

Table 3.3: Statistics of stable users and stable threads in six snapshots from February 2008 to July 2008 of the *BBC Message Boards* network. The results are computed after applying a 10 message threshold filtering to users and threads.

	February	March	April	May	June	July	Average
# users	495	499	490	464	498	515	493
% stable users	59.60	57.27	54.29	58.73	59.27	60.79	58.34
# threads	892	899	981	881	928	1044	937
% stable threads	3.25	3.04	2.22	3.21	2.78	3.08	2.93

### 3.6.2 Interaction-based Link Network

In this section, we show the results of applying our interaction-based social link measure to study the *BBC Message Boards* network. For each snapshot of the network, we compute three graphs. The first graph is created from the direct interactions between users. This graph gives an idea of how users contextually interact with each other in threads. The second graph and the third graph are the results of applying our  $link_{thread}$  measure and cosine similarity on the user-thread association bipartite graph (see Sec. 3.3.2 and Def. 3.4) obtained from that snapshot, respectively.

**Direct interaction network.** For each snapshot  $sn$ , information extracted from replied messages is used to create a direct interaction network  $\mathcal{G}_{direct} = \langle U, E \rangle$ , where nodes  $U$  are users in the snapshot  $sn$  and each edge  $(u_i, u_j) \in E$  is weighted by the  $link_{direct}$  (Eq. 3.8), i.e., the normalized value of the number of messages exchanged between users  $u_i$  and  $u_j$ . We

then compute statistical measures for the graph to get more insights into how users directly communicate in the *BBC Message Boards* network. We find that even though the number of replied messages is very large, as shown in Figure 3.6, such direct interactions happen quite locally between pairs of users. This is indicated by small graph density and node degree measures, which are, on average, about 0.014 and 6.94, respectively. These results imply that relying on direct interactions between users one can only find very few users who are linked to each other. The details of node degrees, graph density, and clustering coefficient computed for the direct interaction graphs of six snapshots from February 2008 to July 2008 are presented in Table 3.4. The distributions of the degrees of nodes in these graphs are shown in Figure 3.9.

Table 3.4: Statistical measures obtained from direct interaction graphs for six snapshots from February 2008 to July 2008 of the *BBC Message Boards* network.

	February	March	April	May	June	July	Average
<b>Mean of node degree</b>	6.55	6.18	7.01	6.70	7.43	7.74	6.93
<b>Graph density</b>	0.013	0.012	0.014	0.014	0.015	0.015	0.014
<b>Clustering coefficient</b>	0.362	0.338	0.374	0.342	0.388	0.390	0.365

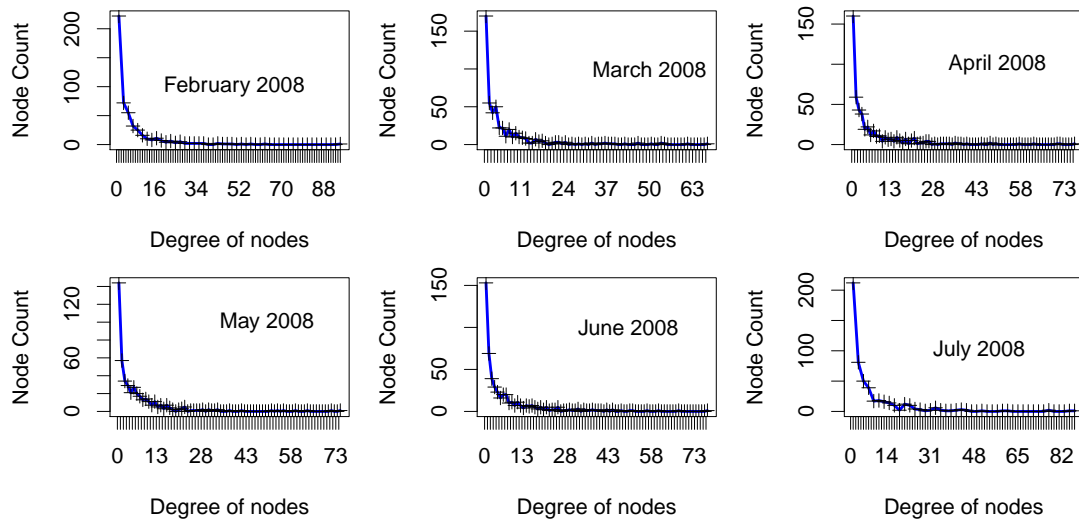


Figure 3.9: Distribution of node degrees in graphs built from direct interactions between users in the *BBC Message Boards* network for six snapshots from February to July 2008.

We apply the modularity-based clustering algorithm [87] for extracting community structures from direct interaction graphs. The results show that over six months from February to July 2008 the direct interactions between users always form large and really cohesive community structures. The number of communities varies from 4 to 6 as shown Figure 3.10, where the PageRank algorithm [92] is applied to measure social influence of each user, denoted by the label size, regarding her direct interactions with others.

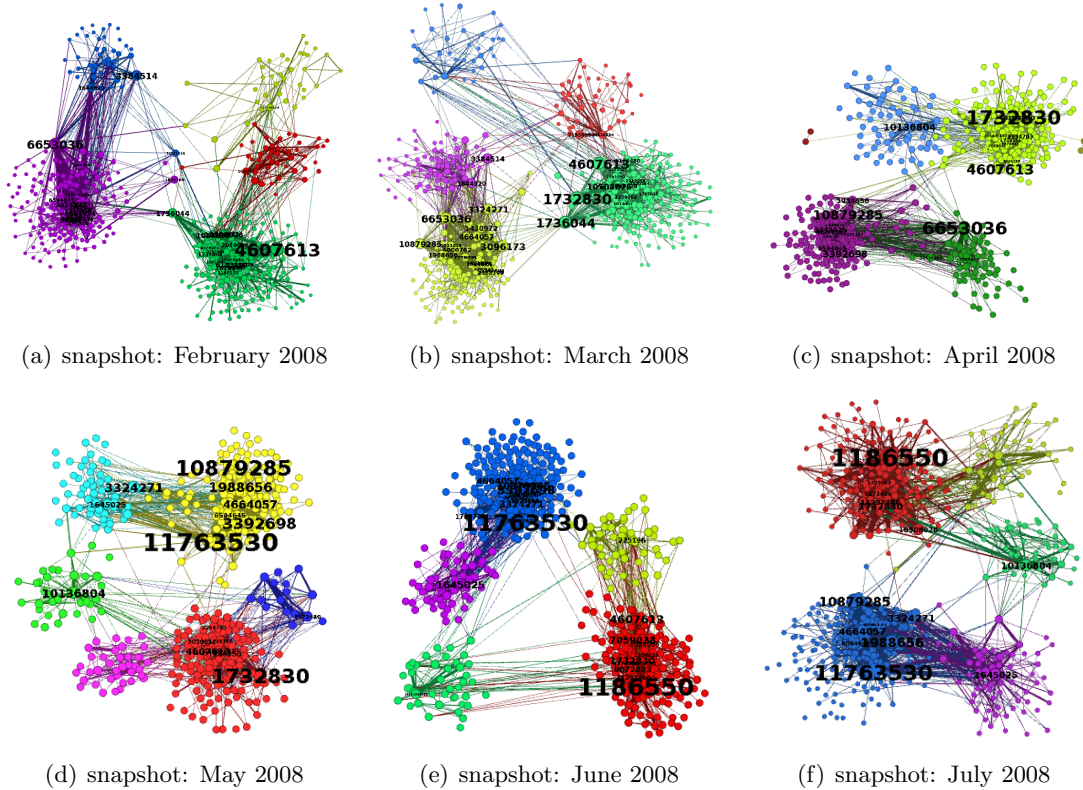


Figure 3.10: Community structures in graphs built from direct interactions between users in selected snapshots of the *BBC Message Boards* network.

**Thread association link vs direct interaction.** We now apply our  $link_{thread}$  measure (Eq. 3.7) to extract social links between users from their participations in threads. Based on the histograms of the link scores as shown in Figure 3.11, a filtering step is first applied to remove links that have a weight less than 0.0005 before analyzing the obtained results. Having such steps accomplished, we find that even though direct interactions between users happen locally, many users are actually linked to each other in the context of posting messages to the same threads of interest. By relating the direct interaction  $link_{direct}$  and the thread association  $link_{thread}$  measures for each pair of users, we find that users who tend to reply to each other only join together in a few threads. This is indicated by many pairs of users having high  $link_{direct}$  but low  $link_{thread}$  scores. There are many users who do not have messages replying to each other but post messages to many common threads. These users are socially linked in terms of having similar interests but might not directly interact with each other, which is highlighted by high  $link_{thread}$  but low  $link_{direct}$  scores between them. Figure 3.12 shows the results obtained from applying  $link_{direct}$  and  $link_{thread}$  measures to two snapshots in June and July 2008.

We further compute statistical measures for graphs built from applying our  $link_{thread}$  measure to each snapshot of the network. The results clearly show that more users are linked to each other. The average of the graph density and the degrees of nodes in such

graphs are about 0.079 and 39.12, respectively. The details of node degrees, clustering coefficient, and density measures of thread association-based link graphs are summarized in Table 3.5. The distributions of the degrees of nodes in such graphs built for six snapshots from February 2008 to July 2008 are shown in Figure 3.13.

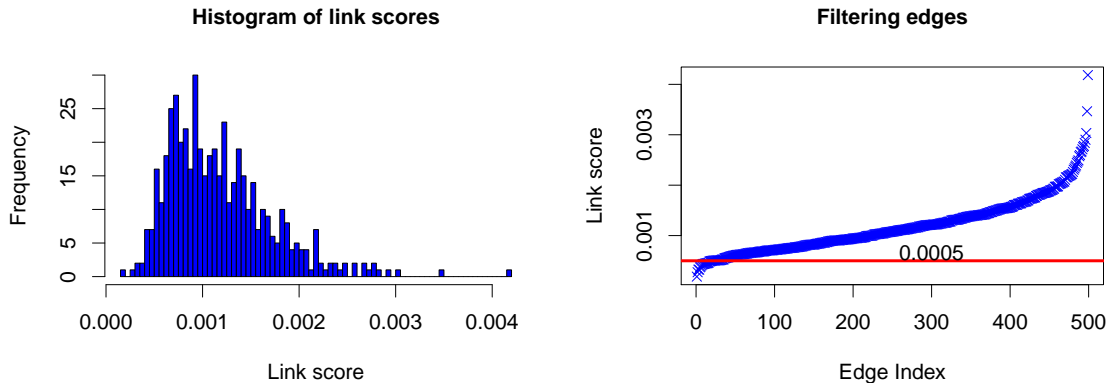
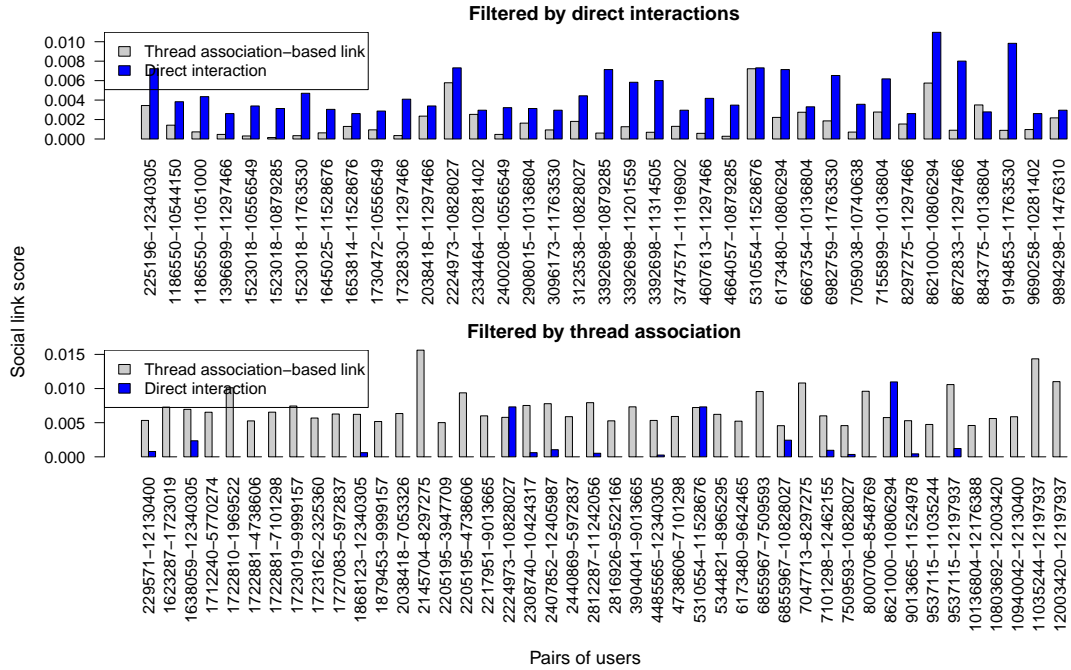


Figure 3.11: Histogram of link scores obtained from applying  $link_{thread}$  measure on selected snapshots of the *BBC Message Boards* network (left) and a threshold is employed to filter edges having weights less than 0.0005 (right).

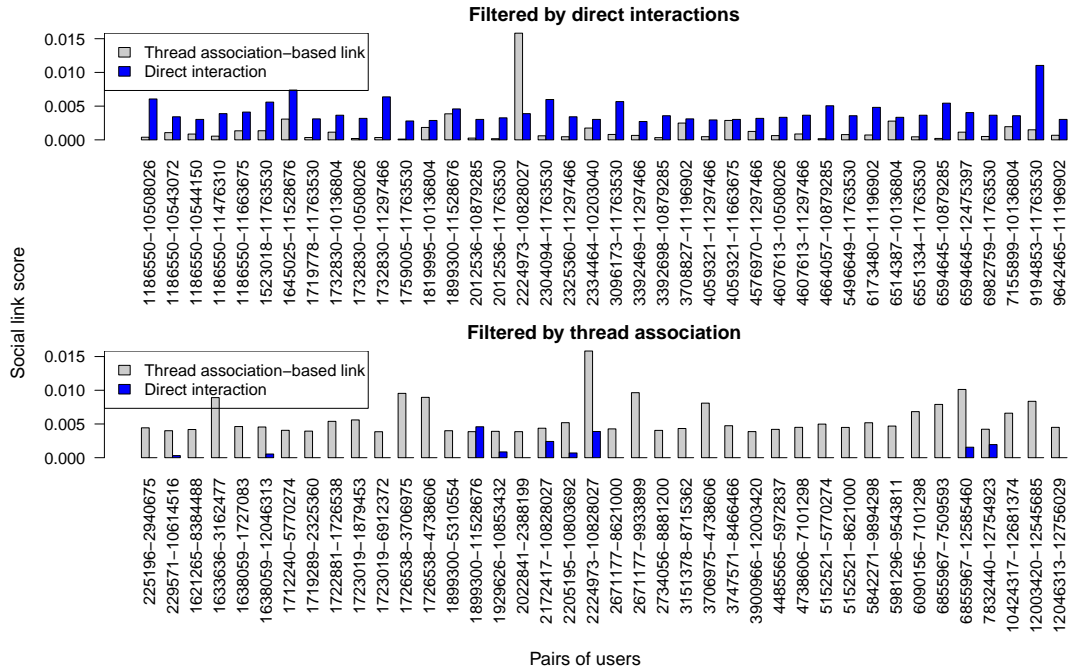
Table 3.5: Statistical measures obtained from thread association-based link graphs of six snapshots from February 2008 to July 2008 of the *BBC Message Boards* network. The results are computed after removing links that have a  $link_{thread}$  weight less than 0.0005.

	February	March	April	May	June	July	Average
<b>Mean of node degree</b>	37.63	37.27	39.64	40.08	38.98	41.13	39.12
<b>Graph density</b>	0.076	0.075	0.081	0.088	0.078	0.080	0.079
<b>Clustering coefficient</b>	0.360	0.360	0.500	0.387	0.383	0.370	0.390

The modularity-based clustering algorithm [87] is again employed to extract community structures. As expected, more communities are found in a thread association-based link graph compared to those extracted from the corresponding direct interaction graph. The main community structures are shown in Figure 3.14.



(a) snapshot: June 2008



(b) snapshot: July 2008

Figure 3.12: Social link scores between users obtained from  $link_{thread}$  and  $link_{direct}$  measures for two snapshots: June 2008 and July 2008. Many pairs of users having high  $link_{direct}$  but low  $link_{thread}$  scores indicate that users who tend to reply to each other only participate in a few number of threads. On the other hand, pairs of users having high  $link_{thread}$  but low  $link_{direct}$  scores highlight that such users have common interests but do not directly interact with each other.



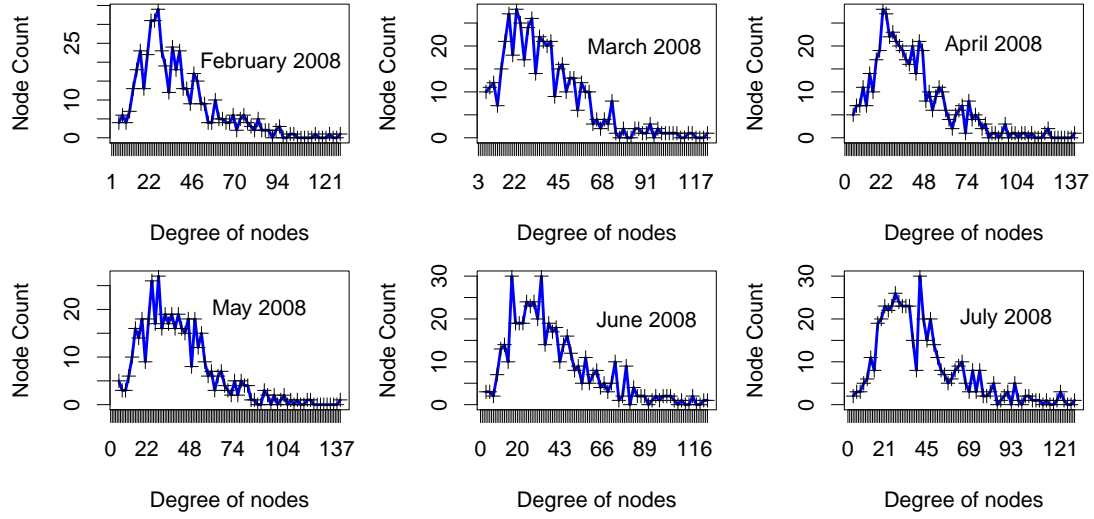


Figure 3.13: Distribution of the degrees of nodes in graphs derived from applying  $link_{thread}$  measure to six monthly snapshots from February to July 2008.

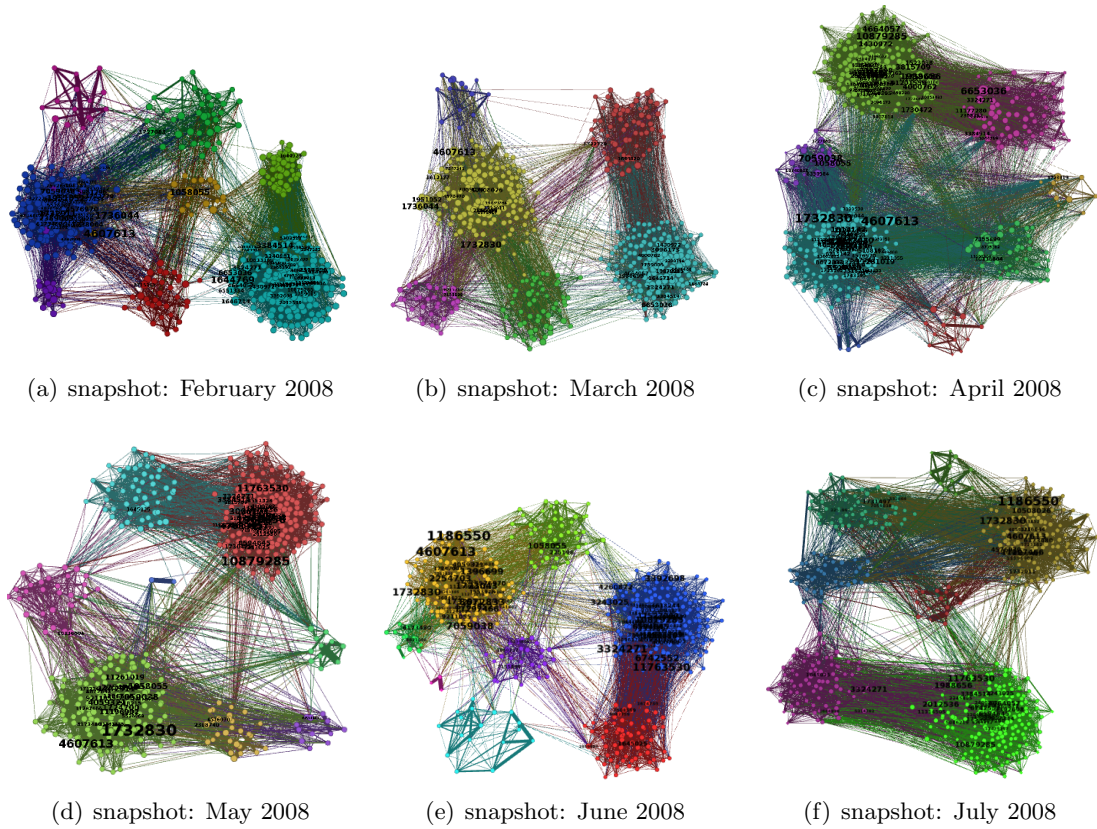


Figure 3.14: Community structures in graphs derived from applying  $link_{thread}$  measure to six monthly snapshots from February to July 2008.

**Qualitative evaluation.** For a qualitative evaluation, we compare the results of our  $link_{thread}$  measure and the results derived from employing cosine similarity for measuring social links. Given a user-thread association matrix  $\mathbf{B}$  (Def. 3.4), the cosine similarity between two users  $u_i$  and  $u_j$  is computed as follows.

$$cosine(u_i, u_j) = \frac{u_i \cdot u_j}{|u_i| \times |u_j|} = \frac{\sum_{z \in Z} \mathbf{B}[u_i, z] \times \mathbf{B}[u_j, z]}{\sqrt{\sum_{z \in Z} \mathbf{B}[u_i, z]^2} \times \sqrt{\sum_{z \in Z} \mathbf{B}[u_j, z]^2}} \quad (3.15)$$

Figure 3.15 shows the histograms of link scores obtained from  $link_{thread}$  measure and cosine similarity measure, respectively, for four selected snapshots from February 2008 to May 2008. It can be observed from the figure, cosine similarity returns long tail histograms, which indicate that quite many users are considered to be very similar. This is because, by employing cosine similarity, only the association of the two users  $u_i$  and  $u_j$  under consideration in threads is taken into account to assign a link score. Therefore, the participation of other users in a thread  $z$  is neglected in measuring the social link for users  $u_i$  and  $u_j$  in  $z$ . Our  $link_{thread}$  measure, however, not only considers how frequent two users  $u_i$  and  $u_j$  participate in  $z$  but also put their association in the context of the participation of other users in  $z$  as well (see Eq. 3.7). By this, our model only returns high link scores for the pairs of users who really participate in the same threads and are distinguished from other users. This is shown in Figure 3.16, where the highest link scores of pairs of users derived from cosine similarity and from our  $link_{thread}$  measures are plotted. In the figure, two filtering procedures are applied to the link scores between pairs of users in each snapshot. A filtering applied to a measure means that only pairs of users that have the highest link scores derived from using that measure are plotted. By doing this, we find that if two users  $u_i$  and  $u_j$  have a high  $link_{thread}$  score then they also have a high link score obtained from cosine similarity. The other direction, however, does not hold. There are pairs of users that have high link scores derived from cosine similarity but have low  $link_{thread}$  scores. These users are *similar* in terms of taking only their association in threads into account but such a similarity is not much distinguished when considered in the context of the participations in threads of other users in the whole network.

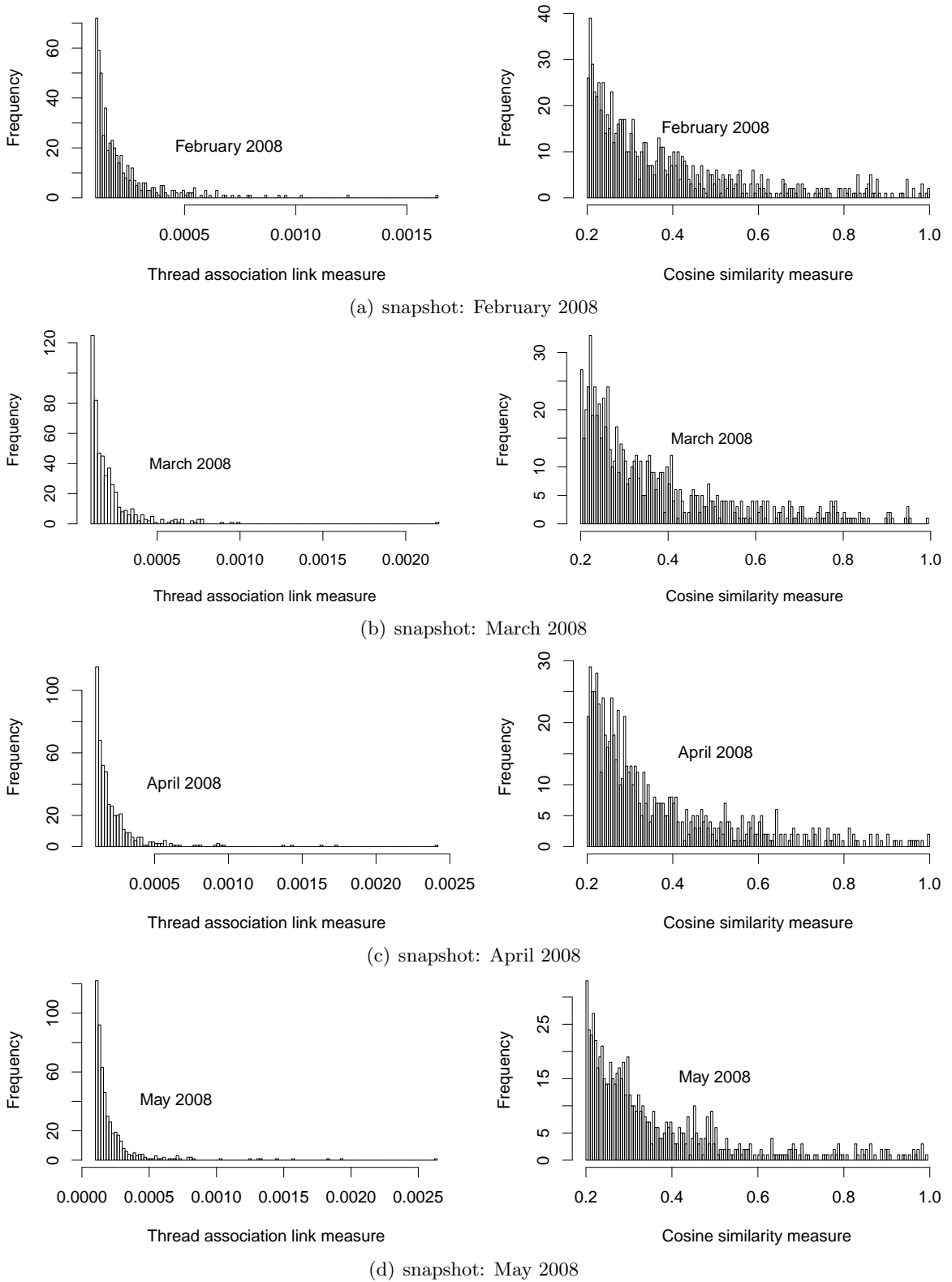
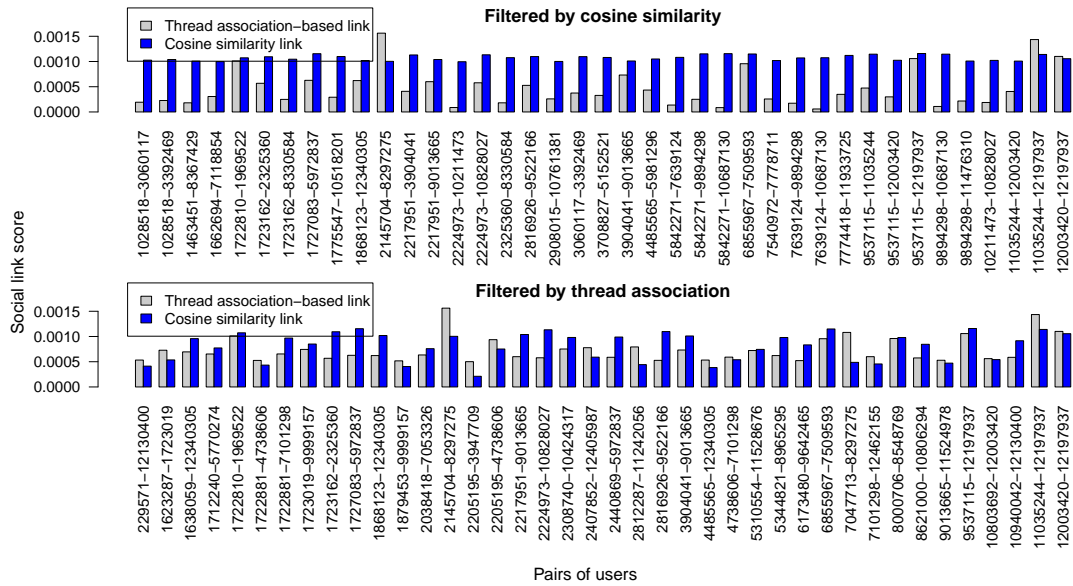
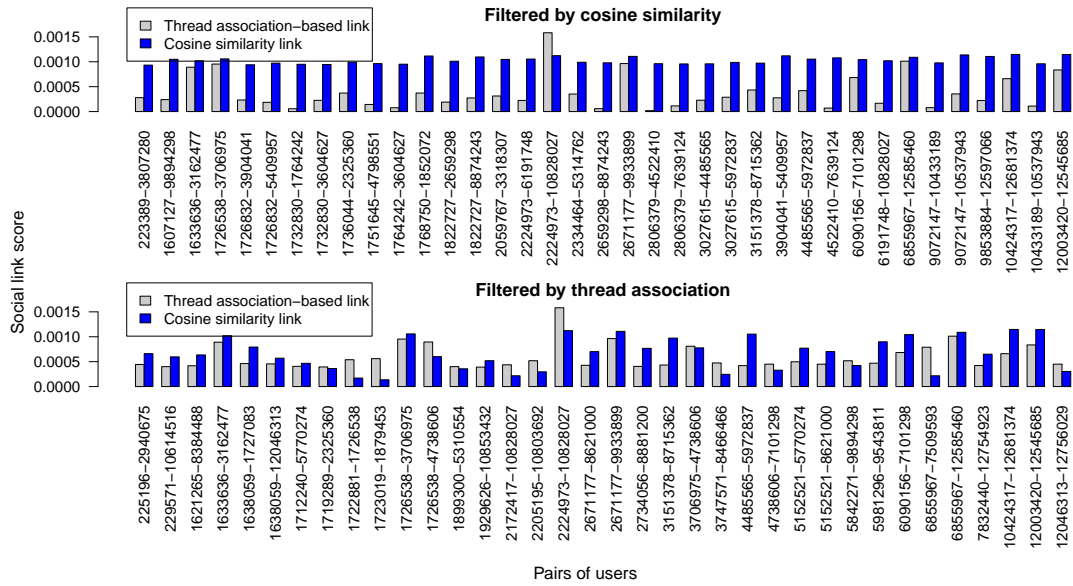


Figure 3.15: Histograms of link scores obtained from  $link_{thread}$  measure (left) and cosine similarity measure (right).



(a) snapshot: June 2008



(b) snapshot: July 2008

Figure 3.16: Comparing link scores derived from  $link_{thread}$  measure and cosine similarity measure in two snapshots: June 2008 and July 2008. Cosine similarity is normalized to sum up to 1 for readability.

### 3.6.3 Latent Semantic-based Network

This section presents the results obtained by applying our latent semantic-based social link measure  $link_{latent}$  (Eq. 3.14) to the *BBC Message Boards* network. We first show that by relying only on the content of the messages to derive social links between users, semantic-based communities can be extracted. We then qualitatively evaluate our model by comparing the results derived with the results obtained from using the typical *TF.IDF* weighting schema. For this purpose, we first employ our weighting schema  $sig(w, u)$  (Eq. 3.12) to derive the significance of terms for users in each snapshot of the network. Singular Value Decomposition is then applied, where the number of dimensions to be reduced is determined using the principle introduced in [100, Chapter 11], i.e., 40% of the dimensions are reduced. Finally, link scores between users are computed using  $link_{latent}$  measure from which a semantic-based link graph is created for each snapshot. Community structures are detected using the modularity-based clustering algorithm [87]. Having these steps accomplished, we find that extracted communities are very cohesive regarding both structural and semantic properties<sup>7</sup>. This is interesting, because no other information was considered in our model except the content of users' postings. In every snapshot from February to July 2008, the number of main communities varies between 4 and 7, among them there are always 4 large communities. By studying the category of messages (provided as ground truth in Table 3.2) posted by users in each community we find that such communities are clearly distinguished by the topics about *Eastern religion*, *Christian religion*, *Ethics and free thought*, *Jewish*, and *TV, Radio and News*. Figure 3.17 shows the proportion of the number of users in semantic-based communities extracted from six selected snapshots.

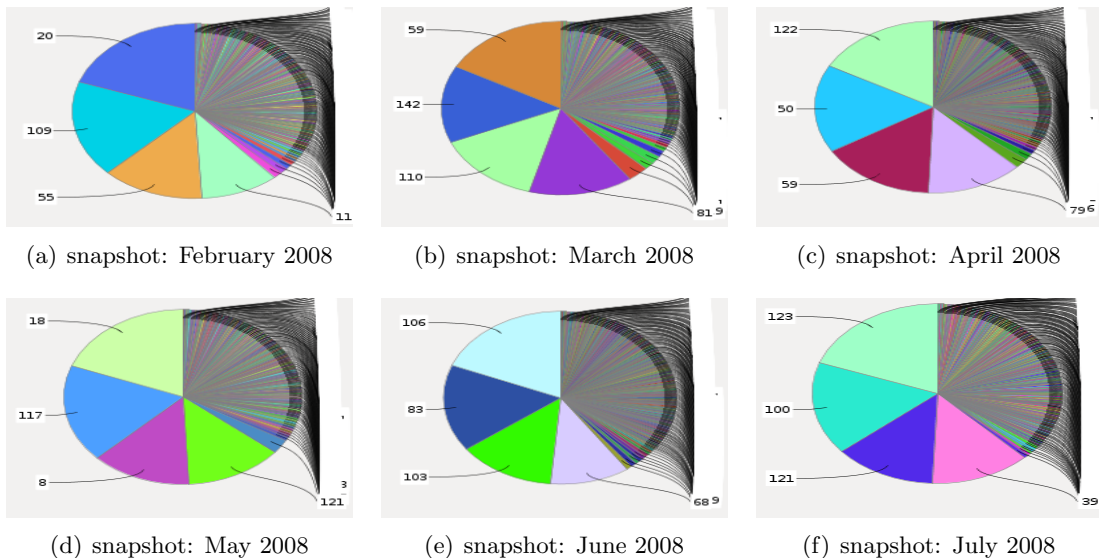


Figure 3.17: Proportion of the number of users in communities extracted from graphs derived using  $link_{latent}$  measure for six snapshots of the *BBC Message Boards* network.

<sup>7</sup>The semantic aspect of communities are determined by extracting messages of users in each communities

**Qualitative evaluation.** We now show that our weighting schema outperforms the *TF.IDF* schema in terms of extracting social characteristics of users from their postings. For this, we analyze the results of the two models applied to the same snapshots of the *BBC Message Boards* network. In order to apply *TF.IDF* schema, all messages of a user in a snapshot are first aggregated to form a single document. By this, each user is characterized by a document and such documents of all users form a corpus. Based on this corpus, the *TF.IDF* schema is employed to compute a term-document matrix that actually plays the role of a term-user matrix  $\mathbf{W}$  (see Section 3.4.1). Singular Value Decomposition is adopted with the same setting as applied in our model. Finally, link scores between users are computed using the same formulas as defined for the  $link_{latent}$  measure (Eq. 3.14). Having such so-called *TF.IDF*-based graphs derived, we then run the modularity-based clustering algorithm [87] to extract community structures. The results show that in each of all six selected snapshots the algorithm only detects 2 or 3 large communities. This implies that many users are considered to be similar when *TF.IDF* is applied. As a consequence, it loses the structure of the semantic-based communities of the network, which are provided by the ground truth (Table 3.2). Figure 3.18 shows the number of users in each community extracted from the corresponding graphs obtained from using our model and using *TF.IDF*, respectively. For each snapshot shown in the figure, the number of large communities extracted from the graph built by our model is close to the number of categorized topics in the network while the number of large *TF.IDF*-based communities in each snapshot varies from 2 to 3.

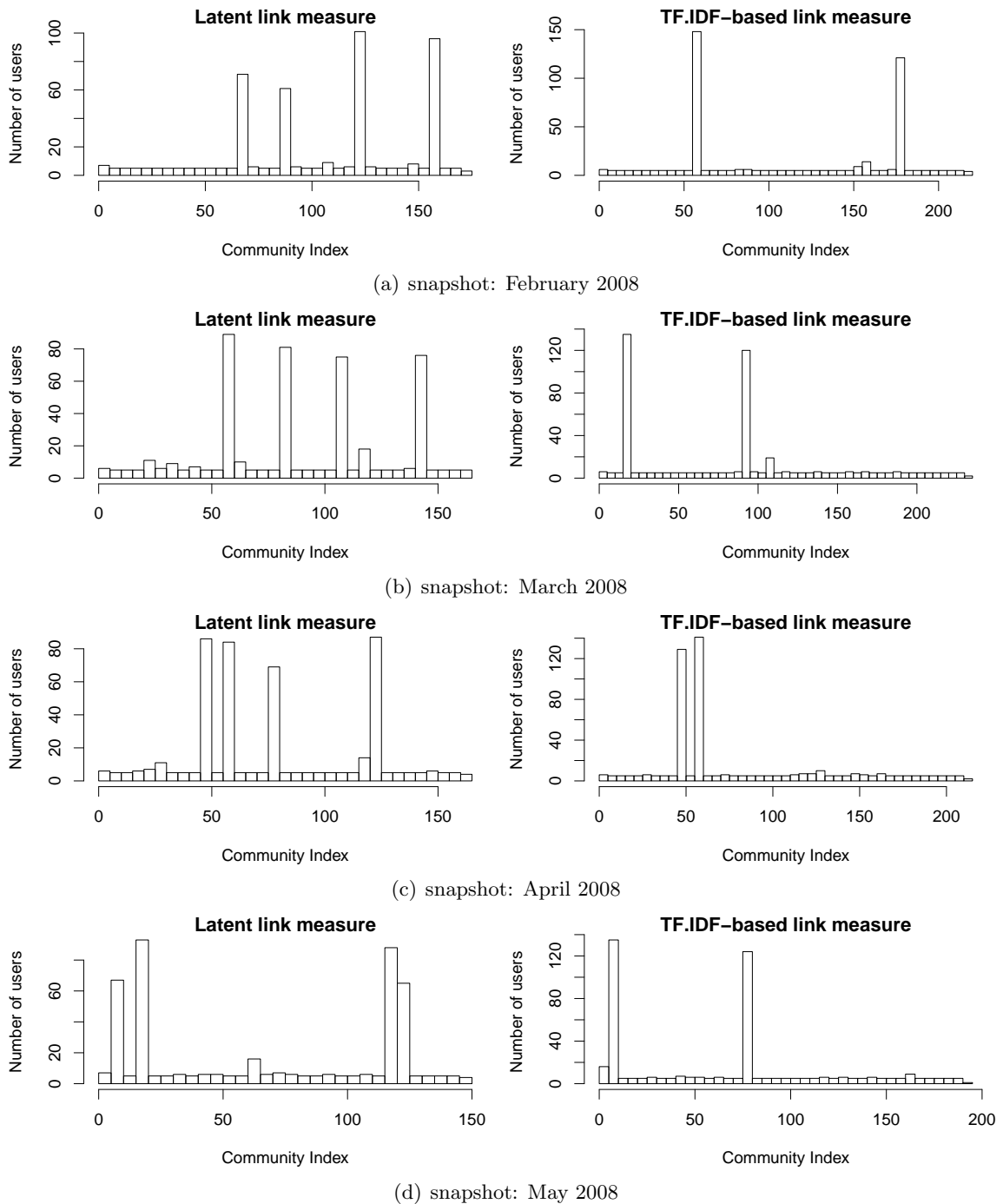


Figure 3.18: Number of users in communities extracted from graphs derived from  $link_{latent}$  measure (left) and from  $TF.IDF$ -based link measure (right). For each snapshot, the number of large communities obtained from our approach is close to the number of categorized topics in the *BBC Message Boards* network while only 2 or 3 large communities are derived when  $TF.IDF$  is employed.

## 3.7 Summary and Discussion

We conclude the chapter by first giving a short summary of the proposed models and then presenting an outlook for extracting and measuring social links between users in location-based social networks (LBSNs).

### 3.7.1 Summary

Extracting social links between users is an important step in social network analysis as its results are used in various applications. In this chapter, we have presented two models for extracting and measuring social links. In the first model ( $link_{int}$ ), relationships between users are identified and measured based on the association of users in threads of discussions ( $link_{thread}$ ) together with their direct interactions ( $link_{direct}$ ). The second model ( $link_{latent}$ ) extracts social characteristics of users reflected in their postings from which so-called semantic-based social links are derived. Since direct interactions are observable and simple to measure, we are more interested in the thread association-based links ( $link_{thread}$ ) and semantic-based links ( $link_{latent}$ ). These two types are referred to as hidden links between users. By using  $link_{thread}$  measure, a high link score between two users indicates that such users frequently occur together in many threads of discussions. For the  $link_{latent}$  measure, a link score is assigned to a pair of users based on the similarity of social characteristics extracted from their postings. The proposed models have been evaluated using a dataset from the *BBC Message Boards* network. The obtained results indicate the utility the models in extracting and measuring hidden social links between users and, thus, further show that more users are socially related than those just observed from direct interactions. It is possible to combine the introduced models to have a social link measure that considers both the interactions between users and the similarity of their social characteristics to assess the relationship between them. By this, a new social link measure can be formally specified as

$$link_{combine}(u_i, u_j) = \beta \times link_{int}(u_i, u_j) + (1 - \beta) \times link_{latent}(u_i, u_j), \quad (3.16)$$

where the value of  $\beta \in [0, 1]$  controls the level of emphasis on the interaction-based social link and the latent semantic-based social link measures.

### 3.7.2 Outlook for LBSN Data

Due to the strong adoption of people worldwide for mobile social networking services, the percentage of social network data that contains information about geographic locations of users is increasing. In such data, location information can be found in the profile of users, in different types of media posted by users including geotagged messages and geotagged photos, or in the history of *check-in* places of users. An intuitive observation is that users sharing a number places they have visited during a certain time period are likely to share



some common interests or are related to each other under some setting (e.g., a group of people traveling together, a group of colleagues in a company). This implies that social links between users can also be extracted from their spatio-temporal mobility history. A method to this could be to model occurrences of users in a snapshot as a user-location bipartite graph  $\mathcal{G} = \langle U, L, E \rangle$  and to derive social links between users based on this graph. The two sets of nodes of the graph are *users*  $U$  and *location nodes*  $L$ . Edges  $E$  of the graph are the connections between users and location nodes.

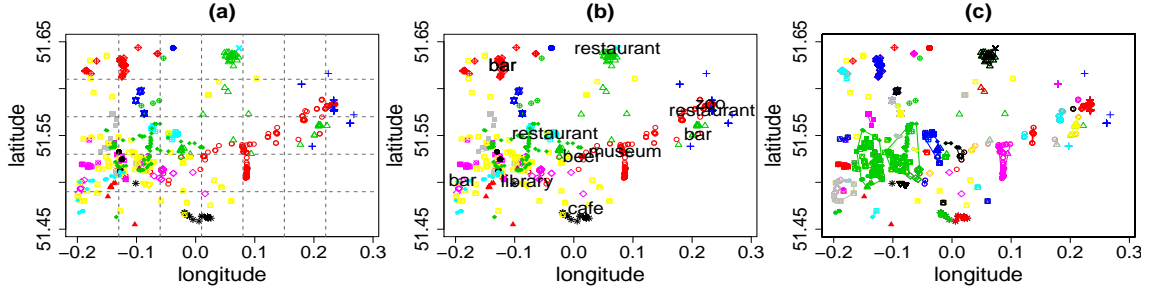


Figure 3.19: An illustration of three methods applied to create location nodes for building a user-location network. (a) a regular grid is created and location nodes are derived from cells of the grid. (b) a set of points of interest (POI) is extracted. A spatial neighborhood of each POI specifies a location node. (c) a density-based clustering algorithm is applied to detect dense clusters. Each cluster is considered a location node.

Generally, each location node  $locNode \in L$  can be a geographic point specified by longitude and latitude coordinates. However, employing such exact geographic coordinates returns a huge number of location nodes. Further, it rarely occurs that different contents are posted exactly from the same location due to the mobility of users and other factors including the limitation of location sensing devices and noise. Indeed, an intuitive observation is that close geographic locations associated with postings often indicate the same (or close by) places from where the postings have been sent. Therefore, a location node should be modeled as a region covering some spatial area rather than only a geographic point. Three methods can be employed to create location nodes for building a user-location network: gridding the spatial area, determining points of interest, and density-based clustering.

**Gridding spatial area.** The first simple strategy is to create a regular grid on the spatial area defined by the bounding box of geographic locations associated with postings of users. The grid is built with an input bandwidth  $r$  specifying the resolution of grid cells. Each cell plays the role of a location node in the user-location network. Figure 3.19 (a) illustrates the gridding method to build location nodes for a small geotagged tweet dataset collected in London in 2012.

**Point of Interest Identification.** The second method is to extract a set of points of interest (POI) from social network data. One can extract frequency-based keywords from texts posted by users and identify POI based on such keywords. That is, only words that occur in user postings more than a given frequency threshold are considered, and the

geographic location associated with each keyword becomes a point of interest. It is possible, however, to apply other methods to specify POI. For example, one can empirically build a list of application-specific words from which POI are derived or one can explicitly specify a number of well-known places in the dataset to be POI. Similar to the gridding method, each identified point of interest together with a specific radius  $r$  define a spatial neighborhood serving as a location node. A sample result of this method applied to the same dataset used in Figure 3.19 (a) is shown in Figure 3.19 (b).

**Density-based Clustering.** The third method is to apply a density-based clustering algorithm with a prior neighborhood radius  $r$  and a threshold of number of neighbors  $minPts$  to find dense clusters. Each cluster is then considered a location node. An illustration of running the density-based clustering [34] on the dataset used in Figures 3.19 (a) and (b) is presented in Figure 3.19 (c). In the figure, each dense cluster is shown with a convex hull covering the geographic locations of users falling in the location node specified by the cluster.

In all three approaches, each location node has a spatial coverage, which is specified by a bandwidth or a radius  $r$  and other parameters depending on the selected algorithm(s). Choosing appropriate values of input parameters is an application specific problem. Generally, adopting a finer granularity of location nodes returns a better model for assessing similarity between users, because only spatially close occurrences of users are clustered into the same location node. However, creating location nodes with a small spatial area requires more computations and does not always give good results. For example, two users visit the same museum and post comments from different locations in that building. Such geographic locations of these users might be assigned to different location nodes, which is unexpected, if location nodes are created with a small spatial area.

Once the location nodes are identified, the user-location network is then derived by connecting users to location nodes based on their occurrence in the spatial area of location nodes. That is, an edge is created between a user node  $u$  and a location node  $locNode$  if the user  $u$  appears at least once in the area of  $locNode$ . Having a user-location network  $\mathcal{G} = \langle U, L, E \rangle$  identified for each snapshot, the next step is to measure how two users are related to each other in terms of their spatio-temporal mobility history. By disregarding the method used to derive the user-location network  $\mathcal{G}$ , each location node  $locNode$  in  $\mathcal{G}$  has a spatial area where geographic locations of some users fall in, as described above. A principle is that users sharing more visited places are more likely to be related to each other. To this extent, one can rely on: (1) the number of location nodes whose spatial area contains geographic locations of two users  $u_i$  and  $u_j$ , and (2) the variance of the number of occurrences of  $u_i$  and  $u_j$  in each shared location node to measure the social link between  $u_i$  and  $u_j$ . Specifically, the larger the number of shared location nodes and the less the variance of the number of occurrences of the two users in each location node, the higher the likelihood that they are socially linked.

## Chapter 4

# Regional LinkTopic Community Extraction

### 4.1 Overview and Objectives

In this chapter, we introduce a model to discover a new type of community called *regional LinkTopic*. A community of this type is formed by users that are geographically close to each other over time, have common interests indicated by the topical similarity of their postings, and are contextually linked to each other. Taking all these features into account to extract communities will obviously return meaningful and practical results. However, it is a challenging task due to the complex constraints to be considered. To date, existing approaches to community detection have paid attention only to the static links and/or the postings of users. The regional aspect and contextual links among users have been neglected in the context of modeling and extracting communities.

To address these gaps and to achieve the goal of extracting regional *LinkTopic* communities, we develop a probabilistic model called *rLinkTopic*. The model jointly considers the spatio-temporal proximity of users with respect to (1) the geotagged messages they post over time, (2) the contextual links embedded in their messages, and (3) message topics to derive communities. This probabilistic approach allows users to have a membership in more than just one community, which is an important feature when discovering communities based on topics. Each community derived is not only characterized by a mixture of topics but also by its geographic and regional properties. Using data from *Twitter*, we demonstrate the effectiveness of our model in extracting regional *LinkTopic* communities, which are described in terms of both geographic locations and topics of interests. The experimental results show that our model outperforms existing models that rely only on links and topics in terms of the perplexity measure and the regional aspect of the communities extracted.

This chapter is structured as follows. Section 4.2 recaps Latent Dirichlet Allocation, a probabilistic model for extracting topics from documents. Section 4.3 formalizes the con-

cepts and definitions underlying our *rLinkTopic* model. In Section 4.4, we discuss the model in detail, including the derivation of the Gibbs sampling rules and the sampling algorithm. A spatial entropy measure for evaluating the geographic localization of communities and the perplexity measure of the model are presented in Section 4.5. The results of the experimental evaluations are shown in Section 4.6. This chapter is concluded with a summary and discussion in Section 4.7.

## 4.2 Topic Models

In the context of topical analysis from text documents and other related applications, Latent Dirichlet Allocation (LDA) [12] is known to be the best model to date. Since its publication in 2003, LDA has been quickly adopted as a powerful tool for extracting clusters of objects in many application domains. These include the topical analysis in text mining [49, 132, 83], object extraction in computer vision [45, 119], and community detection in social network analysis [120, 135, 136]. Even though several models have been introduced as an extension of LDA, it is interesting to note that research communities tend to employ LDA mostly as a black-box. There are few studies contributing to the explanation of the model [44, 51] but still, the authors skipped most of the detailed steps especially for the posterior estimation. Motivated by this and also because the underlying idea of LDA is employed in the *rLinkTopic* model introduced in this chapter, this section gives a more detailed overview of the LDA model.

### 4.2.1 Latent Dirichlet Allocation

LDA is a probabilistic model originally proposed for extracting semantic topics from a corpus of documents. The key idea of the model is that it considers a document as a mixture of topics, a topic being a mixture of terms<sup>1</sup>, and topics are shared among documents [12, 113]. Particularly, given a corpus of documents  $\mathcal{D} = \{d_1, d_2, \dots, d_{|\mathcal{D}|}\}$  built from a vocabulary set consisting of  $|V|$  terms,  $V = \{w_1, w_2, \dots, w_{|V|}\}$ , LDA considers words occurring in any document  $d$  in the corpus to be independently sampled from a common number of topics  $Z = \{z_1, z_2, \dots, z_{|Z|}\}$ . One can, therefore, assume that the topics  $Z$  are shared among documents. Another assumption employed in LDA is that documents as well as words within each document are considered to be exchangeable, respectively. To learn the mixture of topics in a document and the mixture of terms in a topic, a probabilistic framework was introduced, which works as follows.

The mixture of terms in a topic  $z \in Z$  is modeled as a multinomial distribution specified by a multinomial parameter  $\phi_z = \{\phi_{z,w_1}, \phi_{z,w_2}, \dots, \phi_{z,w_{|V|}}\}$ . Each  $\phi_{z,w}$  is the probability that term  $w$  belongs to topic  $z$ , denoted  $P(w|\phi_z)$ , such that  $\sum_{w \in V} P(w|\phi_z) = 1$ . The mixture

---

<sup>1</sup>In this thesis, “term” is used to refer to an element of a vocabulary while “word” is used to indicate a particular observation of a term.

of topics in a document  $d$ , usually referred to as *the topic proportion* of the document, is also modeled as a multinomial parameter  $\theta_d = \{\theta_{d,z_1}, \theta_{d,z_2}, \dots, \theta_{d,z_{|Z|}}\}$ . Each  $\theta_{d,z}$  indicates the likelihood of topic  $z$  in document  $d$ , denoted  $P(z|\theta_d)$ , such that  $\sum_{z \in Z} P(z|\theta_d) = 1$ .

Obviously, if one knows the distribution of terms in topic  $z$  and the topic proportion of document  $d$  beforehand, then the probability that a word  $w$  in  $d$  belongs to topic  $z$  would be

$$P(w, z|\phi_z, \theta_d) = P(z|\theta_d)P(w|\phi_z) = \theta_{d,z}\phi_{z,w}. \quad (4.1)$$

However, generally, we are given a corpus of documents and asked to find some topics in these documents without having knowledge about the distribution of terms in topics and the proportion of topics in documents. In other words, not only the topic that a word should be assigned to but also the distribution of terms in any topic ( $\phi_z$ ) and the topic proportion of any document ( $\theta_d$ ) are hidden. One therefore has to learn such hidden variables from the occurrences of terms in the corpus.

Suppose each of the two variables  $\phi_z$  and  $\theta_d$  is generated by a probability distribution, denoted  $P(\phi_z|\beta)$  and  $P(\theta_d|\alpha)$ , respectively, where  $\alpha$  and  $\beta$  are the hyperparameters of the corresponding distribution; then the joint probability of word  $w$  and topic  $z$  in document  $d$  is

$$P(w, z, \phi_z, \theta_d|\alpha, \beta) = P(\phi_z|\beta)P(\theta_d|\alpha)P(z|\theta_d)P(w|\phi_z), \quad (4.2)$$

and the joint distribution of all words and topics in document  $d$  becomes<sup>2</sup>

$$P(d, \mathbf{z}, \phi, \theta_d|\alpha, \beta) = \prod_{z \in Z} P(\phi_z|\beta) \times P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d)P(w|\phi_{z_w}), \quad (4.3)$$

where  $\phi = \{\phi_z\}$ ,  $\mathbf{z} = \{z_w\}$ ,  $w \in d$ . Each  $z_w \in \mathbf{z}$  is a topic index (i.e.,  $1..|Z|$ ) indicating the topic assignment of word  $w$  in document  $d$ . Finally, the joint distribution of words and topics in the entire corpus, which is referred to as the joint distribution of the LDA model, is

$$P(\mathcal{D}, \mathbf{z}, \phi, \theta|\alpha, \beta) = \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{d \in \mathcal{D}} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d)P(w|\phi_{z_w}), \quad (4.4)$$

where  $\theta = \{\theta_d\}$ ,  $d \in \mathcal{D}$ .

Substituting  $P(z_w|\theta_d)$  and  $P(w|\phi_{z_w})$  in Eq. 4.4 by the respective multinomial components, i.e.,  $\theta_{d,z_w}$  of the topic proportion  $\theta_d$ , and  $\phi_{z_w,w}$  of the distribution  $\phi_{z_w}$  of terms in topic  $z_w$ , we have

$$P(\mathcal{D}, \mathbf{z}, \phi, \theta|\alpha, \beta) = \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{d \in \mathcal{D}} P(\theta_d|\alpha) \prod_{w \in d} \theta_{d,z_w}\phi_{z_w,w}. \quad (4.5)$$

To complete the model, one needs to specify the probability distributions that generate samples of the distribution  $\phi_z$  of terms in a topic, and the topic proportion  $\theta_d$  of a document.

---

<sup>2</sup>In this thesis, the symbol  $\times$  is often used instead of a large parenthesis to separate terms for readability.

As presented above, both  $\phi_z$  and  $\theta_d$  are modeled as multinomial parameters. Therefore, the *Dirichlet* distribution is used as prior of  $\phi_z$  and  $\theta_d$ . This is due to the conjugacy between the *Dirichlet* and *Multinomial* distributions as discussed in Section 2.5.3. Thus, one can now present the joint distribution of the LDA model in a more specific way as<sup>3</sup>

$$P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha; \beta) = \prod_{z \in Z} Dir(\phi_z | \beta) \times \prod_{d \in \mathcal{D}} Dir(\theta_d | \alpha) \prod_{w \in d} \theta_{d, z_w} \phi_{z_w, w}, \quad (4.6)$$

where  $\alpha = \langle \alpha_1, \alpha_2, \dots, \alpha_{|Z|} \rangle$  and  $\beta = \langle \beta_1, \beta_2, \dots, \beta_{|V|} \rangle$  are the hyperparameters of the *Dirichlet* distributions, which present prior knowledge for the topic proportion of a document and the distribution of terms in a topic, respectively.

Figure 4.1 shows the graphical models explaining three main joint distributions in the LDA model. (a) and (b) are the graphical models of Eq. 4.2 and Eq. 4.3, respectively; (c) is the complete graphical model of LDA represented by Eq. 4.4.

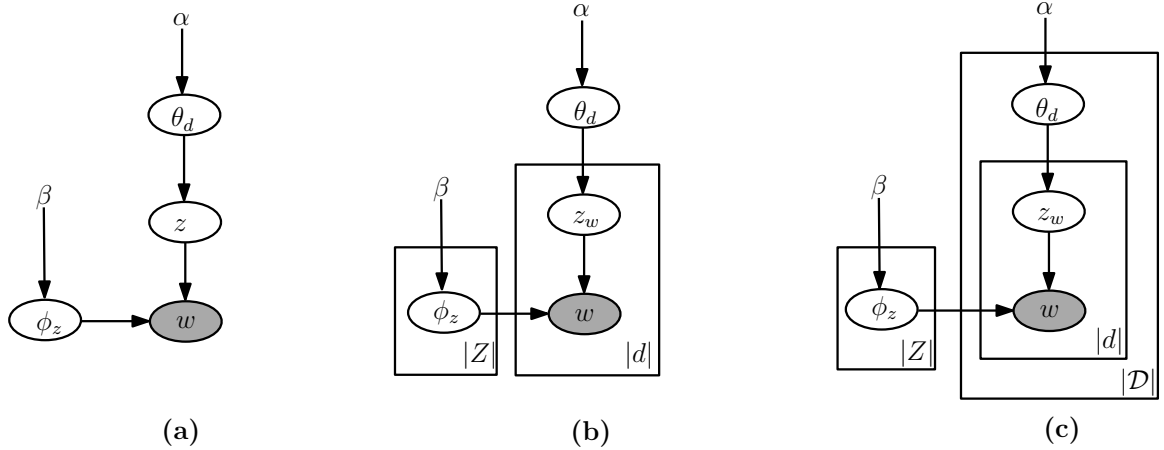


Figure 4.1: Graphical models representing selected joint distributions in the LDA model. (a) is the joint distribution of word  $w$  in topic  $z$  of document  $d$ ; (b) is the joint distribution of all words and topics in document  $d$ ; (c) is the complete graphical model of LDA.

**Generative process.** Having the graphical model shown in Figure 4.1(c), the generative process of the LDA model is as follows.

1. sample the distributions of terms in topics

$$\phi = \{\phi_z \sim Dir_{|V|}(\beta)\}, z \in Z$$

2. for each document  $d$

- 2.1. sample topic proportion  $\theta_d \sim Dir_{|Z|}(\alpha)$

- 2.2. for each word  $w$  in document  $d$

- a. sample a topic index  $z \sim Mult(\theta_d)$

- b. sample term  $w$  in the selected topic  $z$ , i.e.,  $w \sim Mult(\phi_z)$

In the following section, the detailed steps to derive the Gibbs sampling rules for estimating the distributions of hidden variables in LDA are presented.

<sup>3</sup>The notation *Dir* is used as shorthand for the *Dirichlet* distribution.

### 4.2.2 Gibbs Sampling for LDA

There are hidden variables represented by  $\mathbf{z}$  (topic assignments),  $\phi$  (distributions of terms in topics), and  $\theta$  (topic proportions of documents) in the LDA model. The posterior distribution of such variables is analytically obtained using *Bayes' theorem* as in Eq. 4.7. This distribution is, however, intractable to compute due to the marginalization over the hidden variables [12].

$$P(\mathbf{z}, \phi, \theta | \mathcal{D}; \alpha, \beta) = \frac{P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta)}{P(\mathcal{D} | \alpha, \beta)} = \frac{P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta)}{\int_{\phi} \int_{\theta} \sum_{\mathbf{z} \in Z} P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta) d\theta d\phi} \quad (4.7)$$

By applying sampling, the posterior distribution is approximated through the samples of the joint distribution as shown in Eq. 4.8.

$$P(\mathbf{z}, \phi, \theta | \mathcal{D}; \alpha, \beta) = \frac{P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta)}{P(\mathcal{D} | \alpha, \beta)} \propto P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta) \quad (4.8)$$

Generally, implementing a Gibbs sampling algorithm for all variables in the LDA model is straightforward. However, it is inefficient due to the sampling for the multinomial parameters  $\phi$  and  $\theta$ , which can be computed from the topic assignment variables  $\mathbf{z}$ . In other words, it is better to make use of the conjugacy between the *Dirichlet* and the *Multinomial* distributions to integrate out the multinomial parameters  $\theta$  and  $\phi$  in Eq. 4.8 and build a collapsed Gibbs sampling for  $\mathbf{z}$  from which  $\theta$  and  $\phi$  are then derived. In the following, the detailed steps to integrate out  $\theta$  and  $\phi$  are given.

First, from Eq. 4.8, the joint distribution of the topic assignments of all words in the corpus is obtained by

$$P(\mathbf{z} | \mathcal{D}; \alpha, \beta) = \int_{\phi} \int_{\theta} P(\mathbf{z}, \phi, \theta | \mathcal{D}; \alpha, \beta) d\theta d\phi \propto \int_{\phi} \int_{\theta} P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha, \beta) d\theta d\phi. \quad (4.9)$$

It is noted that the second term in Eq. 4.4 can be represented as

$$\prod_{d \in \mathcal{D}} P(\theta_d | \alpha) \prod_{w \in d} P(z_w | \theta_d) P(w | \phi_{z_w}) = \prod_{d \in \mathcal{D}} \prod_{w \in d} P(w | \phi_{z_w}) \times \prod_{d \in \mathcal{D}} P(\theta_d | \alpha) \prod_{w \in d} P(z_w | \theta_d). \quad (4.10)$$

Therefore, the joint distribution of the LDA model (Eq. 4.4) can be rewritten as follows.

$$P(\mathcal{D}, \mathbf{z}, \phi, \theta | \alpha; \beta) = \prod_{z \in Z} P(\phi_z | \beta) \times \prod_{d \in \mathcal{D}} \prod_{w \in d} P(w | \phi_{z_w}) \times \prod_{d \in \mathcal{D}} P(\theta_d | \alpha) \prod_{w \in d} P(z_w | \theta_d) \quad (4.11)$$

By applying Eq. 4.11 to Eq. 4.9 and using the notation  $n_w^{(z)}$  to denote the number of occurrences of term  $w$  assigned to topic  $z$ , we have

$$\begin{aligned}
P(\mathbf{z}|\mathcal{D}; \alpha, \beta) &\propto \int_{\phi} \int_{\theta} \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{d \in \mathcal{D}} \prod_{w \in d} P(w|\phi_{z_w}) \times \prod_{d \in \mathcal{D}} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta d\phi \\
&= \int_{\phi} \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{d \in \mathcal{D}} \prod_{w \in d} P(w|\phi_{z_w}) \times \int_{\theta} \prod_{d \in \mathcal{D}} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta d\phi \\
&= \int_{\theta} \prod_{d \in \mathcal{D}} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta \times \int_{\phi} \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{d \in \mathcal{D}} \prod_{w \in d} P(w|\phi_{z_w}) d\phi \\
&= \prod_{d \in \mathcal{D}} \int_{\theta_d} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta_d \times \int_{\phi} \prod_{z \in Z} P(\phi_z|\beta) \times \prod_{z \in Z} \prod_{w \in V} P(w|\phi_{z_w})^{n_w^{(z)}} d\phi \\
&= \prod_{d \in \mathcal{D}} \int_{\theta_d} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta_d \times \int_{\phi} \prod_{z \in Z} P(\phi_z|\beta) \prod_{w \in V} P(w|\phi_{z_w})^{n_w^{(z)}} d\phi \\
&= \underbrace{\prod_{d \in \mathcal{D}} \int_{\theta_d} P(\theta_d|\alpha) \prod_{w \in d} P(z_w|\theta_d) d\theta_d}_{(*)} \times \underbrace{\prod_{z \in Z} \int_{\phi_z} P(\phi_z|\beta) \prod_{w \in V} P(w|\phi_{z_w})^{n_w^{(z)}} d\phi_z}_{(**)}. \tag{4.12}
\end{aligned}$$

Substituting  $P(\theta_d|\alpha)$  by the *Dirichlet* distribution, and  $P(z_w|\theta_d)$  by the corresponding component  $\theta_{d,z_w}$  of the multinomial parameter  $\theta_d$ , the first term (\*) in Eq. 4.12 becomes

$$\begin{aligned}
(*) &= \prod_{d \in \mathcal{D}} \int_{\theta_d} \overbrace{\frac{\Gamma(\sum_{z \in Z} \alpha_z)}{\prod_{z \in Z} \Gamma(\alpha_z)}}^{\text{just a constant}} \prod_{z \in Z} \theta_{d,z}^{\alpha_z - 1} \prod_{w \in d} \theta_{d,z_w} d\theta_d \\
&\propto \prod_{d \in \mathcal{D}} \int_{\theta_d} \prod_{z \in Z} \theta_{d,z}^{\alpha_z - 1} \prod_{z \in Z} \theta_{d,z}^{n_z^{(d)}} d\theta_d = \prod_{d \in \mathcal{D}} \int_{\theta_d} \prod_{z \in Z} \theta_{d,z}^{n_z^{(d)} + \alpha_z - 1} d\theta_d = \prod_{d \in \mathcal{D}} \frac{\prod_{z \in Z} \Gamma(n_z^{(d)} + \alpha_z)}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)}, \tag{4.13}
\end{aligned}$$

where  $n_z^{(d)}$  is the number of words in document  $d$  that were assigned to topic  $z$ .

Similarly, substituting  $P(\phi_z|\beta)$  by the *Dirichlet* distribution, and  $P(w|\phi_{z_w})$  by the corresponding component  $\phi_{z_w,w}$  of the multinomial parameter  $\phi_{z_w}$ , the second term (\*\*) in Eq. 4.12 becomes

$$\begin{aligned}
(**) &= \prod_{z \in Z} \int_{\phi_z} \overbrace{\frac{\Gamma(\sum_{w \in V} \beta_w)}{\prod_{w \in V} \Gamma(\beta_w)}}^{\text{just a constant}} \prod_{w \in V} \phi_{z,w}^{\beta_w - 1} \prod_{w \in V} \phi_{z,w}^{n_w^{(z)}} d\phi_z \\
&\propto \prod_{z \in Z} \int_{\phi_z} \prod_{w \in V} \phi_{z,w}^{n_w^{(z)} + \beta_w - 1} d\phi_z = \prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \beta_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \beta_w)}. \tag{4.14}
\end{aligned}$$

Substituting the results of Eq. 4.13 and Eq. 4.14 for the corresponding terms in Eq. 4.12, we have

$$P(\mathbf{z}|\mathcal{D}; \alpha, \beta) \propto \underbrace{\prod_{d \in \mathcal{D}} \frac{\prod_{z \in Z} \Gamma(n_z^{(d)} + \alpha_z)}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)}}_{(T_1)} \times \underbrace{\prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \beta_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \beta_w)}}_{(T_2)}. \tag{4.15}$$



Intuitively, the first term  $T_1$  in Eq. 4.15 indicates the joint distribution of topics  $Z$  in documents  $\mathcal{D}$  whereas the second term  $T_2$  is the joint distribution of terms  $V$  in topics  $Z$ .

To derive the likelihood of a word  $w$  in a topic, denoted  $z_w$ , the joint distribution of the topic assignments in Eq. 4.15 is rewritten as

$$P(\mathbf{z}|\mathcal{D}; \alpha, \beta) = P(z_w, \mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta) = P(z_w|\mathbf{z}_{-w}, \mathcal{D}; \alpha, \beta)P(\mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta), \quad (4.16)$$

where  $z_w$  is the topic assignment of word  $w$ , and  $\mathbf{z}_{-w}$  are the topic assignments of all words in the corpus except  $w$ . Therefore, we have

$$P(z_w|\mathbf{z}_{-w}, \mathcal{D}; \alpha, \beta) = \frac{P(z_w, \mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)}{P(\mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)} = \frac{P(\mathbf{z}|\mathcal{D}; \alpha, \beta)}{P(\mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)}. \quad (4.17)$$

Notice that the only difference between the numerator and the denominator in Eq. 4.17 is that the numerator is the joint distribution of the topic assignments of all words whereas the information about the topic assignment of the currently considered word  $w$  is removed in the denominator. Using the notations  $T_1$  and  $T_2$  in Eq. 4.15, we represent Eq. 4.17 as

$$P(z_w|\mathbf{z}_{-w}, \mathcal{D}; \alpha, \beta) = \frac{P(z_w, \mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)}{P(\mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)} = \frac{P(\mathbf{z}|\mathcal{D}; \alpha, \beta)}{P(\mathbf{z}_{-w}|\mathcal{D}; \alpha, \beta)} = \frac{T_1}{T_{-w,1}} \times \frac{T_2}{T_{-w,2}}, \quad (4.18)$$

where  $T_{-w,1}$  and  $T_{-w,2}$  are computed using the same formula as of  $T_1$  and  $T_2$ , respectively, having the topic assignment of the currently considered word  $w$  discarded. In the following, information that is independent of the assignment of word  $w$  to topic  $z_w$  is first removed from  $T_1$  and  $T_2$ . The resulting  $T_1$  and  $T_2$  are then represented in terms of  $T_{-w,1}$  and  $T_{-w,2}$ , respectively, to simplify the formula in Eq. 4.18.

Note that by employing the local Markov assumption the probability that word  $w$  in document  $d$  belongs to topic  $z$  depends only on (1) the occurrences of term  $w$  in  $z$  (i.e., more occurrences of  $w$  in  $z$  intuitively imply that this occurrence  $w$  should be assigned to  $z$  as well), and (2) the presence of other words in document  $d$  in  $z$  (i.e., more words in document  $d$  appearing in  $z$  indicate that  $d$  is likely talking about that topic; therefore,  $w$  should also be assigned to  $z$ ). In other words, such a probability does not depend on the occurrence of other terms in other documents on any topic and it does not depend on the presence of words in  $d$  on other topics.

Suppose the word  $w$  in  $P(z_w|\mathbf{z}_{-w}, \mathcal{D}; \alpha, \beta)$  is in document  $d$ ; then, the information that is independent of the computation of the likelihood of  $w$  in  $z_w$  can be removed from the two terms  $T_1$  and  $T_2$  as follows.

$$\begin{aligned}
T_1 &= \frac{\prod_{z \in Z} \Gamma(n_z^{(d)} + \alpha_z)}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)} \overbrace{\prod_{d' \in \mathcal{D} \setminus d} \frac{\prod_{z \in Z} \Gamma(n_z^{(d')} + \alpha_z)}{\Gamma(\sum_{z \in Z} n_z^{(d')} + \alpha_z)}}^{\text{independent of document } d} \\
&\propto \frac{\prod_{z \in Z} \Gamma(n_z^{(d)} + \alpha_z)}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)} = \frac{\Gamma(n_{z_w}^{(d)} + \alpha_{z_w}) \overbrace{\prod_{z \in Z \setminus z_w} \Gamma(n_z^{(d)} + \alpha_z)}^{\text{independent of topic } z_w}}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)} \propto \frac{\Gamma(n_{z_w}^{(d)} + \alpha_{z_w})}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)} \quad (4.19)
\end{aligned}$$

$$\begin{aligned}
T_2 &= \prod_{z \in Z} \frac{\Gamma(n_w^{(z)} + \beta_w) \overbrace{\prod_{w' \in V \setminus w} \Gamma(n_{w'}^{(z)} + \beta_{w'})}^{\text{independent of term } w}}{\Gamma(\sum_{w' \in V} n_{w'}^{(z)} + \beta_{w'})} \propto \prod_{z \in Z} \frac{\Gamma(n_w^{(z)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{w'}^{(z)} + \beta_{w'})} \\
&= \frac{\Gamma(n_w^{(z_w)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{w'}^{(z_w)} + \beta_{w'})} \prod_{z \in Z \setminus z_w} \frac{\Gamma(n_w^{(z)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{w'}^{(z)} + \beta_{w'})} \propto \frac{\Gamma(n_w^{(z_w)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{w'}^{(z_w)} + \beta_{w'})} \quad (4.20)
\end{aligned}$$

In order to represent  $T_1$  and  $T_2$  in terms of  $T_{-w,1}$  and  $T_{-w,2}$ , we use  $n_{-w,z}^{(d)}$  and  $n_{-w,w}^{(z)}$ , respectively, to indicate the number of words in document  $d$  assigned to topic  $z$ , and the number of times occurrences of term  $w$  assigned to  $z$  excluding the currently considered word  $w$ . Employing these notations,  $T_1$  is now represented as

$$\begin{aligned}
T_1 &\propto \frac{\Gamma(n_{z_w}^{(d)} + \alpha_{z_w})}{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)} = \frac{\Gamma(1 + n_{-w,z_w}^{(d)} + \alpha_{z_w})}{\Gamma(1 + \sum_{z \in Z} n_{-w,z}^{(d)} + \alpha_z)} \quad (4.21) \\
&= \frac{(n_{-w,z_w}^{(d)} + \alpha_{z_w})}{(\sum_{z \in Z} n_{-w,z}^{(d)} + \alpha_z)} \times \frac{\Gamma(n_{-w,z_w}^{(d)} + \alpha_{z_w})}{\Gamma(\sum_{z \in Z} n_{-w,z}^{(d)} + \alpha_z)} = \frac{n_{-w,z_w}^{(d)} + \alpha_{z_w}}{\sum_{z \in Z} n_{-w,z}^{(d)} + \alpha_z} \times T_{-w,1}.
\end{aligned}$$

Similarly,  $T_2$  is simplified by

$$\begin{aligned}
T_2 &\propto \frac{\Gamma(n_w^{(z_w)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{w'}^{(z_w)} + \beta_{w'})} = \frac{\Gamma(1 + n_{-w,w}^{(z_w)} + \beta_w)}{\Gamma(1 + \sum_{w' \in V} n_{-w,w'}^{(z_w)} + \beta_{w'})} \quad (4.22) \\
&= \frac{(n_{-w,w}^{(z_w)} + \beta_w)}{(\sum_{w' \in V} n_{-w,w'}^{(z_w)} + \beta_{w'})} \times \frac{\Gamma(n_{-w,w}^{(z_w)} + \beta_w)}{\Gamma(\sum_{w' \in V} n_{-w,w'}^{(z_w)} + \beta_{w'})} = \frac{n_{-w,w}^{(z_w)} + \beta_w}{\sum_{w' \in V} n_{-w,w'}^{(z_w)} + \beta_{w'}} \times T_{-w,2}.
\end{aligned}$$

Finally, by substituting the results of Eq. 4.21 and Eq. 4.22 for the corresponding terms in Eq. 4.17, we have

$$P(z_w | \mathbf{z}_{-w}, \mathcal{D}; \alpha, \beta) \propto \frac{n_{-w,z_w}^{(d)} + \alpha_{z_w}}{\sum_{z \in Z} n_{-w,z}^{(d)} + \alpha_z} \times \frac{n_{-w,w}^{(z_w)} + \beta_w}{\sum_{w' \in V} n_{-w,w'}^{(z_w)} + \beta_{w'}}. \quad (4.23)$$

Intuitively, Eq. 4.23 states that the probability of word  $w$  in document  $d$  being assigned to topic  $z_w$  is proportional to (1) the number of words in  $d$  already assigned to  $z_w$  and (2) the number of times term  $w$  occurred in  $z_w$ . In other words, the first ratio in Eq. 4.23 expresses the probability of topic  $z_w$  in document  $d$ , and the second ratio expresses the probability of  $w$  in topic  $z_w$ .

Once a sample  $\mathbf{z}$  of the topic assignments is computed, the topic proportion  $\theta_d$  of a document and the distribution  $\phi_z$  of terms in a topic are derived as follows.

**Topic proportion.** Given the topic assignments of words in document  $d$ , denoted  $\mathbf{z}_d$ , the distribution of topics in  $d$  is obtained by

$$\begin{aligned} P(\theta_d | \mathbf{z}_d; \alpha) &= \frac{P(\mathbf{z}_d, \theta_d | \alpha)}{P(\mathbf{z}_d | \alpha)} = \frac{P(\mathbf{z}_d | \theta_d) P(\theta_d | \alpha)}{P(\mathbf{z}_d | \alpha)} = \frac{\prod_{w \in d} P(z_w | \theta_d) P(\theta_d | \alpha)}{\int_{\theta_d} \prod_{w \in d} P(z_w | \theta_d) P(\theta_d | \alpha) d\theta_d} \\ &= \frac{\prod_{z \in Z} \theta_{d,z}^{n_z^{(d)}} \frac{1}{\text{Beta}(\alpha)} \prod_{z \in Z} \theta_{d,z}^{\alpha_z - 1}}{\int_{\theta_d} \prod_{z \in Z} \theta_{d,z}^{n_z^{(d)}} \frac{1}{\text{Beta}(\alpha)} \prod_{z \in Z} \theta_{d,z}^{\alpha_z - 1} d\theta_d} = \frac{\Gamma(\sum_{z \in Z} n_z^{(d)} + \alpha_z)}{\prod_{z \in Z} \Gamma(n_z^{(d)} + \alpha_z)} \prod_{z \in Z} \theta_{d,z}^{n_z^{(d)} + \alpha_z - 1}. \end{aligned} \quad (4.24)$$

The final result of Eq. 4.24 is the updated *Dirichlet* distribution of  $\theta_d$  after observing the words in document  $d$ , i.e.,  $\text{Dir}(\theta_d | n^{(d)} + \alpha)$  where  $n^{(d)} = \langle n_1^{(d)}, n_2^{(d)}, \dots, n_{|Z|}^{(d)} \rangle$  and  $\alpha = \langle \alpha_1, \alpha_2, \dots, \alpha_{|Z|} \rangle$ . Therefore, the likelihood of topic  $z$  in document  $d$  is obtained from the expectation of  $\text{Dir}(\theta_d | n^{(d)} + \alpha)$ , computed by

$$\theta_{d,z} = \frac{n_z^{(d)} + \alpha_z}{\sum_{z' \in Z} n_{z'}^{(d)} + \alpha_{z'}}, \quad d \in \mathcal{D}, z \in Z. \quad (4.25)$$

**Distribution of terms in a topic.** Given words assigned to topic  $z$ , denoted  $\mathbf{z}_z$ , the distribution of terms in  $z$  is derived from

$$\begin{aligned} P(\phi_z | \mathbf{z}_z; \beta) &= \frac{P(\mathbf{z}_z, \phi_z | \beta)}{P(\mathbf{z}_z | \beta)} = \frac{P(\mathbf{z}_z | \phi_z) P(\phi_z | \beta)}{P(\mathbf{z}_z | \beta)} = \frac{\prod_{w \in \mathbf{z}_z} P(w | \phi_z) P(\phi_z | \beta)}{P(\mathbf{z}_z | \beta)} \\ &= \frac{\prod_{w \in V} P(w | \phi_z)^{n_w^{(z)}} P(\phi_z | \beta)}{\int_{\phi_z} \prod_{w \in V} P(w | \phi_z)^{n_w^{(z)}} P(\phi_z | \beta) d\phi_z} = \frac{\prod_{w \in V} \phi_{z,w}^{n_w^{(z)}} \frac{1}{\text{Beta}(\beta)} \prod_{w \in V} \phi_{z,w}^{\beta_w - 1}}{\int_{\phi_z} \prod_{w \in V} \phi_{z,w}^{n_w^{(z)}} \frac{1}{\text{Beta}(\beta)} \prod_{w \in V} \phi_{z,w}^{\beta_w - 1} d\phi_z} \\ &= \frac{\Gamma(\sum_{w \in V} n_w^{(z)} + \beta_w)}{\prod_{w \in V} \Gamma(n_w^{(z)} + \beta_w)} \prod_{w \in V} \phi_{z,w}^{n_w^{(z)} + \beta_w - 1}. \end{aligned} \quad (4.26)$$

The final result of Eq. 4.26 is the *Dirichlet* distribution of  $\phi_z$  with the parameter  $n^{(z)} + \beta$ , where  $n^{(z)} = \langle n_1^{(z)}, n_2^{(z)}, \dots, n_{|V|}^{(z)} \rangle$  and  $\beta = \langle \beta_1, \beta_2, \dots, \beta_{|V|} \rangle$ . Therefore, the likelihood of term  $w$  in topic  $z$  is obtained as the expectation of this distribution, computed by

$$\phi_{z,w} = \frac{n_w^{(z)} + \beta_w}{\sum_{w' \in V} n_{w'}^{(z)} + \beta_{w'}}, \quad z \in Z, w \in V. \quad (4.27)$$

---

**Algorithm 4:** Collapsed Gibbs sampling algorithm for the *LDA* model.

---

**Input:**  
 $\mathcal{D}$ : corpus of documents  
 $|Z|$ : number of topics to be extracted  
 $\alpha, \beta$ : Dirichlet hyperparameters

**Output:**  
 $\phi$ : distributions of terms in topics  
 $\theta$ : topic proportions of documents

```

1  $I := Iterations; BurnIn := BurnInSteps;$ 
2 /* Initialization */
3 foreach  $d \in \mathcal{D}$  do
4   foreach  $w \in d$  do
5      $z \sim uniform();$ 
6     assign  $w$  to  $z$ ;
7 /* Burn-in and update parameters */
8 foreach  $i = 1..I$  do
9   foreach  $d \in \mathcal{D}$  do
10    foreach  $w \in d$  do
11       $z \sim \frac{n_{-w,z}^{(d)} + \alpha_z}{\sum_{z' \in Z} n_{-w,z'}^{(d)} + \alpha_{z'}} \frac{n_{-w,w}^{(z)} + \beta_w}{\sum_{w' \in V} n_{-w,w'}^{(z)} + \beta_{w'}};$ 
12      assign  $w$  to  $z$ ;
13    if  $i > BurnIn$  then
14      update parameters  $\theta$  and  $\phi$  using
15      Eq. 4.25 and Eq. 4.27, respectively;

```

---

**Sampling algorithm.** Having the sampling rule for the topic assignments and formulas for updating the multinomial parameters (i.e., the topic proportion of a document and the distribution of terms in a topic) derived, the collapsed Gibbs sampling algorithm for the LDA model is shown in Algorithm 4. The algorithm first randomly assigns each word to a topic and then applies the sampling rule to build a Markov chain for the topic assignments. After the *Burn-in* stage, the updating rules are employed to derive the multinomial parameters representing the topic proportion of documents and the distribution of terms in topics. Given a predefined number of sampling steps  $I$ , the algorithm has the time complexity  $O(I \times |\mathcal{D}| \times |d| \times |Z|)$ .

LDA was initially designed for the extraction of topics from a corpus of documents. However, it can be employed to cluster observations in a dataset from various applications, often applying these three assumptions: (1) observations are organized in *groups* (e.g., a group is a document); (2) it is desirable to share clusters among groups (e.g., topics are shared among documents); (3) both groups as well as observations in each group are exchangeable. Also, one can extend LDA to applications where each observation in the dataset is described by multiple features. That is, for each observation, more than one

feature needs to be jointly considered to compute the likelihood of the observation in a cluster. These principles of LDA are employed in the *rLinkTopic* model to discover regional *LinkTopic* communities, which is presented in the next section.

## 4.3 Regional LinkTopic Communities

### 4.3.1 Introduction

The past couple of years have witnessed a significant increase in research targeted towards the discovery and analysis of communities in social networks. This interest is driven by various questions raised in different applications domains, ranging from studying the social behavior and ties of users in social networks to targeting users with tailored services and advertisements.

As discussed in Section 2.4, the typical approaches for finding communities rely on the link structure of users, which is presented as a graph. This leads to the application of different graph clustering algorithms to detect such link-based communities. Recent studies, however, pay more attention to finding topical communities. By this, topical analysis is applied to the messages of users to derive topics indicating their interests. The extracted topics are used as another feature besides the link structures to identify relationships between users. The key idea is that by leveraging more common features describing users one can discover more meaningful communities. That is, users in a community exhibit both structural and hidden semantic links to each others. The main approach to extracting communities based on this idea is to develop a probabilistic model simulating a process of generating the observed features associated with users from hidden communities.

In the proposed models (discussed in Section 2.5.6), the two important features, namely the contextual links of users and the regional aspect of communities, have been either neglected or paid only very little attention to. Generally, existing models consider all static links of a user to compute the likelihood of that user in a community. This, however, leads to the problem of losing the semantic context of the links. It is also worth to note that a large proportion of “link users” in a social network is inactive [67, 124]; thus, considering them in the detection of communities is not that meaningful.

Social networks provide users with a *contextual link* feature that can be associated with postings. This feature is thought of as one of the underlying reasons motivating users to spend more time online. For example, a *Twitter* user can mention other users (e.g., using @username) in her messages, reply to or retweet a message posted by users she follows. This is similar to the case on *Facebook* where a user can tag other users in her comments and reply to a comment that she is involved or interested in. Even though a user might have many “link users”, it is intuitive that not all of these are of the same community. This is because structural links to other users are created for several reasons and at different time points but it is likely that such users might not have similar topics of interest, and their

topics change over time. A user tends to tag only others who are involved in a discussion or interested in what she is sharing. Therefore, instead of employing static links among users, one better relies on the contextual links associated with messages to extract communities.

In addition to contextual links, another feature that is helpful in finding more meaningful communities is the spatio-temporal proximity of users. The underlying idea is that co-occurrences of users in time and geographic space indicate some type of social interaction. This holds especially for users having similar interests indicated by their postings and their contextual links. Such an observation is supported by recent studies on the characteristics of social relationships, which show that co-occurrences of users in spatio-temporal proximity imply the existence of social links between them [29, 70, 90, 98]. In this respect, communities tend to be geographically localized. Extracting such regional communities with topical characteristics leads to practical applications especially for recommendations, advertisements, and geographically focused social studies, e.g., [65, 114]. This regional aspect, however, has not been considered in existing studies on the community detection.

To address these shortcomings, we develop a model called *rLinkTopic* to discover (regional) communities. All three features, namely spatio-temporal proximity, contextual links, and postings of users in a social network are leveraged in a probabilistic model. The model considers user occurrences in the network to be created by communities in geographic regions. Specifically, a region determined based on geographic locations of users is modeled as a mixture of communities. A community is a distribution of users who have similar topics and contextually link to each other while posting messages to the network. A community is further characterized by a topic proportion and a *degree* of geographic localization. Finally, a topic is a distribution of terms from a vocabulary.

A general scenario of the generative process is as follows. Occurrences of users are first assigned to regions. Each region is selected in turn to create communities. Each community is selected to generate users and topics. Each topic is selected to generate words in the messages of users. Following this probabilistic principle, a user can be a member of more than one community, and a community can discuss different topics. Also, a community is constrained to a relatively small geographic area, such as a city or neighborhood.

### 4.3.2 Preliminaries

In the following, we outline the data model and the concept of regional *LinkTopic* communities underlying our framework. Notations used throughout this chapter are shown in Table 4.1. As input to the model, we assume occurrences of users  $U$  in a given social network. The concept of user occurrences was given in Def. 3.1. Nevertheless, for the purpose of generality, we aim at developing a model that is able to extract communities in a social network where no thread of discussion is associated with the occurrences of users. Otherwise information about threads is discarded. The user occurrence is therefore reformalized as follows.

Table 4.1: Main notations used in the *rLinkTopic* model for extracting communities based on spatio-temporal proximity, contextual links, and topics of users.

Notation	Description
$U$	set of users in social network, $u$ is a user in $U$
$C$	set of communities, $c$ is a community in $C$
$R$	set of geographic regions, $r$ is a region in $R$
$V$	vocabulary set, $w$ is a term in $V$
$Z$	set of community topics, $z$ is a topic in $Z$
$\theta$	set of community distributions in geographic regions, i.e., $\theta = \{\theta_r\}, r \in R$
$\phi$	set of user distributions for communities $C$ , i.e., $\phi = \{\phi_c\}, c \in C$
$\pi$	set of topic proportions of communities $C$ , i.e., $\pi = \{\pi_c\}, c \in C$
$\varphi$	set of term distributions for topics $Z$ , i.e., $\varphi = \{\varphi_z\}, z \in Z$
$\mathbf{r}$	region assignments of the occurrences of users
$\mathbf{c}$	community assignments of the occurrences of users
$\mathbf{z}$	topic assignments of the messages of users

**Definition 4.1 (User Occurrence)** An occurrence  $o = \langle u, loc, msg, f, t \rangle$  of a user  $u \in U$  in a social network consists of a message  $msg$  posted by  $u$  at a geographic location  $loc$  and at time point  $t$  with an optional set of contextual link users  $f \subseteq U$ . The message  $msg$  contains a set of words from a vocabulary  $V$ .

A user occurrence thus is a formalization of the postings that have (a) a geographic location attached (geotagged) and (b) links to other users. Semantically, we refer to the link users  $f$  in the occurrence  $o$  as *contextual links*. Two messages, one posted by a *Facebook* user and another posted by a *Twitter* user, both having contextual links (indicated by boxes) are shown in Figure 4.2.



Figure 4.2: Examples of postings that contain contextual links. (a) a comment on *Facebook*; (b) a tweet on *Twitter*.

For the extraction of communities from such occurrences of users, we make use of snapshots of the social network. By this, the temporal aspect is taken into account to determine users co-occurring in the network within temporal proximity. The snapshot concept defined in Def. 3.2 is repeated here for a complete formalization of the model.

**Definition 4.2 (Snapshot)** Given a set  $U$  of users in a social network, the set of occurrences of users in  $U$  during a time interval  $\Delta t = [t_s, t_e]$  is called a snapshot of the network, denoted  $sn_t = \{\langle u, loc, msg, f, t \rangle\}$ , where  $u \in U$  and  $t \in \Delta t$ .

Practically, a snapshot can be thought of as a set of, for example, geotagged tweets collected during a 24-hour interval. Note that within a snapshot a user can occur several times, meaning that several postings of a user might appear in one snapshot at possibly different geographic locations.

Having the concept of snapshots formalized, a social network is considered a sequence of snapshots, i.e.,  $SN = \{sn_1, sn_2, \dots, sn_T\}$ , which provides us with the underlying data model for the extraction of the regional *LinkTopic* communities. Following traditional topic modeling, we consider a community as a multinomial distribution of users in  $U$ . Formally, a community  $c$  is represented by a multinomial parameter  $\phi_c = \{\phi_{c,u}\}$ ,  $u \in U$ . Each  $\phi_{c,u}$  is the conditional probability of user  $u$  in community  $c$ , denoted  $P(u|\phi_c)$ , such that  $\sum_{u \in U} P(u|\phi_c) = 1$ . It is noted that the membership of a user in a community is modeled by a likelihood measure, therefore, a user can be a member of more than one community. We detail how regional *LinkTopic* communities are extracted in the following section.

## 4.4 rLinkTopic Probabilistic Model

This section presents the *rLinkTopic* model in detail. As mentioned above, there are three features that are considered for computing the likelihood of a user in a community: contextual links, topics of interest, and spatio-temporal proximity of users. In the following, we first explain how the first two features are employed (Section 4.4.1) and then introduce a geographic region model to address the last feature (Section 4.4.2). The generative process and the posterior estimation for the *rLinkTopic* model are presented in Section 4.4.3 and Section 4.4.4, respectively.

### 4.4.1 Joint Contextual Links and Topics

Given an occurrence  $o = \langle u, loc, msg, f, t \rangle$  of a user  $u$ , the idea is that by jointly searching for the associated features, i.e., user  $u$ , users  $o.f$  that  $u$  interacts with, and the topic indicated by  $o.msg$ , one can find which community this occurrence of  $u$  should be assigned to. The first observed feature to be considered is the set  $o.f$  of links of  $u$  with other users in a community. If  $u$  and the contextual links  $o.f$  occur in the same community then it is a good indicator that this occurrence of  $u$  belongs to that community. In terms of the generative scenario, this property is specified as follows. Once a community  $c$  is sampled for  $u$  regarding the occurrence  $o = \langle u, loc, msg, f, t \rangle$ , all users that  $u$  interacts with in this occurrence are also assigned to community  $c$ . This is intuitive because users in the interaction associated with a posting are clearly related to each other in the context described by this posting.



The second feature is the similarity of the posting *o.msg* with the topics of a community. For this, we make use of a *single topic* feature for each posting of a user. This is because a posting is short and normally addresses exactly one topic that the user is currently interested in. An occurrence *o* should be assigned to a community whose topics include the topic related to the posting *o.msg*.

Users in community *c* share common interests in a mixture of topics *Z* that indicates the topic proportion of the community. This topic proportion is modeled as a multinomial parameter  $\pi_c = \{\pi_{c,z}\}$ ,  $z \in Z$ . Each  $\pi_{c,z}$  is the likelihood of the topic *z* in the community *c*, denoted  $P(z|\pi_c)$ , such that  $\sum_{z \in Z} P(z|\pi_c) = 1$ . A topic *z* is (again) considered a multinomial distribution of terms in the vocabulary *V*, i.e.,  $\varphi_z = \{\varphi_{z,w}\}$ ,  $w \in V$ . Each  $\varphi_{z,w}$  is the likelihood of term *w* in topic *z*, denoted  $P(w|\varphi_z)$ , such that  $\sum_{w \in V} P(w|\varphi_z) = 1$ .

#### 4.4.2 Geographic Region Model

This section introduces an approach to add information about the geographic locations of users to the model so that not only the contextual links and topics but also the spatio-temporal proximity of users are taken into account for extracting communities. The idea is that users co-occurring in a social network within spatio-temporal proximity are more likely to be related compared to others. In the setting of community detection, such users should have a high likelihood to be in the same community, given the existence of contextual links between them and the similarity of their topics. The model therefore needs to use information about the geographic locations of users within each snapshot, i.e., besides the contextual links and topics, to compute their membership in communities. To add this property to the model, we adopt the method developed in the Spatial LDA (SLDA) model [119] to organize occurrences of users in each snapshot to geographic regions. The goal is that the occurrences of nearby users are more likely to be assigned to the same regions. A brief review of the main properties of the SLDA model (for details, see [119]): developed for applications in computer vision, SLDA is an extension of LDA for detecting object classes (topics) in a corpus of images. For this, each object class is modeled as a distribution of image patches (called visual words) that appear close to each other in images. The key idea of SLDA is that, instead of generating directly an image as would be done in the LDA model, it generates an image through sampling close visual words into so-called *documents*. Each *document* is considered to contain hidden object classes. By this, a *document* is a random variable representing a distribution of visual words in spatial proximity and is further modeled as a distribution of the object classes from which visual words have been generated.

In our adoption of the SLDA model, snapshots introduced in the previous section correspond to images, regions to documents, and occurrences of users to visual words. The idea of SLDA is employed in our *rLinkTopic* model as follows. For each snapshot *sn<sub>t</sub>*, a number of so-called *regions* *R<sub>t</sub>* is generated. This is done by modeling the spatial distribution of

occurrences of users in snapshot  $sn_t$  as a mixture of a number of Gaussians with a prior covariance. Each Gaussian is intuitively considered a region consisting of occurrences of users in spatial proximity. Each region  $r \in R_t$  is characterized by three components: a *snapshot index*  $t$  indicating from which snapshot it has been generated, a *representative location* (the mean of the Gaussian), and a *hidden distribution of occurrences* of users.

The steps to identify the representative locations of regions and to sample occurrences of users to regions are as follows. Occurrences of each user  $u$  who occurs more than a predefined number of times, specified by a threshold  $minCount$ , in a snapshot  $sn_t$  are sampled to compute a representative location of  $u$ . This location is used to specify a *center location* of a region. The center location of region  $r$  is denoted  $loc_r$ . The identified center locations are then filtered based on the density of their neighborhood to merge regions. That is, if two center locations are located sufficiently close to each other, specified by a distance threshold  $minRad$ , then they are removed and a new one is derived as the mean point of them. The sampling process to assign occurrence  $o$  to a region is performed by the following uniform Gaussian mixture model.

$$P(o, r) = P(r)P(o|r) \text{ where } P(r) = Uniform() \text{ and } P(o|r) = \mathcal{N}(loc_o|loc_r, \sigma). \quad (4.28)$$

The likelihood of occurrence  $o$  being assigned to region  $r$  depends on the distance between the respective locations  $loc_o$  and  $loc_r$ . The closer the locations, the higher the likelihood that  $o$  will be assigned to  $r$ . Based on this concept of regions, occurrences of users that often co-occur in spatial proximity in a snapshot are more likely to appear in the same regions.

After accomplishing the above steps, a complete set  $R_t = \{r_1, r_2, \dots, r_{|R_t|}\}$  of regions is obtained for snapshot  $sn_t$ . We use  $R$  to denote the set of all regions generated from all snapshots of the network. A sample result of the region assignments computed by our *rLinkTopic* model applied to a *Twitter* dataset collected from England during the time between June 15 and June 20, 2012 is shown in Figure 4.3.

Having such regions identified, the model derives communities from regions instead of snapshots. Particularly, each region is considered a mixture of communities being extracted. As a result, users that co-occur in the network within spatio-temporal proximity tend to be sampled into the same community; this is because their occurrences have been assigned to the same regions. The mixture of communities in a region is represented by a multinomial parameter  $\theta_r = \{\theta_{r,c}\}$ ,  $c \in C$ . Each  $\theta_{r,c}$  is the conditional probability of community  $c$  in region  $r$ , denoted  $P(c|\theta_r)$ , such that  $\sum_{c \in C} P(c|\theta_r) = 1$ .

Putting all together, our generative model samples occurrences of users to regions and, at the same time, discovers communities of users having joint contextual links and similar topics within spatio-temporal proximity.

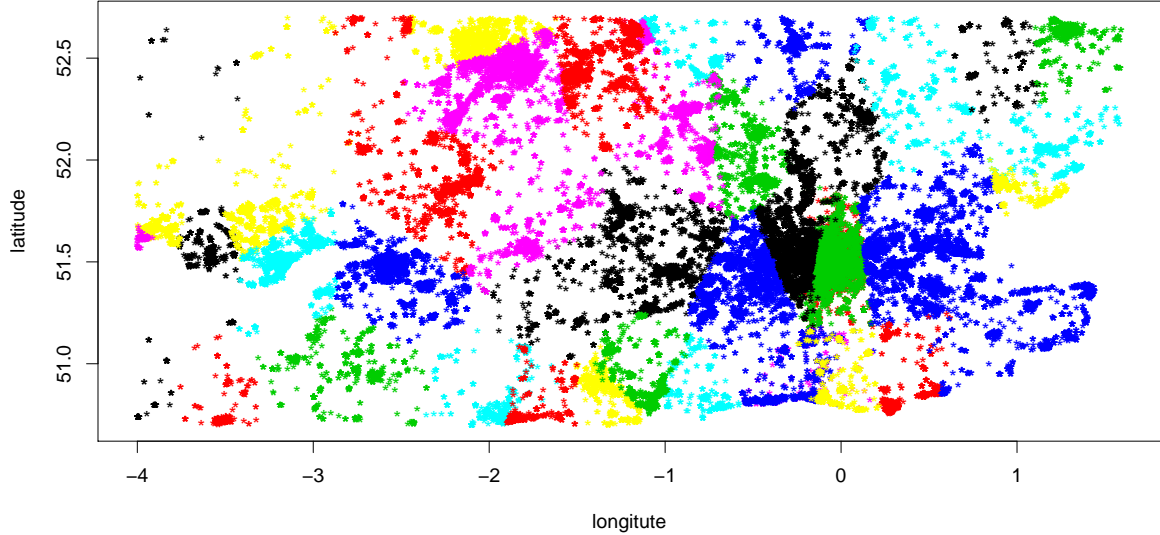


Figure 4.3: A sample result of the region assignments obtained from the *rLinkTopic* model. Regions are distinguished by colors. This is the result of running the model on a *Twitter* dataset collected from England during the time between June 15 and June 20, 2012.

#### 4.4.3 Generative Process

Based on the above descriptions, for presenting the generative process, we first give a short summary of the components in the *rLinkTopic* model as follows.

- a. For each occurrence  $o = \langle u, loc, msg, f, t \rangle$  in snapshot  $sn_t$ , there are four observed features  $u$ ,  $loc$ ,  $msg$ , and  $f$  need to be generated from the model.
- b. Each region  $r$  in snapshot  $sn_t$  has a prior representative location specified by  $loc_r$ , and is formed by a mixture of communities represented by  $\theta_r$ .
- c. Each community  $c$  is a distribution of users, denoted  $\phi_c$ , and has a topic proportion described by  $\pi_c$ .
- d. Each topic  $z$  is a distribution of terms, represented by  $\varphi_z$ .
- e. For each snapshot  $sn_t$ , we employ a uniform distribution  $\eta_t$  to setup a mixture of Gaussians for assigning occurrences of users to the regions created from  $sn_t$ .
- f. All multinomial parameters  $\phi_c$ ,  $\theta_r$ ,  $\pi_c$ , and  $\varphi_z$  in the model are sampled using a *Dirichlet* prior with the corresponding hyperparameters  $\beta$ ,  $\alpha$ ,  $\gamma$ , and  $\mu$ .

The graphical model of *rLinkTopic* is shown in Figure 4.4 where the observed features are represented by shaded variables. The generative process of the *rLinkTopic* model is as follows.

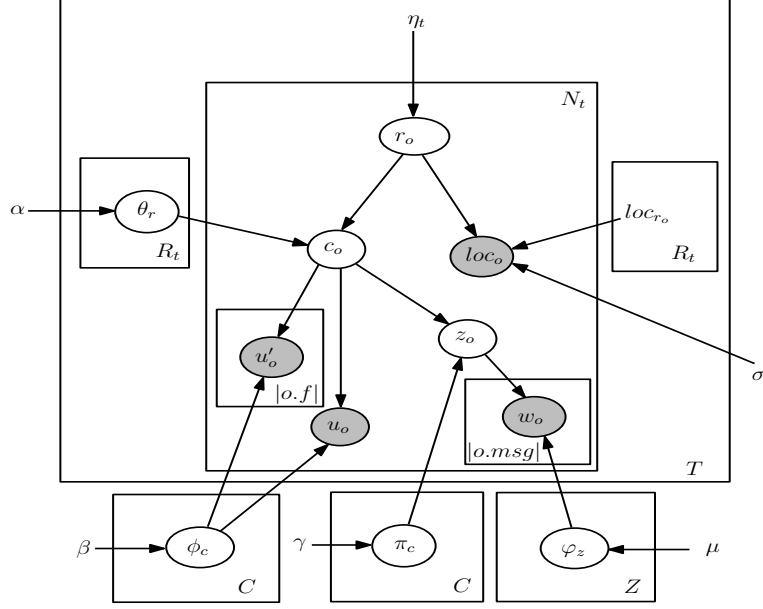


Figure 4.4: Graphical model presenting the generative process of *rLinkTopic* to extract communities based on spatio-temporal proximity, contextual links, and topics of users.

1. sample distributions of users in communities

$$\phi = \{\phi_c \sim \text{Dir}_{|U|}(\beta)\}, c \in C$$

2. sample topic proportions for communities

$$\pi = \{\pi_c \sim \text{Dir}_{|Z|}(\gamma)\}, c \in C$$

3. sample distributions of terms in topics

$$\varphi = \{\varphi_z \sim \text{Dir}_{|V|}(\mu)\}, z \in Z$$

4. for each snapshot  $sn_t$

- 4.1. sample the distribution of communities for each region in  $R_t$

$$\theta_t = \{\theta_r \sim \text{Dir}_{|C|}(\alpha)\}, r \in R_t$$

- 4.2. for each occurrence  $o$  in snapshot  $sn_t$

- a. sample a region  $r_o$  from a uniform distribution  $P(r_o|\eta_t)$

- b. sample the geographic location of  $o$  from the distribution  $P(loc_o|loc_{r_o}, \sigma)$ , which is derived from a Gaussian kernel as follows

$$P(loc_o|loc_{r_o}, \sigma) \propto \exp\left(-\frac{\text{dist}(loc_o, loc_{r_o})}{\sigma^2}\right)$$

- c. sample a community index  $c_o \sim \text{Mult}(\theta_{t, r_o})$

- d. sample the user  $u_o \sim \text{Mult}(\phi_{c_o})$

- e. sample link users  $o.f$

$$f_o = \{u' \sim \text{Mult}(\phi_{c_o})\}, u' \in o.f$$

- f. sample a topic index  $z_o \sim \text{Mult}(\pi_{c_o})$

- g. sample words in the posting  $o.msg$

$$msg_o = \{w \sim \text{Mult}(\varphi_{z_o})\}, w \in o.msg$$

#### 4.4.4 Posterior Estimation for rLinkTopic

In the *rLinkTopic* model, only user  $u_o$  who sends the message, contextual links  $f_o$ , and location  $loc_o$  of user  $u_o$  are observed for a given user occurrence  $o$ . Other variables including the (1) multinomial parameters: the distribution  $\theta_r$  of communities in a region, distribution  $\phi_c$  of users in a community, topic proportion  $\pi_c$  of a community, and distribution  $\varphi_z$  of terms in a topic; and (2) the assignment variables: the region assignment  $r_o$ , community assignment  $c_o$ , and topic assignment  $z_o$ ; are hidden variables. This section presents steps to derive a collapsed Gibbs sampling algorithm to compute these hidden variables.

**Dirichlet prior for multinomial parameters.** The model extracts a set of communities  $C$ , each of which is considered a multinomial distribution of users  $U$ , denoted  $\phi_c = \{\phi_{c,u}\}, u \in U$ . Each  $\phi_c$  is sampled from a *Dirichlet* prior with a  $|U|$ -dimensional hyperparameter  $\beta$ . The joint distribution of  $\phi = \{\phi_c\}$  is

$$P(\phi|\beta) = \prod_{c \in C} P(\phi_c; \beta) \triangleq \prod_{c \in C} Dir(\phi_c; \beta). \quad (4.29)$$

Each community  $c$  has a topic proportion, denoted  $\pi_c = \{\pi_{c,z}\}, z \in Z$ , which is sampled from a *Dirichlet* prior with a  $|Z|$ -dimensional hyperparameter  $\gamma$ . The joint distribution of  $\pi = \{\pi_c\}, c \in C$  is

$$P(\pi|\gamma) = \prod_{c \in C} P(\pi_c; \gamma) \triangleq \prod_{c \in C} Dir(\pi_c; \gamma). \quad (4.30)$$

Each topic  $z$  is a mixture of terms  $V$ , denoted  $\varphi_z = \{\varphi_{z,w}\}, w \in V$ , which is sampled from a *Dirichlet* prior with a  $|V|$ -dimensional hyperparameter  $\mu$ . The joint distribution of  $\varphi = \{\varphi_z\}, z \in Z$  is

$$P(\varphi|\mu) = \prod_{z \in Z} P(\varphi_z; \mu) \triangleq \prod_{z \in Z} Dir(\varphi_z; \mu). \quad (4.31)$$

For each snapshot  $sn_t$ , there are  $|R_t|$  distributions of communities in regions  $R_t$ , i.e.,  $\theta_t = \{\theta_r\}, r \in R_t$ , where  $\theta_r = \{\theta_{r,c}\}, c \in C$ . Each  $\theta_r$  is sampled from a *Dirichlet* prior with a  $|C|$ -dimensional hyperparameter  $\alpha$ . The joint distribution of  $\theta_t$  is

$$P(\theta_t|\alpha) = \prod_{r \in R_t} P(\theta_r; \alpha) \triangleq \prod_{r \in R_t} Dir(\theta_r; \alpha). \quad (4.32)$$

**Joint distribution of the model.** For an occurrence  $o = \langle u, loc, msg, f, t \rangle$  of user  $u$  in snapshot  $sn_t$ , all associated features including the observed and hidden ones are generated by the following procedure.

- The region  $r_o$  that occurrence  $o$  is assigned to is sampled from a uniform distribution  $\eta_t$ . The conditional probability of  $r_o$  is denoted  $P(r_o|\eta_t)$ .
- The likelihood of the geographic location of  $o$  in region  $r_o$  is  $P(loc_o|loc_{r_o}, \sigma)$  that is derived from a Gaussian kernel, where  $\sigma$  is a prior standard deviation.

- The conditional probability of  $c_o$ , i.e., the community that  $u$  belongs to, given the distributions  $\theta_t$  of communities in regions and region  $r_o$ , is  $P(c_o|\theta_{t,r_o})$ .
- The conditional probability of  $u_o$ , i.e., the user  $u$  is sampled for occurrence  $o$ , and contextual links  $o.f$ , given community  $c_o$  and the distributions of users in communities ( $\phi$ ), is  $P(u_o|\phi_{c_o}) \prod_{u' \in o.f} P(u'|\phi_{c_o})$ .
- The conditional probability of  $z_o$ , i.e., the topic that is sampled for message  $msg$ , given the proportions of topics in communities, i.e.,  $\pi$ , and community  $c_o$  is  $P(z_o|\pi_{c_o})$ .
- The conditional probability of message  $o.msg$  in topic  $z_o$  is  $\prod_{w \in o.msg} P(w|\varphi_{z_o})$ .

Therefore, the joint distribution of occurrence  $o = \langle u, loc, msg, f, t \rangle$  in snapshot  $sn_t$  is

$$\begin{aligned}
P(o, \phi, \pi, \varphi, \theta_t, r_o, c_o, z_o | \beta, \gamma, \mu, \alpha, \eta_t, \sigma) = & \quad (4.33) \\
& P(\phi|\beta)P(\pi|\gamma)P(\varphi|\mu) \times \\
& P(\theta_t|\alpha)P(r_o|\eta_t)P(loc_o|loc_{r_o}, \sigma) \times \\
& P(c_o|\theta_{t,r_o})P(u_o|\phi_{c_o}) \prod_{u' \in o.f} P(u'|\phi_{c_o}) \times \\
& P(z_o|\pi_{c_o}) \prod_{w \in o.msg} P(w|\varphi_{z_o}).
\end{aligned}$$

By employing the assumption that occurrences of users are independently and identically distributed, the joint distribution of occurrences in all snapshots  $SN = \{sn_1, sn_2, \dots, sn_T\}$  is

$$\begin{aligned}
P(SN, \phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | \beta, \gamma, \mu, \alpha, \eta, \sigma) = & P(\phi|\beta)P(\pi|\gamma)P(\varphi|\mu) \times \\
\prod_{t=1}^T P(\theta_t|\alpha) \prod_{o \in sn_t} & \underbrace{P(r_o|\eta_t)P(loc_o|loc_{r_o}, \sigma)P(c_o|\theta_{t,r_o})P(u_o|\phi_{c_o}) \prod_{u' \in o.f} P(u'|\phi_{c_o})P(z_o|\pi_{c_o}) \prod_{w \in o.msg} P(w|\varphi_{z_o})}_{(*) \text{ - occurrences in one snapshot}},
\end{aligned} \quad (4.34)$$

where  $\mathbf{r}$ ,  $\mathbf{c}$ , and  $\mathbf{z}$  represent the region assignments, community assignments, and topic assignments, respectively, of all user occurrences in the network.

The marginal probability of the whole network is analytically obtained from the joint distribution by integrating over all multinomial parameters  $\phi$ ,  $\pi$ ,  $\varphi$ , and  $\theta$ ; and summing over regions, communities, and topics, which is mathematically presented as follows<sup>4</sup>.

$$P(SN | \beta, \gamma, \mu, \alpha, \eta, \sigma) = \int_{\theta} \int_{\phi} \int_{\pi} \int_{\varphi} \sum_{r,c,z}^{R,C,Z} P(SN, \phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | \cdot) d\varphi d\pi d\phi d\theta \quad (4.35)$$

<sup>4</sup>In Eq. 4.35 and other equations presented next, if not otherwise specified, the dot “.” is used to denote all hyperparameters in the model, i.e.,  $\cdot \triangleq \beta, \gamma, \mu, \alpha, \eta, \sigma$ .

Applying *Bayes' theorem*, the posterior distribution of the hidden variables  $\phi$ ,  $\pi$ ,  $\varphi$ ,  $\theta$ ,  $\mathbf{r}$ ,  $\mathbf{c}$ , and  $\mathbf{z}$  is analytically obtained by

$$P(\phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) = \frac{P(SN, \phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | \beta, \gamma, \mu, \alpha, \eta, \sigma)}{P(SN | \beta, \gamma, \mu, \alpha, \eta, \sigma)}. \quad (4.36)$$

Because the marginal probability of the network is intractable, the above posterior distribution cannot be exactly derived. As a typical approach, we develop a collapsed Gibbs sampling algorithm to approximate the distributions of the hidden variables from the joint distribution of the model. For this purpose, we first represent the joint distribution of the model as independent terms and then apply the conjugacy between:  $P(\phi | \beta)$  with  $P(u_o | \phi_{c_o})$  and  $P(u'_o \in o.f | \phi_{c_o})$ ;  $P(\theta_t | \alpha)$  with  $P(c_o | \theta_{t, r_o})$ ;  $P(\pi | \gamma)$  with  $P(z_o | \pi_{c_o})$ ; and  $P(\varphi | \mu)$  with  $P(w \in o.msg | \varphi_{z_o})$  to integrate out multinomial parameters  $\phi$ ,  $\theta$ ,  $\pi$ , and  $\varphi$ . The steps in the simplification are presented as follows.

First, the term denoted by (\*) in the joint distribution of the model derived in Eq. 4.34 is restructured as shown in Eq. 4.37.

$$\begin{aligned} (*) &= \underbrace{\prod_{o \in sn_t} P(r_o | \eta) P(loc_o | loc_{r_o}, \sigma)}_{(I)} \times \underbrace{\prod_{o \in sn_t} P(c_o | \theta_{t, r_o})}_{(II)} \times \underbrace{\prod_{o \in sn_t} P(u_o | \phi_{c_o}) \prod_{u' \in o.f} P(u' | \phi_{c_o})}_{(III)} \\ &\times \underbrace{\prod_{o \in sn_t} P(z_o | \pi_{c_o})}_{(IV)} \times \underbrace{\prod_{o \in sn_t} \prod_{w \in o.msg} P(w | \varphi_{z_o})}_{(V)} \end{aligned} \quad (4.37)$$

Note that in Eq. 4.37 (I) does not depend on any multinomial parameter, while each of the other terms depends only on one multinomial parameter. In particular, (II) depends on  $\theta_t$ , (III) depends on  $\phi$ , (IV) depends on  $\pi$ , and (V) depends on  $\varphi$ . Employing such simplifications, the joint distribution of the model is now rewritten as follows.

$$\begin{aligned} P(SN, \phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | \beta, \gamma, \mu, \alpha, \eta, \sigma) &= \prod_{t=1}^T \prod_{o \in sn_t} P(r_o | \eta) P(loc_o | loc_{r_o}, \sigma) \times & (I) \\ &\prod_{t=1}^T P(\theta_t | \alpha) \prod_{o \in sn_t} P(c_o | \theta_{t, r_o}) \times & (II) \\ &P(\phi | \beta) \prod_{t=1}^T \prod_{o \in sn_t} P(u_o | \phi_{c_o}) \prod_{u' \in o.f} P(u' | \phi_{c_o}) \times & (III) \\ &P(\pi | \gamma) \prod_{t=1}^T \prod_{o \in sn_t} P(z_o | \pi_{c_o}) \times & (IV) \\ &P(\varphi | \mu) \prod_{t=1}^T \prod_{o \in sn_t} \prod_{w \in o.msg} P(w | \varphi_{z_o}) & (V) \end{aligned} \quad (4.38)$$

Table 4.2: Notations used to present the count variables in the *rLinkTopic* model.

Symbol	Description
$n_c^{(r)}$	number of occurrences in region $r$ that are assigned to community $c$
$n_u^{(c)}$	number of occurrences of user $u$ that are assigned to community $c$
$n_{f,u}^{(c)}$	number of times user $u$ is contextually linked by other users in community $c$
$n_w^{(z)}$	number of occurrences of term $w$ that are assigned to topic $z$
$n_z^{(c)}$	number of messages in community $c$ that are assigned to topic $z$

By applying Eq. 4.29 to Eq. 4.32, and using the notations defined in Table 4.2, each term in Eq. 4.38 is further simplified as follows<sup>5</sup>:

$$\begin{aligned}
(II) &= \prod_{t=1}^T \prod_{r \in R_t} P(\theta_r | \alpha) \prod_{o \in sn_t} P(c_o | \theta_{t,r_o}) = \prod_{t=1}^T \prod_{r \in R_t} P(\theta_r | \alpha) \overbrace{\prod_{r \in R_t} \prod_{c \in C} P(c | \theta_r)^{n_c^{(r)}}}^{\text{due to } c_o \in C \text{ and } r_o \in R_t} \\
&= \prod_{t=1}^T \prod_{r \in R_t} P(\theta_r | \alpha) \prod_{c \in C} P(c | \theta_r)^{n_c^{(r)}} = \prod_{t=1}^T \prod_{r \in R_t} \frac{1}{\text{Beta}(\alpha)} \prod_{c \in C} \theta_{r,c}^{\alpha_c - 1} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)}} \\
&\propto \prod_{t=1}^T \prod_{r \in R_t} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} \quad (4.39)
\end{aligned}$$

$$\begin{aligned}
(III) &= \prod_{c \in C} P(\phi_c | \beta) \prod_{t=1}^T \prod_{o \in sn_t} P(u_o | \phi_{c_o}) \prod_{u' \in o.f} P(u' | \phi_{c_o}) = \prod_{c \in C} P(\phi_c | \beta) \overbrace{\prod_{c \in C} \prod_{u \in U} P(u | \phi_c)^{n_u^{(c)} + n_{f,u}^{(c)}}}^{\text{due to } c_o \in C \text{ and } u_o \in U} \\
&= \prod_{c \in C} P(\phi_c | \beta) \prod_{u \in U} P(u | \phi_c)^{n_u^{(c)} + n_{f,u}^{(c)}} = \prod_{c \in C} \frac{1}{\text{Beta}(\beta)} \prod_{u \in U} \phi_{c,u}^{\beta_u - 1} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)}} \\
&\propto \prod_{c \in C} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} \quad (4.40)
\end{aligned}$$

$$\begin{aligned}
(IV) &= \prod_{c \in C} P(\pi_c | \gamma) \prod_{t=1}^T \prod_{o \in sn_t} P(z_o | \pi_{c_o}) = \prod_{c \in C} P(\pi_c | \gamma) \overbrace{\prod_{c \in C} \prod_{z \in Z} P(z | \pi_c)^{n_z^{(c)}}}^{\text{due to } c_o \in C \text{ and } z_o \in Z} \\
&= \prod_{c \in C} P(\pi_c | \gamma) \prod_{z \in Z} P(z | \pi_c)^{n_z^{(c)}} = \prod_{c \in C} \frac{1}{\text{Beta}(\gamma)} \prod_{z \in Z} \pi_{c,z}^{\gamma_z - 1} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)}} \\
&\propto \prod_{c \in C} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} \quad (4.41)
\end{aligned}$$

<sup>5</sup>For convenience, the normalizing constant of the *Dirichlet* distribution computed using hyperparameters is presented in terms of the *Beta* function:  $\text{Beta}(X) = \frac{\prod_{x_i \in X} \Gamma(x_i)}{\Gamma(\sum_{x_i \in X} x_i)}$



$$\begin{aligned}
(V) &= \prod_{z \in Z} P(\varphi_z | \mu) \prod_{t=1}^T \prod_{o \in sn_t} \prod_{w \in o.msg} P(w | \varphi_{z_o}) = \prod_{z \in Z} P(\varphi_z | \mu) \overbrace{\prod_{z \in Z} \prod_{w \in V} P(w | \varphi_z)^{n_w^{(z)}}}^{\text{due to } z_o \in Z \text{ and } w \in V} \\
&= \prod_{z \in Z} P(\varphi_z | \mu) \prod_{w \in V} P(w | \varphi_z)^{n_w^{(z)}} = \prod_{z \in Z} \frac{1}{\text{Beta}(\mu)} \prod_{w \in V} \varphi_{z,w}^{\mu_w - 1} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)}} \\
&\propto \prod_{z \in Z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1} \quad (4.42)
\end{aligned}$$

By substituting the results of Eq. 4.39 to Eq. 4.42 for the corresponding terms in Eq. 4.38, the joint distribution of the model becomes

$$\begin{aligned}
P(SN, \phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | \beta, \gamma, \mu, \alpha, \eta, \sigma) &\propto \prod_{t=1}^T \prod_{o \in sn_t} P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma) \times \\
\prod_{t=1}^T \prod_{r \in R_t} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} &\times \prod_{c \in C} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} \times \prod_{c \in C} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} \times \prod_{z \in Z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1}. \quad (4.43)
\end{aligned}$$

Generally, the posterior distribution  $P(\phi, \pi, \varphi, \theta, \mathbf{r}, \mathbf{c}, \mathbf{z} | SN, \beta, \gamma, \mu, \alpha, \eta, \sigma)$  can be approximated by sampling from the above distribution. However, for efficiency purposes, as presented in the discussion of the LDA model (Section 4.2.2), we integrate out multinomial parameters  $\phi, \pi, \varphi$ , and  $\theta$  in order to build a collapsed Gibbs sampling for the model. Particularly, the joint distribution of the region assignments, community assignments, and topic assignments of all occurrences of users in the network is<sup>6</sup>

$$\begin{aligned}
P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) &= \int_{\theta} \int_{\phi} \int_{\pi} \int_{\varphi} P(\phi, \theta, \pi, \varphi, \mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) d\varphi d\pi d\phi d\theta \\
&\propto \prod_{t=1}^T \prod_{o \in sn_t} P(r_o | \eta) P(loc_o | loc_{r_o}, \sigma) \times \int_{\theta} \prod_{t=1}^T \prod_{r \in R_t} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} d\theta \times \\
&\int_{\phi} \prod_{c \in C} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} d\phi \times \int_{\pi} \prod_{c \in C} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} d\pi \times \int_{\varphi} \prod_{z \in Z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1} d\varphi. \quad (4.44)
\end{aligned}$$

Each integral term in Eq. 4.44 is then computed as follows:

$$\int_{\theta} \prod_{t=1}^T \prod_{r \in R_t} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} d\theta = \prod_{t=1}^T \prod_{r \in R_t} \int_{\theta_r} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} d\theta_r = \prod_{r \in R} \frac{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)} \quad (4.45)$$

$$\int_{\phi} \prod_{c \in C} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} d\phi = \prod_{c \in C} \int_{\phi_c} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} d\phi_c = \prod_{c \in C} \frac{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f,u}^{(c)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f,u}^{(c)} + \beta_u)} \quad (4.46)$$

<sup>6</sup>Note that the rule  $\sum_x \sum_y f(x)g(y) = \sum_x (f(x) \sum_y g(y)) = \sum_x f(x) \sum_y g(y)$  is applied in Eq. 4.44.

$$\int_{\pi} \prod_{c \in C} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} d\pi = \prod_{c \in C} \int_{\pi_c} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} d\pi_c = \prod_{c \in C} \frac{\prod_{z \in Z} \Gamma(n_z^{(c)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c)} + \gamma_z)} \quad (4.47)$$

$$\int_{\varphi} \prod_{z \in Z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1} d\varphi = \prod_{z \in Z} \int_{\varphi_z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1} d\varphi_z = \prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)} \quad (4.48)$$

Finally, by applying the results of Eq. 4.45, Eq. 4.46, Eq. 4.47, and Eq. 4.48 to the corresponding terms in Eq. 4.44, the joint distribution of the assignment variables  $\mathbf{r}$ ,  $\mathbf{c}$ , and  $\mathbf{z}$  is obtained:

$$P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) \propto \underbrace{\prod_{t=1}^T \prod_{o \in sn_t} P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma)}_{(T_1)} \times \underbrace{\prod_{r \in R} \frac{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)}}_{(T_2)} \times \underbrace{\prod_{c \in C} \frac{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f,u}^{(c)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f,u}^{(c)} + \beta_u)}}_{(T_3)} \times \underbrace{\prod_{c \in C} \frac{\prod_{z \in Z} \Gamma(n_z^{(c)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c)} + \gamma_z)}}_{(T_4)} \times \underbrace{\prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)}}_{(T_5)} \quad (4.49)$$

One can intuitively interpret the meaning of Eq. 4.49 as follows. The first term  $T_1$  is the joint distribution of geographic locations of users  $U$  in regions  $R$ , which is derived from a uniform Gaussian mixture model. The second term  $T_2$  is the joint distribution of communities  $C$  in regions  $R$ . The third term is the joint distribution of users  $U$  in communities  $C$ . The fourth term  $T_4$  is the joint distribution of topics  $Z$  in communities  $C$ , and the last term  $T_5$  is the joint distribution of terms  $V$  in topics  $Z$ .

Note that  $\mathbf{r}$ ,  $\mathbf{c}$ , and  $\mathbf{z}$  are the region assignments, community assignments, and topic assignments of all occurrences of users, respectively. To derive the Gibbs sampling rules for these assignments for a particular occurrence  $o = \langle u, loc, msg, f, t \rangle$ , given such assignments of other occurrences, the joint distribution in Eq. 4.49 is rewritten as

$$P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) = \frac{P(r_o, c_o, z_o, \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o} | SN; \cdot)}{P(r_o, c_o, z_o | \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \cdot)} P(\mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o} | SN; \cdot), \quad (4.50)$$

where  $r_o$ ,  $c_o$ , and  $z_o$  are the region assignment, community assignment, and topic assignment, respectively, of occurrence  $o$ ; and  $\mathbf{r}_{-o}$ ,  $\mathbf{c}_{-o}$ , and  $\mathbf{z}_{-o}$  are the respective assignments of all other occurrences. The joint distribution of the region assignment, community assignment, and topic assignment of occurrence  $o$  given such assignments of other occurrences is therefore derived from

$$P(r_o, c_o, z_o | \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) = \frac{P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \cdot)}{P(\mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o} | SN; \cdot)}. \quad (4.51)$$

Note that the only difference between the numerator and the denominator in Eq. 4.51 is that the numerator is the full joint distribution of the region, community, and topic assignments of all occurrences whereas such assignments of the currently considered occurrence  $o$  are not being counted in the denominator. By using the notations  $T_1, T_2, T_3, T_4$ , and  $T_5$  in Eq. 4.49 and defining the corresponding terms  $T_{-o,1}, T_{-o,2}, T_{-o,3}, T_{-o,4}$ , and  $T_{-o,5}$  where information about the assignments of  $o$  is removed, the joint distribution of the assignments of  $o$  can be represented as follows.

$$P(r_o, c_o, z_o | \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \cdot) = \frac{P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \cdot)}{P(\mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o} | SN; \cdot)} = \frac{T_1}{T_{-o,1}} \frac{T_2}{T_{-o,2}} \frac{T_3}{T_{-o,3}} \frac{T_4}{T_{-o,4}} \frac{T_5}{T_{-o,5}} \quad (4.52)$$

Since we are interested in the region, community, and topic assignments of the currently considered occurrence  $o = \langle u, loc, msg, f, t \rangle$  in snapshot  $sn_t$ , information in  $P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \cdot)$  and in  $P(\mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o} | SN; \cdot)$  that is independent of such assignments can be discarded. Such independent information is identified based on the underlying assumption of the local Markov property employed in the model.

In the following, independent information in each term  $T_i$  in Eq. 4.52 regarding the region, community, and topic assignments of  $o$  is first removed, and then each resulting term  $T_i$  is presented in terms of  $T_{-o,i}$  for further simplification. Also, the same convention used to define  $T_{-o,i}$  is applied for the notations introduced in Table 4.2. For example,  $n_{-o,c_o}^{(r_o)}$  denotes the number of occurrences in region  $r_o$  that were assigned to community  $c_o$  excluding  $o = \langle u, loc, msg, f, t \rangle$  that is currently considered.

**Region assignment.** There are two terms  $T_1$  and  $T_2$  in  $P(\mathbf{r}, \mathbf{c}, \mathbf{z} | SN; \cdot)$  (Eq. 4.49) that contribute to the likelihood of an occurrence  $o$  in a region  $r$ .  $T_1$  is the joint distribution of the geographic locations of users  $U$  in regions  $R$ .  $T_2$  is the joint distribution of communities  $C$  in regions  $R$ . We consider  $T_1$  here and leave  $T_2$  for the next discussion of the distribution of communities in regions. The likelihood computed by  $T_1$  of occurrence  $o$  in region  $r_o$  depends only on the spatial distance between  $o$  and the representative location of  $r_o$ . The geographic locations of other occurrences are independent of the likelihood of  $o$  in  $r_o$ . Therefore,  $T_{-o,1}$  is independent of  $T_1$ , and  $T_1$  itself can be reduced to retain only the information derived from  $o$ . Particularly, we have

$$\frac{T_1}{T_{-o,1}} \propto T_1 = \prod_{t=1}^T \prod_{o \in sn_t} P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma) \propto P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma). \quad (4.53)$$

**Independence of communities.** The likelihood of community  $c_o$  in region  $r_o$  is independent of all communities in other regions and other communities in region  $r_o$ . Therefore, the second term  $T_2$  in Eq. 4.49 is simplified by

$$\begin{aligned}
T_2 &= \prod_{r \in R} \frac{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)} = \frac{\prod_{c \in C} \Gamma(n_c^{(r_o)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r_o)} + \alpha_c)} \prod_{r \in R \setminus r_o} \frac{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)} \\
&\propto \frac{\prod_{c \in C} \Gamma(n_c^{(r_o)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r_o)} + \alpha_c)} = \frac{\Gamma(n_{c_o}^{(r_o)} + \alpha_{c_o}) \prod_{c \in C \setminus c_o} \Gamma(n_c^{(r_o)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r_o)} + \alpha_c)} \\
&\propto \frac{\Gamma(n_{c_o}^{(r_o)} + \alpha_{c_o})}{\Gamma(\sum_{c \in C} n_c^{(r_o)} + \alpha_c)} = \frac{\Gamma(1 + n_{-o, c_o}^{(r_o)} + \alpha_{c_o})}{\Gamma(1 + \sum_{c \in C} n_{-o, c}^{(r_o)} + \alpha_c)} \\
&= \frac{(n_{-o, c_o}^{(r_o)} + \alpha_{c_o})}{(\sum_{c \in C} n_{-o, c}^{(r_o)} + \alpha_c)} \frac{\Gamma(n_{-o, c_o}^{(r_o)} + \alpha_{c_o})}{\Gamma(\sum_{c \in C} n_{-o, c}^{(r_o)} + \alpha_c)} = \frac{(n_{-o, c_o}^{(r_o)} + \alpha_{c_o})}{(\sum_{c \in C} n_{-o, c}^{(r_o)} + \alpha_c)} \times T_{-o, 2}. \quad (4.54)
\end{aligned}$$

**Community assignment.** The likelihood of occurrence  $o$  in community  $c_o$  only depends on the occurrences of user  $u_o$  in that community. The community assignments of the occurrences of other users, and of the occurrences of user  $u_o$  in other communities are independent of the likelihood of  $o$  in  $c_o$ . Therefore, such independent formation in the third term  $T_3$  in Eq. 4.49 is removed as follows.

$$\begin{aligned}
T_3 &= \prod_{c \in C} \frac{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f, u}^{(c)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f, u}^{(c)} + \beta_u)} = \frac{\prod_{u \in U} \Gamma(n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} \prod_{c \in C \setminus c_o} \frac{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f, u}^{(c)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f, u}^{(c)} + \beta_u)} \\
&\propto \frac{\prod_{u \in U} \Gamma(n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} = \frac{\Gamma(n_{u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o}) \prod_{u \in U \setminus u_o} \Gamma(n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)}{\Gamma(\sum_{u \in U} n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} \\
&\propto \frac{\Gamma(n_{u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o})}{\Gamma(\sum_{u \in U} n_u^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} = \frac{\Gamma(1 + n_{-o, u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o})}{\Gamma(1 + \sum_{u \in U} n_{-o, u}^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} \\
&= \frac{(n_{-o, u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o})}{(\sum_{u \in U} n_{-o, u}^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} \frac{\Gamma(n_{-o, u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o})}{\Gamma(\sum_{u \in U} n_{-o, u}^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} = \frac{(n_{-o, u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o})}{(\sum_{u \in U} n_{-o, u}^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u)} \times T_{-o, 3} \quad (4.55)
\end{aligned}$$

**Independence of topics.** The likelihood of topic  $z_o$  in community  $c_o$  is independent of the topic proportion of other communities and the likelihood of other topics in community  $c_o$ . Therefore, the fourth term  $T_4$  in Eq. 4.49 is reduced as follows.

$$\begin{aligned}
T_4 &= \prod_{c \in C} \frac{\prod_{z \in Z} \Gamma(n_z^{(c)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c)} + \gamma_z)} = \frac{\prod_{z \in Z} \Gamma(n_z^{(c_o)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c_o)} + \gamma_z)} \prod_{c \in C \setminus c_o} \frac{\prod_{z \in Z} \Gamma(n_z^{(c)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c)} + \gamma_z)} \propto \frac{\prod_{z \in Z} \Gamma(n_z^{(c_o)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c_o)} + \gamma_z)} \\
&= \frac{\Gamma(n_{z_o}^{(c_o)} + \gamma_{z_o}) \prod_{z \in Z \setminus z_o} \Gamma(n_z^{(c_o)} + \gamma_z)}{\Gamma(\sum_{z \in Z} n_z^{(c_o)} + \gamma_z)} \propto \frac{\Gamma(n_{z_o}^{(c_o)} + \gamma_{z_o})}{\Gamma(\sum_{z \in Z} n_z^{(c_o)} + \gamma_z)} = \frac{\Gamma(1 + n_{-o, z_o}^{(c_o)} + \gamma_{z_o})}{\Gamma(1 + \sum_{z \in Z} n_{-o, z}^{(c_o)} + \gamma_z)} \\
&= \frac{(n_{-o, z_o}^{(c_o)} + \gamma_{z_o})}{(\sum_{z \in Z} n_{-o, z}^{(c_o)} + \gamma_z)} \frac{\Gamma(n_{-o, z_o}^{(c_o)} + \gamma_{z_o})}{\Gamma(\sum_{z \in Z} n_{-o, z}^{(c_o)} + \gamma_z)} = \frac{(n_{-o, z_o}^{(c_o)} + \gamma_{z_o})}{(\sum_{z \in Z} n_{-o, z}^{(c_o)} + \gamma_z)} \times T_{-o, 4} \quad (4.56)
\end{aligned}$$

**Topic assignment.** The likelihood of any term in other topics, and the likelihood of terms not occurring in message  $o.msg$  in topic  $z_o$  are independent of the assignment of words in  $o.msg$  to  $z_o$ . In other words, the assignment of words in  $o.msg$  to topic  $z_o$  only depends on the likelihood of terms occurring in  $o.msg$  in topic  $z_o$ . Therefore, the last term  $T_5$  in Eq. 4.49 is simplified by

$$\begin{aligned}
T_5 &= \prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)} = \frac{\prod_{w \in V} \Gamma(n_w^{(z_o)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z_o)} + \mu_w)} \prod_{z \in Z \setminus z_o} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)} \\
&\propto \frac{\prod_{w \in V} \Gamma(n_w^{(z_o)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z_o)} + \mu_w)} \propto \frac{\prod_{w \in o.msg} \Gamma(n_w^{(z_o)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z_o)} + \mu_w)} = \frac{\prod_{w \in o.msg} \Gamma(n_w.msg + n_{-w,w}^{(z_o)} + \mu_w)}{\Gamma(n.msg + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w)}, \tag{4.57}
\end{aligned}$$

where (1)  $n_w.msg$  is the number of occurrences of term  $w$  in message  $o.msg$ ; (2)  $n_{-w,w}^{(z_o)}$  is the number occurrences of term  $w$  assigned to topic  $z_o$  excluding the occurrences of  $w$  in  $o.msg$ ; (3)  $n.msg$  is the number of words in  $o.msg$ . By applying the property of the Gamma function, i.e.,  $\Gamma(x+1) = x\Gamma(x)$ , to the numerator of Eq. 4.57, we have

$$\begin{aligned}
\prod_{w \in o.msg} \Gamma(n_w.msg + n_{-w,w}^{(z_o)} + \mu_w) &= \prod_{w \in o.msg} (n_w.msg - 1 + n_{-w,w}^{(z_o)} + \mu_w) \times \\
&\quad (n_w.msg - 2 + n_{-w,w}^{(z_o)} + \mu_w) \dots (n_{-w,w}^{(z_o)} + \mu_w) \Gamma(n_{-w,w}^{(z_o)} + \mu_w) \\
&= \prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i - 1 + n_{-w,w}^{(z_o)} + \mu_w) \times \underbrace{\prod_{w \in o.msg} \Gamma(n_{-w,w}^{(z_o)} + \mu_w)}_{\text{numerator of } T_{-o,5}}. \tag{4.58}
\end{aligned}$$

Similarly, the denominator of Eq. 4.57 is restructured as

$$\begin{aligned}
\Gamma(n.msg + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) &= (n.msg - 1 + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) \times \\
&\quad (n.msg - 2 + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) \dots (\sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) \Gamma(\sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) \\
&= \prod_{i=1}^{n.msg} (i - 1 + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w) \times \underbrace{\Gamma(\sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w)}_{\text{denominator of } T_{-o,5}}. \tag{4.59}
\end{aligned}$$

Substituting the numerator and the denominator of Eq. 4.57 by the results of Eq. 4.58 and Eq. 4.59, respectively, term  $T_5$  becomes

$$T_5 = \prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)}{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)} \propto \frac{\prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i - 1 + n_{-w,w}^{(z_o)} + \mu_w)}{\prod_{i=1}^{n.msg} (i - 1 + \sum_{w \in V} n_{-w,w}^{(z_o)} + \mu_w)} \times T_{-o,5}. \tag{4.60}$$

Finally, by applying the results of Eq. 4.53, Eq. 4.54, Eq. 4.55, Eq. 4.56, and Eq. 4.60 to the corresponding terms in Eq. 4.52, the joint distribution of the region assignment,

community assignment, and topic assignment of occurrence  $o = \langle u, loc, msg, f, t \rangle$  given such assignments of other occurrences is

$$\begin{aligned}
P(r_o, c_o, z_o | \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \beta, \gamma, \mu, \alpha, \eta, \sigma) &= P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma) \times \frac{n_{-o, c_o}^{(r_o)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o, c}^{(r_o)} + \alpha_c} \\
&\times \frac{n_{-o, u_o}^{(c_o)} + n_{f, u_o}^{(c_o)} + \beta_{u_o}}{\sum_{u \in U} n_{-o, u}^{(c_o)} + n_{f, u}^{(c_o)} + \beta_u} \times \frac{n_{-o, z_o}^{(c_o)} + \gamma_{z_o}}{\sum_{z \in Z} n_{-o, z}^{(c_o)} + \gamma_z} \times \frac{\prod_{w \in o, msg} \prod_{i=1}^{n_w \cdot msg} (i - 1 + n_{-w, w}^{(z_o)} + \mu_w)}{\prod_{i=1}^{n \cdot msg} (i - 1 + \sum_{w \in V} n_{-w, w}^{(z_o)} + \mu_w)}.
\end{aligned} \tag{4.61}$$

Based on Eq. 4.61, the sampling rule for each assignment variable, i.e.,  $r_o$ ,  $c_o$ , and  $z_o$ , is then derived by removing terms that are independent of the likelihood of such a particular assignment. For example, to compute the likelihood of occurrence  $o = \langle u, loc, msg, f, t \rangle$  in region  $r$ , given that  $o$  is assigned to community  $c_o$  and topic  $z_o$ , only the first two terms in Eq. 4.61 are taken into account. The first term measures how close the geographic location of occurrence  $o$  is to the representative location of region  $r$ . The second term is the likelihood of community  $c_o$  in region  $r$ . Details of the sampling rules are presented as follows.

### 1. Region assignment sampling rule:

$$\begin{aligned}
P(r_o = r | c_o, z_o, \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \cdot) &= P(r | \eta_t) P(loc_o | loc_r, \sigma) \times \frac{n_{-o, c_o}^{(r)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o, c}^{(r)} + \alpha_c} \\
&\propto \exp\left(-\frac{|loc_o, loc_r|}{\sigma^2}\right) \times \frac{n_{-o, c_o}^{(r)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o, c}^{(r)} + \alpha_c}
\end{aligned} \tag{4.62}$$

where  $n_{-o, c}^{(r)}$  is the number of occurrences of users in region  $r$  that were assigned to community  $c$  excluding  $o$ .

### 2. Community assignment sampling rule:

$$\begin{aligned}
P(c_o = c | r_o, z_o, \mathbf{c}_{-o}, \mathbf{r}_{-o}, \mathbf{z}_{-o}, SN; \cdot) &\propto \frac{n_{-o, u_o}^{(c)} + n_{-o, f, u_o}^{(c)} + \beta_{u_o}}{\sum_{u \in U} n_{-o, u}^{(c)} + n_{-o, f, u}^{(c)} + \beta_u} \times \frac{n_{-o, c}^{(r_o)} + \alpha_c}{\sum_{c' \in C} n_{-o, c'}^{(r_o)} + \alpha_{c'}} \\
&\times \frac{n_{-o, z_o}^{(c)} + \gamma_{z_o}}{\sum_{z \in Z} n_{-o, z}^{(c)} + \gamma_z}
\end{aligned} \tag{4.63}$$

where  $n_{-o, u_o}^{(c)}$  is the number of occurrences of user  $u_o$  that were assigned to community  $c$ ;  $n_{-o, f, u_o}^{(c)}$  is the number of times  $u_o$  is contextually linked by other users in community  $c$ ;  $n_{-o, u}^{(c)}$  and  $n_{-o, f, u}^{(c)}$  are computed similarly to  $n_{-o, u_o}^{(c)}$  and  $n_{-o, f, u_o}^{(c)}$ , respectively, but applied to user  $u$ ;  $n_{-o, z}^{(c)}$  is the number of postings in community  $c$  that were assigned to topic  $z$ . All count variables are computed with the exclusion of occurrence  $o$  that is currently considered.

### 3. Topic assignment sampling rule:

$$P(z_o = z | r_o, c_o, \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, SN; \cdot) \propto \frac{n_{-o,z}^{(c_o)} + \gamma_z}{\sum_{z' \in Z} n_{-o,z'}^{(c_o)} + \gamma_{z'}} \times \frac{\prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i - 1 + n_{-w,w}^{(z)} + \mu_w)}{\prod_{i=1}^{n.msg} (i - 1 + \sum_{w \in V} n_{-w,w}^{(z)} + \mu_w)} \quad (4.64)$$

where  $n_{-w,w}^{(z)}$  is the number of occurrences of term  $w$  that were assigned to topic  $z$  excluding the occurrences of  $w$  in message  $o.msg$ .

**Updating multinomial parameters from assignment samples.** Given a sample  $\langle \mathbf{r}, \mathbf{c}, \mathbf{z} \rangle$  of the region assignments, community assignments, and topic assignments of all occurrences, the posterior distributions of (1) users in a community, i.e.,  $\phi_c$ , (2) communities in a region, i.e.,  $\theta_r$ , (3) topics of a community, i.e.,  $\pi_c$ , and (4) terms in a topic, i.e.,  $\varphi_z$ , are then derived. To be precise, let  $\mathbf{c}_c$  denote the occurrences assigned to community  $c$ ;  $\mathbf{c}_r$  denote the community assignments of occurrences in region  $r$ ;  $\mathbf{z}_c$  denote the topic assignments of messages of community  $c$ ; and  $\mathbf{z}_z$  denote the words assigned to topic  $z$ , then  $\phi_c$ ,  $\theta_r$ ,  $\pi_c$ , and  $\varphi_z$  are obtained as follows.

**1. Distribution of users in a community.** Given the occurrences assigned to community  $c$ , the distribution of users in  $c$  is

$$\begin{aligned} P(\phi_c | \mathbf{c}_c; \beta) &= \frac{P(\mathbf{c}_c, \phi_c | \beta)}{P(\mathbf{c}_c | \beta)} = \frac{P(\mathbf{c}_c | \phi_c) P(\phi_c | \beta)}{\int_{\phi_c} P(\mathbf{c}_c | \phi_c) P(\phi_c | \beta) d\phi_c} \\ &= \frac{\prod_{o \in \mathbf{c}_c} P(u_o | \phi_c) \prod_{u' \in o.f} P(u' | \phi_c) P(\phi_c | \beta)}{\int_{\phi_c} \prod_{o \in \mathbf{c}_c} P(u_o | \phi_c) \prod_{u' \in o.f} P(u' | \phi_c) P(\phi_c | \beta) d\phi_c} \\ &= \frac{\prod_{u \in U} P(u | \phi_c)^{n_u^{(c)} + n_{f,u}^{(c)}} P(\phi_c | \beta)}{\int_{\phi_c} \prod_{u \in U} P(u | \phi_c)^{n_u^{(c)} + n_{f,u}^{(c)}} P(\phi_c | \beta) d\phi_c} = \frac{\prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)}} \frac{1}{Beta(\beta)} \prod_{u \in U} \phi_{c,u}^{\beta_u - 1}}{\int_{\phi_c} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)}} \frac{1}{Beta(\beta)} \prod_{u \in U} \phi_{c,u}^{\beta_u - 1} d\phi_c} \\ &= \frac{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f,u}^{(c)})}{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f,u}^{(c)})} \prod_{u \in U} \phi_{c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u - 1} = Dir(\phi_c | n^{(c)} + \beta), \end{aligned} \quad (4.65)$$

where  $n^{(c)} = \langle n_u^{(c)} + n_{f,u}^{(c)} \rangle$ ,  $u \in U$ .

Having the posterior distribution of  $\phi_c$  identified as the *Dirichlet* distribution  $Dir(\phi_c | n^{(c)} + \beta)$ , the likelihood of user  $u$  in community  $c$ , i.e.,  $\phi_{c,u}$ , is estimated as the expectation of  $Dir(\phi_c | n^{(c)} + \beta)$ , computed as

$$\phi_{c,u} = \frac{n_u^{(c)} + n_{f,u}^{(c)} + \beta_u}{\sum_{u' \in U} n_{u'}^{(c)} + n_{f,u'}^{(c)} + \beta_{u'}}, \quad c \in C, u \in U. \quad (4.66)$$

**2. Distribution of communities in a region.** Given the community assignments of the occurrences in region  $r$ , the distribution of communities in  $r$  is

$$\begin{aligned}
P(\theta_r | \mathbf{c}_r; \alpha) &= \frac{P(\theta_r, \mathbf{c}_r | \alpha)}{P(\mathbf{c}_r | \alpha)} = \frac{P(\mathbf{c}_r | \theta_r) P(\theta_r | \alpha)}{\int_{\theta_r} P(\mathbf{c}_r | \theta_r) P(\theta_r | \alpha)} = \frac{\prod_{o \in r} P(c_o | \theta_r) P(\theta_r | \alpha)}{\int_{\theta_r} \prod_{o \in r} P(c_o | \theta_r) P(\theta_r | \alpha) d\theta_r} \\
&= \frac{\prod_{c \in C} P(c | \theta_r)^{n_c^{(r)}} P(\theta_r | \alpha)}{\int_{\theta_r} \prod_{c \in C} P(c | \theta_r)^{n_c^{(r)}} P(\theta_r | \alpha) d\theta_r} = \frac{\prod_{c \in C} \theta_{r,c}^{n_c^{(r)}} \frac{1}{\text{Beta}(\alpha)} \prod_{c \in C} \theta_{r,c}^{\alpha_c - 1}}{\int_{\theta_r} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)}} \frac{1}{\text{Beta}(\alpha)} \prod_{c \in C} \theta_{r,c}^{\alpha_c - 1} d\theta_r} \\
&= \frac{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)}{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)} \prod_{c \in C} \theta_{r,c}^{n_c^{(r)} + \alpha_c - 1} = \text{Dir}(\theta_r | n^{(r)} + \alpha), \tag{4.67}
\end{aligned}$$

where  $n^{(r)} = \langle n_c^{(r)} \rangle$ ,  $c \in C$ .

The likelihood of community  $c$  in region  $r$  is estimated as the expectation of  $\text{Dir}(\theta_r | n^{(r)} + \alpha)$  for the component  $\theta_{r,c}$ , computed as

$$\theta_{r,c} = \frac{n_c^{(r)} + \alpha_c}{\sum_{c' \in C} n_{c'}^{(r)} + \alpha_{c'}}, r \in R, c \in C. \tag{4.68}$$

**3. Topic proportion of a community.** Given the topic assignments of the messages of community  $c$ , the proportion of topics in  $c$  is

$$\begin{aligned}
P(\pi_c | \mathbf{z}_c; \gamma) &= \frac{P(\mathbf{z}_c, \pi_c | \gamma)}{P(\mathbf{z}_c | \gamma)} = \frac{P(\mathbf{z}_c | \pi_c) P(\pi_c | \gamma)}{P(\mathbf{z}_c | \gamma)} = \frac{\prod_{msg \in \mathbf{z}_c} P(z_{msg} | \pi_c) P(\pi_c | \gamma)}{\int_{\pi_c} \prod_{msg \in \mathbf{z}_c} P(z_{msg} | \pi_c) P(\pi_c | \gamma) d\pi_c} \\
&= \frac{\prod_{z \in Z} P(z | \pi_c)^{n_z^{(c)}} P(\pi_c | \gamma)}{\int_{\pi_c} \prod_{z \in Z} P(z | \pi_c)^{n_z^{(c)}} P(\pi_c | \gamma) d\pi_c} = \frac{\prod_{z \in Z} \pi_{c,z}^{n_z^{(c)}} \frac{1}{\text{Beta}(\gamma)} \prod_{z \in Z} \pi_{c,z}^{\gamma_z - 1}}{\int_{\pi_c} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)}} \frac{1}{\text{Beta}(\gamma)} \prod_{z \in Z} \pi_{c,z}^{\gamma_z - 1} d\pi_c} \\
&= \frac{\Gamma(\sum_{z \in Z} n_z^{(c)} + \gamma_z)}{\prod_{z \in Z} \Gamma(n_z^{(c)} + \gamma_z)} \prod_{z \in Z} \pi_{c,z}^{n_z^{(c)} + \gamma_z - 1} = \text{Dir}(\pi_c | n^{(c)} + \gamma), \tag{4.69}
\end{aligned}$$

where  $n^{(c)} = \langle n_z^{(c)} \rangle$ ,  $z \in Z$ .

The likelihood of topic  $z$  in community  $c$  is obtained from the expectation of  $\text{Dir}(\pi_c | n^{(c)} + \gamma)$  for the component  $\pi_{c,z}$ , computed as

$$\pi_{c,z} = \frac{n_z^{(c)} + \gamma_z}{\sum_{z' \in Z} n_{z'}^{(c)} + \gamma_{z'}}, c \in C, z \in Z. \tag{4.70}$$



**4. Distribution of terms in a topic.** Given words assigned to topic  $z$ , the distribution of terms in  $z$  is

$$\begin{aligned}
P(\varphi_z | \mathbf{z}_z; \mu) &= \frac{P(\mathbf{z}_z, \varphi_z | \mu)}{P(\mathbf{z}_z | \mu)} = \frac{P(\mathbf{z}_z | \varphi_z) P(\varphi_z | \mu)}{P(\mathbf{z}_z | \mu)} = \frac{\prod_{w \in \mathbf{z}_z} P(w | \varphi_z) P(\varphi_z | \mu)}{\int_{\varphi_z} \prod_{w \in \mathbf{z}_z} P(w | \varphi_z) P(\varphi_z | \mu) d\varphi_z} \\
&= \frac{\prod_{w \in V} P(w | \varphi_z)^{n_w^{(z)}} P(\varphi_z | \mu)}{\int_{\varphi_z} \prod_{w \in V} P(w | \varphi_z)^{n_w^{(z)}} P(\varphi_z | \mu) d\varphi_z} = \frac{\prod_{w \in V} \varphi_{z,w}^{n_w^{(z)}} \frac{1}{\text{Beta}(\mu)} \prod_{w \in V} \varphi_{z,w}^{\mu_w - 1}}{\int_{\varphi_z} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)}} \frac{1}{\text{Beta}(\mu)} \prod_{w \in V} \varphi_{z,w}^{\mu_w - 1} d\varphi_z} \\
&= \frac{\Gamma(\sum_{w \in V} n_w^{(z)} + \mu_w)}{\prod_{w \in V} \Gamma(n_w^{(z)} + \mu_w)} \prod_{w \in V} \varphi_{z,w}^{n_w^{(z)} + \mu_w - 1} = \text{Dir}(\varphi_z | n^{(z)} + \mu), \tag{4.71}
\end{aligned}$$

where  $n^{(z)} = \langle n_w^{(z)} \rangle$ ,  $w \in V$ .

The likelihood of term  $w$  in topic  $z$  is obtained from the expectation of  $\text{Dir}(\varphi_z | n^{(z)} + \mu)$  for the component  $\varphi_{z,w}$ , computed as

$$\varphi_{z,w} = \frac{n_w^{(z)} + \mu_w}{\sum_{w' \in V} n_{w'}^{(z)} + \mu_{w'}}, z \in Z, w \in V. \tag{4.72}$$

#### 4.4.5 Gibbs Sampling Algorithm

Having the sampling rules and the formulas for updating the multinomial parameters derived, the Gibbs sampling algorithm for the *rLinkTopic* model is shown in Algorithm 5. The algorithm runs through three stages. In the initialization, each occurrence is randomly assigned to a region, a community, and a topic, respectively. In the second stage, called *Burn-in*, sampling rules are applied to build a Markov chain for assignment variables  $\mathbf{r}$ ,  $\mathbf{c}$  and  $\mathbf{z}$ . In the last stage, the algorithm collects assignment samples and updates the multinomial parameters  $\phi_c$ ,  $\theta_r$ ,  $\pi_c$ , and  $\varphi_z$ . These variables represent the distributions of (1) users in a community, (2) communities in a region, (3) topics of a community, and (4) terms in a topic, respectively. The expectations of  $\phi_c$ ,  $\theta_r$ ,  $\pi_c$ , and  $\varphi_z$  are the output of the model.

**Computational complexity.** Three main tasks of the proposed algorithm are the sampling for the (1) region assignment (line 12), (2) community assignment (line 13), and (3) topic assignment (line 14). For a snapshot  $sn_t$  having  $|R_t|$  regions, the computation for an occurrence  $o$  at a sampling step has time complexity  $O(|R_t| + |C| + |Z|)$ . Therefore, the time complexity of the algorithm for a network of  $T$  snapshots  $SN = \{sn_1, sn_2, \dots, sn_T\}$  and with  $I$  iterations for sampling will be  $O(I \times T \times |sn_t| \times (|R_t| + |C| + |Z|))$ .

---

**Algorithm 5:** Gibbs sampling algorithm for the *rLinkTopic* probabilistic model.

**rLinkTopic**( $SN = \{sn_1, \dots, sn_T\}, |C|, |Z|, \alpha, \beta, \gamma, \mu, minRad, \sigma$ )

---

**Input:**  
 $SN = \{sn_1, sn_2, \dots, sn_T\}$ : snapshots of a social network  
 $|C|$ : number of communities to be extracted  
 $|Z|$ : number of topics associated with communities  
 $minRad$ : a threshold to determine representative locations of regions  
 $\sigma$ : prior standard deviation for Gaussian  
 $\alpha, \beta, \gamma, \mu$ : Dirichlet hyperparameters

**Output:**  
 $\theta$ : distributions of communities in regions  
 $\phi$ : distributions of users in communities  
 $\pi$ : topic proportions of communities  
 $\varphi$ : distributions of terms in topics

- 1  $I := Iterations; BurnIn := BurnInSteps;$
- 2 /\* Initialization \*/
- 3 determineCentreOfRegions( $minRad$ );
- 4 **foreach**  $t = 1..T$  **do**
- 5     **foreach**  $o \in sn_t$  **do**
- 6          $r, c, z \sim uniform()$ ;
- 7         assign  $o$  to  $r, c,$  and  $z$ ;
- 8 /\* Burn-in \*/
- 9 **foreach**  $i = 1..I$  **do**
- 10     **foreach**  $t = 1..T$  **do**
- 11         **foreach**  $o \in sn_t$  **do**
- 12              $r \sim exp(-\frac{|loc_o, loc_r|}{\sigma^2}) \times \frac{n_{-o, c_o}^{(r)} + \alpha_{c_o}}{\sum_{c \in C} (n_{-o, c}^{(r)} + \alpha_c)}$ ;
- 13              $c \sim \frac{n_{-o, u_o}^{(c)} + n_{-o, f, u_o}^{(c)} + \beta_{u_o}}{\sum_{u \in U} (n_{-o, u}^{(c)} + n_{-o, f, u}^{(c)} + \beta_u)} \times \frac{n_{-o, c}^{(r_o)} + \alpha_c}{\sum_{c' \in C} (n_{-o, c'}^{(r_o)} + \alpha_{c'})} \times \frac{n_{-o, z_o}^{(c)} + \gamma_{z_o}}{\sum_{z \in Z} (n_{-o, z}^{(c)} + \gamma_z)}$ ;
- 14              $z \sim \frac{n_{-o, z}^{(c_o)} + \gamma_z}{\sum_{z' \in Z} (n_{-o, z'}^{(c_o)} + \gamma_{z'})} \times \frac{\prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i-1 + n_{-w, w}^{(z)} + \mu_w)}{\prod_{i=1}^{n.msg} (i-1 + \sum_{w \in V} n_{-w, w}^{(z)} + \mu_w)}$ ;
- 15             assign  $o$  to  $r, c, z$ ;
- 16     /\* Update parameters \*/
- 17     **if**  $i > BurnIn$  **then**
- 18         update parameters  $\phi, \theta, \pi, \varphi$  using
- 19         Eq. 4.66, 4.68, 4.70, 4.72, respectively;

---

## 4.5 Evaluation Measures

This section presents two measures applied to evaluate the *rLinkTopic* model. The first measure is introduced to study the regional aspect of communities. The second measure is the perplexity of the model.

### 4.5.1 Spatial Entropy Measure

In information theory, the entropy measure describes how much information is needed on average to encode the observations of a distribution. If the observations are almost random then one needs more information to describe them because the number of possible instances of such observations is large [68, 108]. On the other hand, less information is needed to encode a distribution whose observations are somehow prior. Based on this principle, we introduce a spatial entropy measure to study the geographic localization of communities. Specifically, the measure gives a high (entropy) value to a community whose members are randomly distributed over a large geographic area and gives a small value to a community whose members are located in a small geographic area. Given a community, the spatial entropy is obtained as follows.

For each snapshot, a representative location is first computed for each user  $u$  in the community, i.e., a user  $u$  might occur at different locations during a snapshot. Suppose a user  $u$  has a trajectory of  $k$  geographic locations,  $traj(u) = \{loc_1, loc_2, \dots, loc_k\}$ , then the representative location  $u.loc$  of  $u$  is derived as the centroid of  $traj(u)$ , i.e,  $u.loc = centroid(traj(u))$ . By this, a community  $c$  is regarded as a spatial distribution of the representative locations of users, i.e.,  $\{u.loc | u \in c\}$ . We then apply a simple approach to organize the spatial bounding box of the area of interest i.e., the spatial coverage of the dataset, as a regular grid consisting of spatial cells,  $\mathcal{G} = \{sc\}$ .

To compute spatial entropy, a spatial density, denoted  $p_c(sc)$ , is defined for community  $c$  in cell  $sc$ . This density is the likelihood of finding the users of community  $c$  in the area of cell  $sc$ , computed as

$$p_c(sc) = \frac{|u.loc \in sc | u \in c|}{|c|}. \quad (4.73)$$

The spatial entropy of community  $c$  is then obtained from the spatial density measures derived over all cells of the grid. That is

$$entropy(c) = - \sum_{sc:p_c(sc)>0}^{\mathcal{G}} p_c(sc) \log(p_c(sc)) \in [0, \log(|\mathcal{G}|)]. \quad (4.74)$$

The defined *entropy* measure gets a minimum value 0 if the users of a community are located within one cell, and gets a maximize value  $\log(|\mathcal{G}|)$  if the representative locations of users are uniformly distributed over all cells. Therefore, it can be normalized by  $\log(|\mathcal{G}|)$  to have a value in the range of  $[0, 1]$ . By applying this measure, the spatial characteristics of communities that have different topics of discussion and/or communities located in different geographic areas can be explored. One can further employ this measure to evaluate models

for community detection regarding the geographic localization of communities extracted. These will be discussed in the experimental evaluations presented in Section 4.6.

### 4.5.2 Perplexity Measure

The concept of perplexity comes from the cross entropy measure that is mainly used to evaluate the capacity of a probabilistic model in generating an observed dataset [23]. Given a dataset  $\mathcal{D}$  and two probabilistic models  $P$  and  $M$  developed for  $\mathcal{D}$ , the cross entropy of  $M$  and  $P$  is computed as

$$H_{\mathcal{D}}(P, M) = - \sum_{o \in \mathcal{D}} P(o) \times \log(M(o)). \quad (4.75)$$

The underlying idea is that if  $M$  is identical to  $P$  then the cross entropy becomes the entropy of  $P$ . It further implies that if  $P$  was the true probability distribution of the dataset then one would expect to have a model, e.g.,  $M$ , that has cross entropy  $H_{\mathcal{D}}(P, M)$  close to the entropy of  $P$ . In other words, the cross entropy is at its minimum when the model  $M$  is identical to the true distribution  $P$ . Even though the true distribution  $P$  underlying a dataset is generally unknown, the fact is that the Maximum Likelihood Estimator (MLE) approaches to the true distribution as the number of observations in the dataset goes to infinity. This leads to the concept of *log probability* (or *corpus entropy* as used in text mining literature [75]) of model  $M$ , defined as follows.

$$H_{\mathcal{D}}(M) = - \frac{1}{|\mathcal{D}|} \log(M(\mathcal{D})) = - \frac{1}{|\mathcal{D}|} \sum_{o \in \mathcal{D}} \log(M(o)) \quad (4.76)$$

Note that Eq. 4.76 is the cross entropy of model  $M$  and the MLE for the dataset  $\mathcal{D}$ . To be more precise, let us assume that observations in  $\mathcal{D}$  are of objects  $U = \{u_1, u_2, \dots, u_{|U|}\}$ , and let  $n_u$  denote the number of times  $u$  occurs in  $\mathcal{D}$ . Then, Eq. 4.76 can be rewritten as the formula of the cross entropy of  $M$  and  $MLE$  as follows.

$$\begin{aligned} H_{\mathcal{D}}(M) &= - \frac{1}{|\mathcal{D}|} \sum_{o \in \mathcal{D}} \log(M(o)) = - \sum_{u \in U} \frac{n_u}{|\mathcal{D}|} \times \log(M(u)) \\ &= - \sum_{u \in U} MLE(u) \times \log(M(u)) = H_{\mathcal{D}}(MLE, M). \end{aligned} \quad (4.77)$$

Based on these principles, model  $M$  that has less log probability compared to others that are developed for the same dataset is better in terms of the capacity of generating the data. The perplexity of the model is defined as

$$perplexity(M) = e^{H_{\mathcal{D}}(M)}. \quad (4.78)$$

Applying to our *rLinkTopic* model, the likelihood of a user occurrence is computed using Eq. 4.33 and summing over regions  $R_t$ , communities  $C$ , and topics  $Z$ , as shown in Eq. 4.79.

$$\begin{aligned} P(o|\phi, \pi, \varphi, \theta_t, \beta, \gamma, \mu, \alpha, \eta, \sigma) &= \quad (4.79) \\ \sum_{r \in R_t} P(r|\eta) P(loc_o|loc_r, \sigma) &\sum_{c \in C} P(c|\theta_t, r) P(u_o|\phi_c) \prod_{u' \in o.f} P(u'|\phi_c) \sum_{z \in Z} P(z|\pi_c) \prod_{w \in o.msg} P(w|\varphi_z) \end{aligned}$$

In the experimental evaluations presented in the next section, the perplexity measure is employed to show how the *rLinkTopic* model improves itself while learning community structures as more sampling steps are accomplished, and as it is trained by more data.



## 4.6 Experiments

This section presents experiments to evaluate our *rLinkTopic* model for extracting communities from social network data. We show that interesting and intuitive results are obtained regarding the geographic localization and the topics of communities discovered by the *rLinkTopic* model. We compare *rLinkTopic* with a recent and most related approach called TUCRM [102] in terms of the regional aspect of communities extracted by the two models. We further show that *rLinkTopic* outperforms TURCM in terms of the perplexity measure. All experiments are conducted using *Twitter* data.

### 4.6.1 Twitter Datasets

We collected tweets from US and Europe for around six months from June 1 to November 28, 2012 and extracted all the geotagged tweets for our experiments. That is, in addition to other features, e.g., the `userId`, contextual links, and message, each tweet in the datasets has a geographic coordinate (latitude/longitude) stating from where it has been sent. Relevant statistics of these two datasets are shown in Table 4.3.

Table 4.3: Maps and statistics of *Twitter* datasets used for experimental evaluations.

US dataset	EUROPE dataset
	
Bounding Box: (-122.0,25.0,-65.0,49.0) Number of users: 9.612.945 Number of tweets: 100.587.624	Bounding Box: (-12.8,36.9,38.8,69.2) Number of users: 7.629.360 Number of tweets: 78.015.912

### 4.6.2 Link Structure and Spatial Characteristics of Datasets

We first apply some statistical measures to the two datasets to get an idea of the link structures and spatial distances between the occurrences of *Twitter* users and to find evidence to further support our approach. More precisely, we show that in the datasets the spatial proximity among occurrences of users gives a good indication for social links. That is, the closer two tweets are geographically, the more likely there is an explicit link between respective users. For this purpose, we partition each of the two datasets into 10-day interval snapshots. For each such snapshot, link structures are extracted by first counting the tweets that have either the feature *replyTo* or *mentionedUsers*, resulting in a set of so-called *s-linked* users, denoted  $SL$ . For each user  $u_i \in SL$  there exists a user  $u_j \in SL$  such that  $u_i$  replied to or mentioned  $u_j$ , denoted  $u_i \rightarrow u_j$ . From the set  $SL$ , we then obtain a subset of so-called *bi-linked users* such that if  $u_i \rightarrow u_j$ , then also  $u_j \rightarrow u_i$ , i.e., there is a bidirectional interaction between the two users. Averaged over all snapshots, about 3.66% of users in the US dataset and 2.05% of users in the EUROPE dataset are *s-linked*. The number of bi-linked users in the US dataset is about 1.11%, while in the EUROPE dataset the number is 0.35%. These results imply that even though there are many users, the link structures among these users are very sparse.

To study how geographic locations of users affects the formation or the existence of explicit links between them, we compute the spatial distance between *s-linked* users or rather their geographic occurrences. The result shows that explicit links between *Twitter* users are strongly governed by the spatial distance between them. As shown in Figure 4.5, most of the links occur between users in a distance less than 150 km from each other. This supports our claim that communities are formed by users in geographic regions.

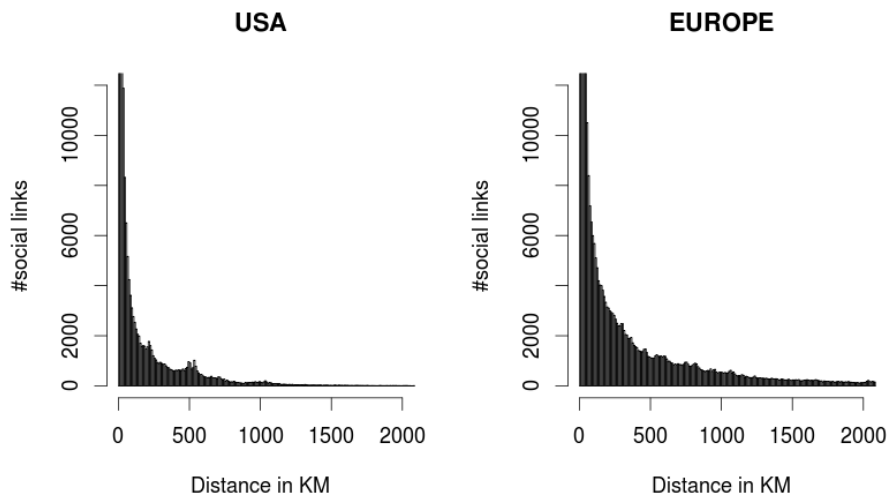


Figure 4.5: Distribution of spatial distances between *Twitter* users having explicit links.

### 4.6.3 Experimental Setup

This section presents the results of applying our *rLinkTopic* model to extract communities of *Twitter* users. We show that discovered communities are associated with intuitive topics<sup>7</sup> and users in each community are geographically localized. Different subsets of the two datasets shown in Table 4.3 are used to conduct the evaluations. The results presented in the following are obtained from 4 selected (sub-)datasets described in Table 4.4. Note that each of the two datasets **Sub-England 01** and **Sub-England 02** is a subset of the dataset **Sub-England 03**. We use these three datasets of the same geographic area to see changes in the community structures discovered from different time intervals.

Table 4.4: Statistics of *Twitter* datasets used to extract regional *LinkTopic* communities. These datasets are created from the **EUROPE** and **US** datasets described in Table 4.3.

Dataset	Bounding Box	Users	Tweets	Terms	Time
<b>Sub-England 01</b>	-4.00,50.70,1.60,52.70	188.312	519.883	222.333	June 15 - Jun 20
<b>Sub-England 02</b>	-4.00,50.70,1.60,52.70	339.095	1.146.598	423.646	June 15 - Jun 30
<b>Sub-England 03</b>	-4.00,50.70,1.60,52.70	740.407	3.665.714	1.120.133	June 10 - July 30
<b>Sub-US</b>	-75.20,40.3,-73.3,41.36	210.361	685.809	309.896	Oct 25 - Nov 10

**Tweet Normalization.** For each geotagged tweet, the following features are used as input to the *rLinkTopic* model: `userId`, `time`, `coordinates`, `contextual links`, and `tweet content`. Some sample tweets are shown in Table 4.5. We apply lexical normalization techniques proposed in [46] to convert abbreviations and slang words to normal word format before cleaning the text, i.e., removing special characters such as `#`, `&`, `$`, removing stop words, and stemming words. Each resulting dataset is then organized into daily snapshots for conducting experiments.

Table 4.5: Sample tweets showing the format of input data for the *rLinkTopic* model.

Userid	Time	Lng	Lat	Contextual links	Message
JKGosling	2012-07-01	-0.203	51.527	MattKingBoo	I love the maps on sale. Get what you want.
SafeDiego86	2012-07-01	0.173	51.433	henrywinter,Dartford	It should be no question of that.
pkfashoni	2012-07-01	-0.196	51.523		Someone tells me a book to read please.

**Data filtering.** For each selected dataset shown in Table 4.4, we further apply two filtering steps to refine the data before running the *rLinkTopic* model to extract communities. Particularly, users who posted less than  $numM$  messages, and terms that occurred less

<sup>7</sup>By *intuitive* we mean that one can empirically classify topics in specific subjects.

than  $numW$  times in the dataset are removed. Details of the filtering parameters applied and the refined datasets are summarized in Table 4.6.

Table 4.6: Statistics of the selected datasets in Table 4.4 after empirically filtering users who sent less than 01 message per day and terms that occurred less than 02 times per day in each dataset.

Dataset	$numM$	$numW$	#Users	#Tweets	#Terms	Time
Sub-England 01	05	10	37.567	456.624	10.811	June 15 - Jun 20
Sub-England 02	15	30	10.643	720.114	7.259	June 15 - June 30
Sub-England 03	50	100	11.739	2.057.895	12.731	June 10 - July 30
Sub-US	15	30	11.914	502.156	8.399	Oct 25 - Nov 10

**Parameter settings.** Table 4.7 shows the input parameters of the  $rLinkTopic$  model. In our experiments, for each dataset, all parameters excepted  $|C|$  and the number of *Burn-In* steps are empirically determined. Particularly, we assign  $|Z| = 20$ ,  $\sigma = 0.033$  (about  $5km$ ),  $minRad = 0.066$  (a region is about  $100km^2$ ). The number of *Burn-in* steps is identified based on the perplexity computed while sampling. Specifically, we find that after a round 800 to 850 iterations, the perplexity starts to be always smaller in the later steps. We employ the heuristic reported in [44] to assign values for the hyperparameters of the *Dirichlet* distributions in the model. Particularly,  $\alpha_c = 50/|C|$  for all  $c \in C$ ,  $\gamma_z = 50/|Z|$  for all  $z \in Z$ ,  $\beta_u = 0.1$  for all  $u \in U$  and  $\mu_w = 0.1$  for all  $w \in V$ . We then run the model with different values of  $|C|$  and use the perplexity measure to select the best one, i.e., the value of  $|C|$  that returns the lowest perplexity of the model. The results presented in the following are computed using the parameter settings in Table 4.7 where the number of communities for each dataset is selected based on the perplexity shown in Figure 4.6.

Table 4.7: Setting values of parameters for the  $rLinkTopic$  model to apply to the selected datasets used in experiments. The number of communities  $|C|$  and the *Burn-In* steps are determined based on the perplexity measure. The heuristic reported in [44] is used to assign values for the hyperparameters  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\mu$ . The two parameters  $\sigma$  and  $minRad$  are empirically determined to build regions.

Dataset	$ C $	$ Z $	$\sigma$	$minRad$	$\alpha_c$	$\beta_u$	$\gamma_z$	$\mu_w$	<i>Burn-In</i> steps
Sub-England 01	70	10	0.033	0.066	0.014	0.01	0.1	0.01	800
Sub-England 02	40	10	0.033	0.066	0.025	0.01	0.1	0.01	820
Sub-England 03	20	10	0.033	0.066	0.050	0.01	0.1	0.01	820
Sub-US	30	10	0.033	0.066	0.033	0.01	0.1	0.01	800



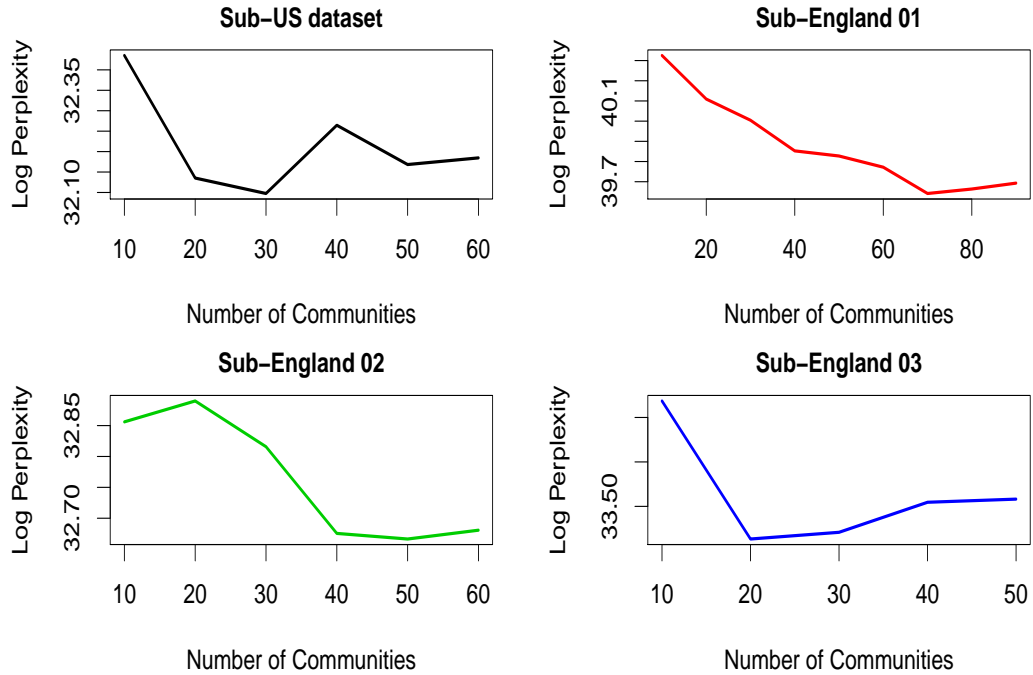


Figure 4.6: Perplexity of the *rLinkTopic* model computed for the selected datasets in Table 4.4. For each dataset, different number of communities are used to determine the best setting.

#### 4.6.4 Regional LinkTopic Communities

**Topics of communities.** Using the above parameter settings for the *rLinkTopic* model to extract communities, we find that topics associated with communities are intuitive even though *Twitter* data are so noisy. Generally, topics extracted in all datasets can be empirically classified into the groups *politics*, *jobs*, *social activities*, *weather*, *music* and *social events*, *social media*, *social networks (SNs)*, and *sports*. There are some topics that contain terms describing different subjects. Such topics are labeled as *general*<sup>8</sup>. The top 15 terms that have the highest likelihood in selected topics associated with communities discovered from the **Sub-US** dataset and **Sub-England 01-03** datasets are shown in Table 4.8 and Table 4.9, respectively.

By studying the topic proportions of communities we further find that each community is associated with at most two topics and between them one topic has much higher likelihood. Topic proportions of selected communities extracted from the **Sub-US** dataset are presented in Figure 4.7 and Table 4.10.

<sup>8</sup>The label associated with a topic is empirically named by the author of this dissertation. The *general* label does not mean the topic is about a general subject but it rather denotes that the topic is not intuitive enough to be classified.

Table 4.8: Eight selected topics associated with communities extracted by the *rLinkTopic* model from the **Sub-US** dataset.

<b>Jobs-Topic: 19</b>		<b>Politics-Topic: 02</b>		<b>Weather-Topic: 11</b>		<b>Charity-Topic: 08</b>	
<b>Term</b>	<b>Likelihood</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>
job	0.1956	insur	0.1638	forecast	0.0905	cake	0.0451
tweetmyjob	0.0604	fastest	0.0827	cloudi	0.0566	home	0.0354
retail	0.0188	job	0.0819	nov	0.0538	breezi	0.0354
manag	0.0175	fairfield	0.0819	shower	0.0536	word	0.0346
alert	0.0170	recruit	0.0819	mostli	0.0448	point	0.0341
sale	0.0126	aflac	0.0819	oct	0.0399	donation	0.0341
hospit	0.0121	grow	0.0819	partli	0.0369	spread	0.0333
prudenti	0.0120	agenc	0.0684	chance	0.0365	alon	0.0317
marketing	0.0119	obama	0.0261	sunni	0.0289	demi	0.0309
account	0.0117	mitt	0.0178	rain	0.0259	rebuild	0.0298
internship	0.0104	ugli	0.0172	sat	0.0205	amc	0.0257
veteranjob	0.0093	alert	0.0135	thu	0.0178	eric	0.0254
insur	0.0079	vote	0.0117	lake	0.0140	maynor	0.0254
assist	0.0078	blue	0.0113	sun	0.0114	east	0.0185
businessmgr	0.0072	economi	0.0105	mon	0.0107	teamheat	0.0182
<b>Restaurant: 09</b>		<b>Social Media: 16</b>		<b>Tourism: 17</b>		<b>School: 18</b>	
<b>Term</b>	<b>Likelihood</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>
coupon	0.0748	instagood	0.0687	airport	0.0288	studytim	0.0725
ridgewood	0.0309	photoofdai	0.0677	station	0.0248	previous	0.0606
restaur	0.0256	instamood	0.0463	intern	0.0231	found	0.0604
funni	0.0159	iger	0.0393	jfk	0.0205	unavail	0.0576
real	0.0159	iphonesia	0.0376	john	0.0179	earlier	0.0536
acn	0.0159	picofthedai	0.0354	kennedi	0.0179	hurri	0.0451
blue	0.0145	iphone4	0.0271	art	0.0171	manag	0.0390
pino	0.0145	iphonegraphi	0.0244	museum	0.0162	stumbl	0.0244
rauti	0.0145	iphoneonli	0.0244	amc	0.0152	brother	0.0240
bgm	0.0145	iphon	0.0240	loew	0.0134	girl	0.0226
pizza	0.0138	instadaili	0.0215	north	0.0130	find	0.0214
dara	0.0121	earth	0.0124	train	0.0129	kid	0.0175
sender	0.0121	stuck	0.0123	york	0.0120	student	0.0154
teamheat	0.0111	ecuador	0.0103	park	0.0116	final	0.0151
ridgefield	0.0092	tweegram	0.0086	modern	0.0096	boi	0.0131

Table 4.9: Eight selected topics associated with communities extracted by the *rLinkTopic* model from three datasets **Sub-England 01-03**.

<b>Jobs-Topic: 01</b>		<b>Arts-Topic: 04</b>		<b>Weather-Topic: 05</b>		<b>SNs-Topic: 07</b>	
<b>Term</b>	<b>Likelihood</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>
job	0.1206	art	0.0539	mph	0.0908	track	0.1148
contract	0.1117	free	0.0423	rain	0.0872	updat	0.1063
develop	0.0326	exhibit	0.0408	wind	0.0869	visit	0.1063
engin	0.0267	chd	0.0386	humid	0.0791	info	0.1063
manag	0.0231	fit	0.0383	temperatur	0.0725	transpond	0.1063
stalban	0.0219	train	0.0275	baromet	0.0652	follow	0.0911
analyst	0.0177	crawlei	0.0216	slowli	0.0475	theo	0.0241
softwar	0.0165	bristol	0.0197	hpa	0.0422	roi	0.0087
consult	0.0141	person	0.0182	rise	0.0402	word	0.0084
rental	0.0137	group	0.0178	fall	0.0361	spread	0.0084
senior	0.0116	artist	0.0176	temp	0.0141	tweet	0.0084
busi	0.0100	raw	0.0149	deg	0.0128	check	0.0084
month	0.0099	event	0.0138	steadi	0.0123	outofcontrol	0.0084
support	0.0085	materi	0.0137	weather	0.0119	krai	0.0066
web	0.0083	buzz	0.0136	pressur	0.0057	swag	0.0065
<b>Traffics-Topic: 06</b>		<b>Football-Topic: 02</b>		<b>Music-Topic: 09</b>		<b>General-Topic: 08</b>	
<b>Term</b>	<b>Likelihood</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>	<b>Term</b>	<b>Likeli.</b>
station	0.0794	work	0.0609	plai	0.0702	work	0.0654
railwai	0.0699	watch	0.0519	radio1	0.0266	watch	0.0514
greater	0.0297	people	0.0410	music	0.0217	people	0.0460
bristol	0.0181	feel	0.0398	live	0.0031	ill	0.0437
cross	0.0155	ill	0.0390	girl	0.0024	feel	0.0423
airport	0.0148	game	0.0388	heart	0.0023	hope	0.0364
ben	0.0134	plai	0.0364	boy	0.0023	home	0.0332
west	0.0133	home	0.0363	life	0.0020	thing	0.0323
king	0.0127	man	0.0356	home	0.0018	great	0.0320
heathrow	0.0127	great	0.0356	station	0.0018	happi	0.0319
lhr	0.0121	hope	0.0354	nice	0.0017	man	0.0318
hounslow	0.0119	wait	0.0341	talk	0.0017	wait	0.0314
midland	0.0096	happi	0.0332	weekend	0.0016	follow	0.0310
cambridg	0.0094	euro2012	0.0305	song	0.0015	girl	0.0299
climb	0.0090	start	0.0305	happi	0.0015	year	0.0297

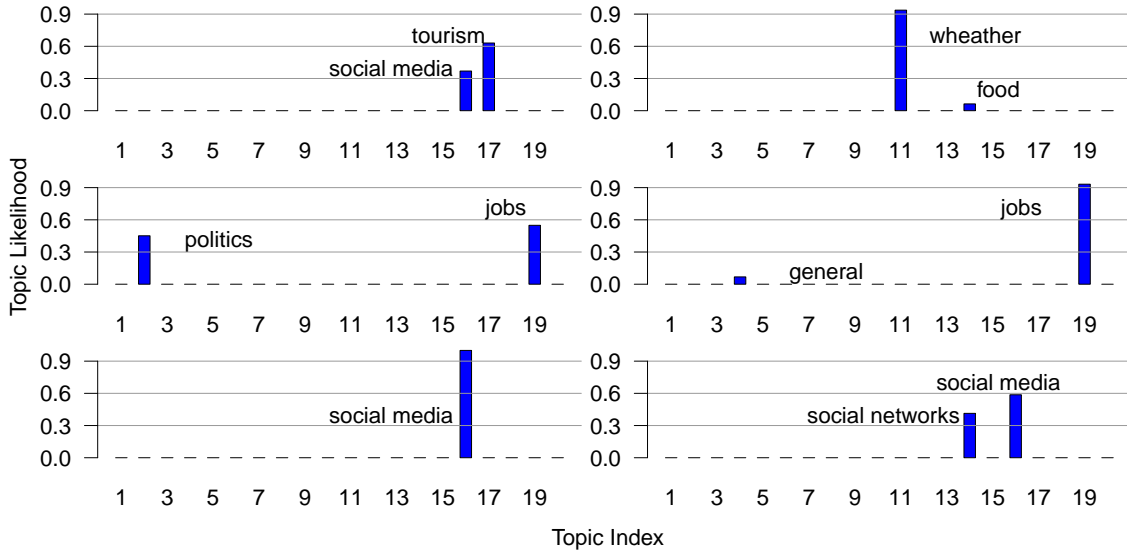


Figure 4.7: Topic proportions of 6 communities extracted from the **Sub-US** dataset. The most prominent topics associated with each community are manually classified.

Table 4.10: Details of the topic proportions of 10 communities extracted from the **Sub-US** dataset. The index of communities, i.e.,  $c \in C$ , and the index of topics, i.e.,  $z \in Z$ , are returned from *rLinkTopic* model. Each column is the topic proportion of a community, i.e.,  $\pi_c = \{P(z|c)\}, z \in Z$ .

Topic Index	Community Index									
	01	02	03	04	05	06	07	08	09	10
01	4.5E-7	4.6E-7	6.0E-7	5.6E-7	4.4E-7	5.4E-7	4.6E-7	4.3E-7	5.2E-7	4.0E-7
02	3.2E-7	1.7E-7	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	<b>0.45</b>	2.9E-7
03	4.6E-7	4.5E-7	6.2E-7	5.4E-7	4.4E-7	5.6E-7	4.8E-7	4.3E-7	5.1E-7	4.1E-7
04	<b>0.99</b>	1.7E-7	4.4E-7	3.9E-7	3.3E-7	<b>0.99</b>	<b>0.99</b>	3.0E-7	3.7E-7	2.9E-7
05	4.6E-7	4.3E-7	4.9E-7	5.1E-7	4.6E-7	6.2E-7	5.3E-7	3.5E-7	5.1E-7	4.3E-7
06	3.9E-7	3.2E-7	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
07	3.2E-7	3.2E-7	4.4E-7	<b>0.48</b>	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
08	3.2E-7	4.7E-7	6.2E-7	5.8E-7	4.7E-7	5.6E-7	4.8E-7	4.5E-7	5.0E-7	4.0E-7
09	3.2E-7	3.2E-7	2.1E-6	3.9E-7	<b>0.99</b>	4.0E-7	3.4E-7	3.0E-7	3.7E-7	<b>0.99</b>
10	3.2E-7	8.4E-7	4.4E-7	5.3E-7	4.5E-7	5.8E-7	4.7E-7	3.8E-7	4.9E-7	4.6E-7
11	3.2E-7	1.6E-7	4.4E-7	<b>0.51</b>	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
12	3.2E-7	4.3E-7	6.4E-7	5.4E-7	4.8E-7	5.9E-7	4.8E-7	3.5E-7	5.2E-7	4.0E-7
13	3.2E-7	4.7E-7	6.4E-7	5.1E-7	4.4E-7	5.8E-7	4.6E-7	4.5E-7	5.4E-7	4.4E-7
14	3.2E-7	3.2E-7	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
15	3.2E-7	3.2E-7	<b>0.99</b>	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
16	3.2E-7	<b>0.36</b>	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	3.7E-7	2.9E-7
17	3.2E-7	<b>0.63</b>	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	<b>0.99</b>	5.5E-7	2.9E-7
18	3.2E-7	4.0E-7	6.1E-7	6.8E-7	4.7E-7	5.5E-7	4.9E-7	4.2E-7	3.7E-7	2.9E-7
19	3.2E-7	3.2E-7	4.4E-7	3.9E-7	3.3E-7	4.0E-7	3.4E-7	3.0E-7	<b>0.54</b>	2.9E-7
20	3.2E-7	4.3E-7	6.2E-7	5.4E-7	4.7E-7	5.8E-7	4.8E-7	3.4E-7	5.2E-7	4.2E-7

**Geographic locations.** In addition to topics, communities discovered by the *rLinkTopic* model exhibit regional characteristics. Particularly, users in each community are spatially located close to each other when occurring in the network. Geographic locations of users in selected communities extracted from the **Sub-US** dataset and **Sub-England 01-03** datasets are shown in Figure 4.8 and Figure 4.9, respectively. In terms of application, one can further explore the geographic area and topics of communities extracted by the *rLinkTopic* model to get more insights into the characteristics of users in specific areas. This, however, is not presented here due to the lack of knowledge about local geographic areas.

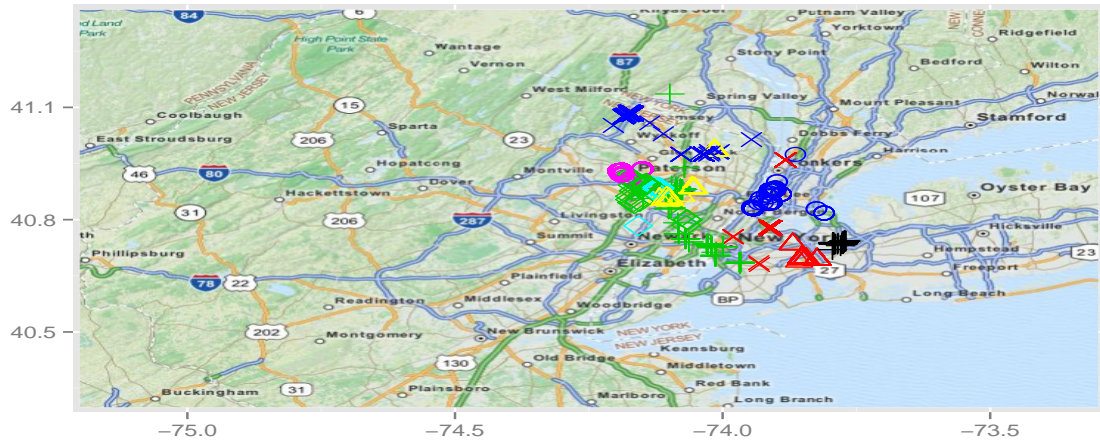
#### 4.6.5 Quantitative Evaluation

We employ the two measures presented in the previous section, i.e., the spatial entropy measure and the perplexity measure, to evaluate the effectiveness of the *rLinkTopic* model in extracting communities. Particularly, by comparing the results of the *rLinkTopic* model with the results of the TURCM model [102] we show that communities extracted by *rLinkTopic* reveal better geographic localization and *rLinkTopic* outperforms TURCM in terms of the perplexity measure.

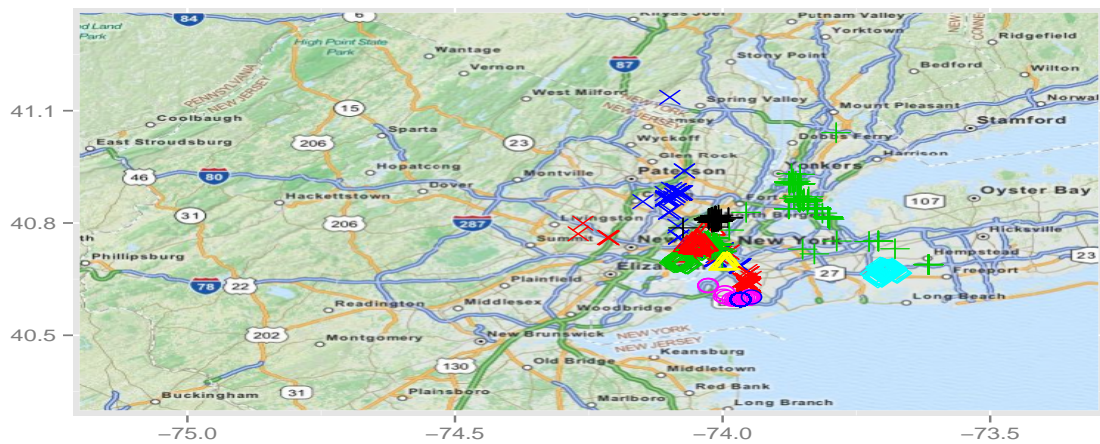
**Geographic localization.** We first run the *rLinkTopic* model with two settings: (1) to extract regional *LinkTopic* communities and (2) to extract (only) *LinkTopic* communities by manually setting the number of regions to 1 as input to the *rLinkTopic* model. This allows the *rLinkTopic* model to extract communities in the way that no region assignments are done. We then employ the TURCM model with different settings for the number of communities to be extracted and select the best setting regarding the perplexity measure. It is noted here that TURCM derives *single topic* communities. That is, the number of communities actually returned from TURCM is  $|C| \times |Z|$ , given the input  $|C|$  and  $|Z|$ . Because of this, the number of communities discovered by TURCM is always greater than the number of *rLinkTopic* communities.

The steps to conduct the comparison are as follows: (1) all communities extracted by the two settings of *rLinkTopic* are considered; (2) communities discovered by TURCM are first manually classified based on their topic and then randomly selected to have the same number as for *rLinkTopic* communities; (3) the spatial entropy measure is applied to compute entropy of communities in two ways: (a) daily entropy, i.e., entropy of a community is computed based on daily occurrences of users; and (b) ten-day entropy, i.e., entropy of a community is computed based on occurrences of users in ten consecutive days.

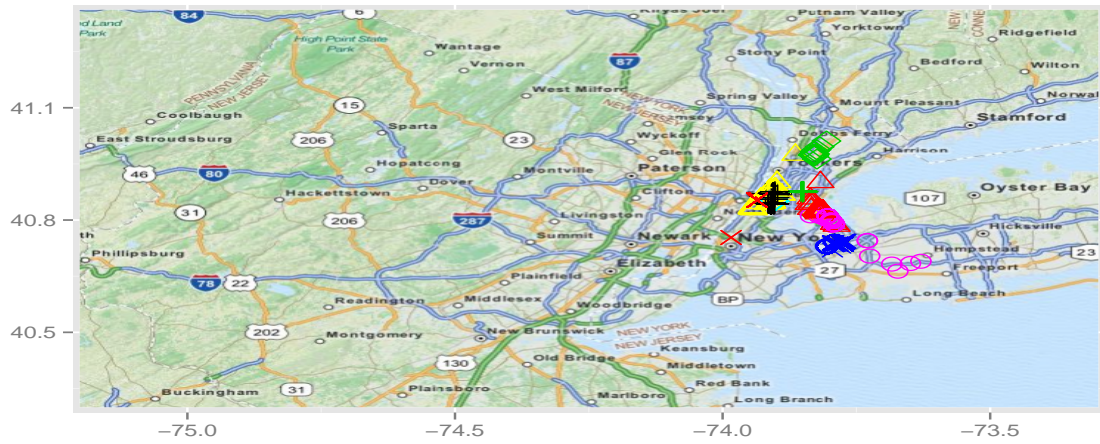
The results show that about half of the number of communities extracted by the TURCM model are comparable with *LinkTopic* communities, i.e., communities extracted by the *rLinkTopic* model without assigning occurrences of users to regions. The obtained regional *LinkTopic* communities always have less entropy compared to TURCM and *LinkTopic* communities. Especially, such a difference is significantly shown by the measure of the



(a) Politics Community



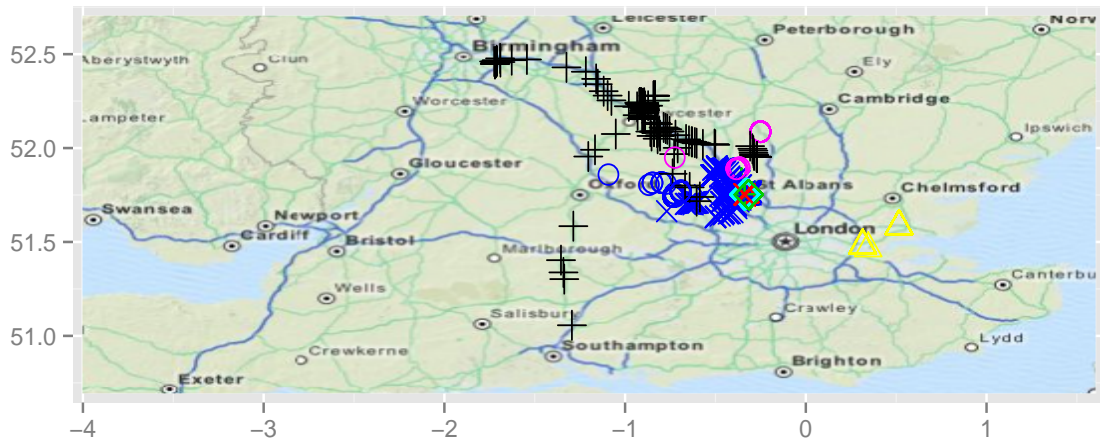
(b) Weather Community



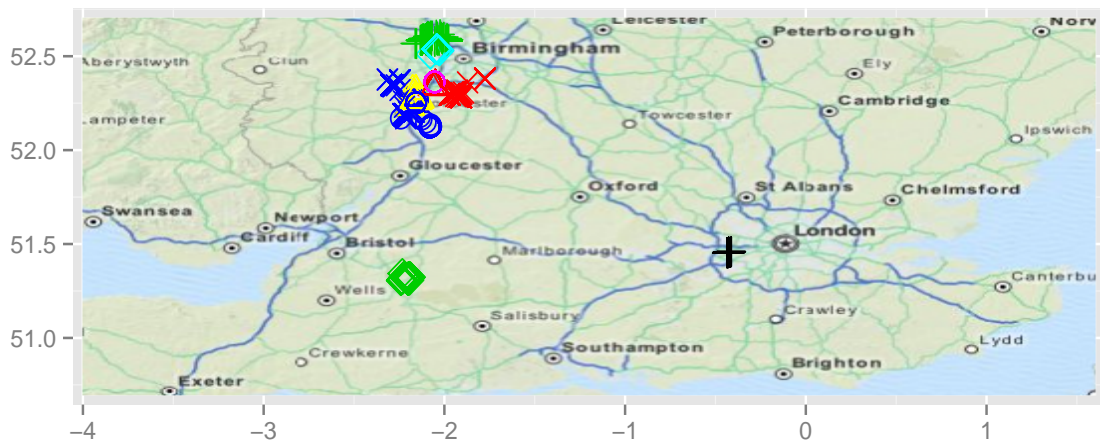
(c) Jobs Community

Figure 4.8: Geographic locations of users in selected communities extracted from the **Sub-US** dataset.

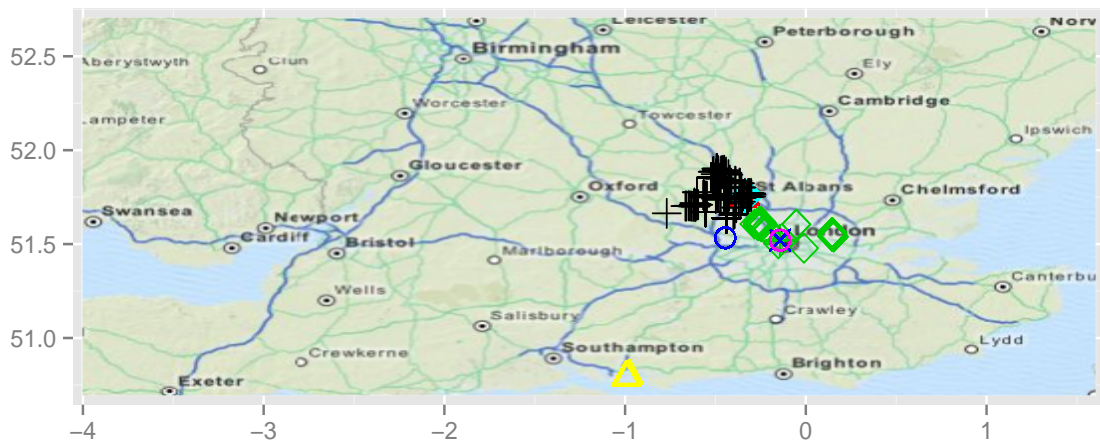




(a) Weather Community (Sub-England 01)



(b) Football Community (Sub-England 02)



(c) Music Community (Sub-England 03)

Figure 4.9: Geographic locations of users in selected communities extracted from datasets **Sub-England 01-03**.

ten-day entropy. These indicate the effectiveness of the *rLinkTopic* model in extracting communities that are geographically localized. Figure 4.10 shows the spatial entropy of selected communities discovered by the (1) *rLinkTopic* model with the parameter setting shown in Table 4.7, (2) *rLinkTopic* model with the same setting except the number of regions set to 1, and (3) the TURCM model.

**Perplexity Analysis.** We compare the perplexity of *rLinkTopic* and TURCM to show the effectiveness of *rLinkTopic* in terms of fitting community structures to the selected datasets. For this purpose, we apply three methods, denoted *M1*, *M2*, and *M3*, to compute the perplexity of the two models.

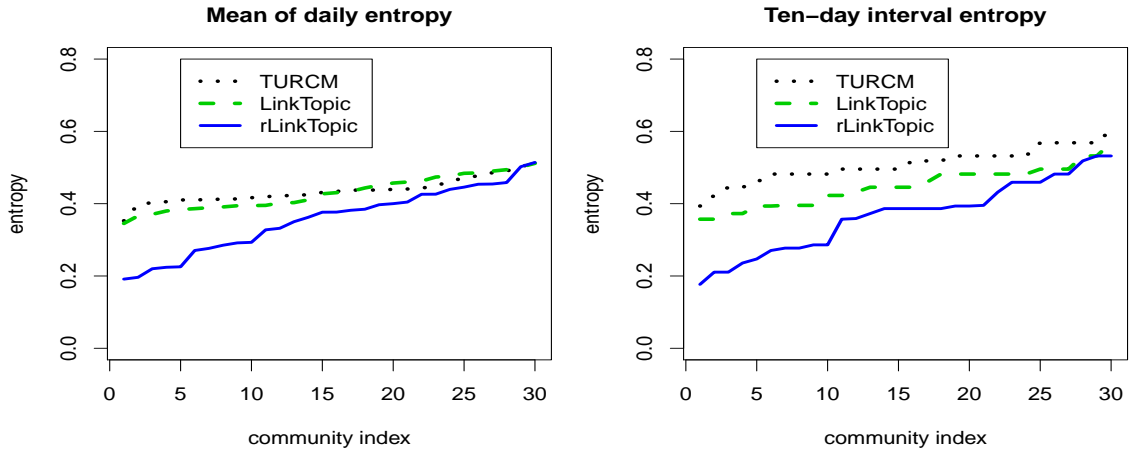
- *M1*: no data are used to train the models before computing the perplexity.
- *M2*: a portion of each daily snapshot is used to train the models.
- *M3*: a number of consecutive (full) snapshots is used to train the models.

For each method, we compute the perplexity at different sampling steps to see how the models learn community structures as more sampling steps are accomplished. For the methods *M2* and *M3*, we further compute the perplexity as more data are used to train the models.

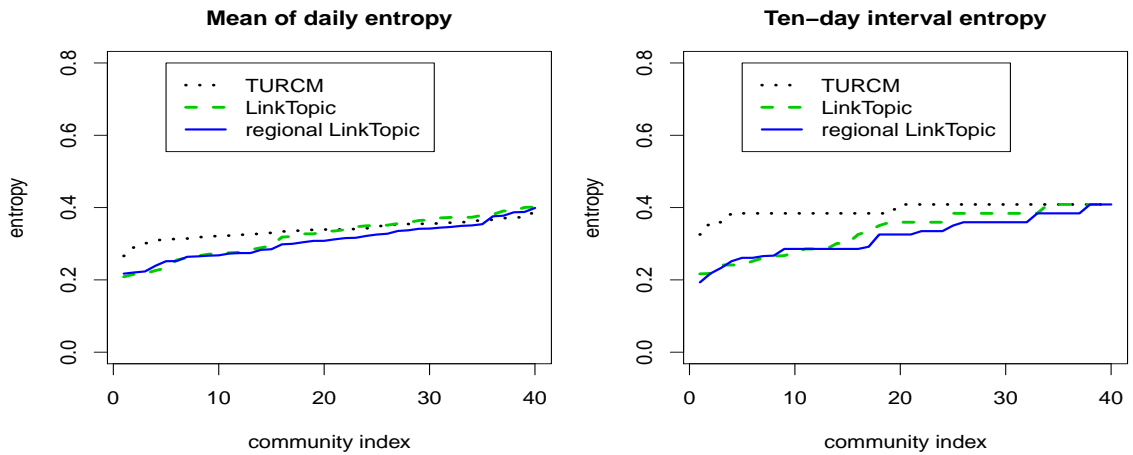
Having the results computed as described, we find that in all cases the perplexity of *rLinkTopic* is much less than the perplexity of TURCM. This implies that employing spatio-temporal proximity of users together with their contextual links in extracting communities derives much better results in terms of fitting the underlying community structures in the datasets. Furthermore, *rLinkTopic* improves the perplexity faster than TURCM when no data are used to train the models. Both models have a similar learning rate when being trained with the same data before computing their perplexity. This holds for both methods *M2* and *M3*. The results computed from the **Sub-England 03** dataset, i.e., the largest one among 4 selected datasets, are selected to support our findings. Figure 4.11 shows the perplexity of both models when no training is applied. It can be observed that the perplexity of the *rLinkTopic* model decreases quickly after the *Burn-in* stage (820 iterations), which means that our model learns community structures faster than TURCM does. Figure 4.12 and Figure 4.13 show the perplexity of the two models when different amounts of data, i.e., different percentages of each daily snapshot and different number of snapshots, are used to train the models.

Based on the behavior of the perplexity of both models, we find that relying on the occurrences of users on a daily basis to learn communities will properly return better results. This claim is clearly supported by the perplexity of the two models presented in Figure 4.14. Specifically, the more daily occurrences are used to train the models the lower the perplexity is obtained. This, however, does not hold when the models are trained by more consecutive (full) snapshots. As shown in Figure 4.14 (right), the perplexity of both models quickly increases again after being trained by 36 daily snapshots.

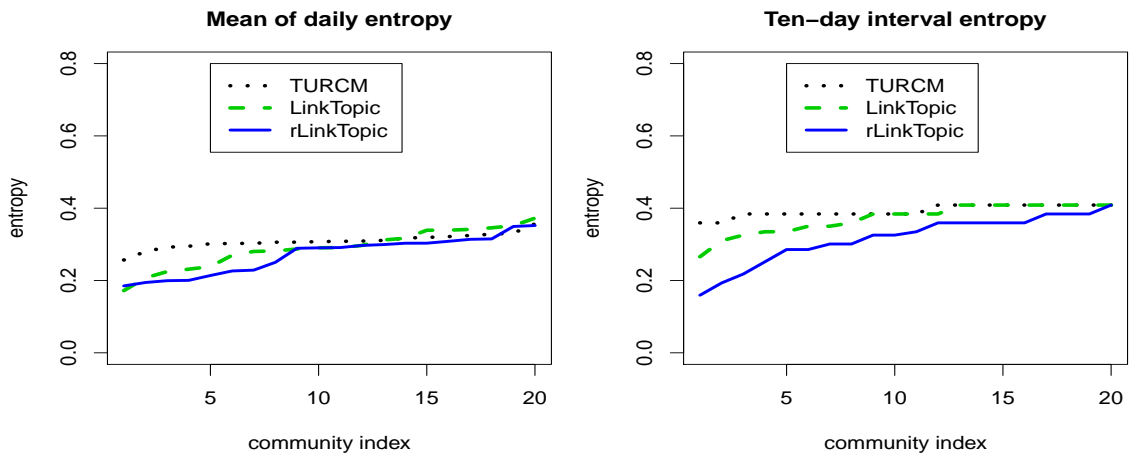




(a) Selected communities in the **Sub-US** dataset



(b) Selected communities in the **Sub-England 02** dataset



(c) Selected communities in the **Sub-England 03** dataset

Figure 4.10: Spatial entropy of communities extracted by the *rLinkTopic* and TURCM models from the **Sub-England 02-03** (a) and **Sub-US** (b) datasets. The indices of communities in each plot are ordered based on the spatial entropy measure.

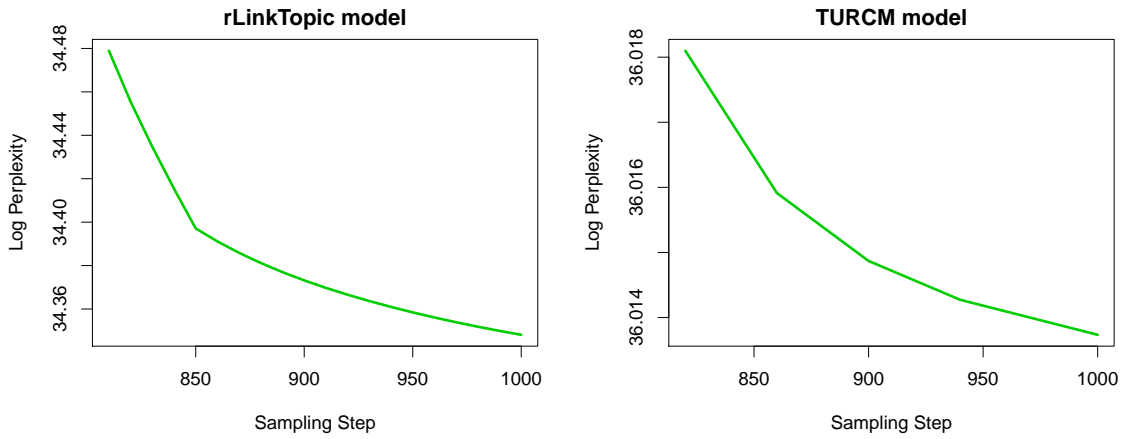


Figure 4.11: Perplexity of *rLinkTopic* and TURCM computed at different sampling steps from the **Sub-England 03** dataset when no data are used to train the models. The perplexity of *rLinkTopic* decreases quickly indicating that *rLinkTopic* improves its learning capacity better than TURCM does.

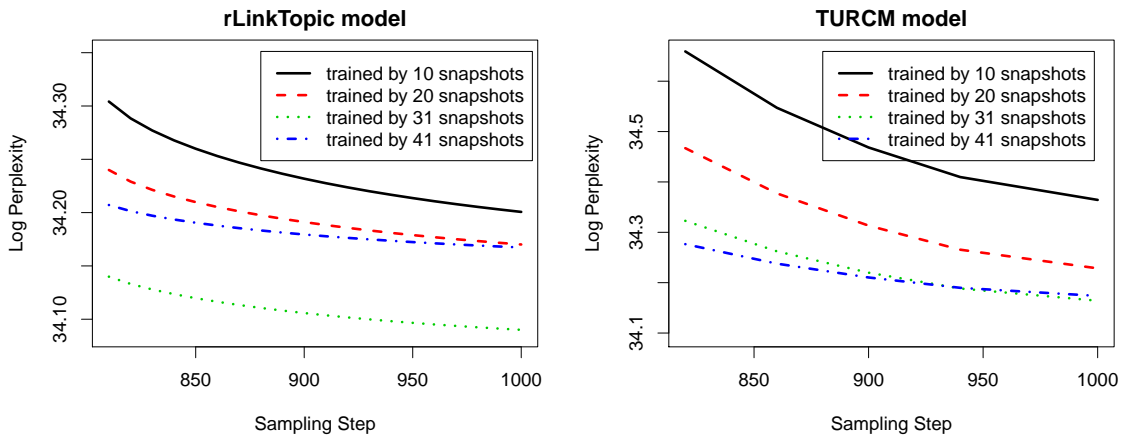


Figure 4.12: Perplexity of the *rLinkTopic* and TURCM models computed at different sampling steps from the **Sub-England 03** dataset when both models are trained by a number of full snapshots.

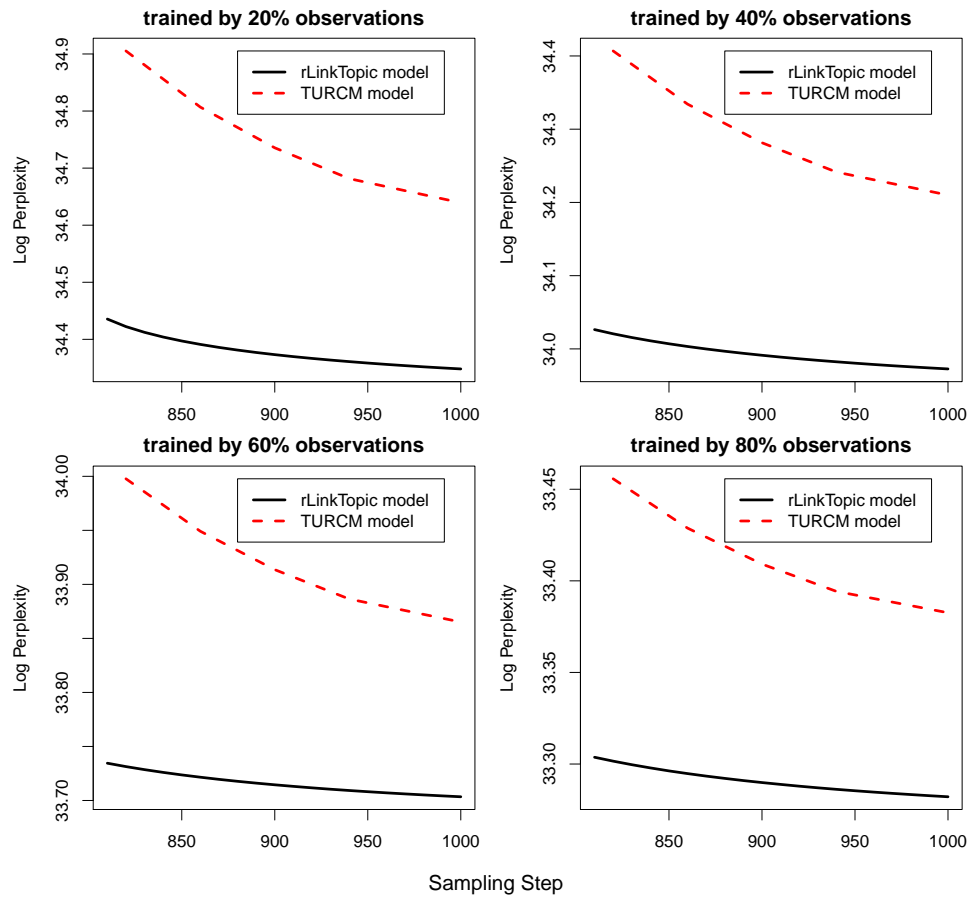


Figure 4.13: Perplexity of *rLinkTopic* model and TURCM model computed at different sampling steps from the **Sub-England 03** dataset when a portion of each daily snapshot is used to train the models.

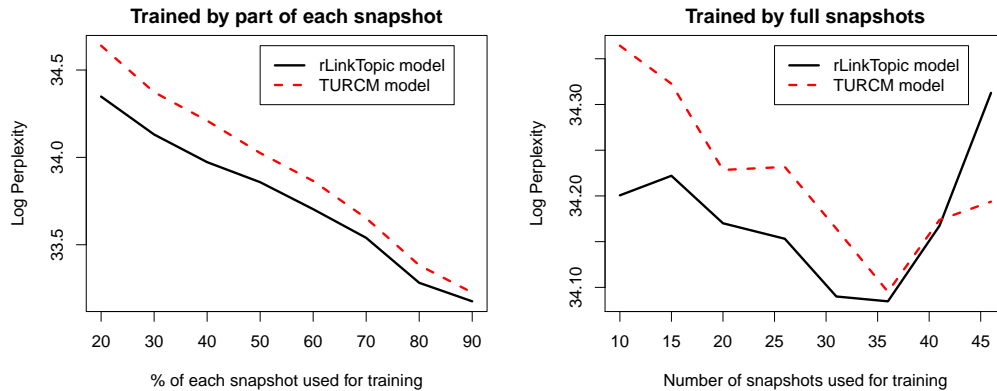


Figure 4.14: Perplexity of *rLinkTopic* model and TURCM model computed at the last sampling step from the **Sub-England 03** dataset. The methods *M2* and *M3* are applied to hold out the data for training the models.

## 4.7 Summary and Discussion

We have introduced a new type of community in social networks, i.e., the community of users occurring in a social network within spatio-temporal proximity, sharing similar topics, and having contextual links while posting messages to the network. A probabilistic model called *rLinkTopic* has been developed to discover such communities. The two important features that have never been considered in existing studies, i.e., the regional aspect of communities and the contextual links of users, are addressed in our model. We conducted extensive evaluations using *Twitter* data. The experimental results show that this approach discovers communities that are characterized not only by the topics but also by regional aspects. This property of communities cannot be obtained by existing approaches to community detection. Also, due to the consideration of the spatio-temporal proximity and the contextual links of users in extracting communities, our model gives much better results in terms of the perplexity measure, when compared to other models.

**Nonparametric extension.** In the *rLinkTopic* model, the number of communities  $|C|$  and the number of topics  $|Z|$  are assumed to be the input parameters. In our experiments, we empirically choose  $|Z|$  and run the model with different values of  $|C|$  to find the best setting regarding the perplexity measure derived. However, different values of  $|Z|$  affect more or less the extracted communities. Furthermore, such a method is intuitively inefficient because one has to run the model different times. This is the common weakness of the parametric Bayesian approach. Fortunately, with the success of the nonparametric Bayesian approach in clustering data [3, 113], this shortcoming of the *rLinkTopic* model can be solved. Particularly, the model can be extended by employing a *Dirichlet process* as prior distribution for each of the parameters  $|C|$  and  $|Z|$ . By this, one neither has to specify the number of communities nor the number of topics to be extracted.

**Dynamics of communities.** Based on our experimental results, we find that community structures, e.g., on *Twitter*, change over time. This observation is obtained from relating the time span of a dataset and the number of communities fitted by the model for that dataset. Particularly, the number of communities fitted by the model for different subsets of a dataset regarding the time interval decreases as more snapshots are used to discover communities. For example, considering the selected datasets shown in Table 4.4, the number of communities fitted by the model for three datasets **Sub-England 01, 02, 03** are 70, 40, and 20, respectively, as shown in Figure 4.6. This implies the dynamics of communities, i.e., more communities are observed for a short time interval but not many communities exist for a long time, and thus brings new questions about the evolution of communities. For example, how to capture changes in the community memberships of users, changes in the topics of communities, and so on. In the next chapter, a comprehensive framework is developed based on the *rLinkTopic* model to address such questions.

## Chapter 5

# Analysis of Community Evolution

### 5.1 Overview and Objectives

Communities in a social network evolve over time due to several reasons. A user is interested in the topics of a community and joins as a new member while some users might leave the community. The happening of social events, e.g., an election, and other phenomena also lead to the evolution of communities. Such an evolution is implied by changes in the features describing a community. These include, for example, users in the community, topics of the community, and geographic locations of the users. Given that a community is characterized by even more features, analyzing its evolution thus is a challenging task. This is because one has to have a complex model that is able to discover communities and to capture changes in as many features describing a community as possible. To date, existing approaches for the analysis of evolving communities attempt to study changes with respect to one feature, which are the community members [7, 22, 73, 74]. The concept of *evolution* is therefore defined only in the context of the user population of a community over time. Because of this, no information is obtained with respect to how other features of the community evolve. From an application perspective, one is usually interested not only in the dynamics of users, e.g., which users are in a community at what time, but also in other features that describe the community over time. These observations motivate our study and development of a comprehensive framework that takes more features of interest into account to study the evolution of communities in social networks. Particularly, in this chapter, we introduce a probabilistic model called *ErLinkTopic* that is an extension of the *rLinkTopic* model developed in the previous chapter for extracting regional *LinkTopic* communities and analyzing their complex evolution. By stating complex evolution, we are particularly interested in changes in the features describing a community as formalized in the *rLinkTopic* model. These include (1) the community membership of users in a community, which is characterized by  $\phi_c$ ; (2) topic proportion of a community, which is characterized by  $\pi_c$ ; and (3) terms occurring in a community topic, which is characterized by  $\varphi_z$ . The idea is that if  $\phi_c$ ,  $\pi_c$ , and  $\varphi_z$  are appropriately derived over time then one can base on the

changes in these variables to formalize community evolution. For example, a community is stable in terms of its members during a time period if there is no changes in  $\phi_c$ . Similarly, the evolution of the prominence of community topics and a topic itself is extracted from  $\pi_c$ , and  $\varphi_z$  over time, respectively. Also, because information about geographic locations is associated with users’ postings, the model further supports the study of changes in the regional and geographic characteristics of communities.

This chapter is organized as follows. Section 5.2 presents the underlying data model and introduces notations used to present the *ErLinkTopic* model. In Section 5.3, we first describe how *rLinkTopic* is extended to build *ErLinkTopic* that can discover communities and, at the same time, capture their evolution (Section 5.3.1). We then give detailed steps to derive a Gibbs sampling algorithm to compute the posterior distribution of the *ErLinkTopic* model (Section 5.3.2). Section 5.4 introduces measures to identify specific changes in the features describing a community. The results of our experimental evaluations using *Twitter* data are presented in Section 5.5. We summarize this chapter and give an outlook for ongoing work in Section 5.6.

## 5.2 Data Model and Notations

In this section, we first describe the data model underlying our *ErLinkTopic* framework and then introduce notations used throughout this chapter.

In the *rLinkTopic* model proposed in the previous chapter, a social network is formalized as a sequence of snapshots. The model relies on the occurrences of users in each snapshot to identify users who occur in the network within spatio-temporal proximity. This *co-occurrence* feature together with the contextual links and the topics of user postings are employed to extract communities. By this, the temporal order of the occurrences of users, i.e., the order of snapshots, is not important and is discarded in the *rLinkTopic* model. Our aim in the development of the *ErLinkTopic* model, however, is to take advantage of the *rLinkTopic* model to extract communities; and, at the same time, to capture community evolution. For the latter aspect, the temporal order is crucial, because it is used to explain the evolution of the characteristics of a community over time. To achieve these goals, we organize network snapshots in a sequence of sliding windows, each of which consists of a number of consecutive snapshots. The general idea is that communities are extracted within each sliding window, i.e., the temporal order of the snapshots in a sliding window is discarded. Information about the community structures obtained from the current sliding window then is employed to derive communities at the next sliding window. By this, we implicitly make an assumption that the community membership of users, topic proportion of communities, and distribution of terms in topics are stable during a sliding window and gradually evolve between two consecutive ones. Adopting the data model introduced in the previous chapter, the concept of sliding windows is formalized as follows.

**Definition 5.1 (Network Sliding Window)** Given a social network  $SN = \{sn_1, sn_2, \dots, sn_T\}$  and a time span  $\Delta t = [t_s, t_e]$ , a sliding window  $\mathcal{W}_t$  of size  $\Delta t$  is a sequence of consecutive snapshots  $\mathcal{W}_t = \{sn_{t_s}, \dots, sn_{t_e}\}$ .

Having the sliding window defined, a social network is now considered a sequence of sliding windows, i.e.,  $SN = \{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_T\}$ , which is the underlying data model for the *ErLinkTopic* framework. Note that for simplicity  $T$  is also used to denote the number of sliding windows in the network. The sliding window data model is illustrated in Figure 5.1, where a slot presents a snapshot and a sequence of consecutive slots indicates a sliding window. The time span of a snapshot and/or a sliding window is identified based on the application and analysis task under consideration. This will be discussed again in the experimental evaluations presented in Section 5.6.

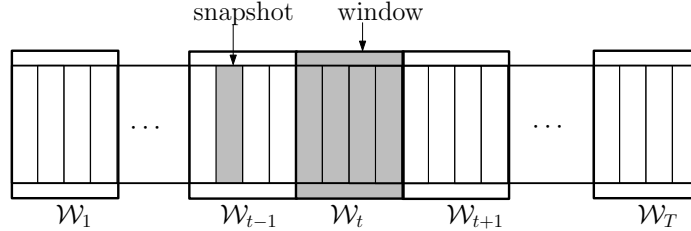


Figure 5.1: Illustration of the sliding window data model underlying the *ErLinkTopic* framework for extracting regional *LinkTopic* communities and analyzing their evolution.

It is noted that by organizing network snapshots in sliding windows one can continuously apply the *rLinkTopic* model to extract communities from a single sliding window, e.g.,  $\mathcal{W}_t$ . This is done by considering only the occurrences of users in the snapshots of the sliding window  $\mathcal{W}_t$  to derive the posterior distribution of the *rLinkTopic* model. In other words, the distribution of communities in regions, distribution of users in communities, topic proportion of communities, and distribution of terms in topics are learned by the *rLinkTopic* model from the occurrences of users in the snapshots belonging to  $\mathcal{W}_t$ . Based on this principle, *rLinkTopic* is extended to *ErLinkTopic* to discover communities and, at the same time, capture their evolution.

To present the *ErLinkTopic* model, the main notations used in the previous chapter for the *rLinkTopic* model are employed and some other notations are introduced, all of which are described in Table 5.1.

It should be emphasized that the subscript  $t$  in variables  $\theta_t$ ,  $\phi_t$ ,  $\pi_t$ ,  $\varphi_t$ ,  $\mathbf{r}_t$ ,  $\mathbf{c}_t$ , and  $\mathbf{z}_t$  introduced in Table 5.1 indicates the time index of the sliding window, not of the snapshot. For instance, the notation  $\phi_{t,c}$  denotes the distribution of users in the community  $c$ , and  $\phi_{t,c,u}$  denotes the likelihood of user  $u$  in the community  $c$ , where  $c$  is obtained from the occurrences of users in the snapshots of the sliding window  $\mathcal{W}_t$ .

Table 5.1: Notations used in the *ErLinkTopic* model for extracting regional *LinkTopic* communities and analyzing their evolution.

Notation	Description
$U$	set of users in social network, $u$ is a user in $U$
$C$	set of communities, $c$ is a community in $C$
$V$	vocabulary set, $w$ is a word in $V$
$Z$	set of community topics, $z$ is a topic in $Z$
$R_{\mathcal{W}_t}$	set of geographic regions created from snapshots of sliding window $\mathcal{W}_t$
$\theta_t$	set of community distributions in geographic regions $R_{\mathcal{W}_t}$ , i.e., $\theta_t = \{\theta_r\}, r \in R_{\mathcal{W}_t}$
$\phi_t$	set of user distributions for communities $C$ at window $\mathcal{W}_t$ , i.e., $\phi_t = \{\phi_{t,c}\}, c \in C$
$\pi_t$	set of topic proportions of communities $C$ at window $\mathcal{W}_t$ , i.e., $\pi_t = \{\pi_{t,c}\}, c \in C$
$\varphi_t$	set of term distributions for topics $Z$ at window $\mathcal{W}_t$ , i.e., $\varphi_t = \{\varphi_{t,z}\}, z \in Z$
$\mathbf{r}_t$	region assignments of the occurrences of users at window $\mathcal{W}_t$
$\mathbf{c}_t$	community assignments of the occurrences of users at window $\mathcal{W}_t$
$\mathbf{z}_t$	topic assignments of the messages of users at window $\mathcal{W}_t$

### 5.3 ErLinkTopic Probabilistic Model

This section presents in detail the *ErLinkTopic* model for extracting regional *LinkTopic* communities and analyzing their evolution. In Section 5.3.1, a discussion explaining how *rLinkTopic* is employed to develop *ErLinkTopic* is given. We present the steps to derive a Gibbs sampling algorithm for the *ErLinkTopic* model in Section 5.3.2.

#### 5.3.1 rLinkTopic to ErLinkTopic

Typically, a two-step approach is applied to study the evolution of communities. In the first step, communities are extracted independently of the occurrences of users at different time points, e.g., snapshots or sliding windows. In the second step, a matching of the communities obtained from consecutive time points is accomplished. Based on the result of the matching, the evolution of communities is then explained. For example, if the *rLinkTopic* model is employed to study community evolution based on this two-step approach, then one would run the model independently on each sliding window to extract communities. Communities obtained from consecutive sliding windows are then matched to find out their evolution. Almost all of existing studies for the analysis of evolving communities follow this strategy [7, 94, 109]. Even that, this typical approach has two main shortcomings. First, the matching procedure always requires extensive computations and the selection of a matching solution is a subjective task. This issue becomes even harder for our setting, because we aim at studying the evolution of multiple features describing a community. The second weakness affecting the result more is that this approach fails to capture the gradual evolution of communities. It is because communities are independently extracted from different sliding windows and none of the obtained information is employed while deriving new communities.



That is, for example, the community structures obtained from the previous sliding window are not used in the extraction of communities at the current sliding window.

Obviously, community memberships of a user at the current sliding window should be derived based on the memberships of that user in communities discovered from the previous sliding window. This happens similarly to the evolution of the topic proportion of a community, and the evolution of terms in a topic. To handle these observations, the *ErLinkTopic* model is developed to discover communities over sliding windows in the way that information about the community structures obtained from a sliding window is used for deriving communities at the next window. That is, the community membership of users, the topic proportion of communities, and the distribution of terms in topics obtained from sliding window  $\mathcal{W}_{t-1}$  are used as prior knowledge provided to compute the corresponding distributions at sliding window  $\mathcal{W}_t$ . This is basically done by extending the *rLinkTopic* model. The key idea in the *rLinkTopic* model is that we employ the conjugacy between the *Dirichlet* distribution and the *Multinomial* distribution to model the features describing a community. Such features include (1) the distribution  $\phi_c$  of users, (2) the topic proportion  $\pi_c$ , (3) the distribution  $\varphi_z$  of terms in a topic associated with  $c$ , and (4) the geographic areas where  $c$  is observed, which is characterized by the likelihood of  $c$  in regions, denoted  $\theta_{r,c}, r \in R$ . As a result, the posterior distribution of each of these variables is also a *Dirichlet* distribution as presented in Section 4.4.4 (Eq. 4.65, Eq. 4.69, Eq. 4.71, Eq. 4.67). Therefore, it is straightforward to extend the *rLinkTopic* model so that it can be used to discover communities and, at the same time, to capture their gradual evolution. More precisely, the scenario of extracting and capturing the evolution of communities over two sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  is as follows. First, applying the *rLinkTopic* model to the occurrences of users in the snapshots of  $\mathcal{W}_{t-1}$  to extract communities from that sliding window. Each identified community  $c$  is characterized by the posterior distributions of the (1) users in  $c$ , denoted  $\phi_{t-1;c}$ , (2) topic proportion of  $c$ , denoted  $\pi_{t-1;c}$ , (3) terms in topics associated with  $c$ , denoted  $\varphi_{t-1;z}, z \in Z$ , and (4) locations of  $c$ , denoted  $\theta_{t;r,c}, r \in R_{\mathcal{W}_{t-1}}$ , derived at sliding window  $\mathcal{W}_{t-1}$ . The estimated value of each of these variables except  $\theta_t$  is then used as an evidence to compute the corresponding variables at the next step for extracting communities from sliding window  $\mathcal{W}_t$ . By this, all features describing a community are obtained over time and their changes are gradually captured. Figure 5.2 shows the graphical model representing the generative process of the *ErLinkTopic* model as described. It is a sequence of *rLinkTopic* models linked to each other. Each block describes the extraction of communities in a sliding window.

Note that in our framework geographic regions in a snapshot are identified from the occurrences of users in that snapshot. Based on this, the structure of regions might change over snapshots. Therefore, we do not model the evolution of the distribution of communities in a region over time. This means we assume  $\theta_t$  to be independent of  $\theta_{t-1}$ . Nevertheless, if regions are fixed over snapshots (e.g., by applying a grid-based approach to modeling re-

gions), then it is straightforward to capture the evolution of the distribution of communities in a region as well.

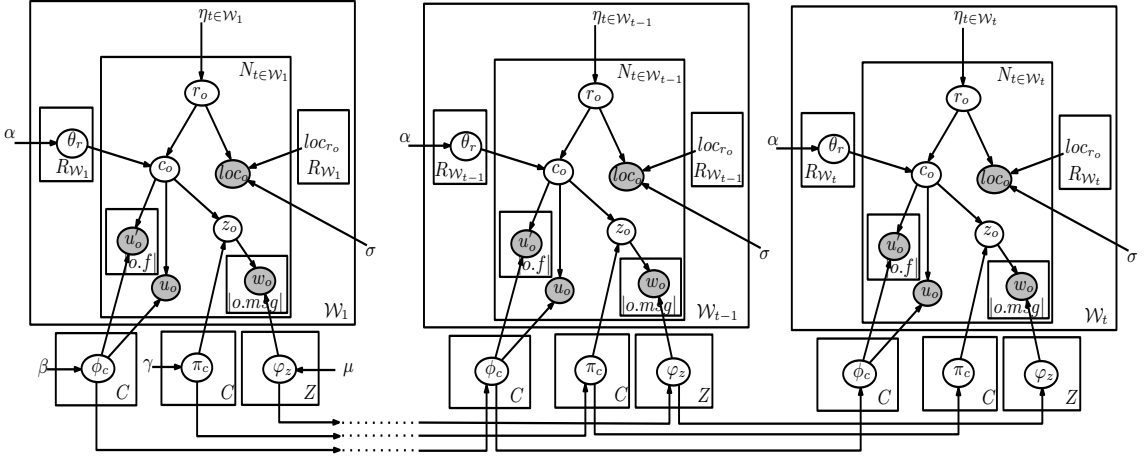


Figure 5.2: Graphical model presenting the generative process of the *ErLinkTopic* model. It consists of a sequence of *rLinkTopic* models linked to each other. Each block represents the extraction of communities in a sliding window.

### 5.3.2 Posterior Estimation for ErLinkTopic Model

There are assumptions implicitly employed in the *ErLinkTopic* model shown in Figure 5.2. First, the distributions  $\phi_t$  of users in communities, the topic proportions  $\pi_t$  of communities, and the distributions  $\varphi_t$  of terms in topics at the current sliding window  $\mathcal{W}_t$  are conditionally independent of the occurrences of users at the previous sliding window  $\mathcal{W}_{t-1}$ , given the corresponding distributions obtained from  $\mathcal{W}_{t-1}$ , i.e.,  $\phi_{t-1}$ ,  $\pi_{t-1}$ , and  $\varphi_{t-1}$ . Second, the occurrences of users in the snapshots of sliding window  $\mathcal{W}_t$  are conditionally independent of all other information, given  $\phi_t$ ,  $\pi_t$ ,  $\varphi_t$ , and  $\theta_t$ . Note that  $\theta_t$  in this model denotes the set of distributions of communities in the regions created from the snapshots of sliding window  $\mathcal{W}_t$ . In terms of notations, this is a little bit different compared to the *rLinkTopic* model where  $\theta_t$  is used to denote the set of distributions of communities in the regions created from snapshot  $t$ . Also, as mentioned above, there is no connection regarding the geographic regions obtained from different snapshots in our model. Having such assumptions employed, the joint distribution of the *ErLinkTopic* model is represented as follows.

$$\begin{aligned}
 P(SN, \phi, \theta, \pi, \varphi, \mathbf{r}, \mathbf{c}, \mathbf{z} | \beta, \gamma, \mu, \alpha, \eta, \sigma) &= P(\mathcal{W}_1, \phi_1, \theta_1, \pi_1, \varphi_1, \mathbf{r}_1, \mathbf{c}_1, \mathbf{z}_1 | \beta, \gamma, \mu, \alpha, \eta, \sigma) \\
 &\times \prod_{t=2}^T P(\mathcal{W}_t, \phi_t, \theta_t, \pi_t, \varphi_t, \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t | \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma)
 \end{aligned} \tag{5.1}$$

Based on Eq. 5.1, the posterior distribution of the model is derived incrementally over sliding windows. Particularly, it is first computed based on the occurrences of users in the snapshots of the first sliding window  $\mathcal{W}_1$  and the hyperparameters of the model. This is actually the posterior estimation of the *rLinkTopic* model applied to the snapshots of  $\mathcal{W}_1$ . For each of the next sliding windows, information about the community structures derived

from the previous step, together with the user occurrences in the snapshots of that sliding window are used to extract communities.

The posterior distribution of the model at sliding window  $\mathcal{W}_t$  ( $t > 1$ ) is computed based on the user occurrences in the snapshots of  $\mathcal{W}_t$  and the posterior distribution derived from  $\mathcal{W}_{t-1}$ , which is presented as follows.

$$P(\phi_t, \theta_t, \pi_t, \varphi_t, \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t \mid \mathcal{W}_t, \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma) = \frac{P(\mathcal{W}_t, \phi_t, \theta_t, \pi_t, \varphi_t, \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t \mid \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma)}{P(\mathcal{W}_t \mid \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma)} \quad (5.2)$$

Using the same strategy applied in the *rLinkTopic* model, we estimate the above posterior distribution by sampling from the joint distribution of the model. More specifically, it is estimated from the joint distribution applied to the user occurrences in the snapshots of sliding window  $\mathcal{W}_t$ , given the information derived from the previous sliding window  $\mathcal{W}_{t-1}$  and the hyperparameters, which is computed as follows.

$$\begin{aligned} P(\mathcal{W}_t, \phi_t, \theta_t, \pi_t, \varphi_t, \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t \mid \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma) &= \prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} P(r_o \mid \eta_t) P(loc_o \mid loc_{r_o}, \sigma) \times \quad (I) \\ &\quad \prod_{sn_t \in \mathcal{W}_t} P(\theta_t \mid \alpha) \prod_{o \in sn_t} P(c_o \mid \theta_t, r_o) \times \quad (II) \\ &\quad P(\phi_t \mid \phi_{t-1}) \prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} P(u_o \mid \phi_t, c_o) \prod_{u' \in o.f} P(u' \mid \phi_t, c_o) \times \quad (III) \\ &\quad P(\pi_t \mid \pi_{t-1}) \prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} P(z_o \mid \pi_t, c_o) \times \quad (IV) \\ &\quad P(\varphi_t \mid \varphi_{t-1}) \prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} \prod_{w \in o.msg} P(w \mid \varphi_t, z_o) \quad (V) \end{aligned} \quad (5.3)$$

Table 5.2: Notations used to present the count variables in the *ErLinkTopic* model. Each variable is computed based on the user occurrences in the snapshots of one sliding window.

Notation	Description
$n_c^{(r)}$	number of occurrences in region $r$ that are assigned to community $c$
$n_u^{(c)}$	number of occurrences of user $u$ that are assigned to community $c$
$n_{f,u}^{(c)}$	number of times user $u$ is contextually linked by other users in community $c$
$n_w^{(z)}$	number of occurrences of term $w$ that are assigned to topic $z$
$n_z^{(c)}$	number of messages in community $c$ that are assigned to topic $z$

Adopting the notations defined in the *rLinkTopic* model, which are repeated in Table 5.2, the above joint distribution is simplified by applying the same steps as presented in the *rLinkTopic* model, i.e., Eq. 4.39, Eq. 4.40, Eq. 4.41, and 4.42, so that the posterior distribution in Eq. 5.2 is then estimated as follows<sup>1</sup>.

<sup>1</sup>Note that the count variables introduced in Table 5.2 are applied to the user occurrences in the snapshots of exactly one sliding window, e.g.,  $\mathcal{W}_t$ .

$$\begin{aligned}
P(\phi_t, \theta_t, \pi_t, \varphi_t, \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t | \mathcal{W}_t; \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma) &\propto \prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma) \times \\
&\prod_{r \in R_{\mathcal{W}_t}} \prod_{c \in C} \theta_{r,c}^{n_{r,c}^{(r)} + \alpha_c - 1} \times \prod_{c \in C} \prod_{u \in U} \phi_{t;c,u}^{n_u^{(c)} + n_{f,u}^{(c)} + \phi_{t-1;c,u} - 1} \times \\
&\prod_{c \in C} \prod_{z \in Z} \pi_{t;c,z}^{n_z^{(c)} + \pi_{t-1;c,z} - 1} \times \prod_{z \in Z} \prod_{w \in V} \varphi_{t;z,w}^{n_w^{(z)} + \varphi_{t-1;z,w} - 1}
\end{aligned} \tag{5.4}$$

By integrating out the multinomial parameters  $\phi_t$ ,  $\pi_t$ ,  $\varphi_t$ , and  $\theta_t$ , the posterior distribution of the region assignments  $\mathbf{r}_t$ , community assignments  $\mathbf{c}_t$ , and topic assignments  $\mathbf{z}_t$  of the user occurrences in the snapshots of sliding window  $\mathcal{W}_t$  becomes

$$\begin{aligned}
P(\mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t | \mathcal{W}_t; \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma) &\propto \underbrace{\prod_{sn_t \in \mathcal{W}_t} \prod_{o \in sn_t} P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma)}_{(T_1)} \times \\
&\underbrace{\prod_{r \in R_{\mathcal{W}_t}} \frac{\prod_{c \in C} \Gamma(n_c^{(r)} + \alpha_c)}{\Gamma(\sum_{c \in C} n_c^{(r)} + \alpha_c)}}_{(T_2)} \times \underbrace{\prod_{c \in C} \frac{\prod_{u \in U} \Gamma(n_u^{(c)} + n_{f,u}^{(c)} + \phi_{t-1;c,u})}{\Gamma(\sum_{u \in U} n_u^{(c)} + n_{f,u}^{(c)} + \phi_{t-1;c,u})}}_{(T_3)} \times \\
&\underbrace{\prod_{c \in C} \frac{\prod_{z \in Z} \Gamma(n_z^{(c)} + \pi_{t-1;c,z})}{\Gamma(\sum_{z \in Z} n_z^{(c)} + \pi_{t-1;c,z})}}_{(T_4)} \times \underbrace{\prod_{z \in Z} \frac{\prod_{w \in V} \Gamma(n_w^{(z)} + \varphi_{t-1;z,w})}{\Gamma(\sum_{w \in V} n_w^{(z)} + \varphi_{t-1;z,w})}}_{(T_5)}.
\end{aligned} \tag{5.5}$$

From Eq. 5.5, the joint distribution of the region assignment  $r_o$ , community assignment  $c_o$ , and topic assignment  $z_o$  of occurrence  $o$  is obtained as follows.

$$\begin{aligned}
P(r_o, c_o, z_o | \mathbf{r}_{t;-o}, \mathbf{c}_{t;-o}, \mathbf{z}_{t;-o}, \mathcal{W}_t; \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, \alpha, \eta, \sigma) &= P(r_o | \eta_t) P(loc_o | loc_{r_o}, \sigma) \times \\
&\frac{n_{-o,c_o}^{(r_o)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o,c}^{(r_o)} + \alpha_c} \times \frac{n_{-o,u_o}^{(c_o)} + n_{f,u_o}^{(c_o)} + \phi_{t-1;c_o,u_o}}{\sum_{u \in U} n_{-o,u}^{(c_o)} + n_{f,u}^{(c_o)} + \phi_{t-1;c_o,u}} \times \\
&\frac{n_{-o,z_o}^{(c_o)} + \pi_{t-1;c_o,z_o}}{\sum_{z \in Z} n_{-o,z}^{(c_o)} + \pi_{t-1;c_o,z}} \times \frac{\prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i-1 + n_{-w,w}^{(z_o)} + \varphi_{t-1;z_o,w})}{\prod_{i=1}^{n.msg} (i-1 + \sum_{w \in V} n_{-w,w}^{(z_o)} + \varphi_{t-1;z_o,w})}
\end{aligned} \tag{5.6}$$

Finally, the sampling rule for each of the assignment variables  $r_o$ ,  $c_o$ , and  $z_o$  is obtained similarly to the corresponding sampling rule in the *rLinkTopic* model, which is presented as follows.

### 1. Sampling rule for region assignment:

$$\begin{aligned}
P(r_o = r | c_o, z_o, \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, \mathcal{W}_t; \cdot) &= P(r | \eta_t) P(loc_o | loc_r, \sigma) \times \frac{n_{-o,c_o}^{(r)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o,c}^{(r)} + \alpha_c} \\
&\propto \exp\left(-\frac{|loc_o, loc_r|}{\sigma^2}\right) \times \frac{n_{-o,c_o}^{(r)} + \alpha_{c_o}}{\sum_{c \in C} n_{-o,c}^{(r)} + \alpha_c}
\end{aligned} \tag{5.7}$$

## 2. Sampling rule for community assignment:

$$P(c_o = c | r_o, z_o, \mathbf{c}_{-o}, \mathbf{r}_{-o}, \mathbf{z}_{-o}, \mathcal{W}_t; \cdot) \propto \frac{n_{-o, u_o}^{(c)} + n_{-o, f.u_o}^{(c)} + \phi_{t-1; c, u_o}}{\sum_{u \in U} n_{-o, u}^{(c)} + n_{-o, f.u}^{(c)} + \phi_{t-1; c, u}} \times \frac{n_{-o, c}^{(r_o)} + \alpha_c}{\sum_{c' \in C} n_{-o, c'}^{(r_o)} + \alpha_{c'}} \times \frac{n_{-o, z_o}^{(c)} + \pi_{t-1; c, z_o}}{\sum_{z \in Z} n_{-o, z}^{(c)} + \pi_{t-1; c, z}} \quad (5.8)$$

## 3. Sampling rule for topic assignment:

$$P(z_o = z | r_o, c_o, \mathbf{r}_{-o}, \mathbf{c}_{-o}, \mathbf{z}_{-o}, \mathcal{W}_t; \cdot) \propto \frac{\prod_{w \in o.msg} \prod_{i=1}^{n_w.msg} (i - 1 + n_{-w, w}^{(z)} + \varphi_{t-1; z_o, w})}{\prod_{i=1}^{n.msg} (i - 1 + \sum_{w \in V} n_{-w, w}^{(z)} + \varphi_{t-1; z_o, w})} \times \frac{n_{-o, z}^{(c_o)} + \pi_{t-1; c_o, z}}{\sum_{z' \in Z} n_{-o, z'}^{(c_o)} + \pi_{t-1; c_o, z'}} \quad (5.9)$$

**Updating multinomial parameters from assignment samples.** Given a sample  $\langle \mathbf{r}_t, \mathbf{c}_t, \mathbf{z}_t \rangle$  of the region, community, and topic assignments of all user occurrences in the snapshots of sliding window  $\mathcal{W}_t$ , the posterior distributions of (1) users in a community  $(\phi_{t;c})$ , (2) communities in a region  $(\theta_{t;r})$ , (3) topics of a community  $(\pi_{t;c})$ , and (4) terms in a topic  $(\varphi_{t;z})$  are derived using the same method applied in the *rLinkTopic* model (Eq. 4.65, Eq. 4.67, Eq. 4.69, Eq. 4.71). Finally, the updating rules for these multinomial parameters at sliding window  $\mathcal{W}_t$  are as follows.

### 1. Distribution of users in a community:

$$\phi_{t;c,u} = \frac{n_u^{(c)} + n_{f.u}^{(c)} + \phi_{t-1;c,u}}{\sum_{u' \in U} n_{u'}^{(c)} + n_{f.u'}^{(c)} + \phi_{t-1;c,u'}} \quad , c \in C, u \in U \quad (5.10)$$

### 2. Distribution of communities in a region:<sup>2</sup>

$$\theta_{t;r,c} = \frac{n_c^{(r)} + \alpha_c}{\sum_{c' \in C} n_{c'}^{(r)} + \alpha_{c'}} \quad , r \in R_{\mathcal{W}_t}, c \in C \quad (5.11)$$

### 3. Topic proportion of a community:

$$\pi_{t;c,z} = \frac{n_z^{(c)} + \pi_{t;c,z}}{\sum_{z' \in Z} n_{z'}^{(c)} + \pi_{t;c,z'}} \quad , c \in C, z \in Z \quad (5.12)$$

### 4. Distribution of terms in a topic:

$$\varphi_{t;z,w} = \frac{n_w^{(z)} + \varphi_{t;z,w}}{\sum_{w' \in V} n_{w'}^{(z)} + \varphi_{t;z,w'}} \quad , z \in Z, w \in V \quad (5.13)$$

---

<sup>2</sup>The distribution of communities in a region at sliding window  $\mathcal{W}_t$  is independent of the information obtained from the previous window, as explained.

**Gibbs sampling algorithm.** The Gibbs sampling algorithm for the *ErLinkTopic* model is shown in Algorithm 6. Input of the algorithm is a sequence of sliding windows  $SN = \{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_T\}$  and the hyperparameters. Hidden variables are first estimated for the first sliding window  $\mathcal{W}_1$  using the *rLinkTopic* model with the given hyperparameters. From the second sliding window, the *rLinkTopic* model is employed in the way that the values of  $\phi_{t-1}$ ,  $\pi_{t-1}$  and  $\varphi_{t-1}$  obtained from the previous sliding window are used as the prior hyperparameters of model. Based on the sequence of each of these variables computed over sliding windows, the evolution of communities regarding the community membership of users, the topic proportion of communities, and the distribution of terms in topics is then analyzed. The main task of the *ErLinkTopic* algorithm is to extract communities from the user occurrences in the snapshots of each sliding window, which is done by the *rLinkTopic* algorithm (lines 2 and 5). Therefore, both algorithms have the same time complexity (see Section 4.4.5).

---

**Algorithm 6:** Gibbs sampling algorithm for the *ErLinkTopic* probabilistic model.  
**ErLinkTopic**( $SN = \{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_T\}, |C|, |Z|, \alpha, \beta, \gamma, \mu, minRad, \sigma$ )

---

**Input:**  
 $SN = \{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_T\}$ : sequence of network sliding windows  
 $|C|$ : number of communities to be extracted  
 $|Z|$ : number of topics associated with communities  
 $minRad$ : a threshold to determine representative locations of regions  
 $\sigma$ : prior standard deviation for Gaussian  
 $\alpha, \beta, \gamma, \mu$ : Dirichlet hyperparameters

**Output:**  
set of evolving communities characterized by:  
(1)  $\theta = \{\theta_1, \theta_2, \dots, \theta_T\}$ : sequence of distributions of communities in regions  
(2)  $\phi = \{\phi_1, \phi_2, \dots, \phi_T\}$ : sequence of distributions of users in communities  
(3)  $\pi = \{\pi_1, \pi_2, \dots, \pi_T\}$ : sequence of topic proportions of communities  
(4)  $\varphi = \{\varphi_1, \varphi_2, \dots, \varphi_T\}$ : sequence of distributions of terms in topics

```

1 /* first sliding window */
2  $\phi_1, \pi_1, \varphi_1, \theta_1 \leftarrow \mathbf{rLinkTopic}(\mathcal{W}_1, |C|, |Z|, \alpha, \beta, \gamma, \mu, minRad, \sigma)$ ;
3 /* from second sliding window */
4 foreach  $t = 2..T$  do
5    $\phi_t, \pi_t, \varphi_t, \theta_t \leftarrow \mathbf{rLinkTopic}(\mathcal{W}_t, |C|, |Z|, \alpha, \phi_{t-1}, \pi_{t-1}, \varphi_{t-1}, minRad, \sigma)$ ;
6   /* detect changes in community memberships of users */
7   detectChangesFrom( $\phi_{t-1}, \phi_t$ );
8   /* detect changes in topic proportions of communities */
9   detectChangesFrom( $\pi_{t-1}, \pi_t$ );
10  /* detect changes in topics of communities */
11  detectChangesFrom( $\varphi_{t-1}, \varphi_t$ );

```

---

## 5.4 Evolution of Communities

This section formalizes the evolution of the features describing communities. Particularly, we introduce methods to study the dynamics and to detect specific evolving phenomena of the community members, topic proportion of communities, and terms in topics over time.

Based on the *ErLinkTopic* algorithm, a community  $c$  at sliding window  $\mathcal{W}_t$  is characterized by the features describing  $c$  at that time. Such features include (1) users in  $c$ , represented by  $\phi_{t,c}$ ; (2) topic proportion of  $c$ , represented by  $\pi_{t,c}$ ; (3) terms in the topics associated with  $c$ , represented by  $\varphi_{t,z}, z \in Z$ ; and (4) where  $c$  is observed, which is characterized by the likelihood of  $c$  in regions, denoted  $\theta_{t,r,c}, r \in \mathcal{R}_{\mathcal{W}_t}$ .

Note that in this study we assume the dynamics of regions over time as explained above. As a consequence,  $\theta_t$  is independent of  $\theta_{t-1}$ , and, because of this, we do not capture the evolution of the likelihood of communities in a region over time. Thus, we are interested only in the gradual changes in each of the three features: users in community  $c$ , topics of community  $c$ , and terms in topics associated with community  $c$  over sliding windows to study the evolution of community  $c$ . These features are encoded in the variables  $\phi_{t,c}$ ,  $\pi_{t,c}$ , and  $\varphi_{t,z}, z \in Z$ , respectively, and again described as follows.

1.  $\phi_{t,c} = \{P(u|c, t)\}, u \in U$  and  $\sum_{u \in U} P(u|c, t) = 1$ , where  $P(u|c, t)$  is the likelihood of user  $u$  in community  $c$  at sliding window  $\mathcal{W}_t$ .
2.  $\pi_{t,c} = \{P(z|c, t)\}, z \in Z$  and  $\sum_{z \in Z} P(z|c, t) = 1$ , where  $P(z|c, t)$  is the likelihood of topic  $z$  in community  $c$  at sliding window  $\mathcal{W}_t$ .
3.  $\varphi_{t,z} = \{P(w|z, t)\}, w \in V$  and  $\sum_{w \in V} P(w|z, t) = 1$ , where  $P(w|z, t)$  is the likelihood of term  $w$  in topic  $z$ .

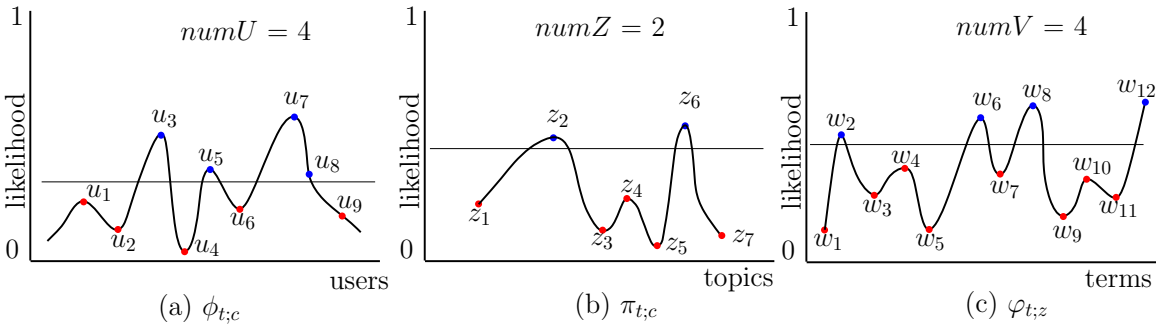


Figure 5.3: Illustration of the (a) likelihood of users in a community  $c$ , (b) topic proportion of a community  $c$ , and (c) likelihood of terms in a topic  $z$  at a sliding window  $\mathcal{W}_t$ . Note that for sake of illustration the likelihood of users in a community, topic proportion of a community, and likelihood of terms in a topic are represented by continuous lines.

Although  $\phi_{t,c}$  consists of the likelihood of all users, we are particularly interested in only a number of users that have the highest likelihood in  $c$ , which are referred to as the *members*

of community  $c$ . It is similar to the topic proportion  $\pi_{t;c}$  and topic  $\varphi_{t;z}$ . That is, only a number of topics that have the highest likelihood in  $c$  are considered the topics of  $c$ ; and only a number of terms that have the highest likelihood in  $z$  are considered terms occurring in  $z$ . Thus, to formalize the evolution of these features, we first assume three predefined cardinality thresholds, namely (1) number of users in a community, denoted  $numU$ ; (2) number of topics of a community, denoted  $numZ$ ; and (3) number of terms in a topic, denoted  $numV$ . These mean we consider (1) a community  $c$  to consist of only  $numU$  users that have the highest likelihood in  $c$ ; (2) the topic proportion of  $c$  to consist of only  $numZ$  topics that have the highest likelihood in  $c$ ; and (3) a topic  $z$  to consist of only  $numV$  terms that have the highest likelihood in  $z$ . Figure 5.3 illustrates the likelihood of users in a community, topic proportion of a community, and likelihood of terms in a topic. In the figure, we also show the selection of community members, community topics, and terms in a topic using  $numU$ ,  $numZ$ , and  $numV$ , respectively, as described.

To study the evolution of these features, the following notations are introduced, given the parameters  $numU$ ,  $numZ$ , and  $numV$ .

1.  $U(c, t, numU)$ : set of  $numU$  users that have the highest likelihood in community  $c$  at sliding window  $\mathcal{W}_t$ .
2.  $Z(c, t, numZ)$ : set of  $numZ$  topics that have the highest likelihood in community  $c$  at  $\mathcal{W}_t$ .
3.  $V(z, t, numV)$ : set of  $numV$  terms that have the highest likelihood in topic  $z$  at  $\mathcal{W}_t$ .

For example, in Figure 5.3 we have:  $U(c, t, numU) = \{u_3, u_5, u_7, u_8\}$ ,  $Z(c, t, numZ) = \{z_2, z_6\}$ , and  $V(z, t, numV) = \{w_2, w_6, w_8, w_{12}\}$ . It is noted that one can select community members, community topics, and terms in topics using a likelihood threshold. That is, instead of defining  $numU$ ,  $numZ$ , and  $numV$  as described, one can rely on the distributions  $\phi_{t;c}$ ,  $\pi_{t;c}$ , and  $\varphi_{t;z}$  to specify the likelihood thresholds  $min_\phi$ ,  $min_\pi$ , and  $min_\varphi$  for deriving community members, community topics, and terms in topics, respectively. By this, the following notations are formalized.

1.  $U(c, t, min_\phi)$ : set of users  $u \in U$  such that  $P(u|c, t) > min_\phi$ .
2.  $Z(c, t, min_\pi)$ : set of topics  $z \in Z$  such that  $P(z|c, t) > min_\pi$ .
3.  $V(z, t, min_\varphi)$ : set of terms  $w \in V$  such that  $P(w|z, t) > min_\varphi$ .

Based on these notations, the evolution of a community with respect to the community members, community topics, and terms in topics is formalized in the following sections.



### 5.4.1 Changes in Community Members

This section presents methods that we apply to study the evolution of community members. We first introduce a measure to assess the frequency of changes in the set of users that have the highest likelihood in a community. We then formalize four main evolving phenomena one might observe from a community over time.

**Dynamics of users.** In our model, the evolution of users in a community is indicated by the frequency of changes in the set  $U(c, t, numU)$  of users over sliding windows. Generally, the difference between the two sets  $U(c, t - 1, numU)$  and  $U(c, t, numU)$  is the result of two possibilities. First, some users having less likelihood in  $c$  at sliding window  $\mathcal{W}_{t-1}$  are becoming more active, e.g, posting more messages or interacting with more users in  $c$ , at sliding window  $\mathcal{W}_t$ . Second, it is also the case that some users change from an active state in  $c$  at  $t - 1$  to an inactive one at  $t$ . Both situations result in the update of the likelihood of users in  $c$ , thus making the difference between  $U(c, t - 1, numU)$  and  $U(c, t, numU)$ .

To capture the dynamics of users in community  $c$  over two consecutive sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$ , we introduce a *user dynamic* measure  $\partial_\phi(c, t - 1, t, numU)$ , computed as follows.

$$\partial_\phi(c, t - 1, t, numU) = \frac{numU - |U(c, t - 1, numU) \cap U(c, t, numU)|}{numU} \in [0, 1] \quad (5.14)$$

A small value of  $\partial_\phi(c, t - 1, t, numU)$  indicates that not many users in community  $c$  change their behavior during the time interval of the two sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$ . On the other hand, if many users in  $c$  change their activities, e.g., moving to other areas, having new “link users” or posting messages about other topics, then a large value of  $\partial_\phi(c, t - 1, t, numU)$  is obtained. This is because such changes of users lead to the update of their likelihood in  $c$  resulting in the difference between  $U(c, t - 1, numU)$  and  $U(c, t, numU)$ . There are two extreme situations corresponding to the values 0 and 1 of  $\partial_\phi(c, t - 1, t, numU)$ . These are respectively called *stable* and *separating* phenomena of users in a community.

**Stability of users.** A community  $c$  is *stable* between two sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  if the users in  $c$  at sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  are the same, i.e.,  $U(c, t - 1, numU) = U(c, t, numU)$ . The defined measure  $\partial_\phi(c, t - 1, t, numU)$  therefore has the value 0 for this case. One should note that we identify the stability based on the identities of users in the two sets  $U(c, t - 1, numU)$  and  $U(c, t, numU)$ . This means the likelihood of users in  $c$  might change between  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  but all users in  $U(c, t - 1, numU)$  still appear in  $U(c, t, numU)$ .

**Separation of users.** A community  $c$  is *separated* if all users in  $c$  are changed between  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$ , i.e.,  $U(c, t - 1, numU) \cap U(c, t, numU) = \emptyset$ . In this case  $\partial_\phi(c, t - 1, t, numU) = 1$ . It is noted that in our approach, a community is characterized by not only the users but also the topics and the geographic locations. This means, even though the community is separated regarding its members at  $t$ , i.e., users in  $U(c, t, numU)$  are all new compared

to users in  $U(c, t - 1, numU)$ , community  $c$  is still *alive* in terms of the topics associated with it. One can therefore assume the user separation is a *renew* event of the community members.

In addition to the stability and separation, there are other evolving events can be observed based on changes in the likelihood of users in a community. Two of which are the increase and the decrease of the community members, described as follows.

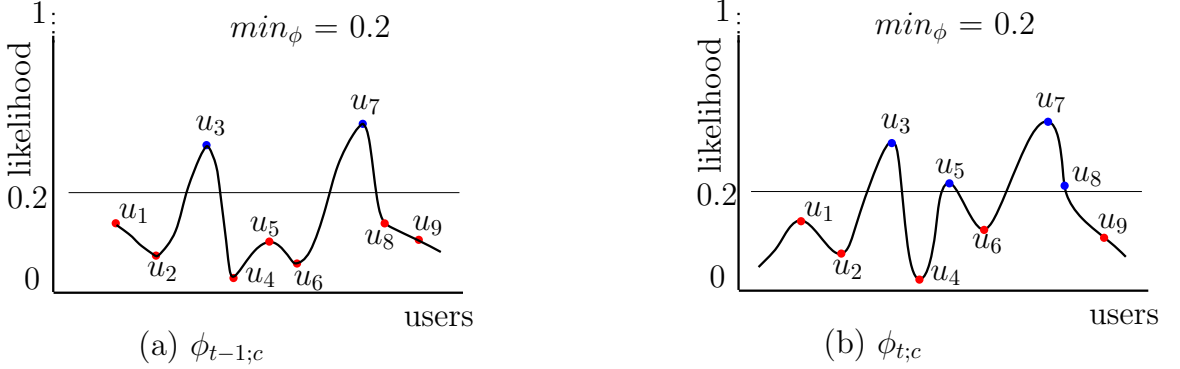


Figure 5.4: Changes in the memberships of users in a community over two consecutive sliding windows. More users having a high membership at  $t$  indicate the growth of the community.

**Growth of communities.** The growth of a community between two sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  means that there are more users at  $\mathcal{W}_t$  that have a high membership in community  $c$  compared to such users at  $\mathcal{W}_{t-1}$ . Such a growth event is illustrated in Figure 5.4, where the number of community members increases from two users ( $u_3$  and  $u_7$ ) at  $t - 1$  to four users ( $u_3, u_5, u_7$  and  $u_8$ ) at  $t$ , given a likelihood threshold  $min_\phi = 0.2$ . To identify this event of a community  $c$ , one can rely on the histograms of  $\phi_{t-1;c}$  and  $\phi_{t;c}$  to determine a likelihood threshold  $min_\phi$  for selecting the members of  $c$  at  $t - 1$  and  $t$ , denoted  $U(c, t - 1, min_\phi)$  and  $U(c, t, min_\phi)$ , respectively. That is, instead of selecting users based on the predefined cardinality  $numU$  of the community, a likelihood threshold  $min_\phi$  is applied to identify community members. By comparing the users in  $U(c, t - 1, min_\phi)$  and  $U(c, t, min_\phi)$  one can then extract the growth event. Particularly, a community  $c$  grows from  $t - 1$  to  $t$  if  $U(c, t - 1, min_\phi) \subset U(c, t, min_\phi)$ .

**Shrinkage of communities.** Contrary to the growth event, a community  $c$  might shrink because of the leaving of some members. This phenomenon is indicated by changes in the memberships of users in  $c$  in the inverse direction of the growth event. That is, more users having a high membership in community  $c$  at sliding window  $\mathcal{W}_{t-1}$  than at sliding window  $\mathcal{W}_t$ . Figure 5.5 illustrates the shrinkage of a community. This event can be detected using the same method described above for the identification of the growth of a community. Particularly, given a likelihood threshold  $min_\phi$ , a community  $c$  shrinks between two sliding windows  $\mathcal{W}_{t-1}$  and  $\mathcal{W}_t$  if  $U(c, t, min_\phi) \subset U(c, t - 1, min_\phi)$ .

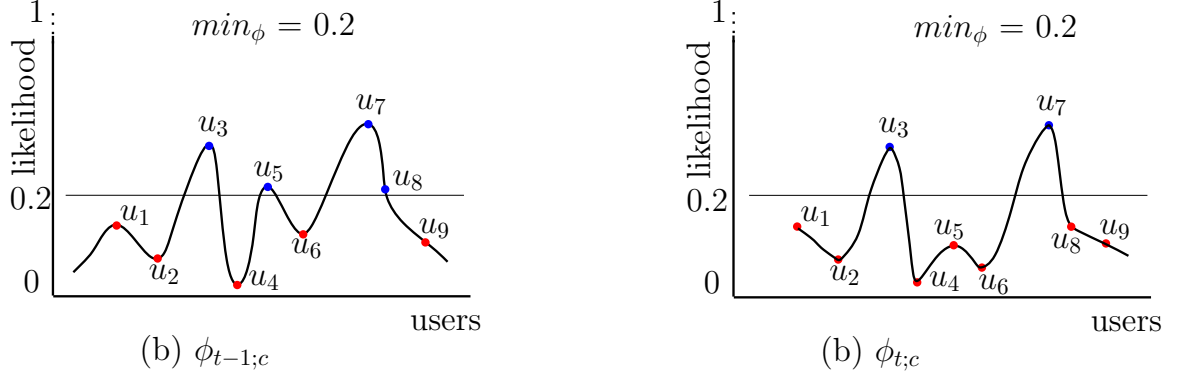


Figure 5.5: Changes in the community memberships of users in a community over two consecutive sliding windows. Fewer users having a high membership at sliding window  $t$  indicate the shrinkage of the community.

#### 5.4.2 Changes in Topics of Communities

The evolution of a community is also indicated by changes in the prominence of the topics discussed by community members. A community is associated with a number of topics where the prominence of each topic is characterized in the topic proportion of the community. Over time, the likelihood of each topic in the community can change. For example, a topic about weather is discussed a lot by community members during a time span before and after a tropical storm hits a country. This topic, however, might become less prominent on other days. Such changes in the prominence of the topics associated with a community  $c$  are implied by the difference between the two sets  $Z(c, t - 1, numZ)$  and  $Z(c, t, numZ)$ . Applying the same method for analyzing the dynamics of community members, we first have the *topic-prominence dynamic* measure  $\partial_\pi(c, t - 1, t, numZ)$  to determine the frequency of updating the prominence of the topics associated with community  $c$ .

$$\partial_\pi(c, t - 1, t, numZ) = \frac{numZ - |Z(c, t - 1, numZ) \cap Z(c, t, numZ)|}{numZ} \in [0, 1] \quad (5.15)$$

Based on  $\partial_\pi(c, t - 1, t, numZ)$  or the two sets  $Z(c, t - 1, numZ)$  and  $Z(c, t, numZ)$ , the *stability* (i.e., the prominence of associated topics does not change), *generalization* (i.e., more topics become prominent), and *specification* (i.e., fewer topics become prominent) in terms of the prominence of topics of a community are captured. For example, a community  $c$  is stable in terms of the prominence of topics discussed by its members if the likelihood of topics in the topic proportion  $\pi_c$  does not change. This phenomenon can be observed by checking whether  $Z(c, t - 1, numZ)$  and  $Z(c, t, numZ)$  are the same.

In addition to the dynamics of the prominence of topics associated with a community, terms describing a topic itself change over time also. For example, the frequency of terms used to describe the weather topic changes over seasons. Such a frequency of changes of

terms occurring in a topic  $z$  is obtained by the *term dynamic* measure  $\partial_\varphi(z, t - 1, t, numV)$ .

$$\partial_\varphi(z, t - 1, t, numV) = \frac{numV - |V(z, t - 1, numV) \cap V(z, t, numV)|}{numV} \in [0, 1] \quad (5.16)$$

By adopting the same method applied to study the dynamics of community members and of the prominence of community topics, we can formalize the *stability*, *emergence* (i.e., specific terms describing the topic become prominent), and *vanishing* (i.e., no terms is prominent) phenomena of a topic over time.

## 5.5 Experiments

This section presents the experimental results of applying our approach to extracting and analyzing the evolution of (regional *LinkTopic*) communities in social networks. Particularly, by using *Twitter* data, we show the effectiveness and efficiency of the *ErLinkTopic* model in terms of discovering communities and, at the same time, capturing changes in the features describing communities. Our framework is implemented in Java. All experiments are run on an Intel(R) Core(TM) i7-4770 CPU @ 3.40G with 16GB RAM, running Ubuntu 64bit.

### 5.5.1 Twitter Datasets

We use two six-month interval subsets created from the **EUROPE** and **US** *Twitter* datasets presented in the previous chapter (Section 4.6.1) for conducting the experiments. The first subset is called **Sub-England** dataset and the second subset is called **Sub-US** dataset. For each dataset, we first compute the histogram of the number of occurrences of users and terms, respectively, to get an idea of the properties of these datasets. Based on this, a filtering step is applied so that users posting less than 180 messages, i.e., on average 1 message a day, and terms occurring less than 360 times, i.e., on average 2 time a day, are removed from the **Sub-US** dataset. Such numbers applied to filter users and terms in the **Sub-England** dataset are 180 and 540, respectively. Relevant statistics of the two datasets before and after filtering users and terms are summarized in Table 5.3.

Table 5.3: Statistics of *Twitter* datasets used to evaluate the *ErLinkTopic* model in extracting regional *LinkTopic* communities and analyzing their evolution. These datasets are created from the **EUROPE** and **US** datasets described in the previous chapter (Section 4.6.1, Table 4.3).

Dataset	Users/Filtered	Tweets/Filtered	Terms/Filtered	Time
<b>Sub-England</b>	1.720.956/18.264	13.114.353 /6.572.764	2.915.851/15.215	June 01 - Nov 28
<b>Sub-US</b>	980.924/14.756	6.301.435/3.654.000	2.135.098/16.260	June 01 - Nov 28

### 5.5.2 Experimental Setup

The main objective of our experiments is to extract communities and capture their evolution from which to study how the features describing a community evolve over time. Besides this, it is also necessary to verify the efficiency of the *ErLinkTopic* model regarding the computational complexity. For these purposes, we empirically organize each of our datasets in three different sliding window intervals: 1 week, 2 weeks, and 1 month. Each sliding window is further structured into daily snapshots. We then employ the same method applied in the *rLinkTopic* model presented in Section 4.6 to setup values for the input parameters. Particularly, the number of communities to be extracted and the number of *Burn-in* steps are identified based on the perplexity measure. Other parameters are empirically determined. The values assigned to the main parameters for each time interval setting of sliding windows applied to each dataset are summarized in Table 5.4.

Table 5.4: Setting values for the input parameters in the *ErLinkTopic* model applied to each *Twitter* dataset used in the experiments.

Dataset	1 week				2 week				1 month			
	$ C $	$ Z $	$\sigma$	$Rad$	$ C $	$ Z $	$\sigma$	$Rad$	$ C $	$ Z $	$\sigma$	$Rad$
<b>Sub-England</b>	70	20	0.033	0.066	40	20	0.033	0.066	30	20	0.033	0.066
<b>Sub-US</b>	40	20	0.033	0.066	30	20	0.033	0.066	25	20	0.033	0.066

### 5.5.3 Dynamic Measure Analysis

Based on the results extracted from the three different settings of sliding windows, i.e., 1-week interval, 2-week interval, and 1-month interval, we study the dynamics of communities in terms of changes in (1) the members of each community using the user dynamic measure  $\partial_\phi(c, t - 1, t, numU)$ , (2) the prominence of topics associated with each community using the topic-prominence dynamic measure  $\partial_\pi(c, t - 1, t, numZ)$ , and (2) terms occurring in each community topic using the term dynamic measure  $\partial_\varphi(z, t - 1, t, numW)$ . We visualize the community membership of users in each community and the likelihood of terms in each topic to determine appropriate values for  $numU$  and  $numW$ , respectively. By studying the community membership of users, we find two prevalent points at  $numU = 5$  and  $numU = 30$  where the likelihood of users in every community strongly decreases. However, the top 5 users in all communities change frequently at every sliding window. We therefore select  $numU = 30$  for evaluating the dynamics of users in communities. Applying the same method we determine that a good value for  $numW$  is 20. The community membership of users in selected communities and the likelihood of terms in selected topics extracted from the two datasets are shown in Figures 5.6, 5.7, 5.8. Finally, we choose  $numZ = 5$  for

measuring the dynamics of the prominence of community topics. The following findings are obtained from both two datasets.

1. Communities evolve gradually over a short time interval of sliding windows. This evolving trend applies to all three features of interests, i.e., community members, community topics, and terms describing a topic. Changes to these features happen more often when longer time intervals are employed to form a sliding window. This finding confirms that social networks and especially communities in social networks are dynamic structures.
2. Community members evolve faster than community topics, which is indicated by a larger value of  $\partial_\phi(c, t - 1, t, numU)$  compared to the value of  $\partial_\pi(c, t - 1, t, numZ)$  or  $\partial_\phi(z, t - 1, t, numW)$ . This implies that the topics discussed by a community are more stable regarding both the topic prominence and terms describing topics even though users might change topics of interest and leave a community and join other communities more often. The dynamic measures of six communities extracted from the **Sub-US** dataset and five communities extracted from the **Sub-England** dataset are presented in Table 5.5 and Table 5.6, respectively. Figure 5.9 and Figure 5.10 show the user dynamic measure of ten communities and term dynamic measure of four topics, all extracted from the **Sub-US** dataset.

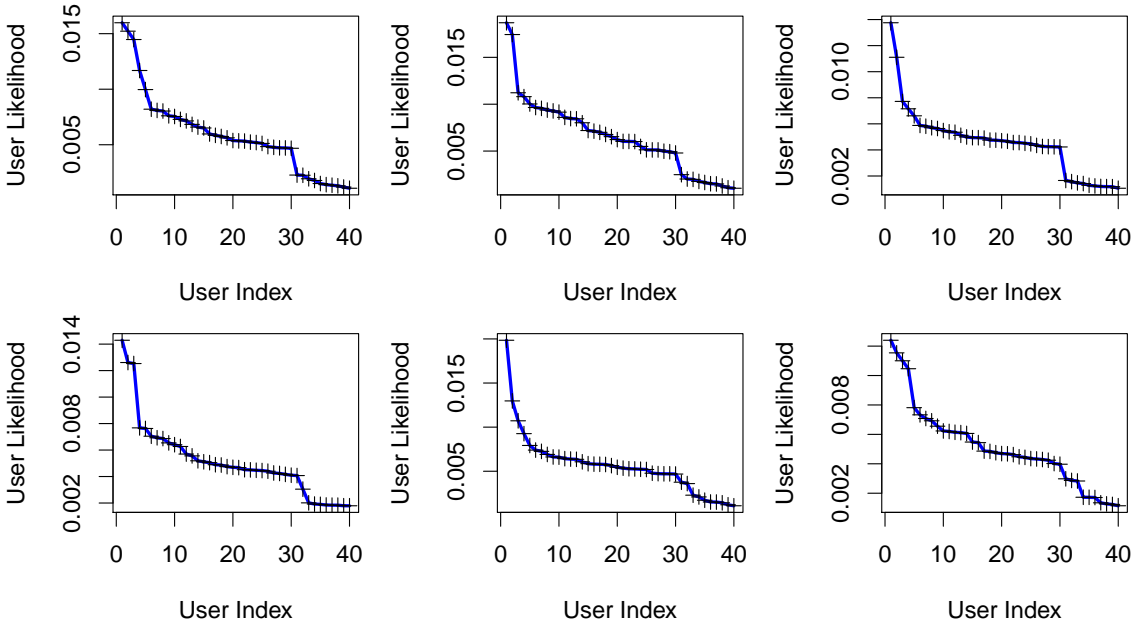


Figure 5.6: Distributions of the likelihood of users in six selected communities extracted from the **Sub-US** dataset. For each community, the likelihood of users decreases strongly at around the 5<sup>th</sup> and the 30<sup>th</sup> users.

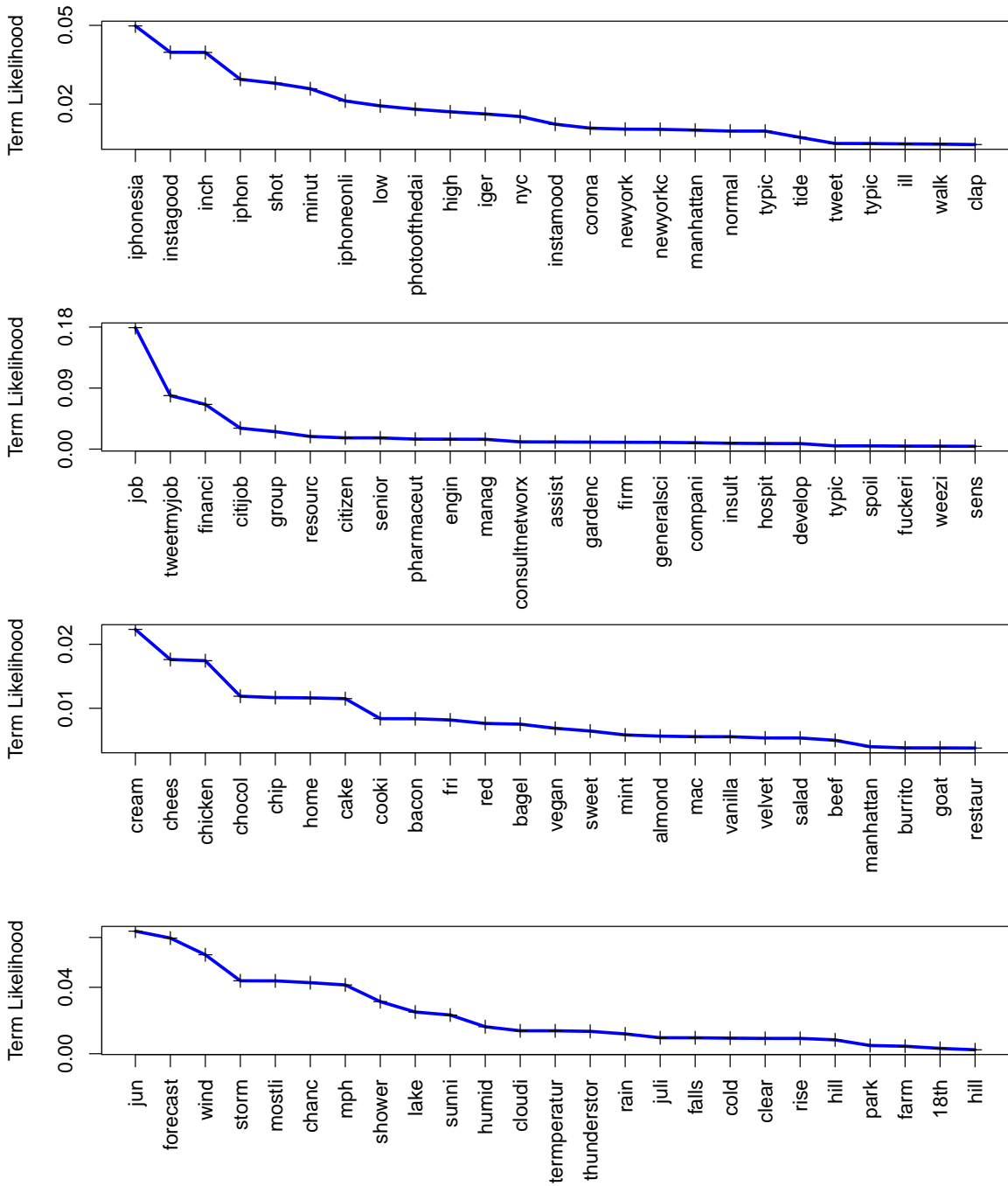


Figure 5.7: Distributions of the likelihood of terms in selected topics associated with communities extracted from the **Sub-US** dataset. For each topic, the likelihood of terms decreases strongly at around the 4<sup>th</sup> and the 20<sup>th</sup> terms.

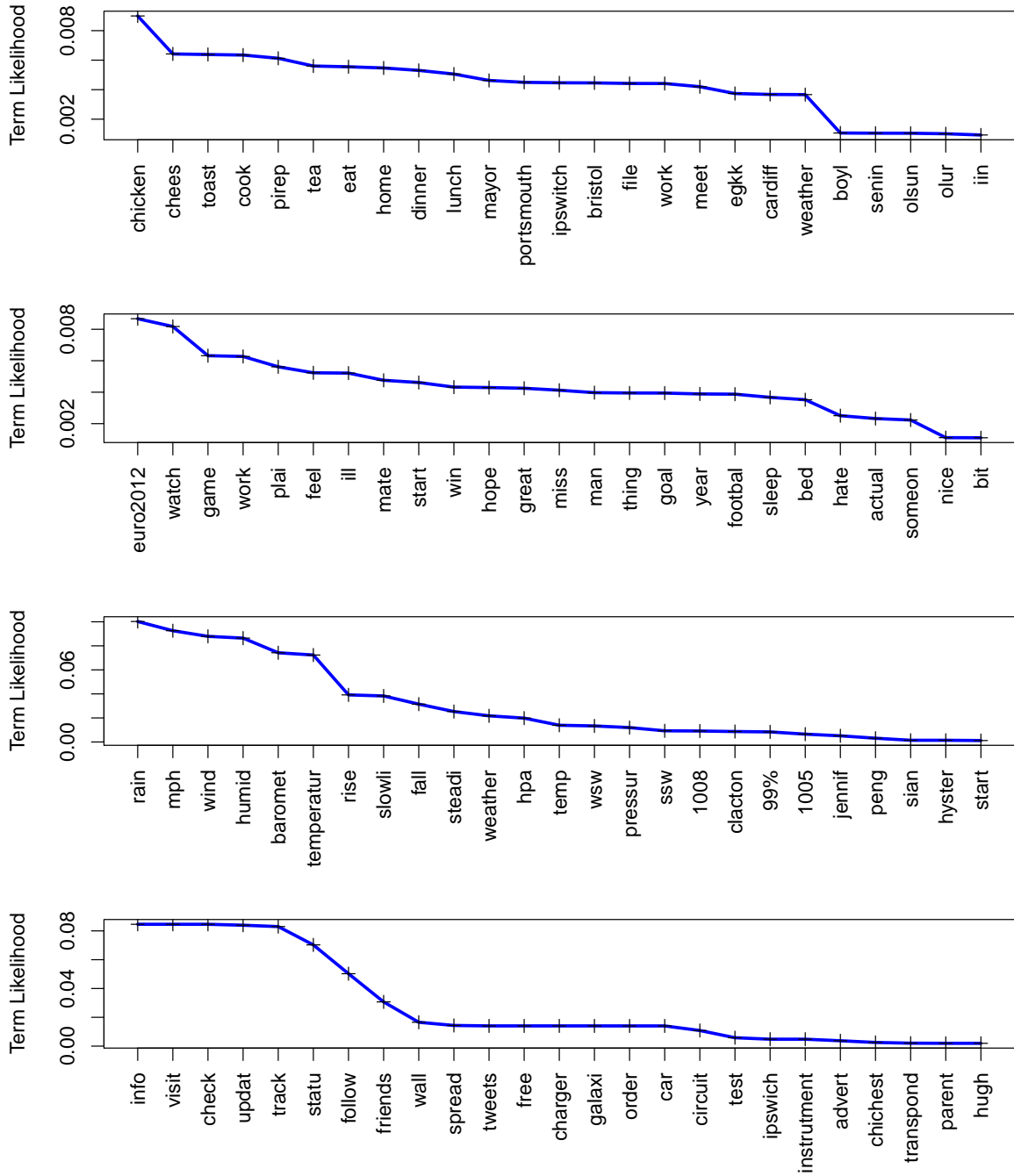


Figure 5.8: Distributions of the likelihood of terms in selected topics associated with communities extracted from the **Sub-England** dataset. For each topic, the likelihood of terms decreases strongly at around the 7<sup>th</sup> and the 20<sup>th</sup> terms.



Table 5.5: Dynamic measures computed at the first five sliding windows for six selected communities extracted from the **Sub-US** dataset. These communities are manually labeled based on the topic that is the most prominence in all sliding windows.

<b>Politics communities:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.40	0.20	0.35	0.73	0.60	0.40	0.93	0.40	0.30
02	0.60	0.20	0.40	0.76	0.40	0.40	0.93	0.40	0.40
03	0.63	0.40	0.25	0.70	0.40	0.35	0.96	0.40	0.65
04	0.53	0.40	0.35	0.63	0.40	0.60	0.93	0.40	0.70
05	0.66	0.0	0.45	0.76	0.20	0.35	0.70	0.40	0.75
<b>Average</b>	<b>0.56</b>	<b>0.24</b>	<b>0.36</b>	<b>0.71</b>	<b>0.40</b>	<b>0.41</b>	<b>0.89</b>	<b>0.40</b>	<b>0.56</b>
01	0.56	0.20	0.20	0.76	0.40	0.30	0.86	0.40	0.55
02	0.76	0.20	0.30	0.70	0.20	0.25	0.96	0.40	0.68
03	0.70	0.20	0.20	0.73	0.20	0.10	0.96	0.40	0.60
04	0.66	0.0	0.15	0.66	0.40	0.15	0.86	0.60	0.72
05	0.56	0.0	0.20	0.63	0.30	0.30	0.90	0.60	0.62
<b>Average</b>	<b>0.65</b>	<b>0.12</b>	<b>0.21</b>	<b>0.70</b>	<b>0.30</b>	<b>0.22</b>	<b>0.91</b>	<b>0.48</b>	<b>0.63</b>
<b>Job communities:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.66	0.10	0.20	0.76	0.40	0.40	0.86	0.60	0.35
02	0.63	0.20	0.25	0.86	0.40	0.40	1.00	0.40	0.45
03	0.76	0.20	0.20	0.86	0.20	0.35	0.93	0.60	0.60
04	0.66	0.0	0.25	0.93	0.60	0.60	1.00	0.20	0.70
05	0.76	0.0	0.15	0.80	0.80	0.10	0.86	0.40	0.80
<b>Average</b>	<b>0.69</b>	<b>0.10</b>	<b>0.21</b>	<b>0.84</b>	<b>0.48</b>	<b>0.37</b>	<b>0.93</b>	<b>0.44</b>	<b>0.58</b>
01	0.76	0.20	0.20	0.75	0.60	0.35	0.85	0.40	0.60
02	0.63	0.20	0.25	0.73	0.20	0.40	0.80	0.40	0.65
03	0.66	0.0	0.20	0.80	0.60	0.65	0.93	0.60	0.55
04	0.70	0.0	0.25	0.76	0.20	0.55	0.96	0.40	0.70
05	0.60	0.0	0.15	0.63	0.40	0.55	0.93	0.50	0.50
<b>Average</b>	<b>0.67</b>	<b>0.08</b>	<b>0.21</b>	<b>0.73</b>	<b>0.40</b>	<b>0.50</b>	<b>0.89</b>	<b>0.46</b>	<b>0.60</b>
<b>Weather communities:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.63	0.30	0.25	0.63	0.60	0.40	0.90	0.40	0.40
02	0.70	0.0	0.45	0.70	0.60	0.45	1.00	0.20	0.70
03	0.66	0.0	0.50	0.76	0.20	0.50	0.93	0.60	0.75
04	0.66	0.0	0.40	0.86	0.80	0.55	0.96	0.0	0.70
05	0.76	0.0	0.30	0.66	0.60	0.45	0.93	0.60	0.70
<b>Average</b>	<b>0.68</b>	<b>0.06</b>	<b>0.38</b>	<b>0.72</b>	<b>0.56</b>	<b>0.47</b>	<b>0.94</b>	<b>0.36</b>	<b>0.65</b>
01	0.66	0.20	0.45	0.73	0.40	0.50	0.83	0.40	0.55
02	0.50	0.30	0.55	0.76	0.40	0.40	0.93	0.40	0.50
03	0.63	0.0	0.25	0.80	0.10	0.60	1.00	0.40	0.55
04	0.50	0.0	0.30	0.73	0.20	0.55	0.86	0.20	0.65
05	0.56	0.20	0.15	0.70	0.40	0.60	0.93	0.40	0.70
<b>Average</b>	<b>0.59</b>	<b>0.14</b>	<b>0.34</b>	<b>0.74</b>	<b>0.30</b>	<b>0.53</b>	<b>0.91</b>	<b>0.36</b>	<b>0.59</b>

Table 5.6: Dynamic measures computed at the first five sliding windows for five selected communities extracted from the **Sub-England** dataset. These communities are manually labeled based on the topic that is the most prominence in all sliding windows.

<b>Football community:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.40	0.0	0.35	0.63	0.20	0.50	0.73	0.40	0.60
02	0.53	0.20	0.40	0.73	0.0	0.45	0.83	0.20	0.50
03	0.50	0.0	0.35	0.76	0.20	0.35	0.86	0.20	0.65
04	0.53	0.20	0.45	0.80	0.0	0.50	0.83	0.20	0.60
05	0.46	0.0	0.45	0.83	0.20	0.60	0.70	0.40	0.65
<b>Average</b>	<b>0.48</b>	<b>0.08</b>	<b>0.40</b>	<b>0.75</b>	<b>0.12</b>	<b>0.48</b>	<b>0.79</b>	<b>0.28</b>	<b>0.60</b>
<b>Social media community:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.46	0.0	0.20	0.66	0.0	0.25	0.76	0.20	0.35
02	0.53	0.0	0.25	0.70	0.0	0.35	0.86	0.40	0.45
03	0.66	0.20	0.25	0.76	0.20	0.30	0.83	0.20	0.60
04	0.66	0.0	0.35	0.86	0.0	0.40	0.80	0.20	0.50
05	0.56	0.20	0.15	0.86	0.40	0.25	0.86	0.20	0.40
<b>Average</b>	<b>0.57</b>	<b>0.08</b>	<b>0.24</b>	<b>0.76</b>	<b>0.12</b>	<b>0.31</b>	<b>0.82</b>	<b>0.24</b>	<b>0.46</b>
<b>Weather community:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.45	0.20	0.20	0.76	0.20	0.45	0.75	0.40	0.50
02	0.51	0.0	0.30	0.80	0.20	0.35	0.80	0.20	0.40
03	0.53	0.0	0.22	0.73	0.0	0.30	0.85	0.20	0.55
04	0.60	0.20	0.40	0.73	0.40	0.40	0.75	0.20	0.65
05	0.55	0.20	0.10	0.60	0.20	0.55	0.83	0.40	0.50
<b>Average</b>	<b>0.53</b>	<b>0.12</b>	<b>0.24</b>	<b>0.72</b>	<b>0.20</b>	<b>0.41</b>	<b>0.80</b>	<b>0.32</b>	<b>0.52</b>
<b>Food community:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.45	0.20	0.10	0.73	0.20	0.40	0.80	0.20	0.50
02	0.50	0.0	0.30	0.66	0.0	0.75	0.83	0.20	0.40
03	0.30	0.20	0.20	0.76	0.30	0.35	0.73	0.40	0.55
04	0.50	0.20	0.15	0.83	0.20	0.25	0.90	0.20	0.30
05	0.53	0.0	0.20	0.63	0.0	0.50	0.85	0.40	0.60
<b>Average</b>	<b>0.46</b>	<b>0.12</b>	<b>0.19</b>	<b>0.72</b>	<b>0.14</b>	<b>0.45</b>	<b>0.82</b>	<b>0.28</b>	<b>0.47</b>
<b>Music and event community:</b>									
Sliding Window	1-week interval			2-week interval			1-month interval		
	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$	$\partial_\phi$	$\partial_\pi$	$\partial_\varphi$
01	0.30	0.0	0.20	0.63	0.0	0.25	0.72	0.20	0.40
02	0.40	0.20	0.30	0.73	0.20	0.45	0.80	0.20	0.60
03	0.45	0.0	0.32	0.76	0.20	0.80	0.65	0.20	0.55
04	0.41	0.0	0.20	0.80	0.0	0.35	0.85	0.40	0.45
05	0.50	0.20	0.35	0.73	0.40	0.50	0.80	0.40	0.40
<b>Average</b>	<b>0.41</b>	<b>0.08</b>	<b>0.27</b>	<b>0.73</b>	<b>0.16</b>	<b>0.47</b>	<b>0.76</b>	<b>0.28</b>	<b>0.48</b>

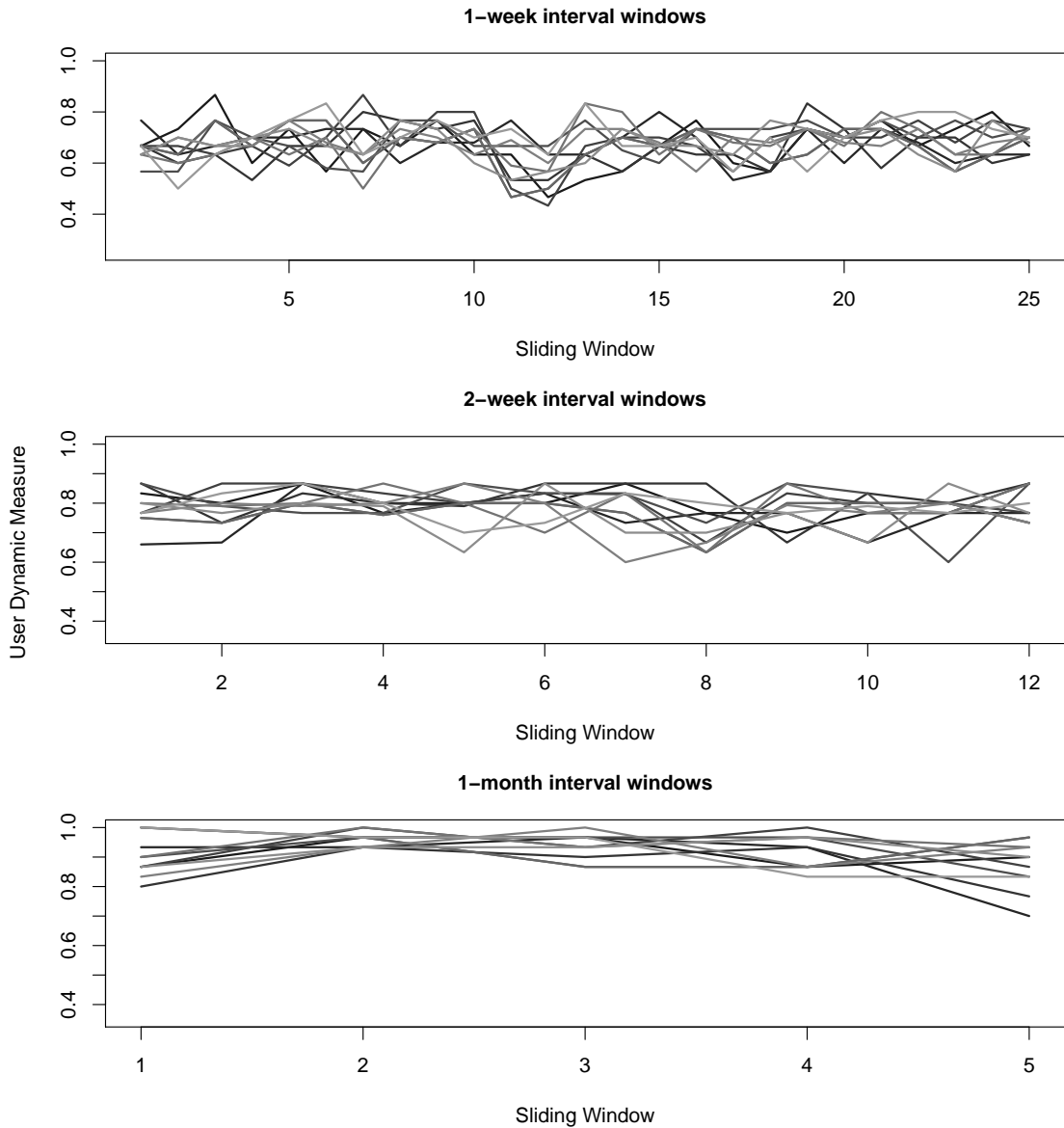


Figure 5.9: User dynamic measure computed for ten communities discovered from the **Sub-US** dataset with three different time intervals of sliding windows. Larger values of the user dynamic measure are observed as a longer time interval is employed to create a sliding window. This indicates that the likelihood of users in a community changes gradually over short time.

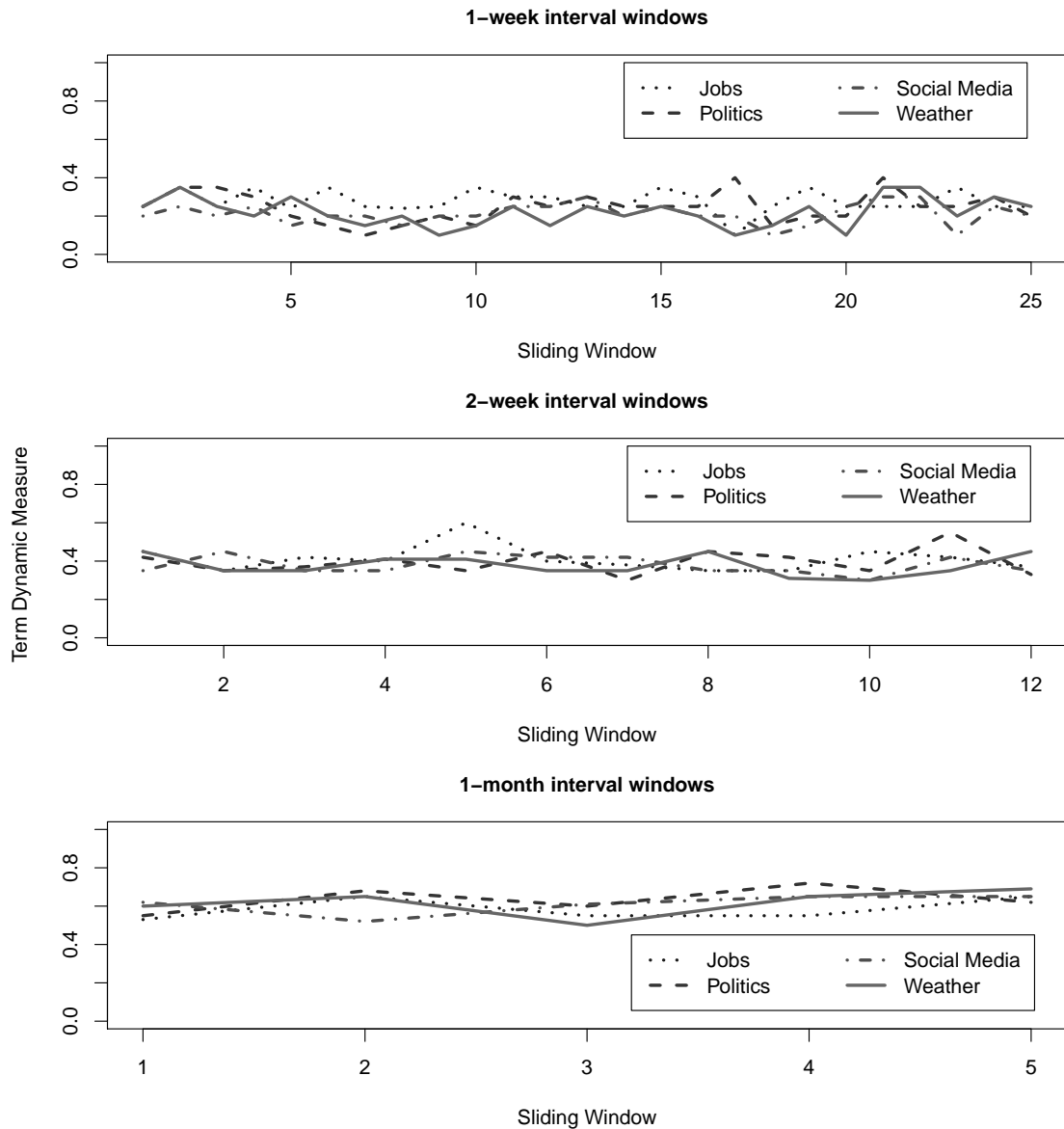


Figure 5.10: Term dynamic measure computed for four topics associated with communities discovered from the **Sub-US** dataset with three different time intervals of sliding windows. By comparing the user dynamic measure and term dynamic measure (see Figure 5.9), it is observed that changes in the community members happen more often than changes in terms describing community topics.

### 5.5.4 Selected Evolving Communities

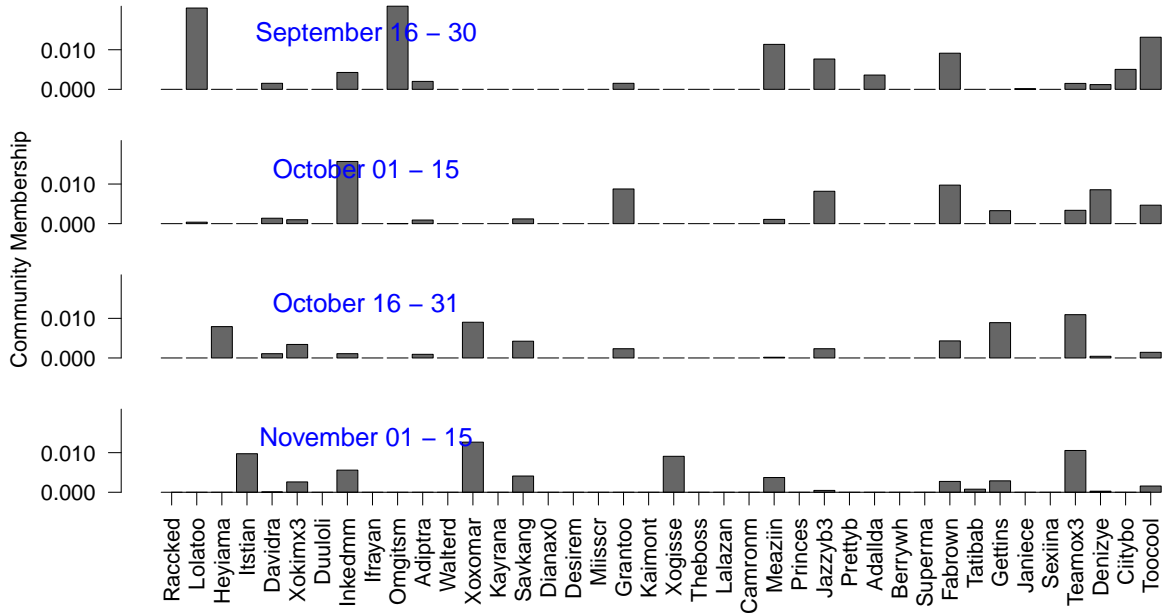
Example communities extracted from the **Sub-US** dataset are presented in this section to demonstrate the effectiveness of the *ErLinkTopic* model in extracting evolving communities. For this purpose, topics associated with communities extracted by the model are first manually classified into the groups *politics*, *jobs*, *social activities*, *weather*, *music and social events*, *social media*, *social networks*, *sports*, and *general*. A topic is labeled as *general* if terms occurring in that topic are about different subjects making it unclear for a classification. Example terms describing some selected topics are summarized in Table 5.7.

Table 5.7: Example terms occurring in selected topics associated with communities extracted from the **Sub-US** dataset.

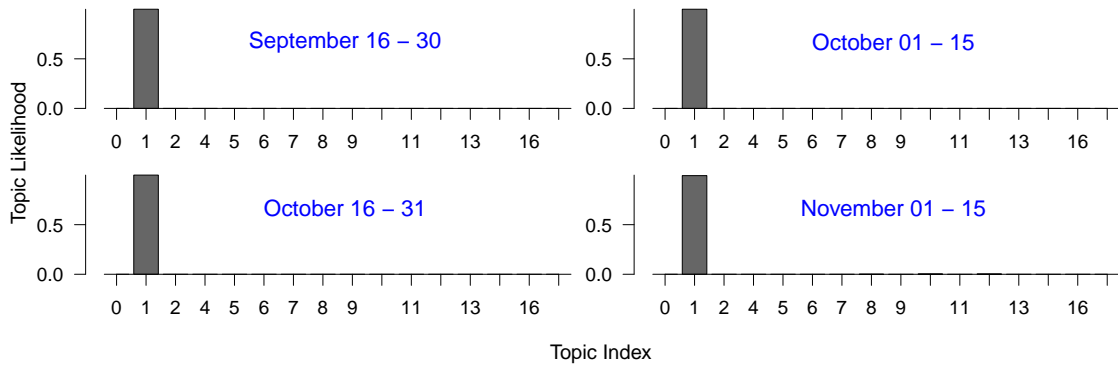
Topic label	Example terms
politics	vote, insur (insurance), job, fairfield, agenc (agency), obama (Barack Obama), mitt (Mitt Romney), blue (blue party), economi
jobs	job, retail, manag (manager), sale, hospital, marketing, account, internship, assist (assistant), businessmgr (business management)
social media	instagood, photooftheday (photo of the day), instagram, iphone, iphonegraphi, earth, iphonesia
social networks	update, follow, track, visit, info (information), spread, tweet, check, wall, friends
weather	forecast, sunni (sunny), cloudi (cloudy), rain, shower, mostli (mostly), chanc (chance), wind, temperatur (temperature), rise, storm, cold
music and events	plai (play), music, live, girl, boy, station, song, happi (happy), weekend, life, radio
food and restaurant	coupon, restaurant, rauti, pizza, menu, pric (price), tast (taste)
sports	plai (play), watch, game, great, hope, start, people, happi (happy), man

We then manually label each community based on the prominence of topics associated with it. Generally, each community is associated with at most two topics at a time point. The evolution of each community is characterized by changes in the community membership of users, the prominence of topics, and the likelihood of terms in each topic as well. Evolving phenomena that are observed from communities extracted from our datasets include the stability, generalization, specification, and shifting of the prominence of topics associated with a community; the growth and shrinkage of community members; and the stability of terms describing topics. In our experiments, we rarely find the stability of community members, especially when a sliding window of more than 2-week interval is applied. This indicates that users in social networks in general and particularly *Twitter* users are dynamic in terms of posting messages associated with contextual links of different topics reflecting their complex life and changing geographic locations over time. In the following, selected communities that exhibit some specific evolving phenomena are described.

**Stability of topic prominence.** Almost all communities exhibit a stability regarding the prominence of topics for a while. The time period of such stability varies from different communities but reaches a maximum of two months. Figure 5.11 shows a community characterized by a topic about music and social events from September 16, 2012 to November 15, 2012. As shown in the Figure 5.11(b), during this period only the topic indexed 1 that is manually classified as “music and social events” is prominent for this community.



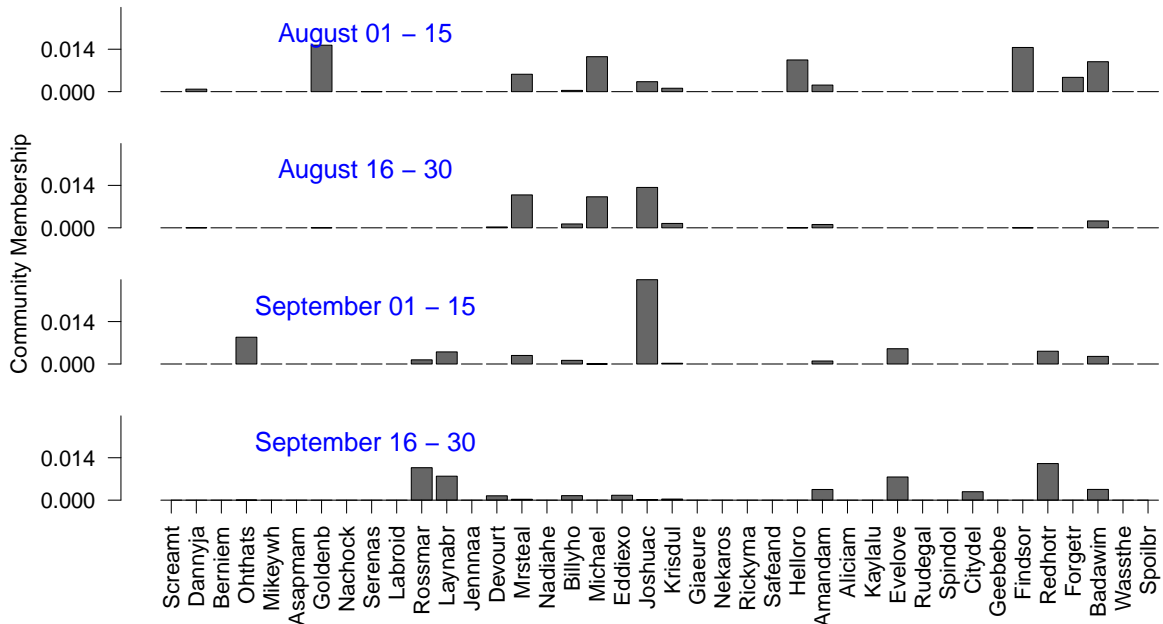
(a) Membership of users in the community



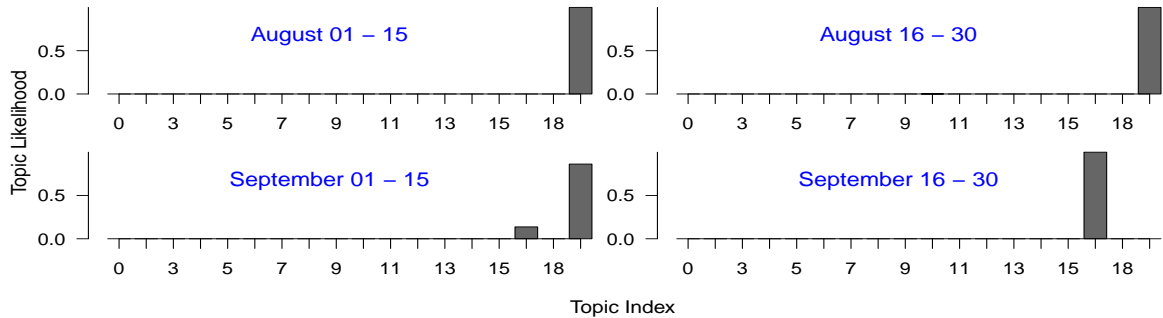
(b) Prominence of a topic about *music and social events* associated with the community

Figure 5.11: The evolution of community members (above) and the stability of the prominence of a topic about music and social events (below) of a community discovered from the **Sub-US** dataset.

**Shifting of topics of interest.** There are communities that gradually change the topics of interest at some point in time. That is, the likelihood of the topic characterizing the community (i.e., the topic having the highest likelihood in the community) starts to decrease and another topic becomes more prominent at the same time. In this regard, we find an interesting trend from the **Sub-US** dataset that communities characterized by a *job* topic tend to shift their interest to politics before the election in the US in 2012. Figure 5.12 shows an example. At first, this community is associated with a topic described by terms about jobs (the topic indexed 19) during August 2012. The shifting of topics happens at the beginning of September 2012, where the likelihood of the topic described by terms about politics (the topic indexed 16) increases. By the end of September 2012, the community is characterized by only the *politics* topic.



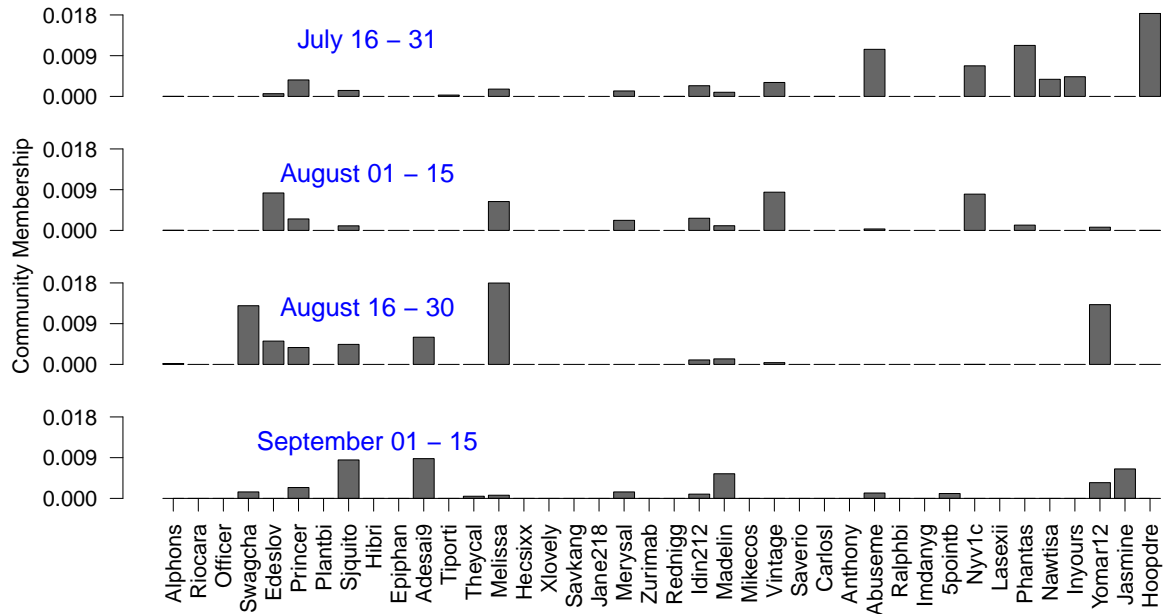
(a) Community membership of users



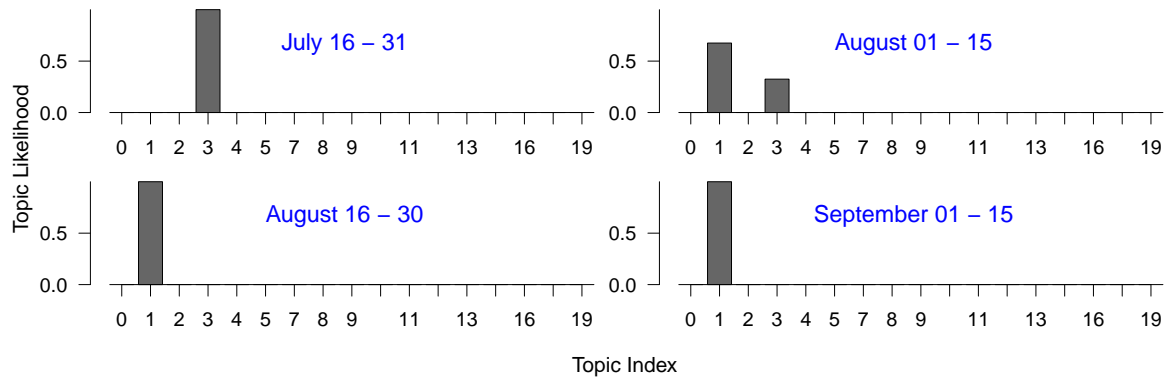
(b) Prominence of topics associated with the community

Figure 5.12: The evolution of community members and the shifting of the prominence of a topic about jobs (indexed 19) to a topic about politics (indexed 16) of a community discovered from the **Sub-US** dataset.

Another example of the shifting of topic prominence from a *general* topic to a topic about music and social events is shown in Figure 5.13. For this community, it is first associated with the topic indexed 3, which is very general (i.e., words occurring in this topic are about many subjects). The shifting phenomenon then happens at the beginning of August 2012, where the community is also characterized by another topic about music and social events (the topic indexed 1). Since middle of August, only this new emerging topic is associated with the community.



(a) Community membership of users

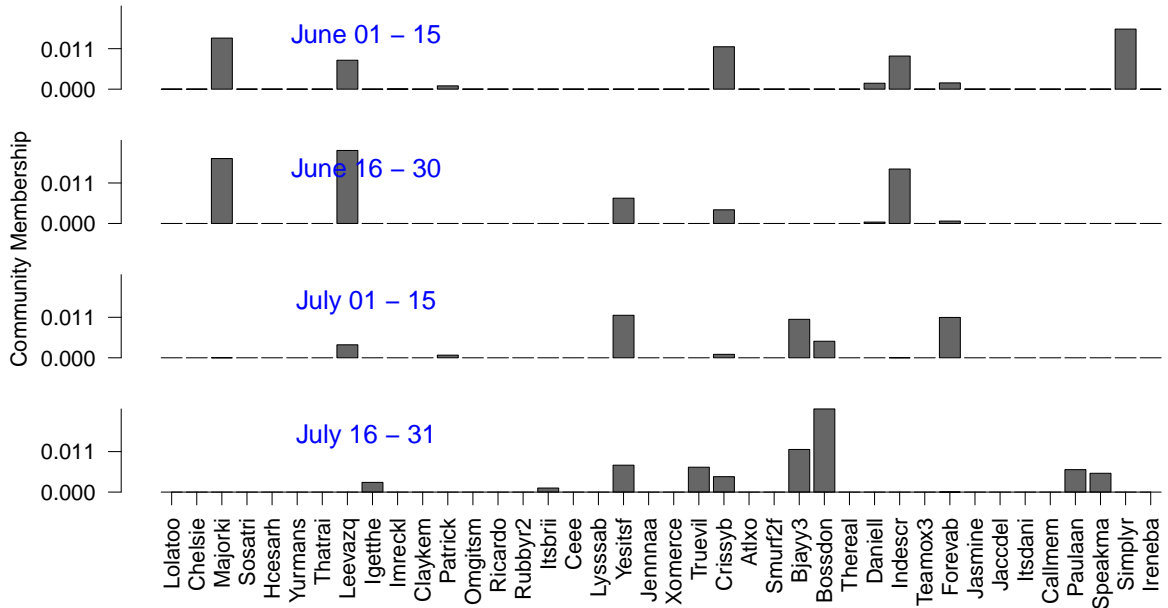


(b) Prominence of topics associated with the community

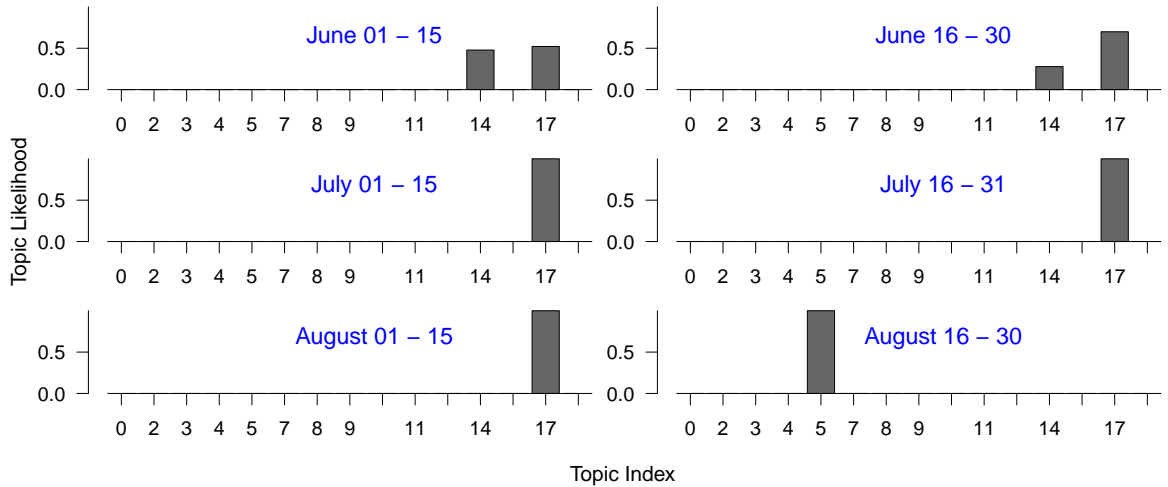
Figure 5.13: The evolution of community members and the shifting of the prominence of a *general* topic (indexed 3) to a topic about music and social events (indexed 1) of a community discovered from the **Sub-US** dataset.



**Specification.** An example of communities that change the interest from more topics to fewer topics, i.e., a community becomes more *topical specific*, is shown in Figure 5.14. Here, two topics indexed 14 and 17, one is about social networks (e.g., update, check-in, follows,...) and another is more about social media (e.g., iphone, pictureoftheday, shot,...), characterize the community for the first two weeks in June 2012. Then, the topic about social networks becomes less prominent and the community finally described by only the topic about social media.



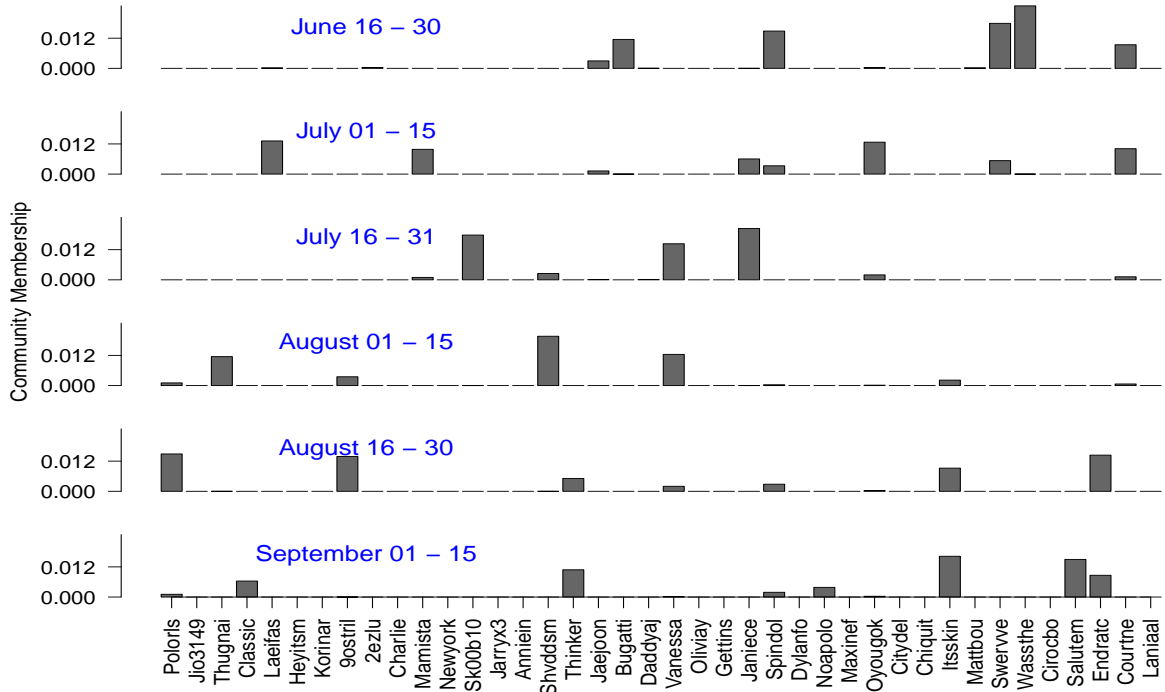
(a) Community membership of users



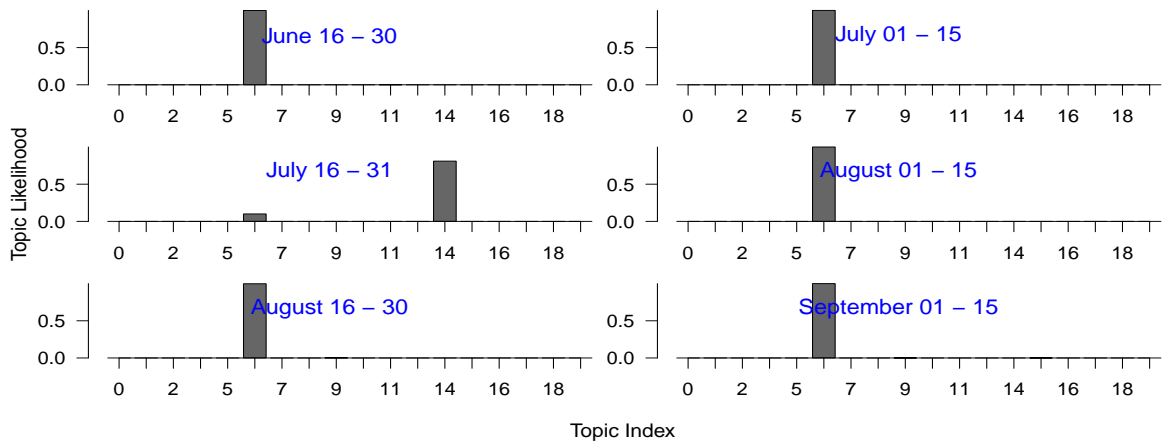
(b) Prominence of topics associated with the community

Figure 5.14: The evolution of community members and changes in the prominence of topics of a community discovered from the **Sub-US** dataset. The community becomes more *topical specific* due to the changes in the number of prominent topics from two to one.

**Change to a new topic and then return to old topic.** We also find an interesting phenomenon that a community might quickly change the interest to another topic for a while and then turns back to the topic characterizing it before. An example of this phenomenon is shown in Figure 5.15. This community is associated with a topic about weather (indexed 6) for almost all the time. However, during the period of the last two weeks in July 2012 users in this community post messages about social networks (indexed 14).



(a) Community membership of users



(b) Prominence of topics associated with the community

Figure 5.15: This community is characterized by a topic about weather for almost of the time from June 16, 2012 to September 15, 2012 except the last two weeks in July where it quickly turns the interest to social networks.

### 5.5.5 Evolution of Topics Associated with Communities

This section describes the evolution of example topics that exhibit changes in the likelihood of specific terms reflecting some real-world phenomena and events.

**Topic about weather.** The evolution of terms occurring in the topic about weather is shown in Figure 5.16. Based on the likelihood of terms at different points in time, it is observable that in June and July 2012, the weather topic is more clear compared to other time periods. Furthermore, the likelihood of some specific terms changes over seasons or reflects a related weather phenomenon happening. For example, “storm”, “chanc” (chance), “wind”, “forecast”, “mph” (miles per hour), and “rain” have a high likelihood in the topic during the last two weeks in June 2012. Actually, there was a storm happening in the New York area at that time<sup>3</sup>. The term “thundersto” (thunderstorm) occurs more in the first two weeks in August 2012, at that time severe thunderstorms happened in New York also. In addition, the evolution of the likelihood of “falls” and “cold” follows the trend that such terms are used to describe weather over seasons in a year.

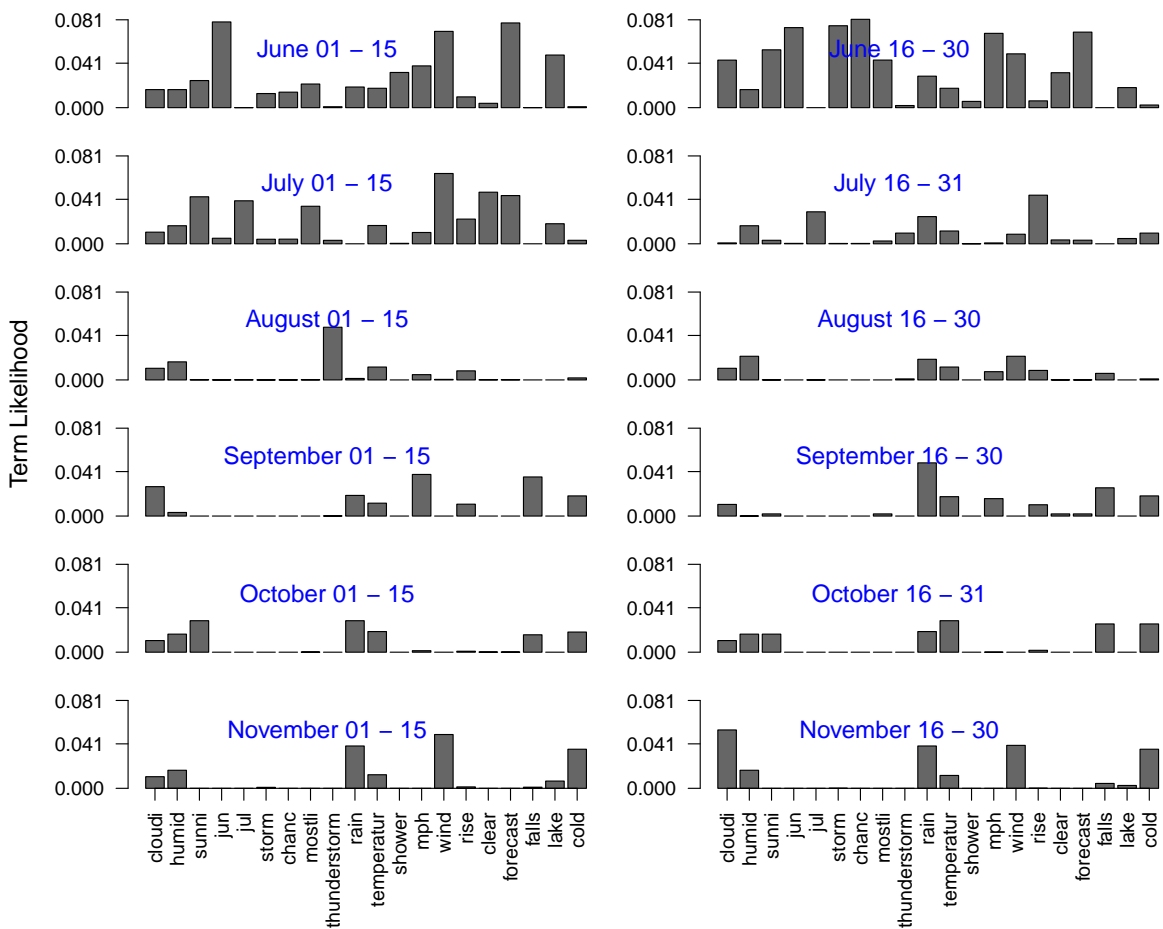


Figure 5.16: The evolution of a topic about weather discovered from the **Sub-US** dataset.

<sup>3</sup><http://www.erh.noaa.gov/okx/stormtotals.html>[Accessed January 2014]

**Topic about politics.** The evolution of terms occurring in the *politics* topic is shown in Figure 5.17. As expected, changes in the likelihood of terms exhibit a reflection of the presidential election in the US in 2012. It is observable that as time approaches the election schedule (December 2012) the likelihood of the related terms increases sufficiently.

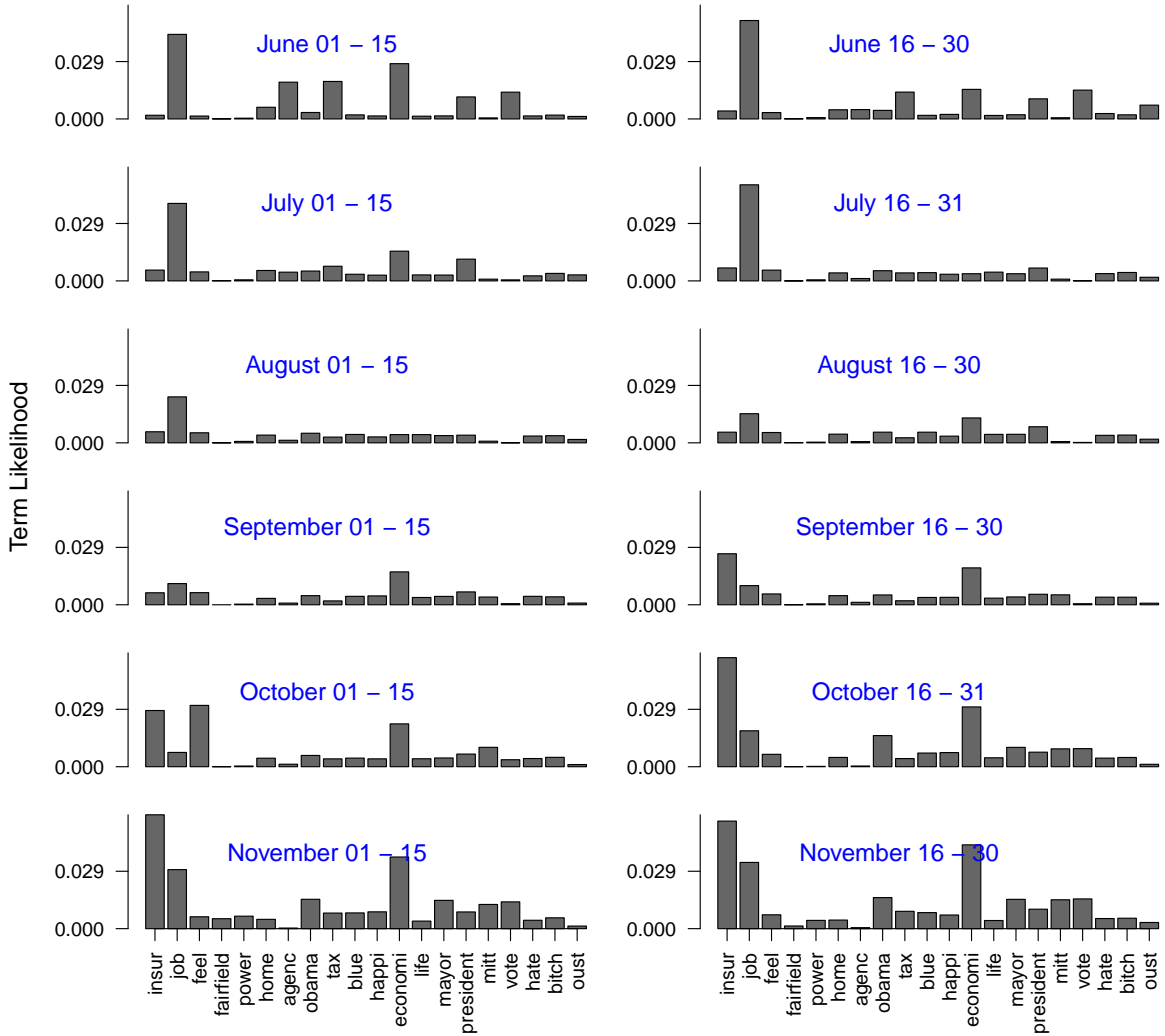


Figure 5.17: The evolution of a topic about politics discovered from the **Sub-US** dataset.

**Topic about jobs.** The evolution of the likelihood of terms in the topic about jobs is not that clear as in the topics about weather and politics described above. However, there are some terms that co-occur at specific points in time that might be of interest. For example, “job”, “financi” (financial), and “degree” occur more in June 2012. Terms about specific job positions such as “develop” (development), “senior”, “assist” (assistant), and “engin” (engineer) are prominent in August and September 2012. This topic is not much clear in October and November 2012 as not many terms describing jobs are prominent during these two months, which implies a vanishing phenomenon of the topic.

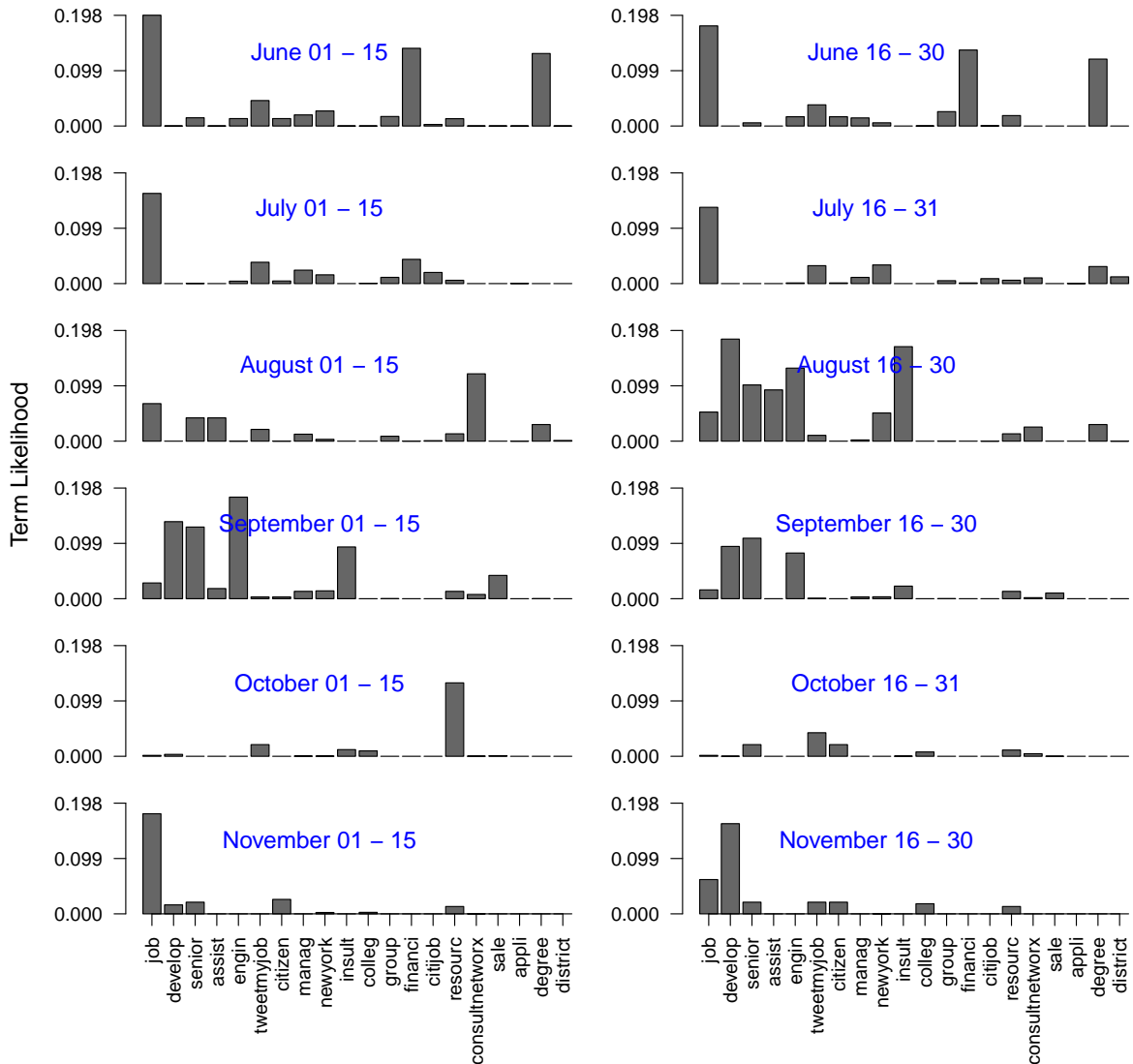


Figure 5.18: The evolution of a topic about jobs discovered from the **Sub-US** dataset.

### 5.5.6 Evaluation of Runtime

This section discusses the running time of the *rLinkTopic* algorithm applied to the datasets used in the experiments presented. Particularly, for each time interval of sliding windows, we measure the running time of the algorithm using three different settings of the number of iterations for sampling. In the first setting, the model is run with 820 steps for the *Burn-In* stage and 180 steps for collecting assignment samples and updating multinomial parameters. The results (i.e., the communities, topics, and their evolution) presented in this chapter are derived from this configuration. In the second setting, 700 steps for the *Burn-In* stage and 100 steps for collecting assignment samples and updating multinomial parameters are employed. Such steps of iterations for the last setting are 600 and 100, respectively.

The results show that for each dataset the model takes almost the same time when it is run with different time intervals of sliding windows, given that the same number of communities  $|C|$  and number of topics  $|Z|$  are assigned to the model. This is shown by the rows marked by (\*) in Table 5.8. Also, the running time of the algorithm increases linearly to the number of iterations and the number of communities applied. Details of the evaluations are summarized in Table 5.8 and Figures 5.19, 5.20.

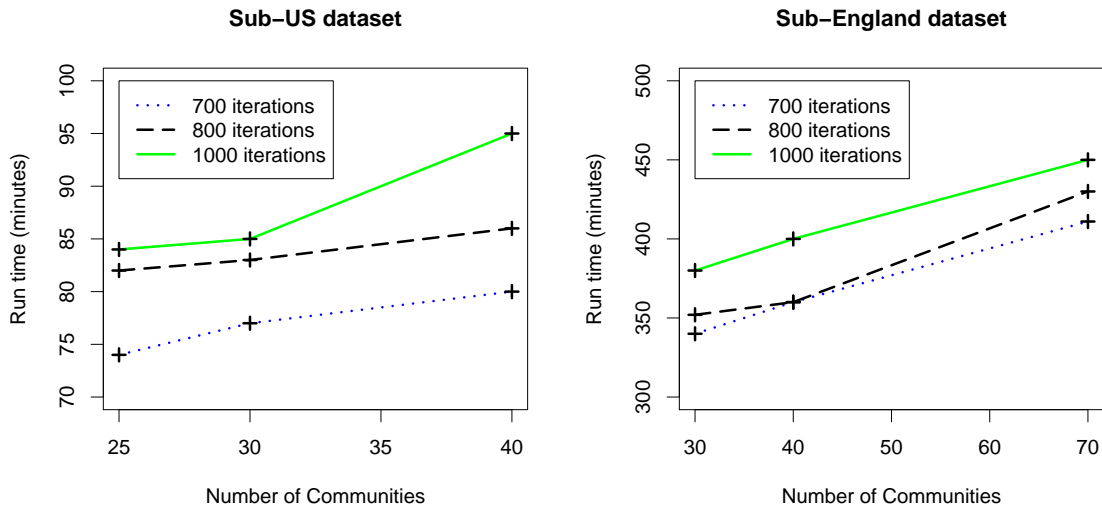
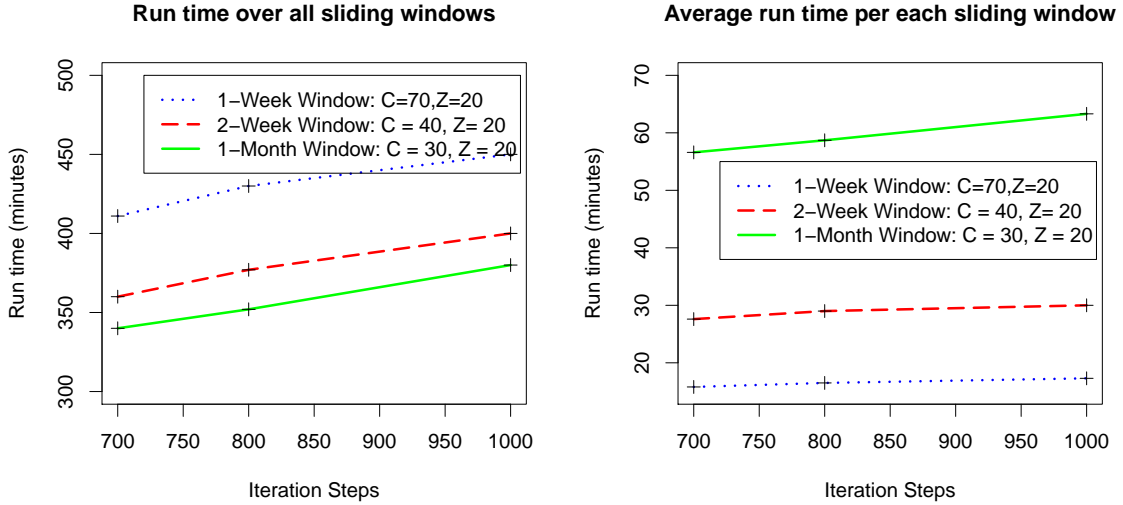
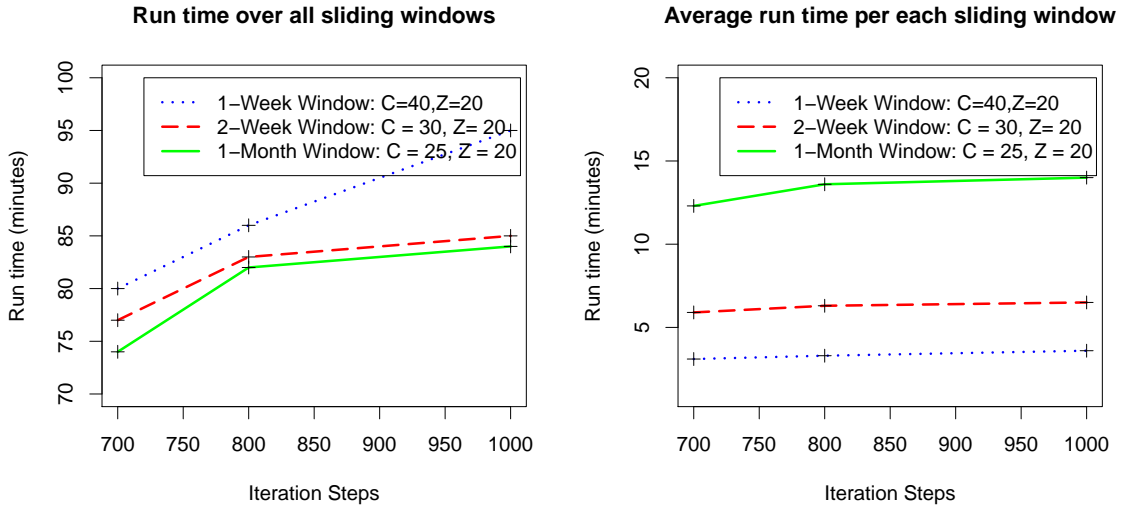


Figure 5.19: Running time of the *ErLinkTopic* algorithm applied to the **Sub-US** dataset (left) and **Sub-England** dataset (right) when different number of communities are extracted. Three settings of the number of iterations (700, 800, and 1000) are employed to measure running time.



(a) Sub-England dataset



(b) Sub-US dataset

Figure 5.20: Running time of the *ErLinkTopic* algorithm applied to the **Sub-England** dataset (a) and **Sub-US** dataset (b). Three time intervals (1 week, 2 weeks, and 1 month) are employed to create sliding windows. For each time interval, three settings of the number of iterations (700, 800, and 1000) are used in the *ErLinkTopic* algorithm. The detailed measurements are shown in Table 5.8.

Table 5.8: Parameter settings and the corresponding running time of the *ErLinkTopic* algorithm when applied to the **Sub-England** and **Sub-US** datasets. Noted that the parameters in the rows marked by (\*) for each dataset are the same except the time interval of sliding windows. The running time shown in these rows indicates that the size of windows does not affect the computational efficiency of the algorithm.

<b>Sub-England dataset:</b>						
	$ C $	$ Z $	<i>Burn-In</i>	<i>Sampling steps</i>	<i>Time (Minutes)</i>	<i>Time/Window</i>
<b>1-week window</b>	<b>70</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>450.0</b>	<b>17.3(*)</b>
	70	20	700	100	430.0	16.5
	70	20	600	100	411.0	15.8
<b>2-week window</b>	<b>40</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>400.0</b>	<b>30.0</b>
	40	20	700	100	377.0	29.0
	40	20	600	100	360.0	27.6
<b>1-month window</b>	<b>30</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>380.0</b>	<b>63.3</b>
	30	20	700	100	352.0	58.7
	30	20	600	100	340.0	56.6
	70	20	820	180	452	75.3(*)
<b>Sub-US dataset:</b>						
	$ C $	$ Z $	<i>Burn-In</i>	<i>Sampling steps</i>	<i>Time (Minutes)</i>	<i>Time/Window</i>
<b>1-week window</b>	<b>40</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>95.0</b>	<b>3.6(*)</b>
	40	20	700	100	86.0	3.3
	40	20	600	100	80.0	3.1
<b>2-week window</b>	<b>30</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>85.0</b>	<b>6.5</b>
	30	20	700	100	83.0	6.3
	30	20	600	100	77.0	5.9
<b>1-month window</b>	<b>25</b>	<b>20</b>	<b>820</b>	<b>180</b>	<b>84.0</b>	<b>14.0</b>
	25	20	700	100	82.0	13.6
	25	20	600	100	74.0	12.3
	40	20	820	180	95.0	15.8(*)

## 5.6 Summary and Discussion

**Summary.** Understanding how communities evolve over time have become a hot topic in the field of social network analysis due to the wide range of its applications. In this context, several approaches have been introduced to capture changes in the community members. Our claim is that a community is characterized by complex features, not only the identity of users. Examples include the topics of interest, and the regional and geographic characteristics. Studying changes in such features of communities also provides informative findings for related applications. This leads to the main goal of the study in this chapter, which is to capture the evolution of complex features describing communities. Particularly, we have extended the *rLinkTopic* model developed in Chapter 4 to build a complete framework called *ErLinkTopic* model. The model is able to extract regional *LinkTopic* communities and to capture gradual changes in three features describing each community, i.e., community



members, the prominence of topics describing communities, and terms describing such topics. It further supports the study of regional and geographic characteristics of communities as well as changes in such features. Experimental evaluations have been conducted using *Twitter* data to evaluate the model in terms of its effectiveness and efficiency in extracting communities and capturing changes in the features describing each community.

**Open issues.** There are aspects in the proposed framework that we would like to study in order to improve the model. First, in this framework, regions are derived from the density of geographic locations of users within each snapshot. This implies an assumption that regions might change over time. Because of this, the model ignores the evolution of the community distribution in each region. There should be an improvement for the model in a way that it is able to capture region evolution as well. Second, even though our model supports the study of changes in the regional and geographic characteristics of communities, the results of such analysis tasks have not been discussed in the experiments. The reason is that we need more knowledge about local geographic areas to be able to give comments on the evolution of such features of communities. Third, due to the lack of ground truth in real-world datasets, evaluating the results of extracting feature-based communities and analyzing their evolution is a challenging task. In this work, case studies have been employed. However, it is not a comprehensive method to evaluate the results because no quantitative measures are derived. There might be two possible solutions for this. The first method is to develop a Bayesian prediction model and adapt it to the framework to predict some features of users in a sub-dataset and compare the predicted results with the features of users. This method was employed in [74]. The second method, as usual, is to develop annotated benchmarks, which is more challenging due to the complex constraints in extracting and analyzing feature-based communities. Finally, in our framework, we assume there are no changes in the number of communities  $|C|$  and the number of topics  $|Z|$  across time. As discussed in the last section of Chapter 4, it should be more appropriate if a *Dirichlet* process is employed so that these constraints are relaxed.



## Chapter 6

# Conclusions and Future Work

Social networking has become part of our life. Hundreds of millions of users are active in social networks and create massive amounts of rich-feature data on a daily basis. Thus, social networks are becoming the largest data repositories that capture information reflecting real-life activities of people worldwide. Extracting knowledge from such data is apparently useful for understanding human beings and society in general, and potentially facilitates to obtain more benefits from various applications that take advantages of knowing the behavior of users through social networks. This makes research studies in the field of social network analysis to be popularized. In the context of this dissertation, social links and communities were targeted as hidden structures obtained from investigating different features embedded in user-generated data. Those imply meaningful and interpretable relationships between users. Particularly, the thesis presents new models and algorithms for the measurements of hidden social links between users and the extraction and analysis of feature-based communities in social networks. In the following, we first summarize the key results of the dissertation and then give an outlook for further studies.

### 6.1 Summary

The thesis begins with a discussion of the motivation, the goals and challenges, and the background and related works, which were presented in the first two chapters. In Chapter 3, the concepts of *user occurrence*, *snapshot*, and *social network* were first formalized. These build the underlying data model used to address the problems presented in the whole dissertation. We then developed two approaches for extracting and measuring social links between users; one is based on the participations of users in threads of a blog or a forum network, and another one is derived from applying latent semantic extraction to the postings of users. The models were evaluated using the data collected from the *BBC Message Boards* network. Based on the results obtained, we emphasize that meaningful and interpretable relationships between users can be extracted from the features describing their activities in social networks.

The other major contributions of this dissertation were presented in Chapter 4 and Chapter 5, where a comprehensive framework for extracting and analyzing a new type of feature-based community called regional *LinkTopic* was developed. In Chapter 4, a probabilistic model, *rLinkTopic*, was introduced for extracting regional *LinkTopic* communities from a social network during a period of time. The model was then extended to *ErLinkTopic* in Chapter 5 to address the complex evolution of such communities over time. Technically, both *rLinkTopic* and *ErLinkTopic* are very complex because different features describing users (geographic locations, contextual links, and topics) are taken into account to discover regional *LinkTopic* communities as well as to capture their evolution. Different datasets collected from *Twitter* were used to evaluate the utility, effectiveness, and efficiency of the models and the obtained results were discussed to highlight the advantages our approach.

## 6.2 Future Work

As discussed in the last section of each of the three previous chapters, several open issues can be targeted as extensions to our work presented in this dissertation. We briefly review such remarks and suggest some other aspects for further investigation.

- **Learning social links.** For the extraction and measurements of hidden social links, we have discussed the possibility of employing the history of spatio-temporal mobility of users. A further exploration is to take all the interactions between users, the topics of user postings, and the history of spatio-temporal mobility of users into account to derive a social link measure. Such a sophisticated model is helpful especially for specific applications, for example, detecting criminal or terrorist groups and investigating their communication patterns [126]. In such applications, multiple features describing activities of users are carefully analyzed to derive social link weights between users.
- **Learning communities.** In our framework for extracting and analyzing dynamic regional *LinkTopic* communities, the parametric Bayesian approach has been employed to develop the *rLinkTopic* and *ErLinkTopic* models. A more practical approach is to adapt these models to a nonparametric Bayesian framework so that the number of communities and the number of topics associated with a community are automatically learned by the models. This leads to another issue that needs to be investigated, which is the scalability of the models when applied to large-scale datasets. In addition, the models should be able to capture the evolution of geographic regions so that changes in the distribution of communities in regions are captured as well. Finally, conducting more experiments using data from other social networks, instead of using only *Twitter* data, might give more insights into the results obtained, for example, to evaluate the reliability of the results, as well as the flexibility of the models.

- Other applications.** Extracting communities from social networks is actually an instance of a more general problem known as mining patterns from data. There are several types of patterns that have been defined and explored. Examples include frequent itemsets and sequential patterns in transactional databases [1, 2] and various spatio-temporal patterns in spatial and temporal data [57, 127]. Among these, co-location patterns [20, 56] defined as *a subset of features whose instances are frequently located together in spatial proximity* is most relevant to the concept of regional *LinkTopic* communities. While existing approaches for mining co-location patterns are limited in utility due to the predefined spatial neighborhood (i.e., a distance threshold is used to identify neighbor objects) and topological structures (e.g., clique co-location, star co-location) employed, adapting our probabilistic approach as presented in the *rLinkTopic* model to extract co-location patterns is a promising idea. By this, given a dataset  $\mathcal{D}$  consisting of spatial objects categorized by the features  $F = \{f_1, f_2, \dots, f_n\}$ , a co-location pattern  $c$  is modeled as a multinomial distribution over the features in  $F$ , which is characterized by a variable  $\phi_c = \{P(f|c)\}$  such that  $\sum_{f \in F} P(f|c) = 1$ . Each  $P(f|c)$  is the likelihood of feature  $f$  in pattern  $c$ . Based on the spatial distribution of objects in  $\mathcal{D}$ , the model derives the likelihood for features in patterns such that the features having objects located close to each other obtain a high membership in the same pattern. Following this, no predefined constraints employed in the typical approaches for mining co-location patterns are necessary. Nevertheless, there are open issues such as how to determine the number of patterns and how to measure the prevalence of patterns.

# Bibliography

- [1] Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, June 1993.
- [2] Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering, ICDE '95*, pages 3–14, Washington, DC, USA, 1995. IEEE Computer Society.
- [3] Amr Ahmed and Eric P. Xing. Dynamic non-parametric mixture models and the recurrent Chinese restaurant process: with applications to evolutionary clustering. In *Proceedings of the SIAM International Conference on Data Mining*, pages 219–230. SIAM, 2008.
- [4] Yong-Yeol Ahn, Seungyeop Han, Haewoon Kwak, Sue Moon, and Hawoong Jeong. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th International Conference on World Wide Web, WWW '07*, pages 835–844, New York, NY, USA, 2007. ACM.
- [5] Richard D. Alba. A graph-theoretic definition of a sociometric clique. *Journal of Mathematical Sociology*, 3:3–113, 1973.
- [6] Hélio Almeida, Dorgival Guedes, Wagner Meira, and Mohammed J. Zaki. Is there a best quality metric for graph clusters ? In *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I, ECML PKDD'11*, pages 44–59, Berlin, Heidelberg, 2011. Springer-Verlag.
- [7] Sitaram Asur, Srinivasan Parthasarathy, and Duygu Ucar. An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Trans. Knowl. Discov. Data*, 3(4):16:1–16:36, December 2009.
- [8] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- [9] José M. Bernardo and Adrian F. M. Smith. *Bayesian Theory*. Wiley, 1 edition, 1994.
- [10] Michael S. Bernstein, Eytan Bakshy, Moira Burke, and Brian Karrer. Quantifying the invisible audience in social networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, pages 21–30, New York, NY, USA, 2013. ACM.
- [11] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

- [12] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.
- [13] I. Bomze, M. Budinich, P. Pardalos, and M. Pelillo. The maximum clique problem. In D.-Z. Du and P. M. Pardalos, editors, *Handbook of Combinatorial Optimization, volume 4*. Kluwer Academic Publishers, 1999.
- [14] Francesco Bonchi, Carlos Castillo, Aristides Gionis, and Alejandro Jaimes. Social network analysis and mining for business applications. *ACM Trans. Intell. Syst. Technol.*, 2(3):22:1–22:37, May 2011.
- [15] Stephen P. Borgatti, Ajay Mehra, Daniel J. Brass, and Giuseppe Labianca. Network analysis in the social sciences. *Science*, 323(5916):892–895, 2009.
- [16] U. Brandes, M. Gaertler, and D. Wagner. Engineering graph clustering: models and experimental evaluation. *ACM Journal of Experimental Algorithmics*, 12(1.1), 2008.
- [17] Ulrik Brandes. On variants of shortest-path betweenness centrality and their generic computation. *Journal of Social Networks*, 30(2):136–145, 2008.
- [18] Coen Bron and Joep Kerbosch. Algorithm 457: Finding all cliques of an undirected graph. *Commun. ACM*, 16(9):575–577, September 1973.
- [19] Ronald S. Burt. Models of network structure. *Annual Review of Sociology*, 6:79–141, 1980.
- [20] Tran Van Canh and Michael Gertz. A constraint neighborhood based approach for collocation pattern mining. In *Proceedings of the 2012 Fourth International Conference on Knowledge and Systems Engineering, KSE '12*, pages 128–135, Washington, DC, USA, 2012. IEEE Computer Society.
- [21] Tran Van Canh and Michael Gertz. A spatial LDA model for discovering regional communities. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '13*, pages 162–168, New York, NY, USA, 2013. ACM.
- [22] Deepayan Chakrabarti, Ravi Kumar, and Andrew Tomkins. Evolutionary clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06*, pages 554–560, New York, USA, 2006. ACM.
- [23] Stanley Chen, Douglas Beeferman, and Ronald Rosenfeld. Evaluation Metrics for Language Models. In *DARPA Broadcast News Transcription and Understanding Workshop (BNTUW)*, Lansdowne, Virginia, USA, February 1998.
- [24] Yun Chi, Xiaodan Song, Dengyong Zhou, Koji Hino, and Belle L. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07*, pages 153–162, New York, NY, USA, 2007. ACM.
- [25] Yun Chi, Xiaodan Song, Dengyong Zhou, Koji Hino, and Belle L. Tseng. On evolutionary spectral clustering. *ACM Trans. Knowl. Discov. Data*, 3(4):17:1–17:30, December 2009.

- [26] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1082–1090, New York, NY, USA, 2011. ACM.
- [27] Hyunwoo Chun, Haewoon Kwak, Young-Ho Eom, Yong-Yeol Ahn, Sue Moon, and Hawoong Jeong. Comparison of online social relations in volume vs interaction: a case study of cyworld. In *Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement*, IMC '08, pages 57–70, New York, NY, USA, 2008. ACM.
- [28] Gianni Costa and Riccardo Ortale. A Bayesian hierarchical approach for exploratory analysis of communities and roles in social networks. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 194–201. IEEE Computer Society, 2012.
- [29] David J Crandall, Lars Backstrom, Dan Cosley, Siddharth Suri, Daniel Huttenlocher, and Jon Kleinberg. Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences of the United States of America*, 107(52):22436–22441, 2010.
- [30] Chavdar Dangalchev. Residual closeness in networks. *Physica A: Statistical Mechanics and its Applications*, 365(2):556 – 564, 2006.
- [31] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.
- [32] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.
- [33] Inderjit S. Dhillon and Suvrit Sra. Generalized nonnegative matrix approximations with Bregman divergences. In *Neural Information Proc. Systems*, pages 283–290, 2005.
- [34] Martin Ester, Hans peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD '96, pages 226–231. AAAI Press, 1996.
- [35] M. Everett and S.P. Borgatti. Extending centrality. In J Scott PJ Carington and S Wasserman, editors, *Models and Methods in Social Network Analysis*, pages 57–75. New York: CUP, 2005.
- [36] S. Fortunato. Quality functions in community detection. In *Proceedings of SPIE International Conference Fluctuations and Noise*, Florence, Italy, 2007.
- [37] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(35):75 – 174, 2010.
- [38] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, January 2007.



- [39] Linton C. Freeman. *The Development of Social Network Analysis: a study in the Sociology of Science*. Vancouver, CA: Empirical Press., 2004.
- [40] Marco Gaertler. Clustering. In *Network Analysis*. Volume 3418 of *Lecture Notes in Computer Science*, pages 178–215, Springer, 2004.
- [41] Stuart Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(6):721–741, 1984.
- [42] Minas Gjoka, Maciej Kurant, Carter T. Butts, and Athina Markopoulou. Practical recommendations on crawling online social networks. *IEEE J. Sel. Areas Commun. on Measurement of Internet Topologies*, 2011.
- [43] Neil Zhenqiang Gong, Wenchang Xu, Ling Huang, Prateek Mittal, Emil Stefanov, Vyas Sekar, and Dawn Song. Evolution of social-attribute networks: measurements, modeling, and implications using google+. *CoRR*, abs/1209.0835, 2012.
- [44] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April 2004.
- [45] Tom S. F. Haines and Tao Xiang. Video topic modelling with behavioural segmentation. In *Proceedings of the 1st ACM International Workshop on Multimodal Pervasive Video Analysis*, MPVA '10, pages 53–58, New York, NY, USA, 2010. ACM.
- [46] Bo Han, Paul Cook, and Timothy Baldwin. Lexical normalization for social media text. *ACM Trans. Intell. Syst. Technol.*, 4(1):5:1–5:27, February 2013.
- [47] Jiawei Han. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
- [48] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, April 1970.
- [49] Qi He, Bi Chen, Jian Pei, Baojun Qiu, Prasenjit Mitra, and Lee Giles. Detecting topic evolution in scientific literature: How can citations help? In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 957–966, New York, NY, USA, 2009. ACM.
- [50] David Heath and William Sudderth. De Finetti’s Theorem on Exchangeable Variables. *American Statistician*, 30:188–189, 1976.
- [51] Gregor Heinrich. Parameter estimation for text analysis. Technical report, University of Leipzig, Germany, 2008.
- [52] Keith Henderson and Tina Eliassi-Rad. Applying latent Dirichlet allocation to group discovery in large graphs. In *Proceedings of the 2009 ACM symposium on Applied Computing*, SAC '09, pages 1456–1461, New York, NY, USA, 2009. ACM.
- [53] Adel Hlaoui and Shengrui Wang. A direct approach to graph clustering. In *Neural Networks and Computational Intelligence'04*, pages 158–163, 2004.
- [54] Jake M Hofman and Chris H Wiggins. A Bayesian approach to network modularity. *Physical Review Letters*, 100(25):1–4, 2007.

- [55] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50–57, New York, NY, USA, 1999. ACM.
- [56] Yan Huang, Shashi Shekhar, and Hui Xiong. Discovering collocation patterns from spatial data sets: a general approach. *IEEE Transactions on Knowledge and Data Engineering*, 16(12):1472–1485, 2004.
- [57] Yan Huang, Liqin Zhang, and Pusheng Zhang. A framework for mining sequential patterns from spatio-temporal event data sets. *IEEE Trans. on Knowl. and Data Eng.*, 20(4):433–448, 2008.
- [58] Jing Jiang, Christo Wilson, Xiao Wang, Wenpeng Sha, Peng Huang, Yafei Dai, and Ben Y. Zhao. Understanding latent interactions in online social networks. *ACM Trans. Web*, 7(4):18:1–18:39, November 2013.
- [59] Long Jin, Yang Chen, Tianyi Wang, Pan Hui, and A.V. Vasilakos. Understanding user behavior in online social networks: a survey. *Communications Magazine, IEEE*, 51(9):144–150, September 2013.
- [60] Michael I. Jordan. Graphical models. *Journal of Statistical Science*, 19:140–155, 2004.
- [61] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, November 1999.
- [62] G. Karypis, Eui-Hong Han, and V. Kumar. Chameleon: hierarchical clustering using dynamic modeling. *Computer*, 32(8):68–75, August 1999.
- [63] Brian W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *The Bell System Technical Journal*, 49:291–307, 1970 1970.
- [64] D. Koller, N. Friedman, L. Getoor, and B. Taskar. Graphical Models in a Nutshell. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2007.
- [65] Chandan Kumar, Dirk Ahlers, Wilko Heuten, and Susanne Boll. Interactive exploration of geographic regions with web-based keyword distributions. In *EuroHCIR*, pages 11–14, 2013.
- [66] Ravi Kumar, Jasmine Novak, and Andrew Tomkins. Structure and evolution of online social networks. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 611–617, New York, NY, USA, 2006. ACM.
- [67] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 591–600, New York, NY, USA, 2010. ACM.
- [68] DidierG. Leibovici. Defining spatial entropy from multivariate distributions of co-occurrences. In KathleenStewart Hornsby, Christophe Claramunt, Michel Denis, and Grard Ligozat, editors, *Spatial Information Theory*, volume 5756 of *Lecture Notes in Computer Science*, pages 392–404. Springer Berlin Heidelberg, 2009.

- [69] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. Statistical properties of community structure in large social and information networks. In *Proceedings of the 17th International Conference on World Wide Web, WWW '08*, pages 695–704, New York, NY, USA, 2008. ACM.
- [70] Rong-Hua Li, Jianquan Liu, Jeffrey Xu Yu, Hanxiong Chen, and Hiroyuki Kitagawa. Co-occurrence prediction in a large location-based social network. *Frontiers of Computer Science*, 7(2):185–194, 2013.
- [71] Kwan Hui Lim and Amitava Datta. Finding Twitter communities with common interests using following links of celebrities. In *Proceedings of the 3rd International Workshop on Modeling social media, MSM '12*, pages 25–32, New York, NY, USA, 2012. ACM.
- [72] Yu-Ru Lin, Yun Chi, Shenghuo Zhu, Hari Sundaram, and Belle L. Tseng. Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. In *Proceedings of the 17th International Conference on World Wide Web, WWW '08*, pages 685–694, New York, NY, USA, 2008. ACM.
- [73] Yu-Ru Lin, Yun Chi, Shenghuo Zhu, Hari Sundaram, and Belle L. Tseng. Analyzing communities and their evolutions in dynamic social networks. *ACM Trans. Knowl. Discov. Data*, 3(2):8:1–8:31, April 2009.
- [74] Yu-Ru Lin, Jimeng Sun, Hari Sundaram, Aisling Kelliher, Paul Castro, and Ravi Konuru. Community discovery via metagraph factorization. *ACM Trans. Knowl. Discov. Data*, 5(3):17:1–17:44, August 2011.
- [75] Yuan Lin, Hongfei Lin, Jiajin Wu, and Kan Xu. Learning to rank with cross entropy. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management, CIKM '11*, pages 2057–2060, New York, USA, 2011. ACM.
- [76] Pasquale Lops, Marco de Gemmis, Giovanni Semeraro, Fedelucio Narducci, and Cataldo Musto. Leveraging the LinkedIn social network data for extracting content-based user profiles. In *Proceedings of the fifth ACM Conference on Recommender Systems, RecSys '11*, pages 293–296, New York, NY, USA, 2011. ACM.
- [77] R. Duncan Luce and Albert D. Perry. A method of matrix analysis of group structure. *Prychometrika*, 14(2):95–116, October 1949.
- [78] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.
- [79] Andrew McCallum, Andrs Corrada-emmanuel, and Xuerui Wang. Topic and role discovery in social networks. In *IJCAI*, pages 786–791, 2005.
- [80] Nicholas Metropolis and Stanislaw M. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, September 1949.
- [81] Rebekah A. Pure Metzger Miriam J., Christo Wilson and Ben Y. Zhao. Invisible interactions: What latent social interaction can tell us about social relationships in

- social network sites. *Networked Sociability and Individualism: Technology for Personal and Professional Relationships*, pages 79–102, 2012.
- [82] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, IMC '07*, pages 29–42, New York, NY, USA, 2007. ACM.
- [83] Samaneh Moghaddam and Martin Ester. On the design of LDA models for aspect-based opinion mining. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12*, pages 803–812, New York, NY, USA, 2012. ACM.
- [84] Robert Mokken. Cliques, clubs and clans. *Quality and Quantity: International Journal of Methodology*, 13(2):161–173, 1979.
- [85] Tsuyoshi Murata and Sakiko Moriyasu. Link prediction based on structural properties of online social networks. *New Generation Computing*, 26(3):245–257, 2008.
- [86] Nagarajan Natarajan, Prithviraj Sen, and Vineet Chaoji. Community detection in content-sharing social networks. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM '13*, pages 82–89, New York, NY, USA, 2013. ACM.
- [87] M. E. J. Newman. The structure and function of complex networks. *SIAM review*, 45:167–256, 2003.
- [88] M E J Newman and M Girvan. Finding and evaluating community structure in networks. *Pattern Recognition Letters*, 69(5):413–421, 2004.
- [89] M.E.J. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):025102, 2001.
- [90] J.-P. Onnela, S. Arbesman, M.C. González, A-L Barabási, and N.A. Christakis. Geographic constraints on social network groups. *PLOs ONE*, 7:e16939, 2011.
- [91] Daniel Ortiz-Arroyo. Discovering sets of key players in social networks. In *Computational Social Network Analysis*, Computer Communications and Networks, pages 27–47. Springer London, 2010.
- [92] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: bringing order to the web. In *Proceedings of the 7th International World Wide Web Conference*, pages 161–172, Brisbane, Australia, 1998.
- [93] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, June 2005.
- [94] Gergely Palla, Albert Işzl Barabasi, Tams Vicsek, and Budapest Hungary. Quantifying social group evolution. *Nature*, 446:664–667, April 2007.
- [95] Thelwall M. Paltoglou G. and Buckley. Online textual communications annotated with grades of emotion strength. In *Proceedings of the 3rd International Workshop on Emotion. EMOTION2010*, 2010.

- [96] Arindam Banerjee Nishith Pathak and Kendrick Erickson. Social topic models for community extraction. In *The 2nd SNA-KDD Workshop 08 (SNA-KDD08)*, Las Vegas, Nevada, USA, August 24 2008.
- [97] Yue Peng, Zhenyu Li, and Gaogang Xie. Weighting the edges in interactive online social network graphs. In *IEEE International Conference on Network Protocol (ICNP), Poster, Japan, 2010*.
- [98] Daniele Quercia, Licia Capra, and Jon Crowcroft. The social world of Twitter: topics, geography, and emotions. In *ICWSM*. The AAAI Press, 2012.
- [99] U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3), 2007.
- [100] Anand Rajaraman and Jeffrey David Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA, 2011.
- [101] Matthew J. Rattigan, Marc Maier, and David Jensen. Graph clustering with network structure indices. In *Proceedings of the 24th International Conference on Machine learning, ICML '07*, pages 783–790, New York, NY, USA, 2007. ACM.
- [102] Mrinmaya Sachan, Danish Contractor, Tanveer A. Faruque, and L. Venkata Subramaniam. Using content and interactions for discovering communities in social networks. In *Proceedings of the 21st International Conference on World Wide Web, WWW '12*, pages 331–340, New York, NY, USA, 2012. ACM.
- [103] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November 1975.
- [104] Jari Saramäki and Kimmo Kaski. Scale-free networks generated by random walkers. *Physica A*, 341:80–86, 2004.
- [105] Salvatore Scellato. *Spatial properties of online social services: measurement, analysis and applications*. PhD thesis, University of Cambridge, 2012.
- [106] Mikkel N. Schmidt and Morten Mørup. Nonparametric bayesian modeling of complex networks: An introduction. *IEEE Signal Process. Mag.*, 30(3):110–128, 2013.
- [107] John Scott. *Social Network Analysis: A Handbook*. Sage Publications, second edition, 2000.
- [108] Christian Sengstock, Michael Gertz, and Tran Van Canh. Spatial interestingness measures for co-location pattern mining. *2012 IEEE 12th International Conference on Data Mining Workshops*, 0:821–826, 2012.
- [109] Myra Spiliopoulou, Irene Ntoutsi, Yannis Theodoridis, and Rene Schult. Monic: modeling and monitoring cluster transitions. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge discovery and Data mining, KDD '06*, pages 706–711, New York, NY, USA, 2006. ACM.
- [110] Steffen Staab, Pedro Domingos, Peter Mika, Jennifer Golbeck, Li Ding, Tim Finin, Anupam Joshi, Andrzej Nowak, and Robin R. Vallacher. Social networks applied. *IEEE Intelligent Systems*, 20(1):80–93, January 2005.

- [111] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009:4:2–4:2, January 2009.
- [112] Erik B. Sudderth. *Graphical Models for Visual Object Recognition and Tracking*. PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2006.
- [113] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101, 2004.
- [114] Bart Thomee and Adam Rae. Uncovering locally characterizing regions within geo-tagged data. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1285–1296, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.
- [115] S. M. van Dongen. *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht, The Netherlands, 2000.
- [116] Alexei Vázquez. Growing network with local rules: preferential attachment, clustering hierarchy, and degree correlations. *Physical Review E*, 67(5), May 2003.
- [117] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the evolution of user interaction in facebook. In *Proceedings of the 2Nd ACM Workshop on Online Social Networks*, WOSN '09, pages 37–42, New York, USA, 2009. ACM.
- [118] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, January 2008.
- [119] Xiaogang Wang and Eric Grimson. Spatial latent Dirichlet allocation. In *Proceedings of Neural Information Processing Systems Conference (NIPS) 2007*, 2007.
- [120] Xuerui Wang, Natasha Mohanty, and Andrew McCallum. Group and topic discovery from relations and text. In *Proceedings of the 3rd International Workshop on Link Discovery*, LinkKDD '05, pages 28–35, New York, NY, USA, 2005. ACM.
- [121] Stanley Wasserman and Katherine Faust. *Social Network Analysis: methods and applications*. Cambridge University Press, 1 edition, 1994.
- [122] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6684):440–442, June 1998.
- [123] U.K. Wiil, J. Gniadek, and N. Memon. Measuring link importance in terrorist networks. In *Proceedings of the 2010 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '10, pages 225–232, IEEE Computer Society, 2010.
- [124] Christo Wilson, Bryce Boe, Alessandra Sala, Krishna P.N. Puttaswamy, and Ben Y. Zhao. User interactions in social networks and their implications. In *Proceedings of the 4th ACM European Conference on Computer Systems*, EuroSys '09, pages 205–218, New York, NY, USA, 2009. ACM.
- [125] Jierui Xie, Stephen Kelley, and Boleslaw K. Szymanski. Overlapping community detection in networks: the state-of-the-art and comparative study. *ACM Comput. Surv.*, 45(4):43:1–43:35, August 2013.

- [126] Christopher C. Yang and Xuning Tang. Social networks integration and privacy preservation using subgraph generalization. In *Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics*, CSI-KDD '09, pages 53–61, New York, NY, USA, 2009. ACM.
- [127] Hui Yang, Srinivasan Parthasarathy, and Sameep Mehta. A generalized framework for mining spatio-temporal patterns in scientific data. In *Patterns in Scientific Data, ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pages 716–721, 2005.
- [128] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Detecting communities and their evolutions in dynamic social networks: a Bayesian approach. *Machine Learning*, 82:157–189, 2011. 10.1007/s10994-010-5214-7.
- [129] Zhijun Yin, Liangliang Cao, Quanquan Gu, and Jiawei Han. Latent community topic analysis: integration of community discovery with topic modeling. *ACM Trans. Intell. Syst. Technol.*, 3(4):63:1–63:21, September 2012.
- [130] Kai Yu, Shipeng Yu, and Volker Tresp. Soft clustering on graphs. In *Advances in Neural Information Processing Systems*, page 05, 2005.
- [131] Zengfeng Zeng and Bin Wu. Detecting probabilistic community with topic modeling on sampling subgraphs. In *Proceedings of the 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '12, pages 623–630. IEEE Computer Society, 2012.
- [132] Chenyi Zhang and Jianling Sun. Large scale microblog mining using distributed mblda. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 1035–1042, New York, NY, USA, 2012. ACM.
- [133] Haizheng Zhang, C. Lee Giles, Henry C. Foley, and John Yen. Probabilistic community discovery using hierarchical latent gaussian mixture model. In *Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 1*, AAAI'07, pages 663–668. AAAI Press, 2007.
- [134] Haizheng Zhang, Baojun Qiu, C. Lee Giles, Henry C. Foley, and John Yen. An LDA-based community structure discovery approach for large-scale social networks. In *Proceedings of Intelligence and Security Informatics*, ISI '07, pages 200–207, 2007.
- [135] Guoqing Zheng, Jinwen Guo, Lichun Yang, Shengliang Xu, Shenghua Bao, Zhong Su, Dingyi Han, and Yong Yu. Mining topics on participations for community discovery. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 445–454, New York, NY, USA, 2011. ACM.
- [136] Ding Zhou, Eren Manavoglu, Jia Li, C. Lee Giles, and Hongyuan Zha. Probabilistic models for discovering e-communities. In *Proceedings of the 15th International Conference on World Wide Web*, WWW '06, pages 173–182, New York, NY, USA, 2006. ACM.